CSPFA

Cisco Secure PIX Firewall Advanced

Student Guide

Version 3.2

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706 USA

The products and specifications, configurations, and other technical information regarding the products in this manual are subject to change without notice. All statements, technical information, and recommendations in this manual are believed to be accurate but are presented without warranty of any kind, express or implied. You must take full responsibility for their application of any products specified in this manual.

LICENSE

PLEASE READ THESE TERMS AND CONDITIONS CAREFULLY BEFORE USING THE MANUAL, DOCUMENTATION, AND/OR SOFTWARE (“MATERIALS”). BY USING THE MATERIALS YOU AGREE TO BE BOUND BY THE TERMS AND CONDITIONS OF THIS LICENSE. IF YOU DO NOT AGREE WITH THE TERMS OF THIS LICENSE, PROMPTLY RETURN THE UNUSED MATERIALS (WITH PROOF OF PAYMENT) TO THE PLACE OF PURCHASE FOR A FULL REFUND.

Cisco Systems, Inc. (“Cisco”) and its suppliers grant to you (“You”) a nonexclusive and nontransferable license to use the Cisco Materials solely for Your own personal use. If the Materials include Cisco software (“Software”), Cisco grants to You a nonexclusive and nontransferable license to use the Software in object code form solely on a single central processing unit owned or leased by You or otherwise embedded in equipment provided by Cisco. You may make one (1) archival copy of the Software provided You affix to such copy all copyright, confidentiality, and proprietary notices that appear on the original. EXCEPT AS EXPRESSLY AUTHORIZED ABOVE, YOU SHALL NOT: COPY, IN WHOLE OR IN PART, MATERIALS; MODIFY THE SOFTWARE; REVERSE COMPILE OR REVERSE ASSEMBLE ALL OR ANY PORTION OF THE SOFTWARE; OR RENT, LEASE, DISTRIBUTE, SELL, OR CREATE DERIVATIVE WORKS OF THE MATERIALS.

You agree that aspects of the licensed Materials, including the specific design and structure of individual programs, constitute trade secrets and/or copyrighted material of Cisco. You agree not to disclose, provide, or otherwise make available such trade secrets or copyrighted material in any form to any third party without the prior written consent of Cisco. You agree to implement reasonable security measures to protect such trade secrets and copyrighted Material. Title to the Materials shall remain solely with Cisco.

This License is effective until terminated. You may terminate this License at any time by destroying all copies of the Materials. This License will terminate immediately without notice from Cisco if You fail to comply with any provision of this License. Upon termination, You must destroy all copies of the Materials.

Software, including technical data, is subject to U.S. export control laws, including the U.S. Export Administration Act and its associated regulations, and may be subject to export or import regulations in other countries. You agree to comply strictly with all such regulations and acknowledge that it has the responsibility to obtain licenses to export, re-export, or import Software.

This License shall be governed by and construed in accordance with the laws of the State of California, United States of America, as if performed wholly within the state and without giving effect to the principles of conflict of law. If any portion hereof is found to be void or unenforceable, the remaining provisions of this License shall remain in full force and effect. This License constitutes the entire License between the parties with respect to the use of the Materials

Restricted Rights - Cisco’s software is provided to non-DOD agencies with RESTRICTED RIGHTS and its supporting documentation is provided with LIMITED RIGHTS. Use, duplication, or disclosure by the U.S. Government is subject to the restrictions as set forth in subparagraph “C” of the Commercial Computer Software - Restricted Rights clause at FAR 52.227-19. In the event the sale is to a DOD agency, the U.S. Government’s rights in software, supporting documentation, and technical data are governed by the restrictions in the Technical Data Commercial Items clause at DFARS 252.227-7015 and DFARS 227.7202.

DISCLAIMER OF WARRANTY. ALL MATERIALS ARE PROVIDED “AS IS” WITH ALL FAULTS. CISCO AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. In no event shall Cisco’s or its suppliers’ liability to You, whether in contract, tort (including negligence), or otherwise, exceed the price paid by You. The foregoing limitations shall apply even if the above-stated warranty fails of its essential purpose.

The following information is for FCC compliance of Class A devices: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio-frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case users will be required to correct the interference at their own expense.

The following information is for FCC compliance of Class B devices: The equipment described in this manual generates and may radiate radio-frequency energy. If it is not installed in accordance with Cisco’s installation instructions, it may cause interference with radio and television reception. This equipment has been tested and found to comply with the limits for a Class B digital device in accordance with the specifications in part 15 of the FCC rules. These specifications are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation.

You can determine whether your equipment is causing interference by turning it off. If the interference stops, it was probably caused by the Cisco equipment or one of its peripheral devices. If the equipment causes interference to radio or television reception, try to correct the interference by using one or more of the following measures:

- Turn the television or radio antenna until the interference stops.
- Move the equipment to one side or the other of the television or radio.
- Move the equipment farther away from the television or radio.
- Plug the equipment into an outlet that is on a different circuit from the television or radio. (That is, make certain the equipment and the television or radio are on circuits controlled by different circuit breakers or fuses.)

Modifications to this product not authorized by Cisco Systems, Inc. could void the FCC approval and negate your authority to operate the product.

The following third-party software may be included with your product and will be subject to the software license agreement:

CiscoWorks software and documentation are based in part on HP OpenView under license from the Hewlett-Packard Company. HP OpenView is a trademark of the Hewlett-Packard Company. Copyright © 1992, 1993 Hewlett-Packard Company.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

Network Time Protocol (NTP). Copyright © 1992, David L. Mills. The University of Delaware makes no representations about the suitability of this software for any purpose.

Point-to-Point Protocol. Copyright © 1989, Carnegie-Mellon University. All rights reserved. The name of the University may not be used to endorse or promote products derived from this software without specific prior written permission.

The Cisco implementation of TN3270 is an adaptation of the TN3270, curses, and termcap programs developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981-1988, Regents of the University of California.

Cisco incorporates Fastmac and TrueView software and the RingRunner chip in some Token Ring products. Fastmac software is licensed to Cisco by Madge Networks Limited, and the RingRunner chip is licensed to Cisco by Madge NV. Fastmac, RingRunner, and TrueView are trademarks and in some jurisdictions registered trademarks of Madge Networks Limited. Copyright © 1995, Madge Networks Limited. All rights reserved.

XRemote is a trademark of Network Computing Devices, Inc. Copyright © 1989, Network Computing Devices, Inc., Mountain View, California. NCD makes no representations about the suitability of this software for any purpose.

The X Window System is a trademark of the X Consortium, Cambridge, Massachusetts. All rights reserved.

Cisco Systems has more than 200 offices in the following countries and regions. Addresses, phone numbers, and fax numbers are listed on the Cisco Web site at www.cisco.com/go/offices.

Argentina Australia Austria Belgium Brazil Bulgaria Canada Chile China PRC Colombia Costa Rica Croatia Czech Republic Denmark Dubai, UAE Finland France Germany Greece Hong Kong SAR Hungary India Indonesia Ireland Israel Italy Japan Korea Luxembourg Malaysia Mexico The Netherlands New Zealand Norway Peru Philippines Poland Portugal Puerto Rico Romania Russia Saudi Arabia Scotland Singapore Slovakia Slovenia South Africa Spain Sweden Switzerland Taiwan Thailand Turkey Ukraine United Kingdom United States Venezuela Vietnam Zimbabwe

Cisco Secure PIX Firewall Advanced, Revision 3.1: Instructor Preparation Guide

Table of Contents

1

Course Introduction

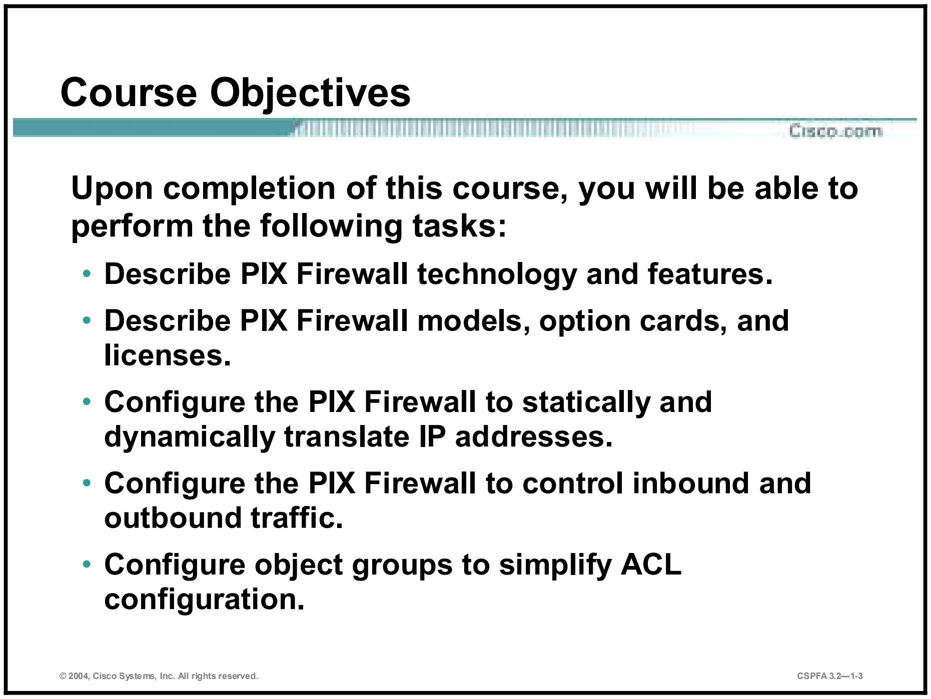
Overview

This lesson includes the following topics:

- Course objectives
- Course agenda
- Participant responsibilities
- General administration
- Graphic symbols
- Participant introductions
- Lab topology overview

Course Objectives

This topic introduces the course and the course objectives.

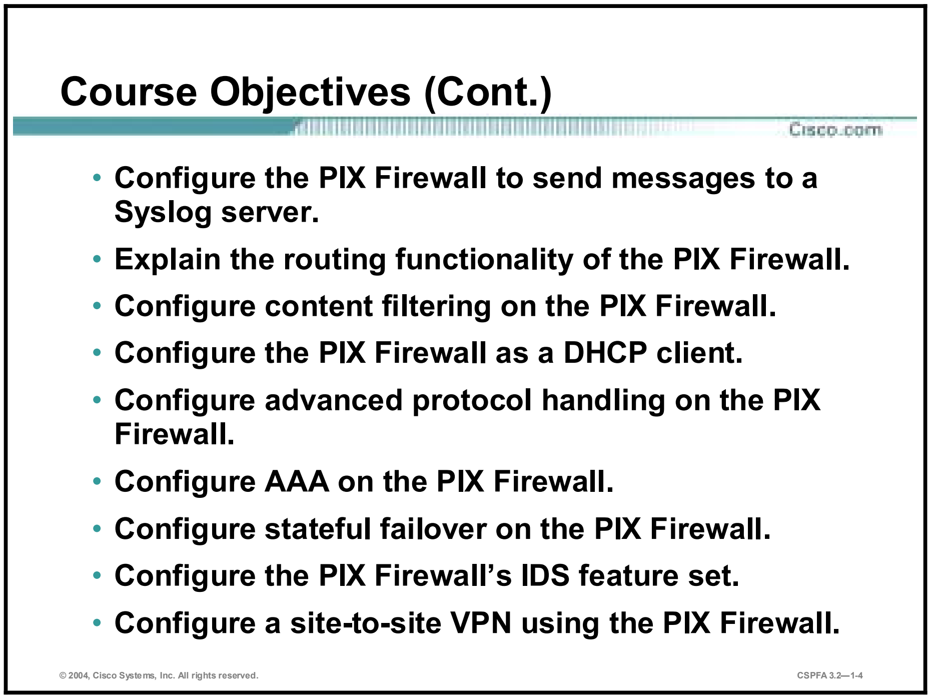


Course Objectives

Upon completion of this course, you will be able to perform the following tasks:

- **Describe PIX Firewall technology and features.**
- **Describe PIX Firewall models, option cards, and licenses.**
- **Configure the PIX Firewall to statically and dynamically translate IP addresses.**
- **Configure the PIX Firewall to control inbound and outbound traffic.**
- **Configure object groups to simplify ACL configuration.**

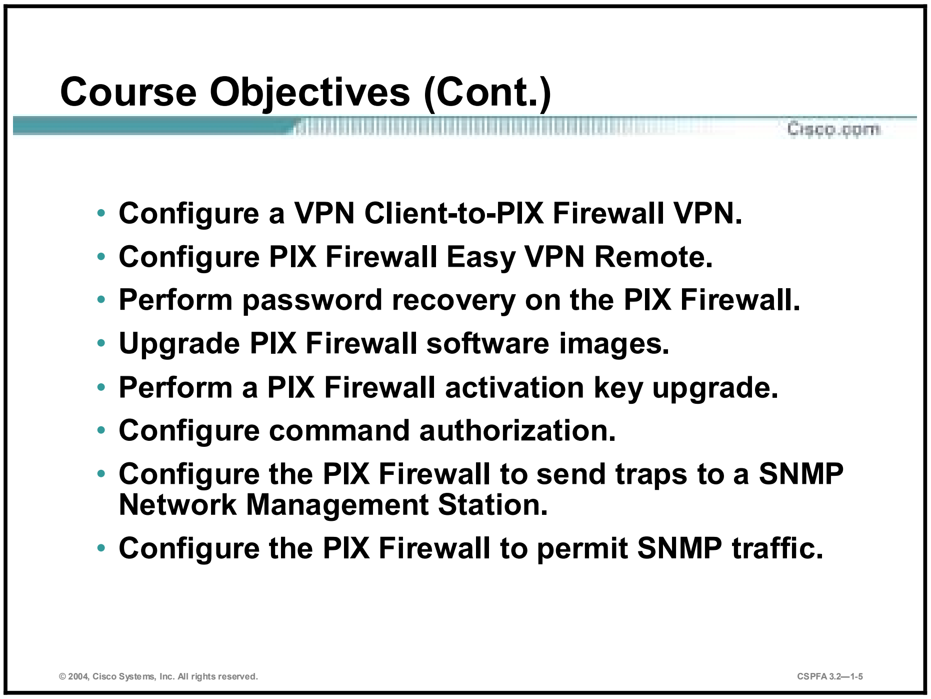
CSPFA 3.2—1-3



Course Objectives (Cont.)

- **Configure the PIX Firewall to send messages to a Syslog server.**
- **Explain the routing functionality of the PIX Firewall.**
- **Configure content filtering on the PIX Firewall.**
- **Configure the PIX Firewall as a DHCP client.**
- **Configure advanced protocol handling on the PIX Firewall.**
- **Configure AAA on the PIX Firewall.**
- **Configure stateful failover on the PIX Firewall.**
- **Configure the PIX Firewall’s IDS feature set.**
- **Configure a site-to-site VPN using the PIX Firewall.**

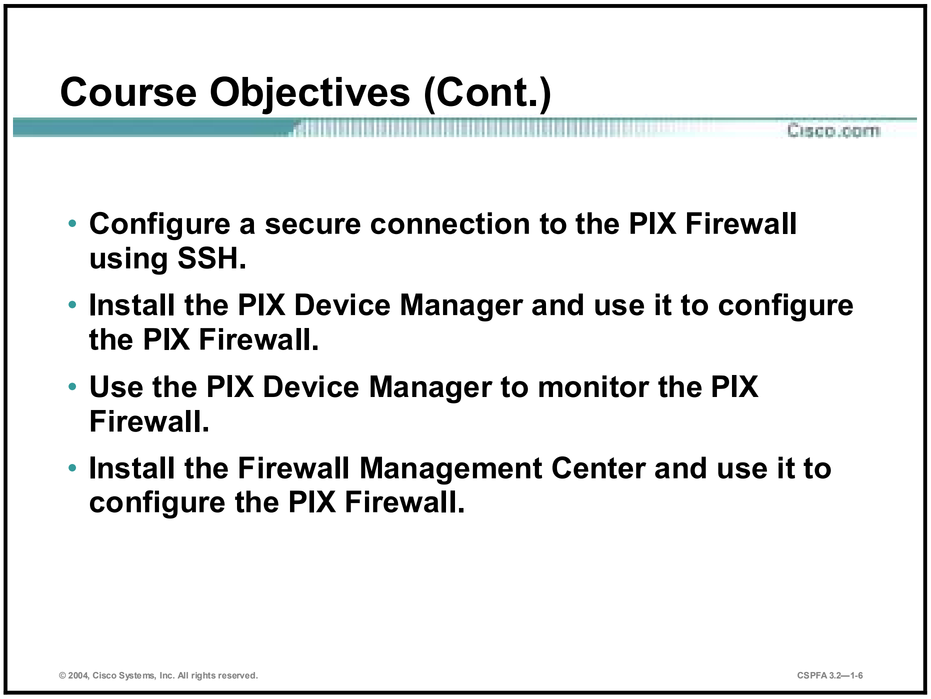
CSPFA 3.2—1-4



Course Objectives (Cont.)

- Configure a VPN Client-to-PIX Firewall VPN.
- Configure PIX Firewall Easy VPN Remote.
- Perform password recovery on the PIX Firewall.
- Upgrade PIX Firewall software images.
- Perform a PIX Firewall activation key upgrade.
- Configure command authorization.
- Configure the PIX Firewall to send traps to a SNMP Network Management Station.
- Configure the PIX Firewall to permit SNMP traffic.

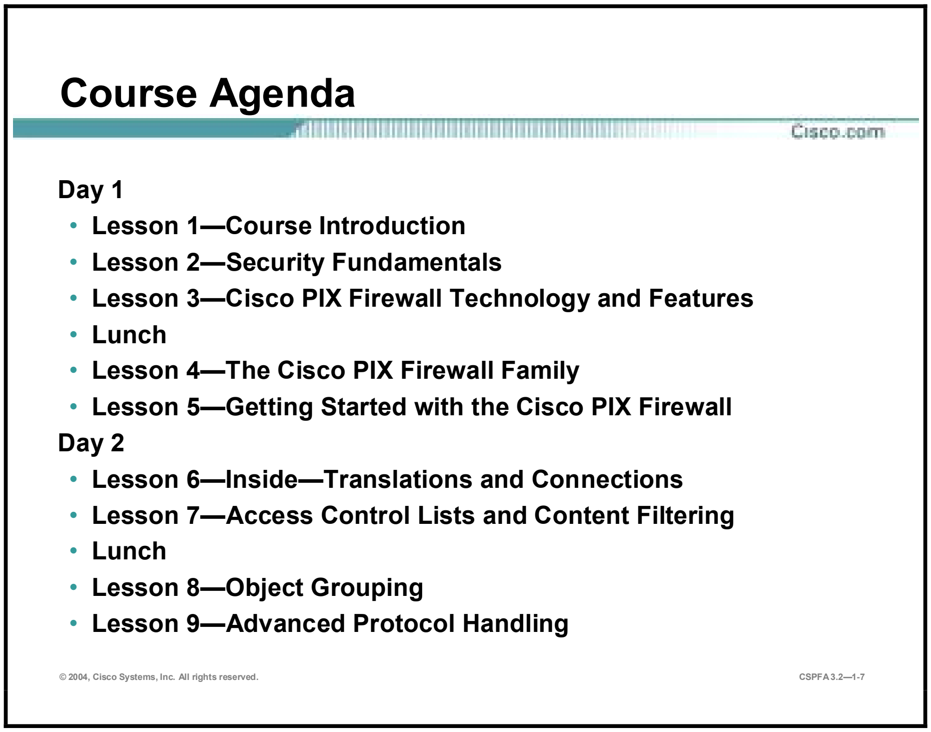
CSPFA 3.2—1-5



Course Objectives (Cont.)

- Configure a secure connection to the PIX Firewall using SSH.
- Install the PIX Device Manager and use it to configure the PIX Firewall.
- Use the PIX Device Manager to monitor the PIX Firewall.
- Install the Firewall Management Center and use it to configure the PIX Firewall.

CSPFA 3.2—1-6



Course Agenda

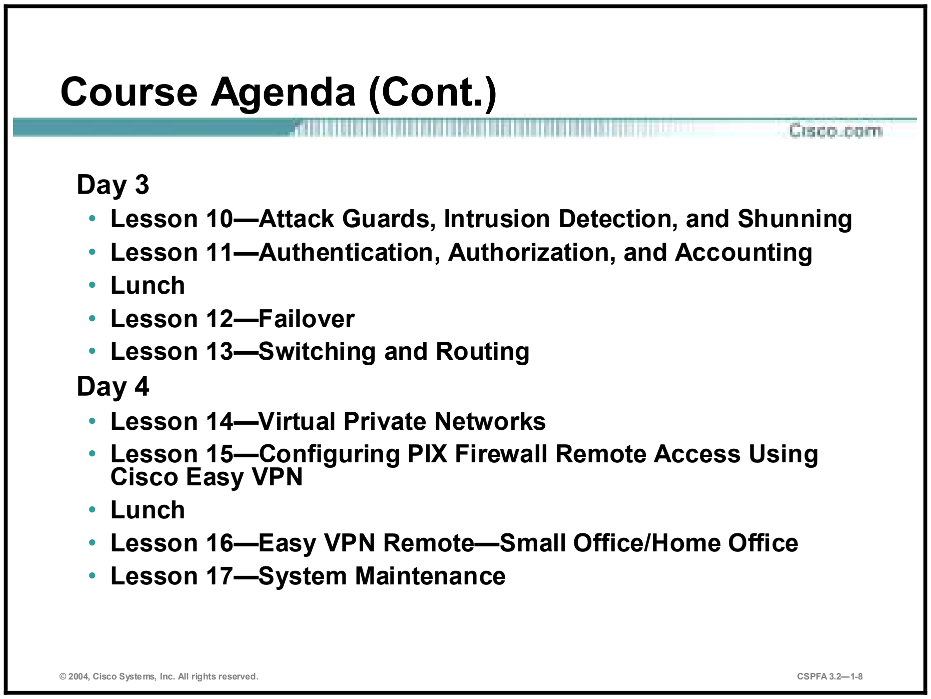
Day 1

- Lesson 1—Course Introduction
- Lesson 2—Security Fundamentals
- Lesson 3—Cisco PIX Firewall Technology and Features
- Lunch
- Lesson 4—The Cisco PIX Firewall Family
- Lesson 5—Getting Started with the Cisco PIX Firewall

Day 2

- Lesson 6—Inside—Translations and Connections
- Lesson 7—Access Control Lists and Content Filtering
- Lunch
- Lesson 8—Object Grouping
- Lesson 9—Advanced Protocol Handling

CSPFA 3.2—1-7



Course Agenda (Cont.)

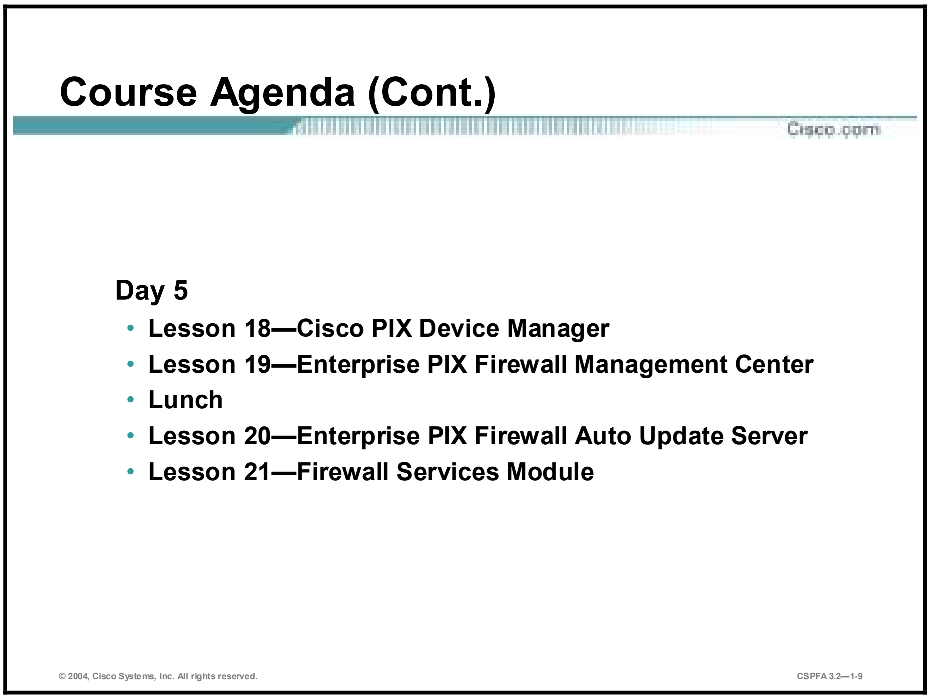
Day 3

- Lesson 10—Attack Guards, Intrusion Detection, and Shunning
- Lesson 11—Authentication, Authorization, and Accounting
- Lunch
- Lesson 12—Failover
- Lesson 13—Switching and Routing

Day 4

- Lesson 14—Virtual Private Networks
- Lesson 15—Configuring PIX Firewall Remote Access Using Cisco Easy VPN
- Lunch
- Lesson 16—Easy VPN Remote—Small Office/Home Office
- Lesson 17—System Maintenance

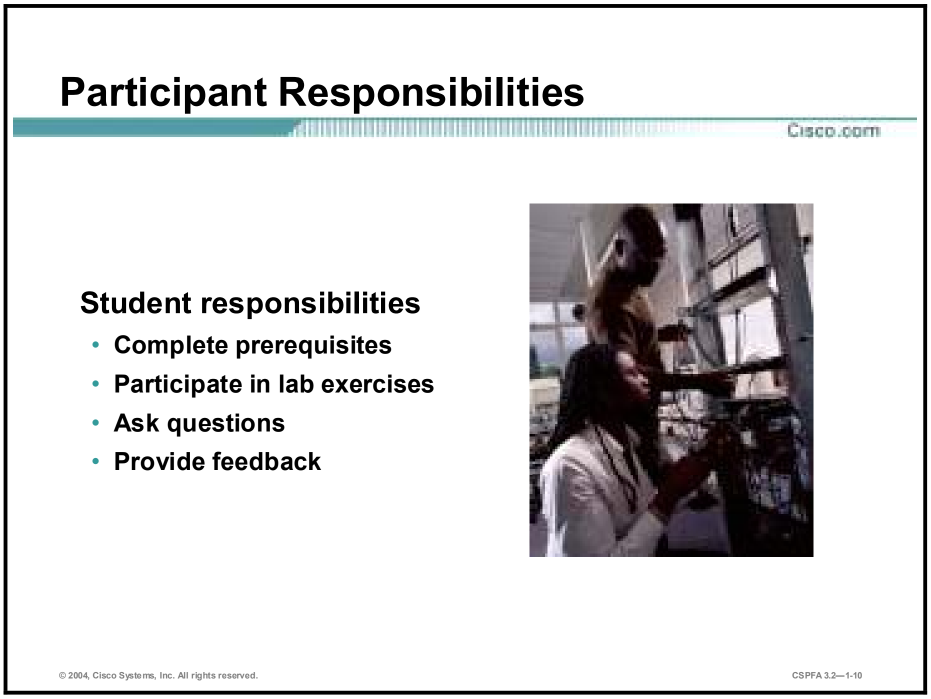
CSPFA 3.2—1-8



Course Agenda (Cont.)

Day 5

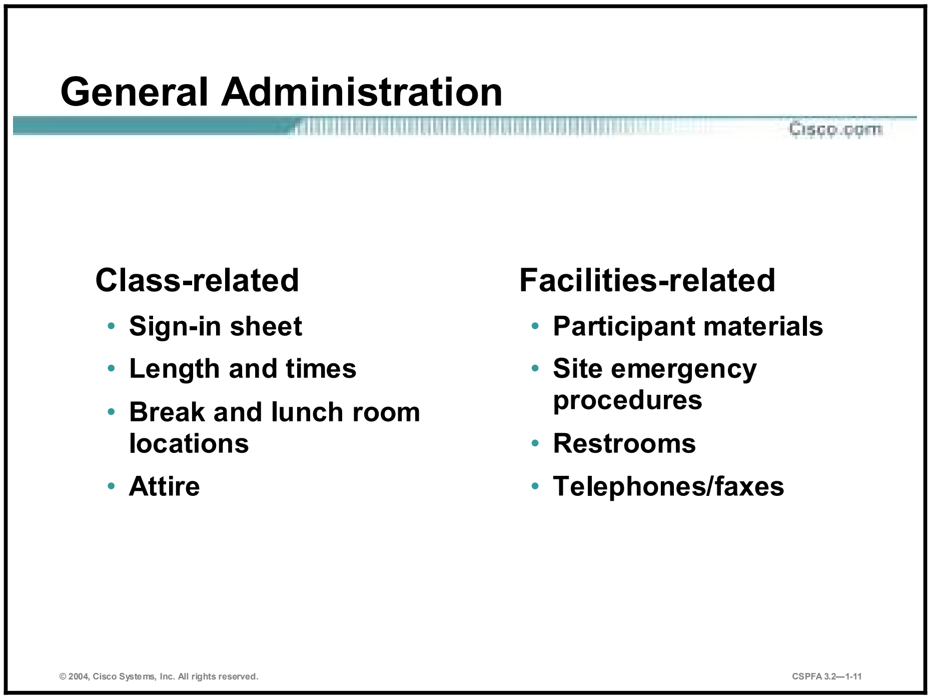
- Lesson 18—Cisco PIX Device Manager
- Lesson 19—Enterprise PIX Firewall Management Center
- Lunch
- Lesson 20—Enterprise PIX Firewall Auto Update Server
- Lesson 21—Firewall Services Module



Participant Responsibilities

Student responsibilities

- Complete prerequisites
- Participate in lab exercises
- Ask questions
- Provide feedback



General Administration

Cisco.com

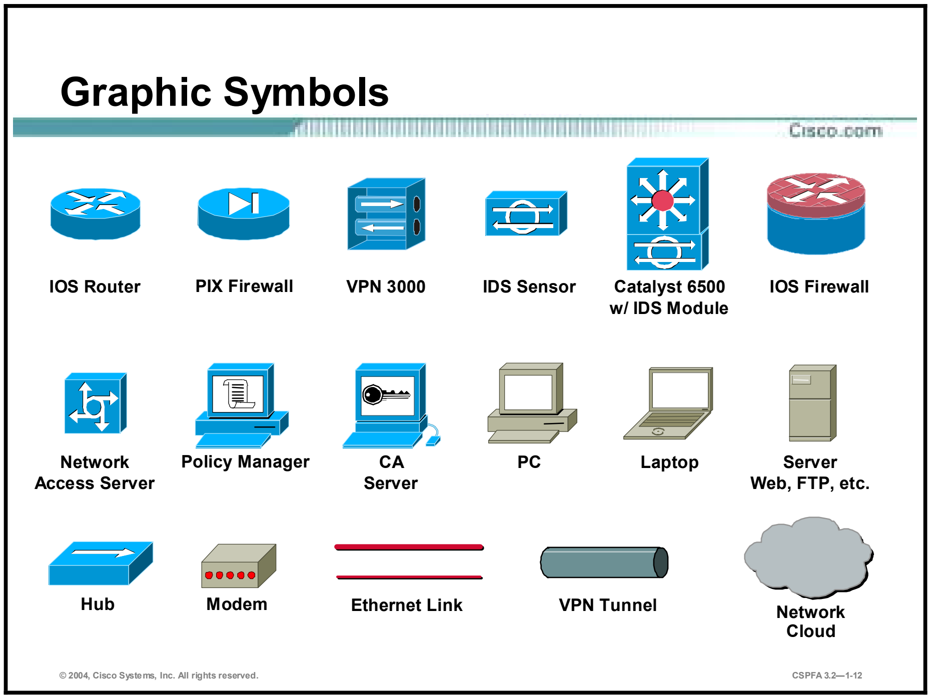
Class-related

- Sign-in sheet
- Length and times
- Break and lunch room locations
- Attire

Facilities-related

- Participant materials
- Site emergency procedures
- Restrooms
- Telephones/faxes

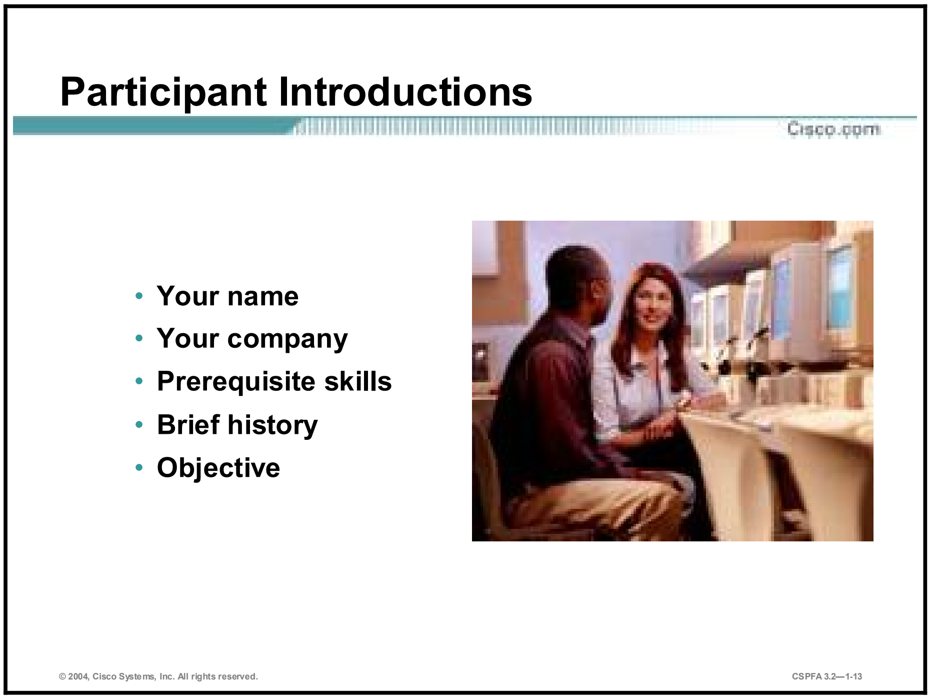
CSPFA 3.2—1-11



Graphic Symbols

Cisco.com

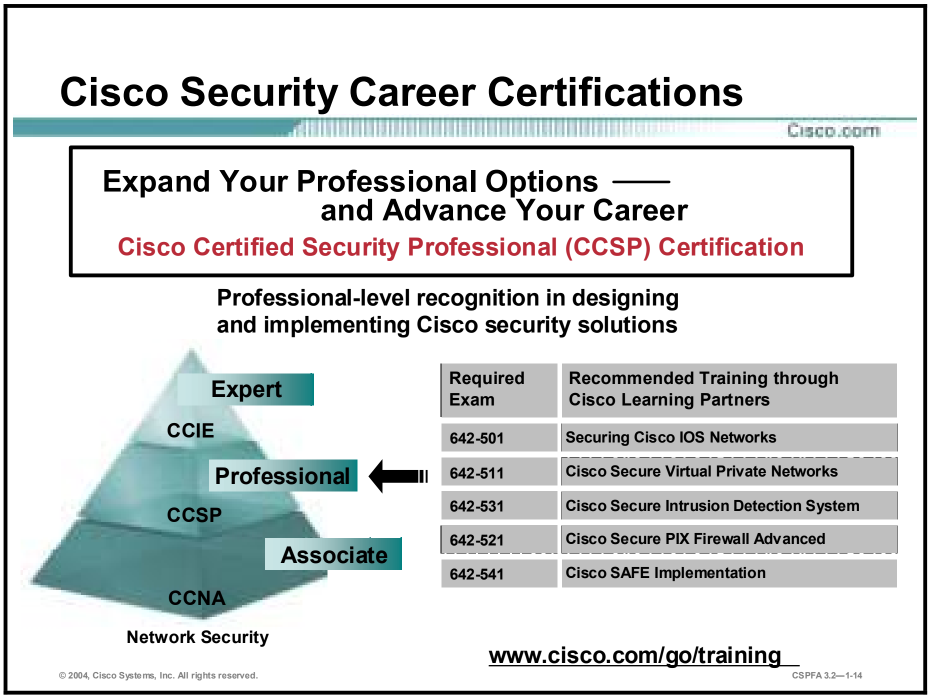
CSPFA 3.2—1-12



Participant Introductions

Cisco.com

- Your name
- Your company
- Prerequisite skills
- Brief history
- Objective



Cisco Security Career Certifications

Cisco.com

Expand Your Professional Options —— and Advance Your Career

Cisco Certified Security Professional (CCSP) Certification

Professional-level recognition in designing and implementing Cisco security solutions

Expert

CCIE

Professional

CCSP

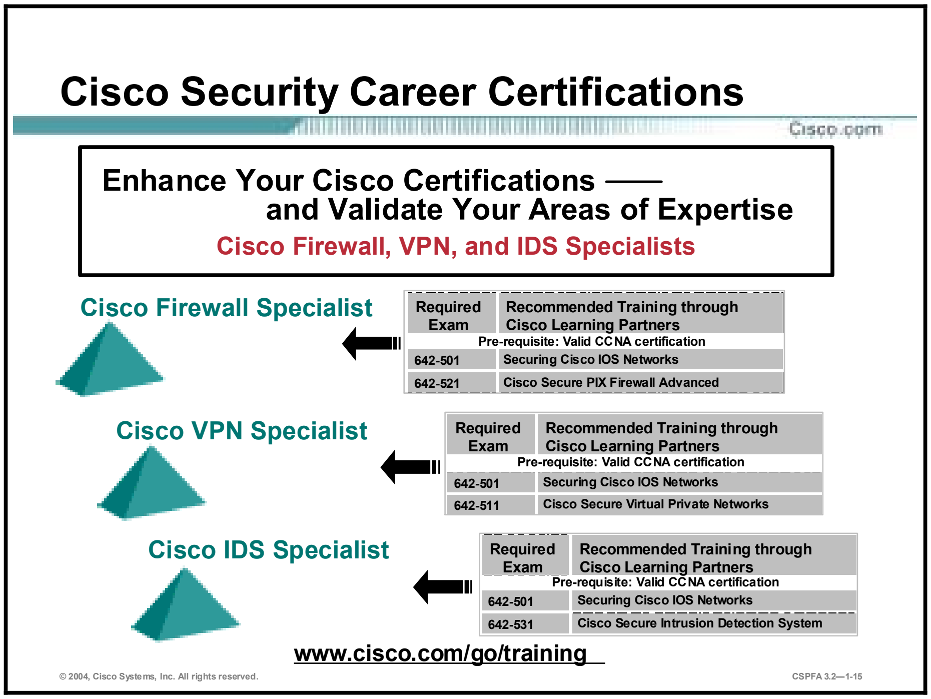
Associate

CCNA

Network Security

Required Exam	Recommended Training through Cisco Learning Partners
642-501	Securing Cisco IOS Networks
642-511	Cisco Secure Virtual Private Networks
642-531	Cisco Secure Intrusion Detection System
642-521	Cisco Secure PIX Firewall Advanced
642-541	Cisco SAFE Implementation

www.cisco.com/go/training



Cisco Security Career Certifications

Enhance Your Cisco Certifications —— and Validate Your Areas of Expertise

Cisco Firewall, VPN, and IDS Specialists

Cisco Firewall Specialist

Required Exam	Recommended Training through Cisco Learning Partners
Pre-requisite: Valid CCNA certification	
642-501	Securing Cisco IOS Networks
642-521	Cisco Secure PIX Firewall Advanced

Cisco VPN Specialist

Required Exam	Recommended Training through Cisco Learning Partners
Pre-requisite: Valid CCNA certification	
642-501	Securing Cisco IOS Networks
642-511	Cisco Secure Virtual Private Networks

Cisco IDS Specialist

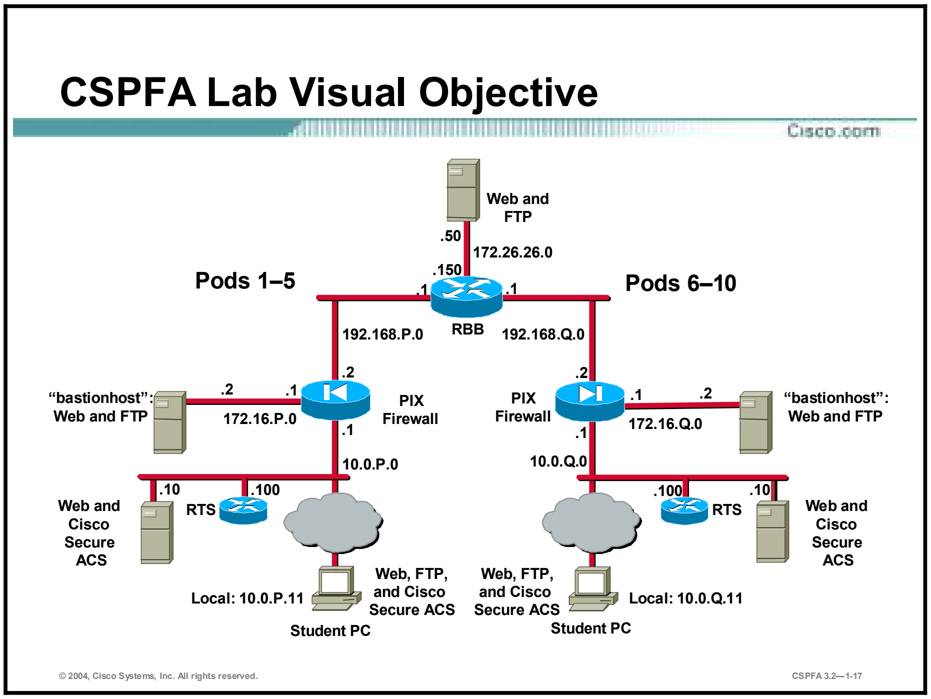
Required Exam	Recommended Training through Cisco Learning Partners
Pre-requisite: Valid CCNA certification	
642-501	Securing Cisco IOS Networks
642-531	Cisco Secure Intrusion Detection System

www.cisco.com/go/training

CSPFA 3.2—1-15

Lab Topology Overview

This topic explains the lab topology used in this course.

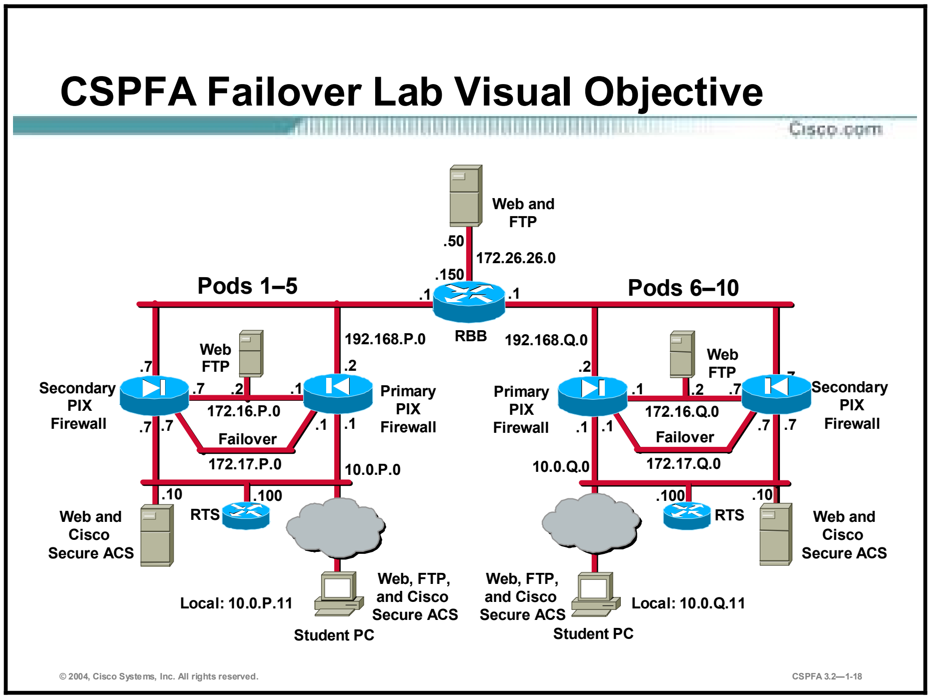


CSPFA Lab Visual Objective

CSPFA 3.2—1-17

Each pair of students will be assigned a pod. In general, you will be setting up virtual private networks (VPNs) between your pod (pod P) and your assigned peer pod (pod Q).

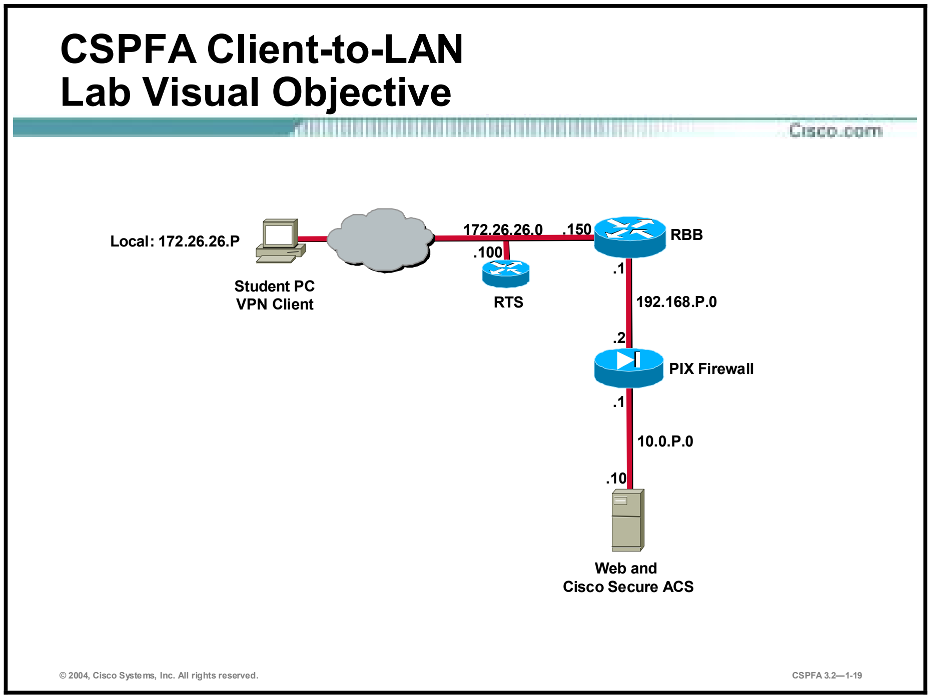
Note	The P in a command indicates your pod number. The Q in a command indicates the pod number of your peer router.



CSPFA Failover Lab Visual Objective

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—1-18



CSPFA Client-to-LAN Lab Visual Objective

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—1-19

2

Security Fundamentals

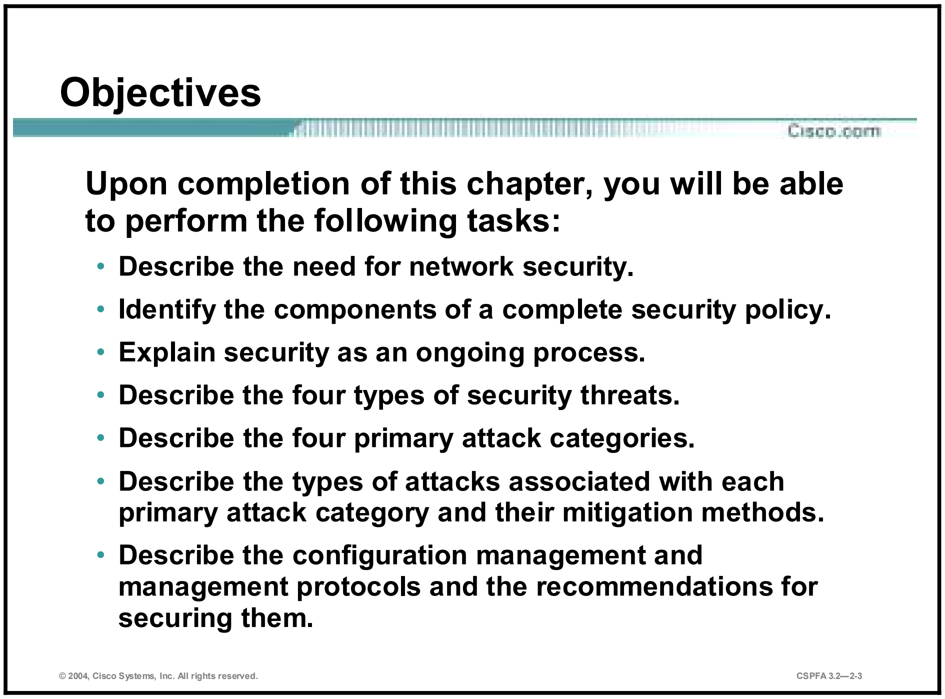
Overview

This lesson describes security fundamentals. It includes the following topics:

- Objectives
- Need for network security
- Network security policy
- Primary network threats and attacks
- Reconnaissance attacks and mitigation
- Access attacks and mitigation
- Denial of service attacks and mitigation
- Worm, virus, and Trojan horse attacks and mitigation
- Management protocols and functions
- Summary

Objectives

This topic lists the lesson's objectives.



Objectives

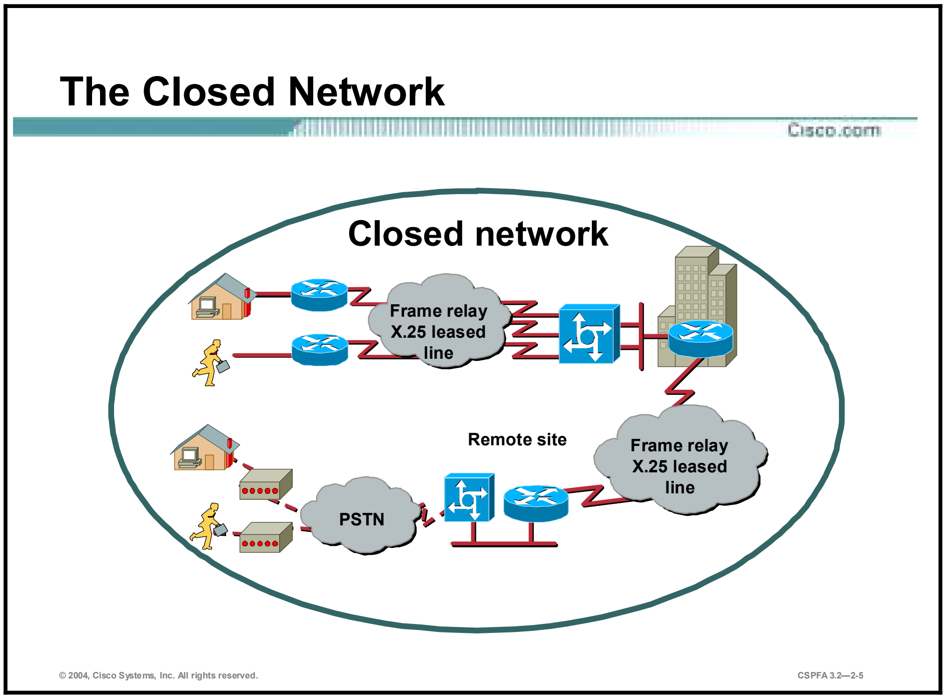
Upon completion of this chapter, you will be able to perform the following tasks:

- Describe the need for network security.
- Identify the components of a complete security policy.
- Explain security as an ongoing process.
- Describe the four types of security threats.
- Describe the four primary attack categories.
- Describe the types of attacks associated with each primary attack category and their mitigation methods.
- Describe the configuration management and management protocols and the recommendations for securing them.

CSPFA 3.2—2-3

Need for Network Security

Over the past few years, Internet-enabled business, or e-business, has drastically improved companies' efficiency and revenue growth. E-business applications such as e-commerce, supply-chain management, and remote access enable companies to streamline processes, lower operating costs, and increase customer satisfaction. Such applications require mission-critical networks that accommodate voice, video, and data traffic, and these networks must be scalable to support increasing numbers of users and the need for greater capacity and performance. However, as networks enable more and more applications and are available to more and more users, they become ever more vulnerable to a wider range of security threats. To combat those threats and ensure that e-business transactions are not compromised, security technology must play a major role in today's networks.

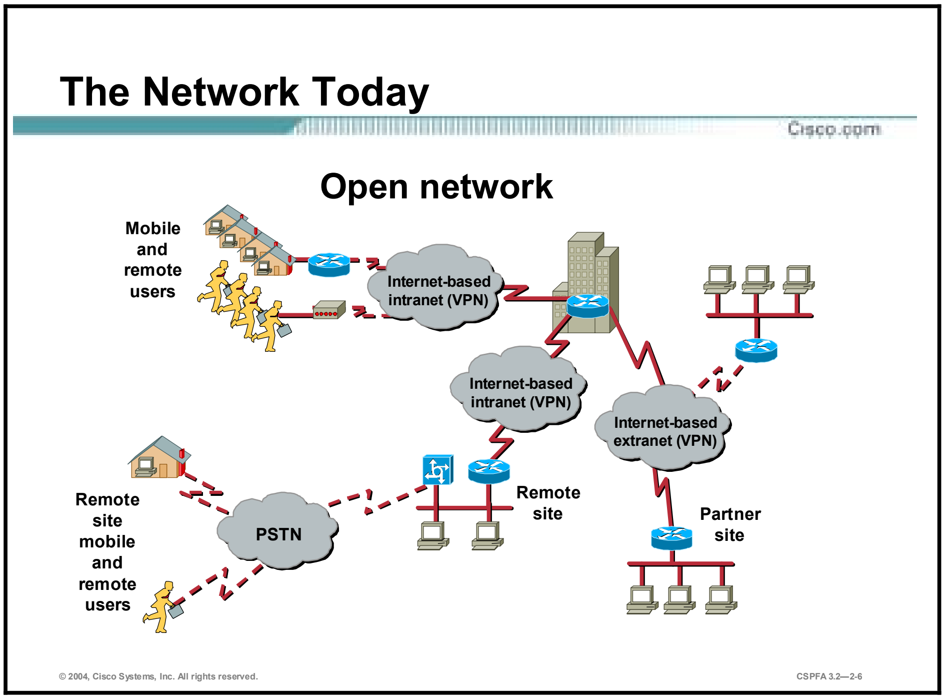


The Closed Network

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—2-5

The closed network typically consists of a network designed and implemented in a corporate environment, and it provides connectivity only to known parties and sites without connecting to public networks. Networks were designed this way in the past and thought to be reasonably secure because there was no outside connectivity.



The Network Today

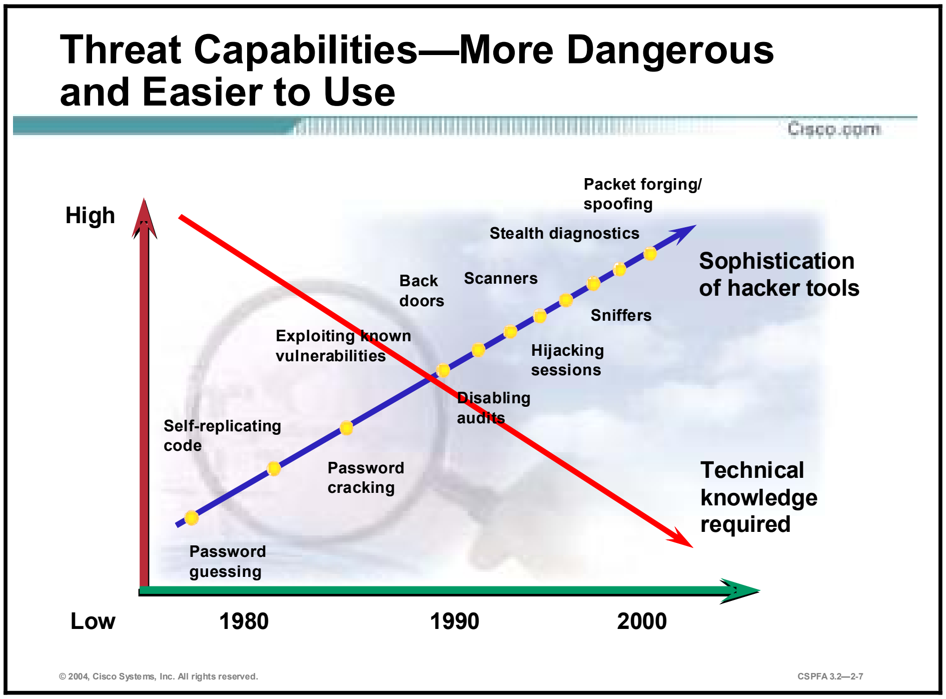
Cisco.com

Open network

Mobile and remote users
Internet-based intranet (VPN)
Internet-based intranet (VPN)
Internet-based extranet (VPN)
Remote site mobile and remote users
PSTN
Remote site
Partner site

CSPFA 3.2—2-6

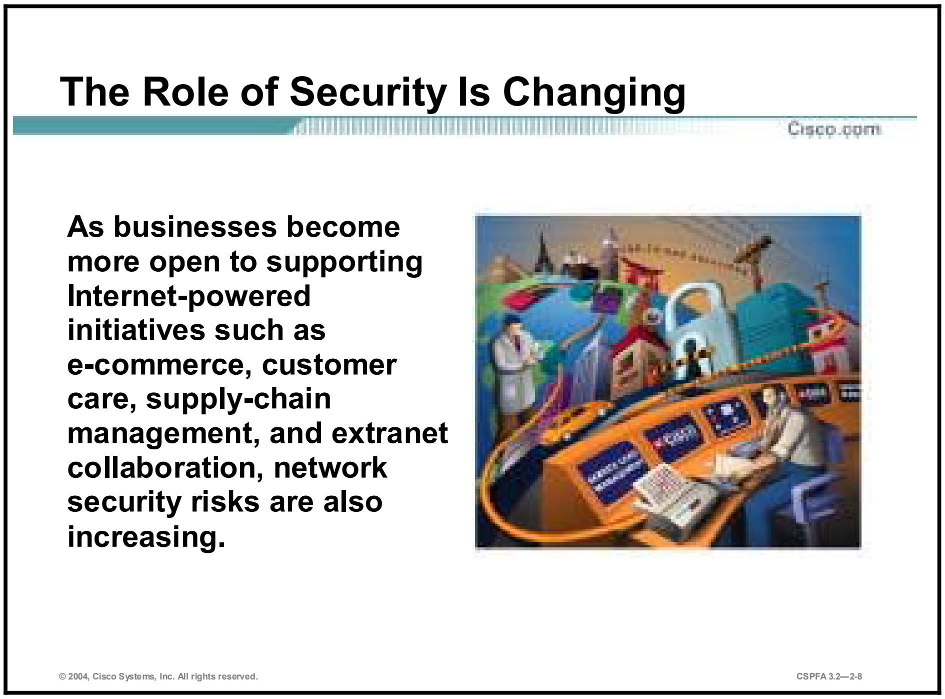
The networks of today are designed with availability to the Internet and public networks, which is a major requirement. Most of today’s networks have several access points to other networks both public and private; therefore, securing these networks has become fundamentally important.



Threat Capabilities—More Dangerous and Easier to Use

CSPFA 3.2—2-7

With the development of large open networks there has been a huge increase in security threats in the past 20 years. Not only have hackers discovered more vulnerabilities, but the tools used to hack a network have become simpler and the technical knowledge required has decreased. There are downloadable applications available that require little or no hacking knowledge to implement. There are also applications intended for troubleshooting a network that when used improperly can pose severe threats.



The Role of Security Is Changing

Cisco.com

As businesses become more open to supporting Internet-powered initiatives such as e-commerce, customer care, supply-chain management, and extranet collaboration, network security risks are also increasing.

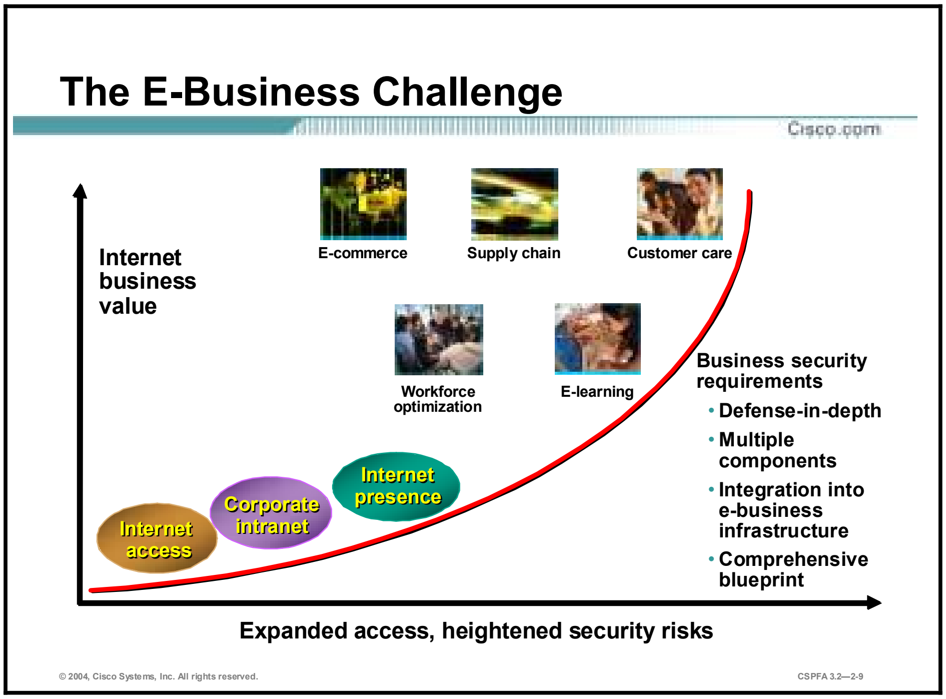
© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—2-8

Security has moved to the forefront of network management and implementation. It is necessary for the survival of many businesses to allow open access to network resources and ensure that the data and resources are as secure as possible.

Security is becoming more important because of the following:

- Required for e-business—The importance of e-business and the need for private data to traverse public networks has increased the need for network security.
- Required for communicating and doing business safely in potentially unsafe environments—Today's business environment requires communication with many public networks and systems, which produces the need for as much security as is possible.
- Networks require development and implementation of a corporate-wide security policy—Establishing a security policy should be the first step in migrating a network to a secure infrastructure.



The E-Business Challenge

Cisco.com

Internet business value

E-commerce

Supply chain

Customer care

Workforce optimization

E-learning

Internet access

Corporate intranet

Internet presence

Business security requirements

- Defense-in-depth
- Multiple components
- Integration into e-business infrastructure
- Comprehensive blueprint

Expanded access, heightened security risks

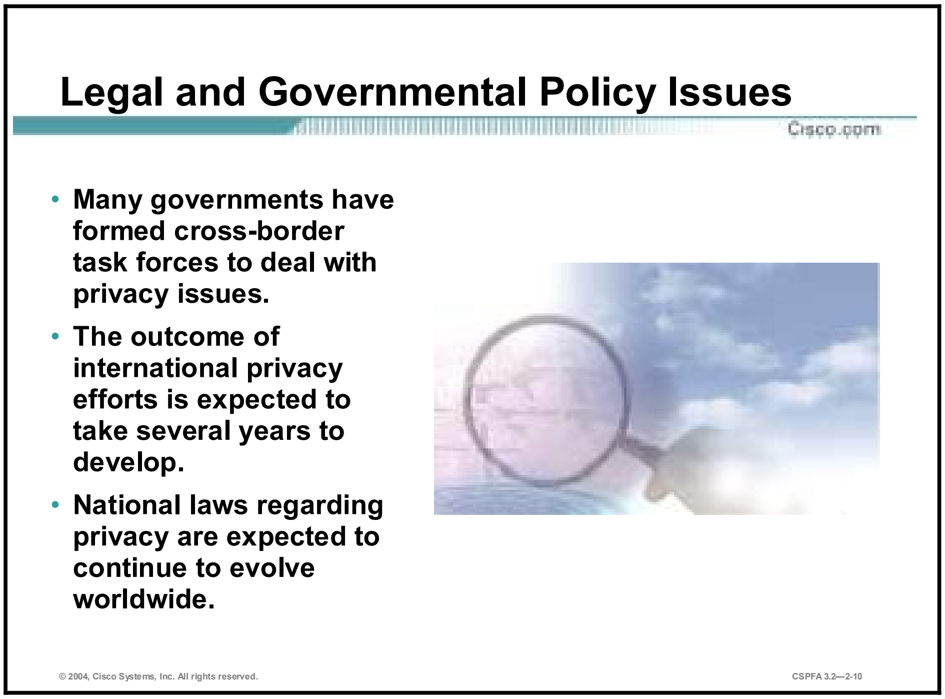
© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—2-9

Security must be a fundamental component of any e-business strategy. As enterprise network managers open their networks to more users and applications, they also expose these networks to greater risk. The result has been an increase in business security requirements.

The Internet has radically shifted expectations of companies' abilities to build stronger relationships with customers, suppliers, partners, and employees. Driving companies to become more agile and competitive, e-business is giving birth to exciting new applications for e-commerce, supply-chain management, customer care, workforce optimization, and e-learning—applications that streamline and improve processes, speed up turnaround times, lower costs, and increase user satisfaction.

E-business requires mission-critical networks that accommodate ever-increasing constituencies and demands for greater capacity and performance. These networks also need to handle voice, video, and data traffic as networks converge into multiservice environments.

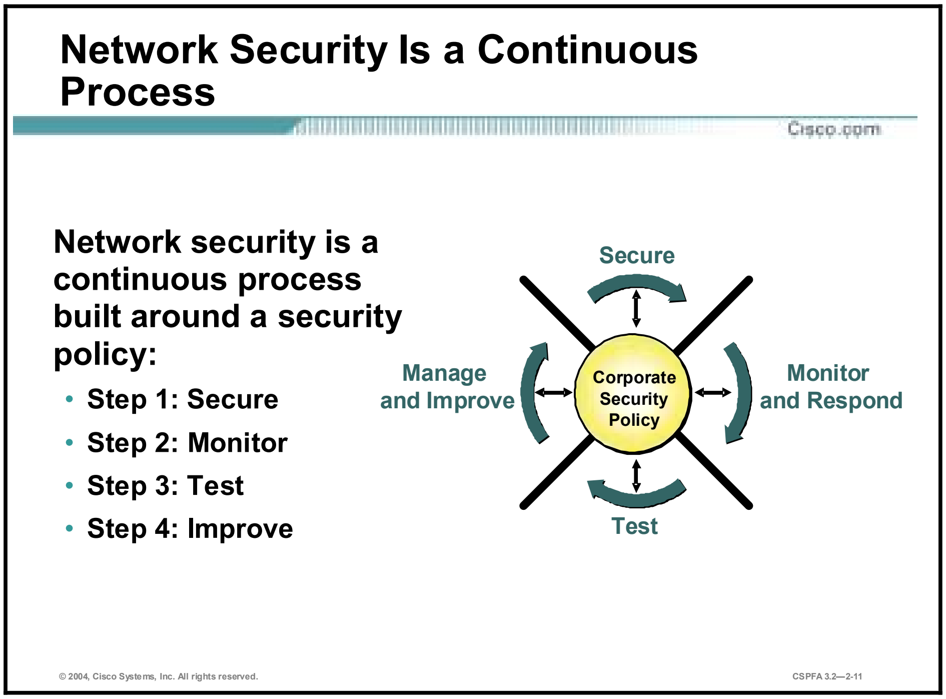


Legal and Governmental Policy Issues

- Many governments have formed cross-border task forces to deal with privacy issues.
- The outcome of international privacy efforts is expected to take several years to develop.
- National laws regarding privacy are expected to continue to evolve worldwide.

CSPFA 3.2—2-10

As concerns about privacy increase, many governments have formed cross-border task forces to deal with privacy issues. International privacy efforts are expected to take several years to develop and even longer to implement globally. National laws regarding privacy are expected to continue to evolve worldwide.



Network Security Is a Continuous Process

Cisco.com

Network security is a continuous process built around a security policy:

- **Step 1: Secure**
- **Step 2: Monitor**
- **Step 3: Test**
- **Step 4: Improve**

Secure
Manage and Improve
Corporate Security Policy
Monitor and Respond
Test

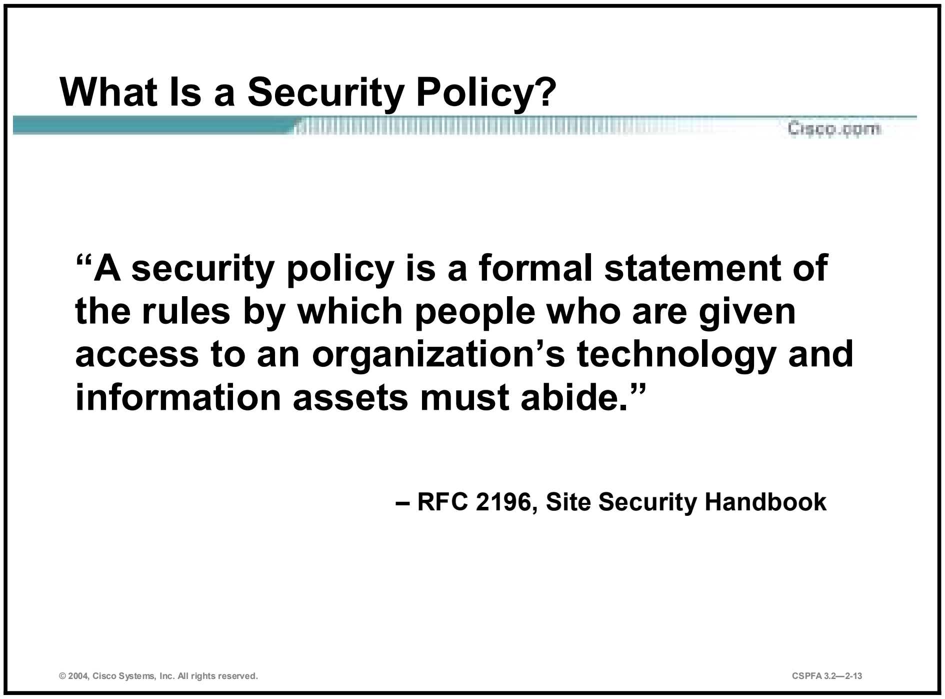
CSPFA 3.2—2-11

After setting appropriate policies, a company or organization must methodically consider security as part of normal network operations. This process could be as simple as configuring routers to not accept unauthorized addresses or services, or as complex as installing firewalls, intrusion detection systems (IDSs), centralized authentication servers, and encrypted virtual private networks (VPNs). Network security is a continuing process:

- Secure—The following are methods used to secure a network:
  - Authentication
  - Encryption
  - Firewalls
  - Vulnerability patching
- Monitor—To ensure that a network remains secure, it is important to monitor the state of security preparation. Network vulnerability scanners can proactively identify areas of weakness, and IDSs can monitor and respond to security events as they occur. Using security monitoring solutions, organizations can obtain unprecedented visibility into both the network data stream and the security posture of the network.
- Test—Testing security is as important as monitoring. Without testing the security solutions in place, it is impossible to know about existing or new attacks. The hacker community is an ever-changing environment. You can perform this testing yourself or outsource it to a third party such as the Cisco Security Posture Assessment (SPA) group.
- Improve—Monitoring and testing provides the data necessary to improve network security. Administrators and engineers should use the information from the monitor and test phases to make improvements to the security implementation as well as to adjust the security policy as vulnerabilities and risks are identified.

Network Security Policy

A security policy can be as simple as an acceptable use policy for network resources or it can be several hundred pages in length and detail every element of connectivity and associated policies.



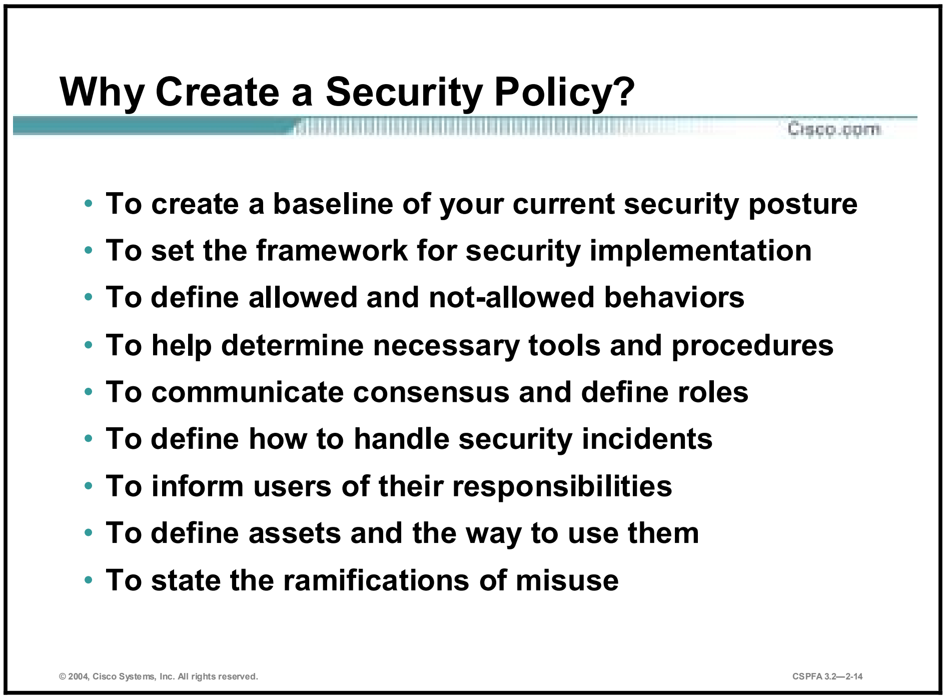
What Is a Security Policy?

"A security policy is a formal statement of the rules by which people who are given access to an organization's technology and information assets must abide."

– RFC 2196, Site Security Handbook

CSPFA 3.2—2-13

According to the Site Security Handbook (RFC 2196), "A security policy is a formal statement of the rules by which people who are given access to an organization's technology and information assets must abide." It further states, "A security policy is essentially a document summarizing how the corporation will use and protect its computing and network resources."



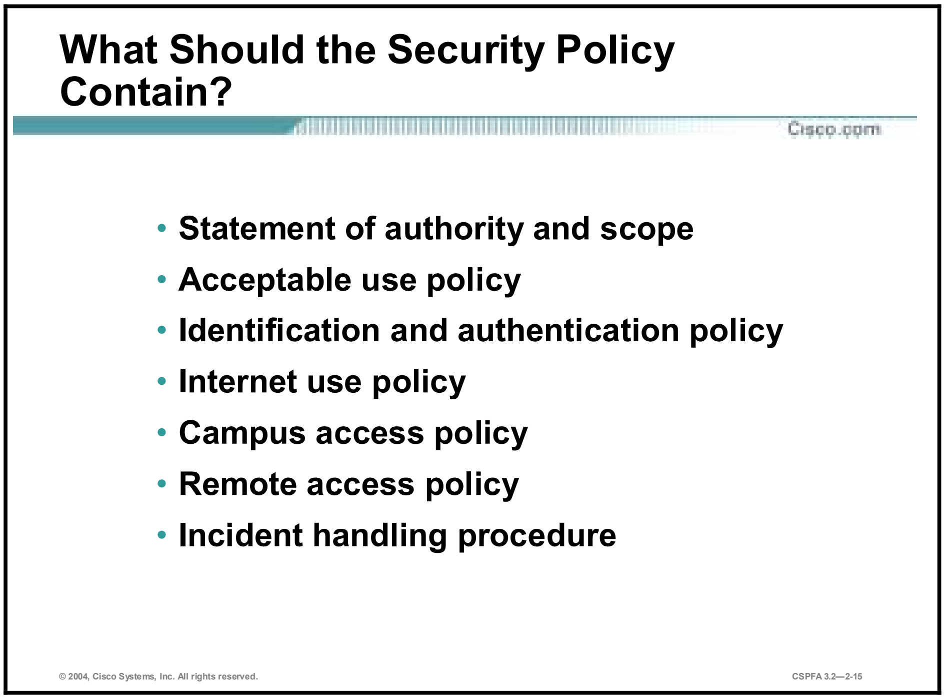
Why Create a Security Policy?

- To create a baseline of your current security posture
- To set the framework for security implementation
- To define allowed and not-allowed behaviors
- To help determine necessary tools and procedures
- To communicate consensus and define roles
- To define how to handle security incidents
- To inform users of their responsibilities
- To define assets and the way to use them
- To state the ramifications of misuse

CSPFA 3.2—2-14

Security policies provide many benefits and are worth the time and effort needed to develop them. Developing a security policy:

- Provides a process for auditing existing network security.
- Provides a general security framework for implementing network security.
- Defines which behavior is and is not allowed.
- Helps determine which tools and procedures are needed for the organization.
- Helps communicate consensus among a group of key decision makers and define responsibilities of users and administrators.
- Defines a process for handling network security incidents.
- Enables global security implementation and enforcement. Computer security is now an enterprise-wide issue, and computing sites are expected to conform to the network security policy.
- Creates a basis for legal action if necessary.



What Should the Security Policy Contain?

Cisco.com

- Statement of authority and scope
- Acceptable use policy
- Identification and authentication policy
- Internet use policy
- Campus access policy
- Remote access policy
- Incident handling procedure

© 2004, Cisco Systems, Inc. All rights reserved.

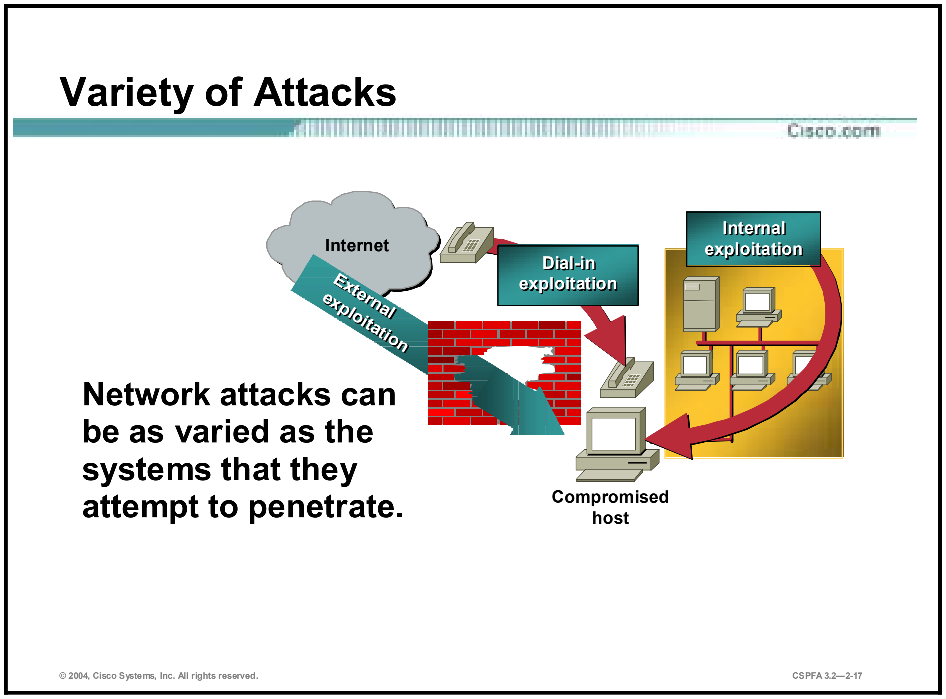
CSPFA 3.2—2-15

The following are some of the key policy components:

- Statement of authority and scope—This topic specifies who sponsors the security policy and what areas the policy covers.
- Acceptable use policy—This topic specifies what the company will and will not allow regarding its information infrastructure.
- Identification and authentication policy—This topic specifies what technologies, equipment, or combination of the two the company will use to ensure that only authorized individuals have access to its data.
- Internet access policy—This topic specifies what the company considers ethical and proper use of its Internet access capabilities.
- Campus access policy—This topic specifies how on-campus users will use the company's data infrastructure.
- Remote access policy—This topic specifies how remote users will access the company's data infrastructure.
- Incident handling procedure—This topic specifies how the company will create an incident response team and the procedures it will use during and after an incident.

Primary Network Threats and Attacks

This topic provides an overview of primary network threats and attacks.



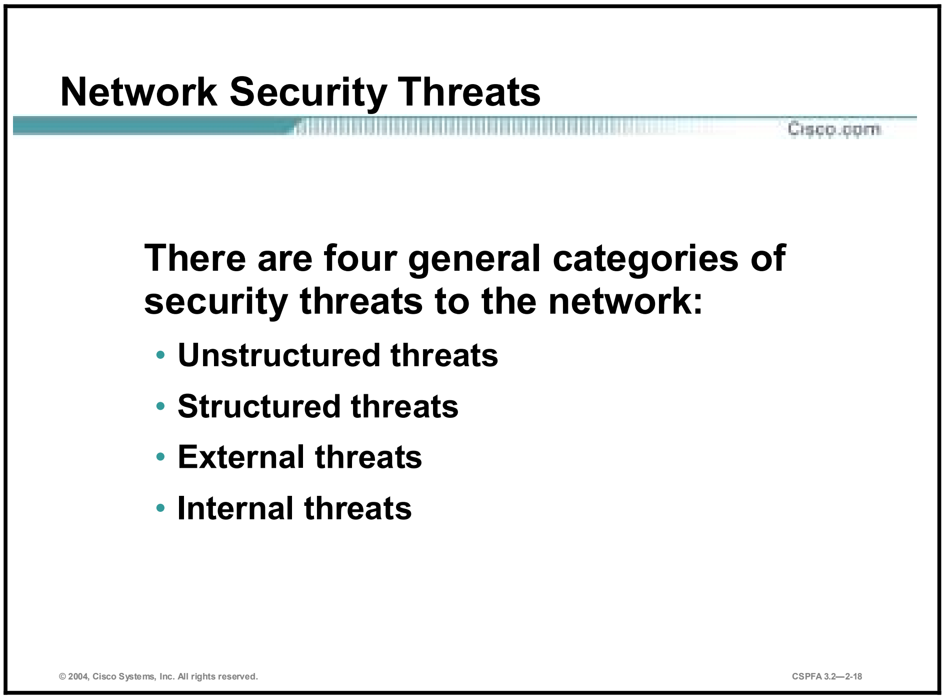
Variety of Attacks

Cisco.com

Network attacks can be as varied as the systems that they attempt to penetrate.

CSPFA 3.2—2-17

Without proper protection, any part of any network can be susceptible to attacks or unauthorized activity. Routers, switches, and hosts can all be violated by professional hackers, company competitors, or even internal employees. In fact, according to several studies, more than half of all network attacks are waged internally. The Computer Security Institute (CSI) in San Francisco, California, estimates that between 60 and 80 percent of network misuse comes from inside the enterprises where the misuse has taken place. To determine the best ways to protect against attacks, IT managers should understand the many types of attacks that can be instigated and the damage that these attacks can cause to e-business infrastructures.



Network Security Threats

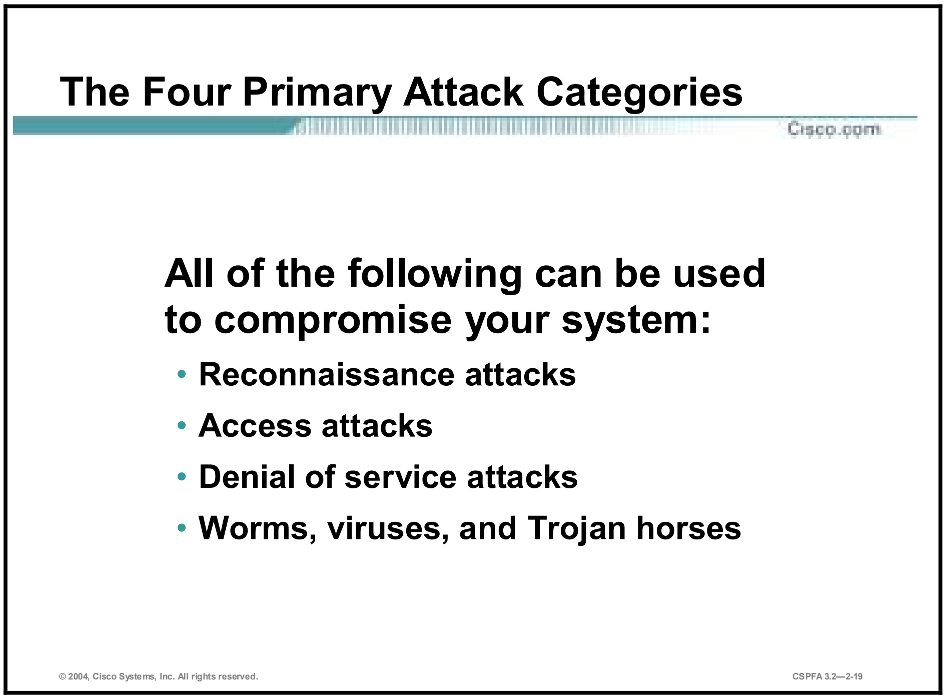
There are four general categories of security threats to the network:

- **Unstructured threats**
- **Structured threats**
- **External threats**
- **Internal threats**

CSPFA 3.2—2-18

There are four general threats to network security:

- Unstructured threats—These threats primarily consist of random hackers using various common tools, such as malicious shell scripts, password crackers, credit card number generators, and dialer daemons. Although hackers in this category may have malicious intent, many are more interested in the intellectual challenge of cracking safeguards than in creating havoc.
- Structured threats—These threats are created by hackers who are more highly motivated and technically competent. Typically, such hackers act alone or in small groups to understand, develop, and use sophisticated hacking techniques to penetrate unsuspecting businesses. These groups are often involved in the major fraud and theft cases reported to law enforcement agencies. Occasionally, such hackers are hired by organized crime, industry competitors, or state-sponsored intelligence collection organizations.
- External threats—These threats consist of structured and unstructured threats originating from an external source. These threats may have malicious and destructive intent, or they may simply be errors that generate a threat.
- Internal threats—These threats typically involve disgruntled former or current employees. Although internal threats may seem more ominous than threats from external sources, security measures are available for reducing vulnerabilities to internal threats and responding when attacks occur.



The Four Primary Attack Categories

Cisco.com

All of the following can be used to compromise your system:

- **Reconnaissance attacks**
- **Access attacks**
- **Denial of service attacks**
- **Worms, viruses, and Trojan horses**

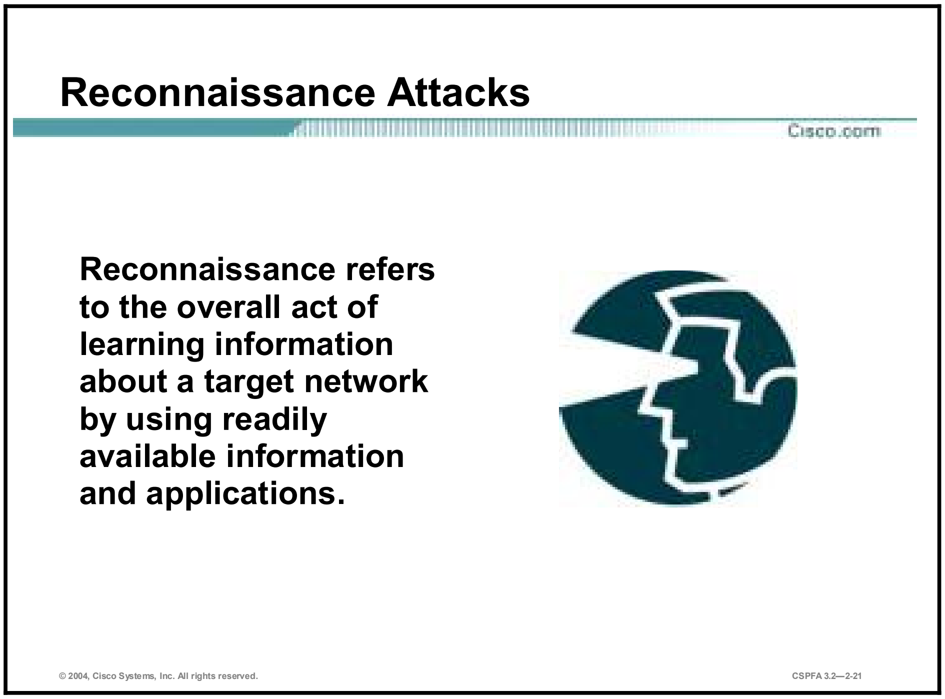
CSPFA 3.2—2-19

There are four types of network attacks:

- Reconnaissance attacks—An intruder attempts to discover and map systems, services, and vulnerabilities.
- Access attacks—An intruder attacks networks or systems to retrieve data, gain access, or escalate access privileges.
- Denial of service (DoS) attacks—An intruder attacks your network in a way that damages or corrupts your computer system or denies you and others access to your networks, systems, or services.
- Worms, viruses, and Trojan horses—Malicious software is inserted onto a host in order to damage a system, corrupt a system, replicate itself, or deny services or access to networks, systems, or services.

Reconnaissance Attacks and Mitigation

This topic describes reconnaissance attacks and their mitigation.



Reconnaissance Attacks

Cisco.com

Reconnaissance refers to the overall act of learning information about a target network by using readily available information and applications.

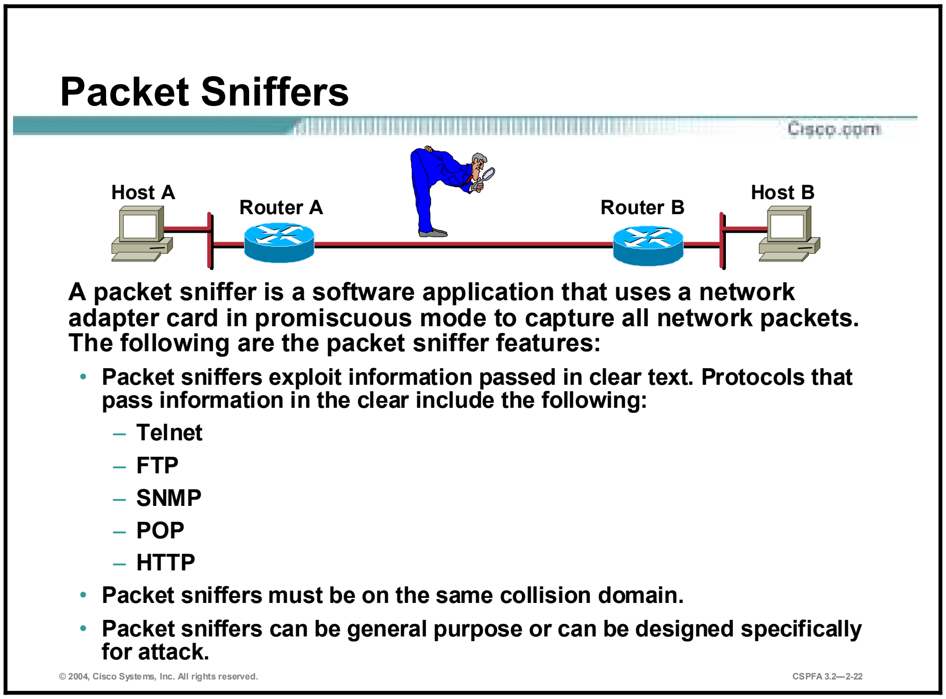
CSPFA 3.2—2-21

Reconnaissance is the unauthorized discovery and mapping of systems, services, or vulnerabilities. It is also known as information gathering and, in most cases, precedes an actual access or DoS attack. The malicious intruder typically conducts a ping sweep of the target network first to determine which IP addresses are alive. After this has been accomplished, the intruder determines which services or ports are active on the live IP addresses. From this information, the intruder queries the ports to determine the application type and version as well as the type and version of the operating system running on the target host.

Reconnaissance is somewhat analogous to a thief casing a neighborhood for vulnerable homes to break into, such as an unoccupied residence, a house with an easy-to-open door or window, and so on. In many cases the intruders go as far as “rattling the door handle,” not to go in immediately if it is opened, but to discover vulnerable services that they can exploit later when there is less likelihood that anyone is looking.

Reconnaissance attacks can consist of the following:

- Packet sniffers
- Port scans
- Ping sweeps
- Internet information queries



Packet Sniffers

A packet sniffer is a software application that uses a network adapter card in promiscuous mode to capture all network packets. The following are the packet sniffer features:

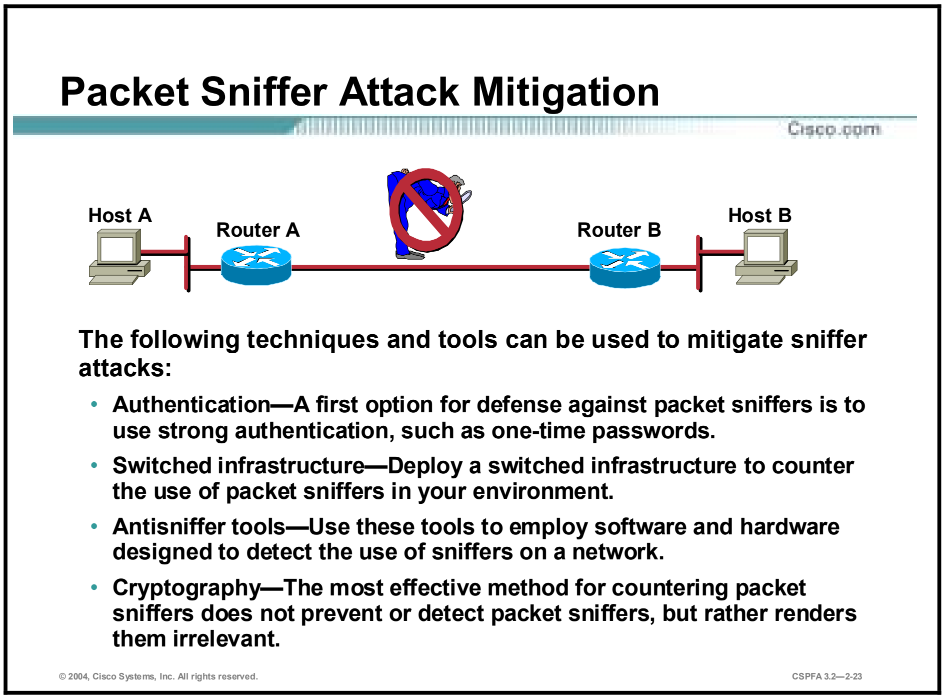
- **Packet sniffers exploit information passed in clear text. Protocols that pass information in the clear include the following:**
  - **Telnet**
  - **FTP**
  - **SNMP**
  - **POP**
  - **HTTP**
- **Packet sniffers must be on the same collision domain.**
- **Packet sniffers can be general purpose or can be designed specifically for attack.**

CSPFA 3.2—2-22

A packet sniffer is a software application that uses a network adapter card in promiscuous mode (a mode in which the network adapter card sends all packets received on the physical network wire to an application for processing) to capture all network packets that are sent across a LAN.

Several network applications distribute network packets in clear text; that is, the information sent across the network is not encrypted. Because the network packets are not encrypted, they can be processed and understood by any application that can pick them up off the network and process them.

A network protocol specifies how packets are identified and labeled, which enables a computer to determine whether a packet is intended for it. Because the specifications for network protocols, such as TCP/IP, are widely published, a third party can easily interpret the network packets and develop a packet sniffer. (The real threat today results from the numerous freeware and shareware packet sniffers that are available, which do not require the user to understand anything about the underlying protocols.)


Cisco.com
Host A
Router A
Router B
Host B

Packet Sniffer Attack Mitigation

The following techniques and tools can be used to mitigate sniffer attacks:

- **Authentication—A first option for defense against packet sniffers is to use strong authentication, such as one-time passwords.**
- **Switched infrastructure—Deploy a switched infrastructure to counter the use of packet sniffers in your environment.**
- **Antisniffer tools—Use these tools to employ software and hardware designed to detect the use of sniffers on a network.**
- **Cryptography—The most effective method for countering packet sniffers does not prevent or detect packet sniffers, but rather renders them irrelevant.**

CSPFA 3.2—2-23

The following techniques and tools can be used to mitigate packet sniffer attacks:

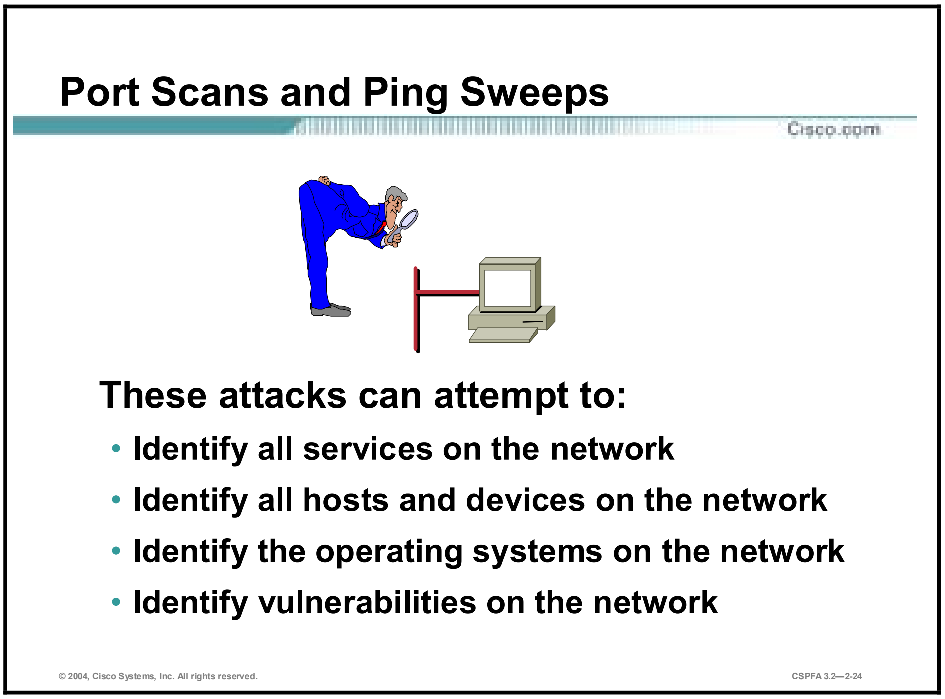
- Authentication—Using strong authentication is a first option for defense against packet sniffers. Strong authentication can be broadly defined as a method of authenticating users that cannot easily be circumvented. A common example of strong authentication is one-time passwords (OTPs).

  An OTP is a type of two-factor authentication. Two-factor authentication involves using something you have combined with something you know. Automated teller machines (ATMs) use two-factor authentication. A customer needs both an ATM card and a personal identification number (PIN) to make transactions. With OTPs you need a PIN and your token card to authenticate to a device or software application. A token card is a hardware or software device that generates new, seemingly random, passwords at specified intervals (usually 60 seconds). A user combines that password with a PIN to create a unique password that works only for one instance of authentication. If a hacker learns that password by using a packet sniffer, the information is useless because the password has already expired. Note that this mitigation technique is effective only against a sniffer implementation that is designed to grab passwords. Sniffers deployed to learn sensitive information (such as e-mail messages) will still be effective.

- Switched infrastructure—This technique can be used to counter the use of packet sniffers in your network environment. For example, if an entire organization deploys switched Ethernet, hackers can gain access only to the traffic that flows on the specific port to which they connect. A switched infrastructure obviously does not eliminate the threat of packet sniffers, but it can greatly reduce their effectiveness.

- Antisniffer tools—Software and hardware designed to detect the use of sniffers on a network can be employed. Such software and hardware does not completely eliminate the threat, but like many network security tools, they are part of the overall system. These so-called antisniffers detect changes in the response time of hosts to determine whether the hosts are processing more traffic than their own. One such network security software tool, which is available from Security Software Technologies, is called AntiSniff.

- Cryptography—Rendering packet sniffers irrelevant is the most effective method for countering packet sniffers, even more effective than preventing or detecting packet sniffers. If a communication channel is cryptographically secure, the only data a packet sniffer will detect is cipher text (a seemingly random string of bits) and not the original message. The Cisco deployment of network-level cryptography is based on IPSec, which is a standard method for networking devices to communicate privately using IP. Other cryptographic protocols for network management include Secure Shell Protocol (SSH) and Secure Sockets Layer (SSL).



Port Scans and Ping Sweeps

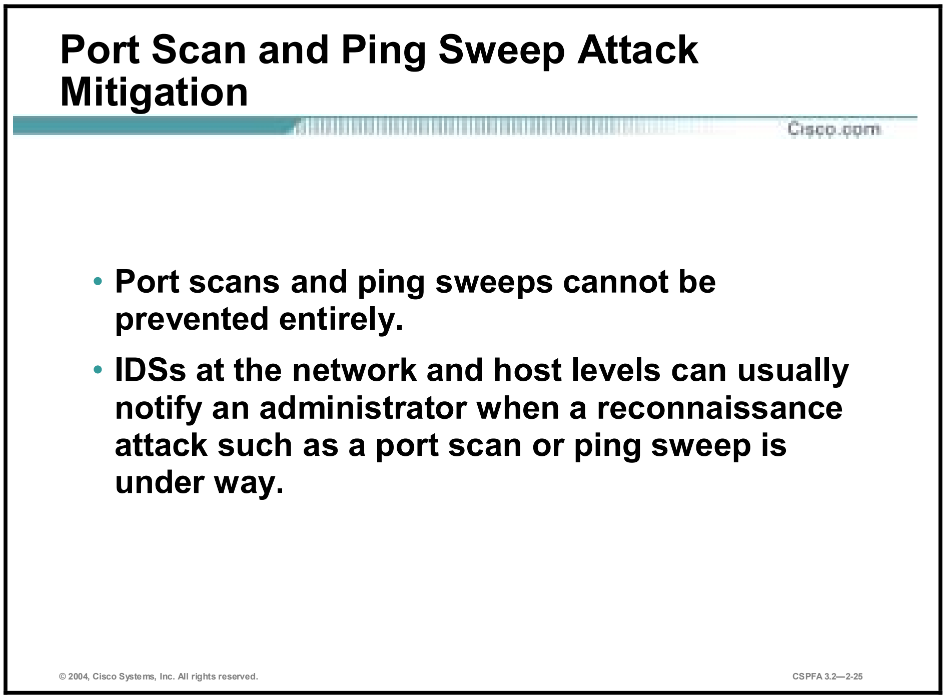
Cisco.com

These attacks can attempt to:

- Identify all services on the network
- Identify all hosts and devices on the network
- Identify the operating systems on the network
- Identify vulnerabilities on the network

CSPFA 3.2—2-24

Port scans and ping sweeps are typically applications built to run various tests against a host or device in order to identify vulnerable services. The information is gathered by examining IP addressing and port or banner data from both TCP and UDP ports.



Port Scan and Ping Sweep Attack Mitigation

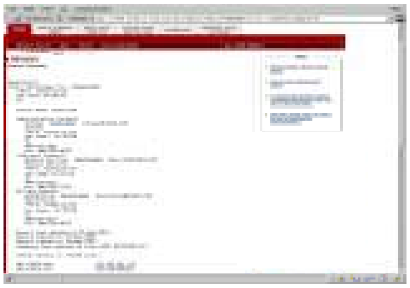
- **Port scans and ping sweeps cannot be prevented entirely.**
- **IDSs at the network and host levels can usually notify an administrator when a reconnaissance attack such as a port scan or ping sweep is under way.**

CSPFA 3.2—2-25

If ICMP echo and echo reply are turned off on edge routers, for example, ping sweeps can be stopped, but at the expense of network diagnostic data. However, port scans can easily be run without full ping sweeps; they simply take longer because they need to scan IP addresses that might not be live. IDSs at the network and host levels can usually notify an administrator when a reconnaissance attack is under way. This warning allows the administrator to better prepare for the coming attack or to notify the Internet service provider (ISP) that is hosting the system launching the reconnaissance probe.

Internet Information Queries

Sample IP address query



Sample domain name query

CSPFA 3.2—2-26

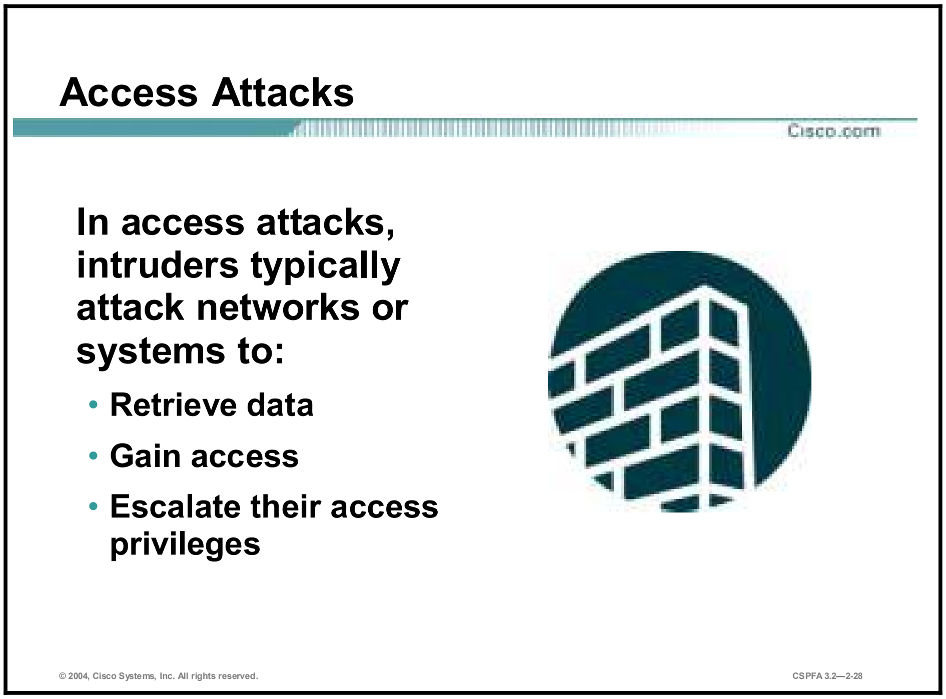
The figure demonstrates how existing Internet tools can be used for network reconnaissance (for example, an IP address query or a Domain Name System [DNS] query).

DNS queries can reveal such information as who owns a particular domain and what addresses have been assigned to that domain. Ping sweeps of the addresses revealed by the DNS queries can present a picture of the live hosts in a particular environment. After such a list is generated, port scanning tools can cycle through all well-known ports to provide a complete list of all services running on the hosts discovered by the ping sweep. Finally, the hackers can examine the characteristics of the applications that are running on the hosts. This step can lead to specific information that is useful when the hacker attempts to compromise that service.

IP address queries can reveal information such as who owns a particular IP address or range of addresses and what domain is associated with them.

Access Attacks and Mitigation

This topic describes specific access attacks and their mitigation.



Access Attacks

Cisco.com

In access attacks, intruders typically attack networks or systems to:

- **Retrieve data**
- **Gain access**
- **Escalate their access privileges**

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—2-28

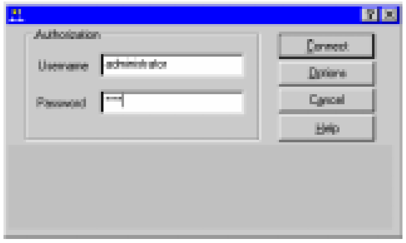
Access attacks exploit known vulnerabilities in authentication services, FTP services, and Web services to gain entry to Web accounts, confidential databases, and other sensitive information. Access attacks can consist of the following:

- Password attacks
- Trust exploitation
- Port redirection
- Man-in-the-middle attacks

Password Attacks

Hackers can implement password attacks using several methods:

- **Brute-force attacks**
- **Trojan horse programs**
- **IP spoofing**
- **Packet sniffers**



Authorization

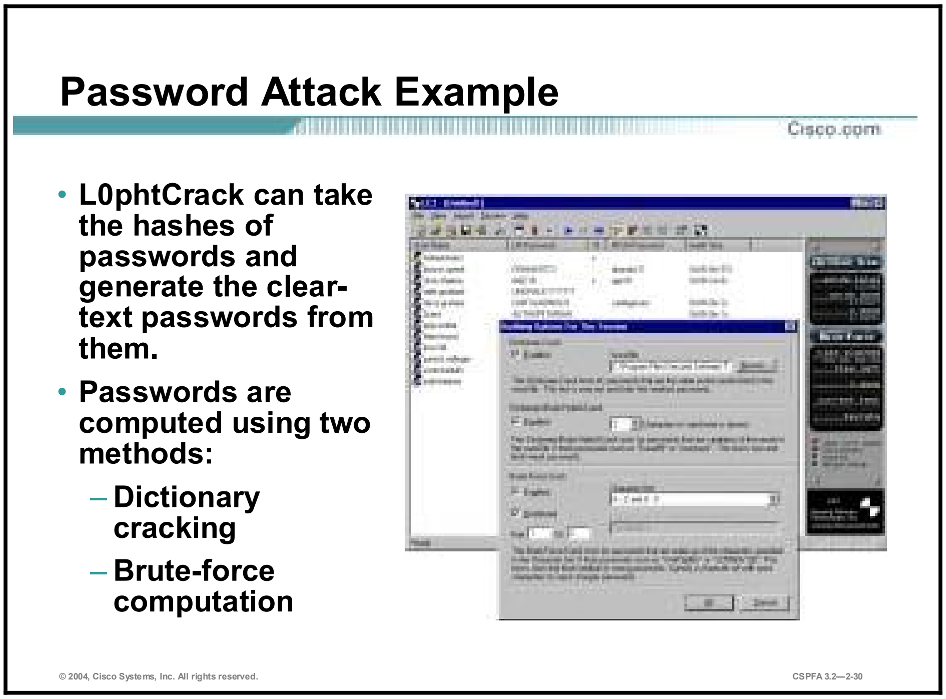
Username: administrator

Password: ****

CSPFA 3.2—2-29

Password attacks can be implemented using several methods, including brute-force attacks, Trojan horse programs, IP spoofing, and packet sniffers. Although packet sniffers and IP spoofing can yield user accounts and passwords, password attacks usually refer to repeated attempts to identify a user account, password, or both. These repeated attempts are called brute-force attacks.

Often a brute-force attack is performed using a program that runs across the network and attempts to log in to a shared resource, such as a server. When an attacker gains access to a resource, he or she has the same access rights as the user whose account has been compromised. If this account has sufficient privileges, the attacker can create a back door for future access, without concern for any status and password changes to the compromised user account.



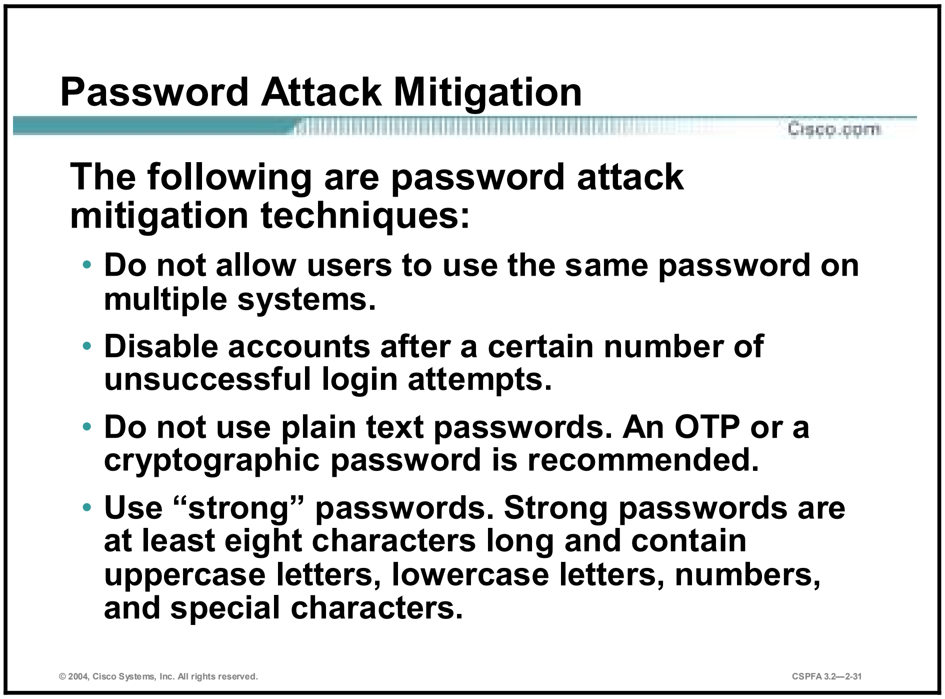
Password Attack Example

- L0phtCrack can take the hashes of passwords and generate the clear-text passwords from them.
- Passwords are computed using two methods:
  - Dictionary cracking
  - Brute-force computation

Just as with packet sniffer and IP spoofing attacks, a brute-force password attack can provide access to accounts that can be used to modify critical network files and services. An example that compromises your network's integrity is an attacker modifying the routing tables for your network. By doing so, the attacker ensures that all network packets are routed to him or her before they are transmitted to their final destination. In such a case, an attacker can monitor all network traffic, effectively becoming a man in the middle.

The following are the two methods for computing passwords with L0phtCrack:

- Dictionary cracking—The password hashes for all of the words in a dictionary file are computed and compared against all of the password hashes for the users. This method is extremely fast and finds very simple passwords.
- Brute-force computation—This method uses a particular character set, such as A–Z or A–Z plus 0–9, and computes the hash for every possible password made up of those characters. It will always compute the password if that password is made up of the character set you have selected to test. The downside is that time is required for completion of this type of attack.



Password Attack Mitigation

Cisco.com

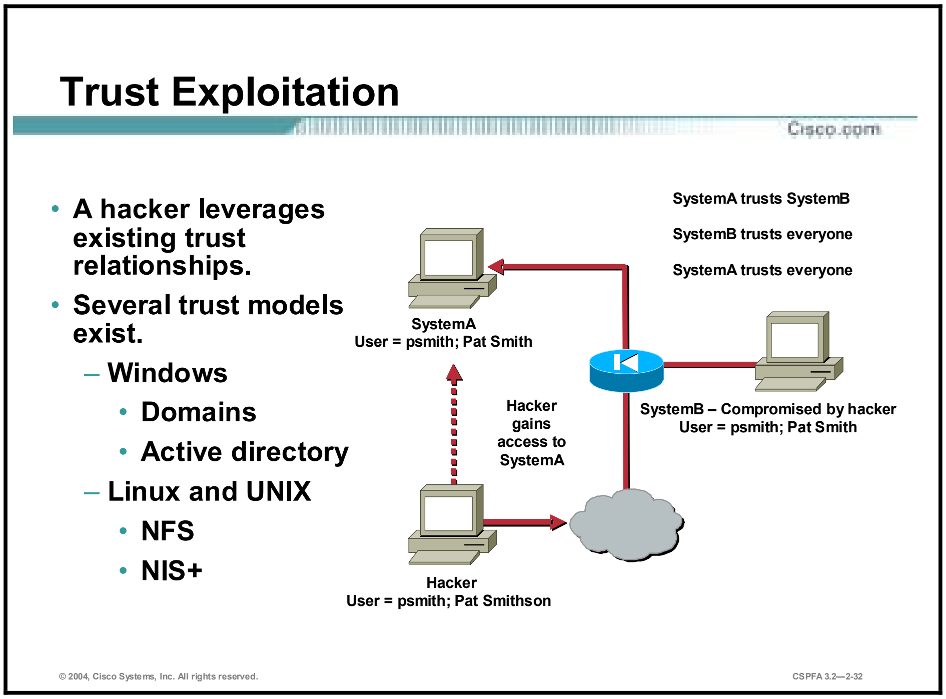
The following are password attack mitigation techniques:

- Do not allow users to use the same password on multiple systems.
- Disable accounts after a certain number of unsuccessful login attempts.
- Do not use plain text passwords. An OTP or a cryptographic password is recommended.
- Use “strong” passwords. Strong passwords are at least eight characters long and contain uppercase letters, lowercase letters, numbers, and special characters.

CSPFA 3.2—2-31

The following are password attack mitigation techniques:

- Do not allow users to have the same password on multiple systems—Most users will use the same password for each system they access, and often personal system passwords will be the same as well.
- Disable accounts after a specific number of unsuccessful logins—This practice helps to prevent continuous password attempts.
- Do not use plain-text passwords—Use of either an OTP or encrypted password is recommended.
- Use “strong” passwords—Many systems now provide strong password support and can restrict a user to the use of strong passwords only. Strong passwords are at least eight characters long and contain uppercase letters, lowercase letters, numbers, and special characters.



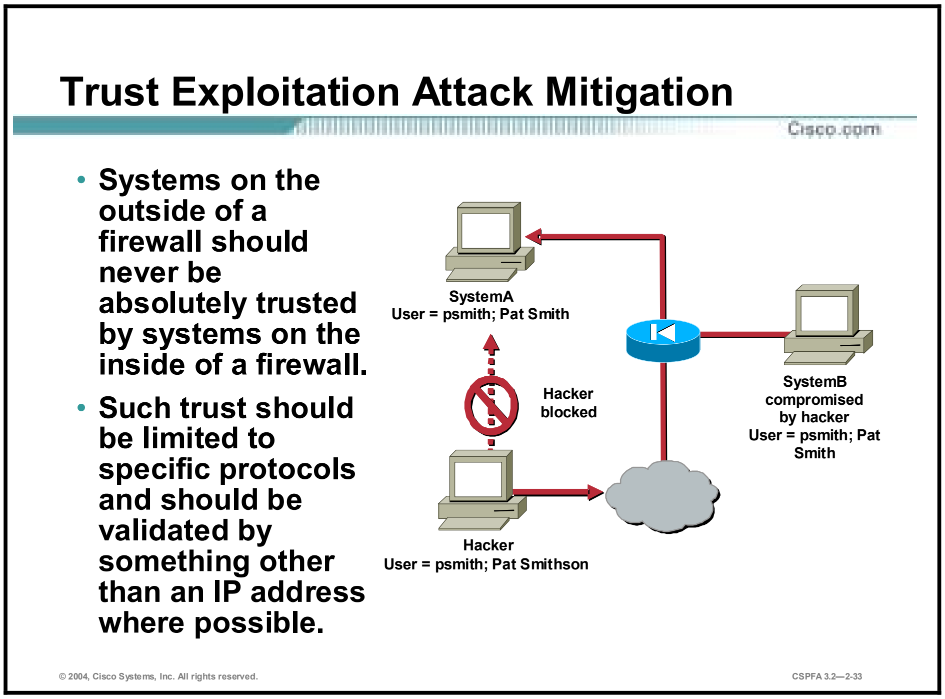
Trust Exploitation

Cisco.com

- A hacker leverages existing trust relationships.
- Several trust models exist.
  - Windows
    - Domains
    - Active directory
  - Linux and UNIX
    - NFS
    - NIS+

CSPFA 3.2—2-32

Although it is not an attack in itself, trust exploitation refers to an individual's taking advantage of a trust relationship within a network. The classic example is a perimeter network connection from a corporation. These network segments often house DNS, Simple Mail Transfer Protocol (SMTP), and HTTP servers. Because they all reside on the same segment, a compromise of one system can lead to the compromise of other systems if those other systems in turn trust systems attached to the same network. Another example is a system on the outside of a firewall that has a trust relationship with a system on the inside of a firewall. When the outside system is compromised, the attacker can leverage that trust relationship to attack the inside network.



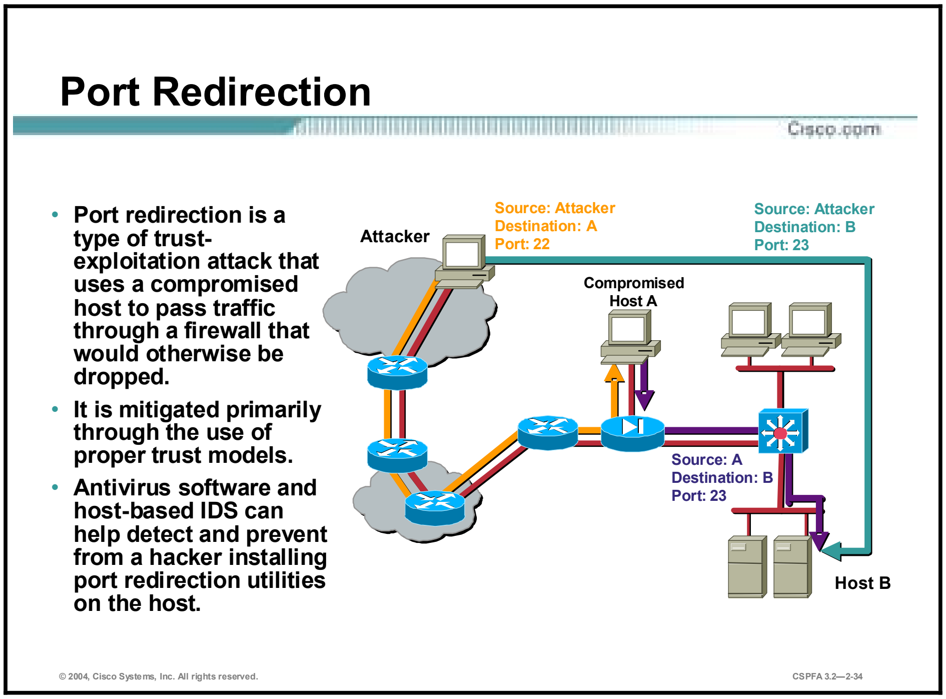
Trust Exploitation Attack Mitigation

- Systems on the outside of a firewall should never be absolutely trusted by systems on the inside of a firewall.
- Such trust should be limited to specific protocols and should be validated by something other than an IP address where possible.

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—2-33

You can mitigate trust exploitation-based attacks through tight constraints on trust levels within a network. Systems on the outside of a firewall should never be absolutely trusted by systems on the inside of a firewall. Such trust should be limited to specific protocols and should be authenticated by something other than an IP address where possible.



Port Redirection

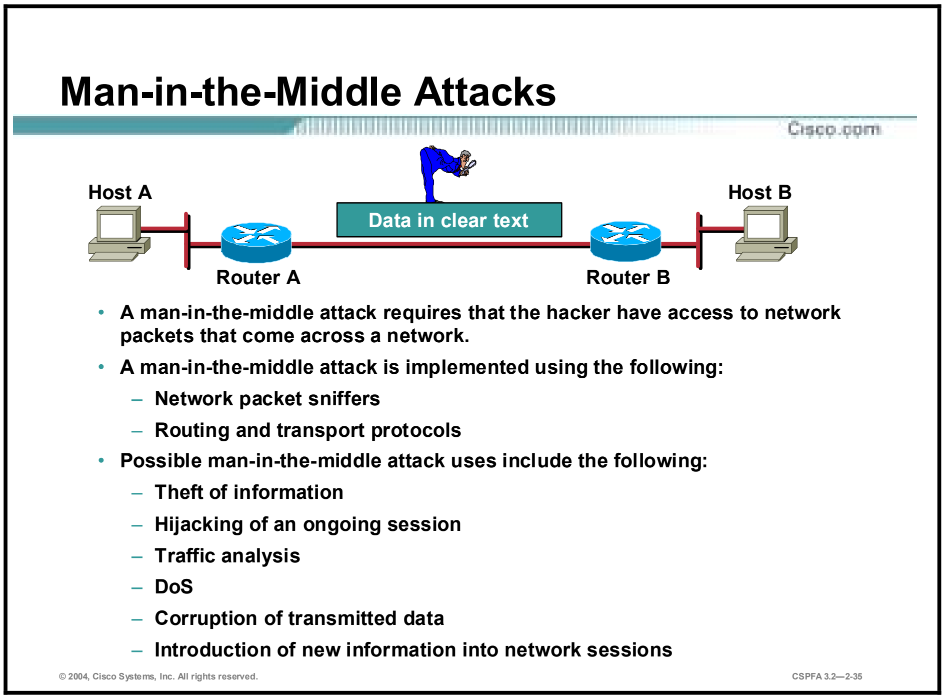
Cisco.com

- Port redirection is a type of trust-exploitation attack that uses a compromised host to pass traffic through a firewall that would otherwise be dropped.
- It is mitigated primarily through the use of proper trust models.
- Antivirus software and host-based IDS can help detect and prevent from a hacker installing port redirection utilities on the host.

CSPFA 3.2—2-34

Port redirection attacks are a type of trust exploitation attack that uses a compromised host to pass traffic through a firewall that would otherwise be dropped. Consider a firewall with three interfaces and a host on each interface. The host on the outside can reach the host on the public services segment (commonly referred to as a Demilitarized Zone [DMZ]), but not the host on the inside. The host on the public services segment can reach the host on both the outside and the inside. If hackers were able to compromise the public services segment host, they could install software to redirect traffic from the outside host directly to the inside host. Though neither communication violates the rules implemented in the firewall, the outside host has now achieved connectivity to the inside host through the port redirection process on the public services host. An example of an application that can provide this type of access is netcat.

Port redirection can be mitigated primarily through the use of proper trust models, which are network specific (as mentioned earlier). Assuming a system under attack, a host-based IDS can help detect a hacker and prevent installation of such utilities on a host.



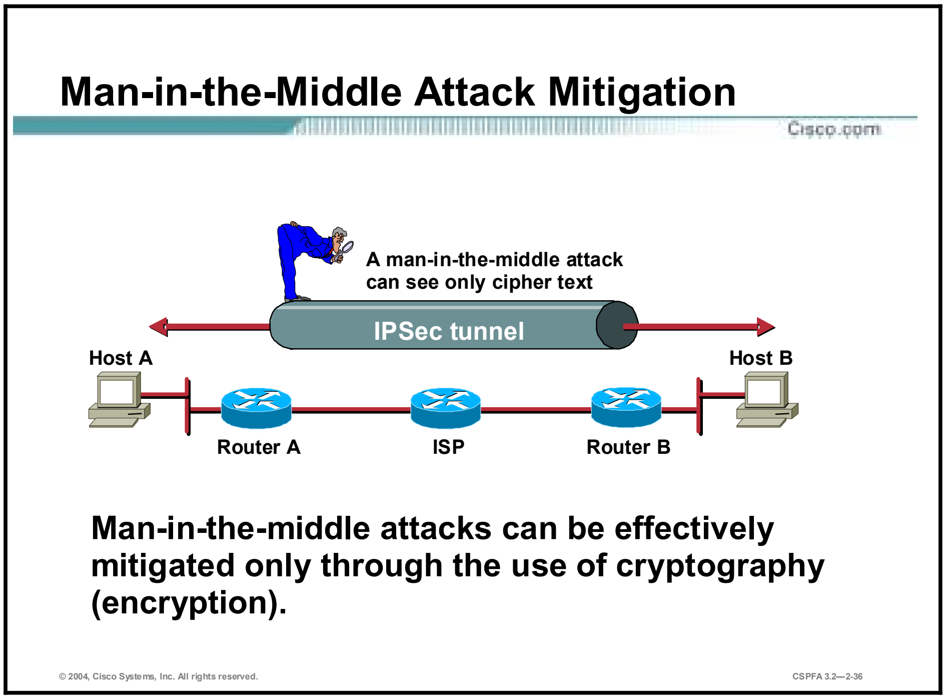
Man-in-the-Middle Attacks

- A man-in-the-middle attack requires that the hacker have access to network packets that come across a network.
- A man-in-the-middle attack is implemented using the following:
  - Network packet sniffers
  - Routing and transport protocols
- Possible man-in-the-middle attack uses include the following:
  - Theft of information
  - Hijacking of an ongoing session
  - Traffic analysis
  - DoS
  - Corruption of transmitted data
  - Introduction of new information into network sessions

CSPFA 3.2—2-35

A man-in-the-middle attack requires that the attacker have access to network packets that come across the network. Such attacks are often implemented using network packet sniffers and routing and transport protocols. The possible uses of such attacks are theft of information, hijacking of an ongoing session to gain access to your internal network resources, traffic analysis to derive information about your network and its users, denial of service, corruption of transmitted data, and introduction of new information into network sessions.

An example of a man-in-the-middle attack could be someone who is working for your ISP and who can gain access to all network packets transferred between your network and any other network.



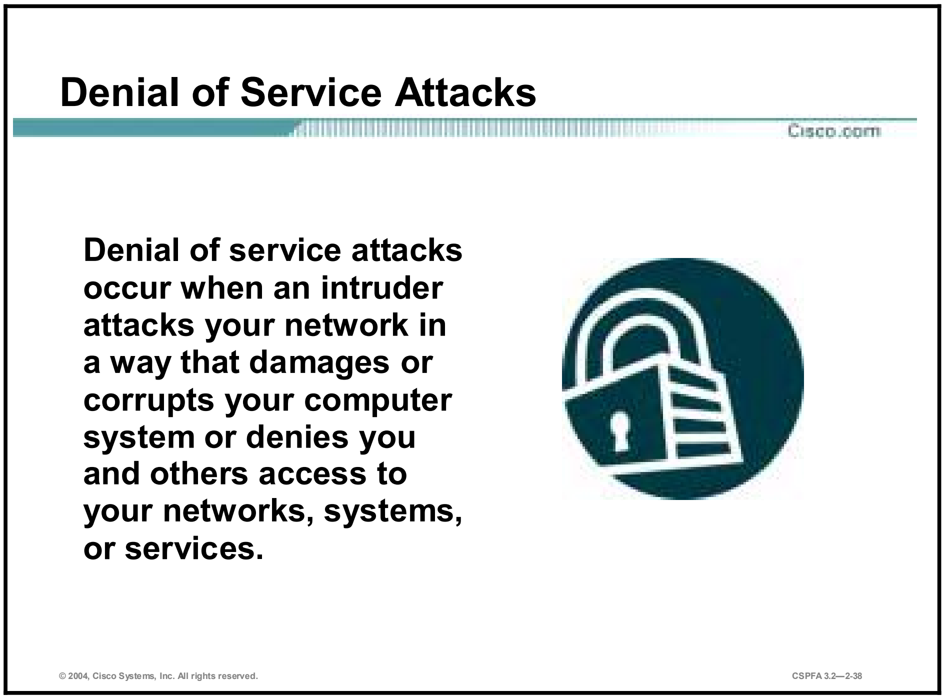
Man-in-the-Middle Attack Mitigation

Man-in-the-middle attacks can be effectively mitigated only through the use of cryptography (encryption).

Man-in-the-middle attack mitigation is achieved, as shown in the figure, by encrypting traffic in an IPSec tunnel, which would allow the hacker to see only cipher text.

Denial of Service Attacks and Mitigation

This topic describes specific DoS attacks and their mitigation.



Denial of Service Attacks

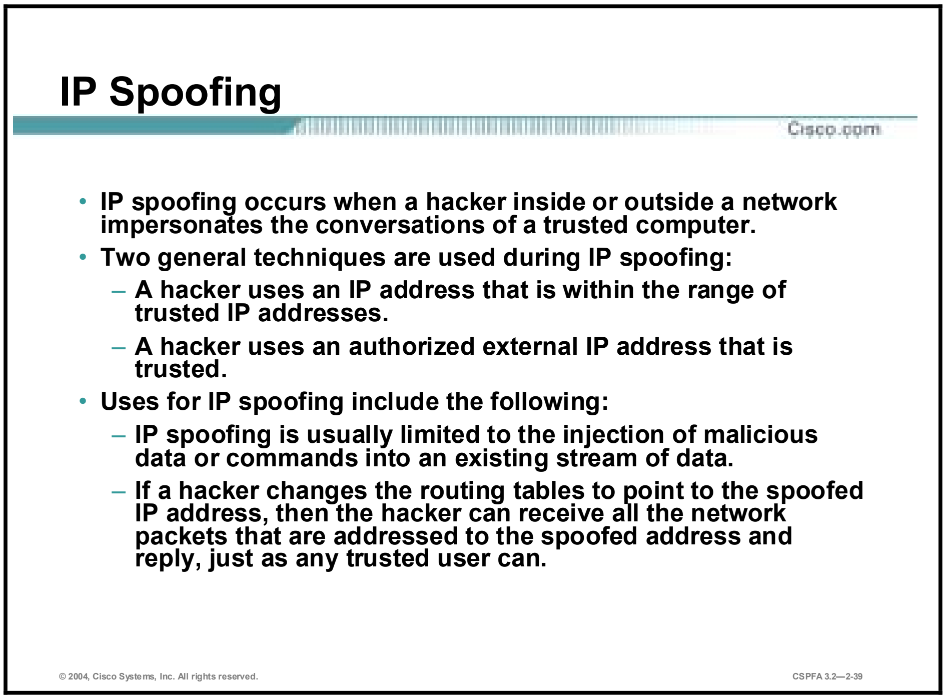
Cisco.com

Denial of service attacks occur when an intruder attacks your network in a way that damages or corrupts your computer system or denies you and others access to your networks, systems, or services.

CSPFA 3.2—2-38

Certainly the most publicized form of attack, DoS attacks are also among the most difficult to completely eliminate. Even within the hacker community, DoS attacks are regarded as trivial and considered bad form because they require so little effort to execute. Still, because of their ease of implementation and potentially significant damage, DoS attacks deserve special attention from security administrators. If you are interested in learning more about DoS attacks, researching the methods employed by some of the better-known attacks can be useful. DoS attacks can consist of the following:

- IP spoofing
- Distributed denial of service (DDoS)



IP Spoofing

Cisco.com

- IP spoofing occurs when a hacker inside or outside a network impersonates the conversations of a trusted computer.
- Two general techniques are used during IP spoofing:
  - A hacker uses an IP address that is within the range of trusted IP addresses.
  - A hacker uses an authorized external IP address that is trusted.
- Uses for IP spoofing include the following:
  - IP spoofing is usually limited to the injection of malicious data or commands into an existing stream of data.
  - If a hacker changes the routing tables to point to the spoofed IP address, then the hacker can receive all the network packets that are addressed to the spoofed address and reply, just as any trusted user can.

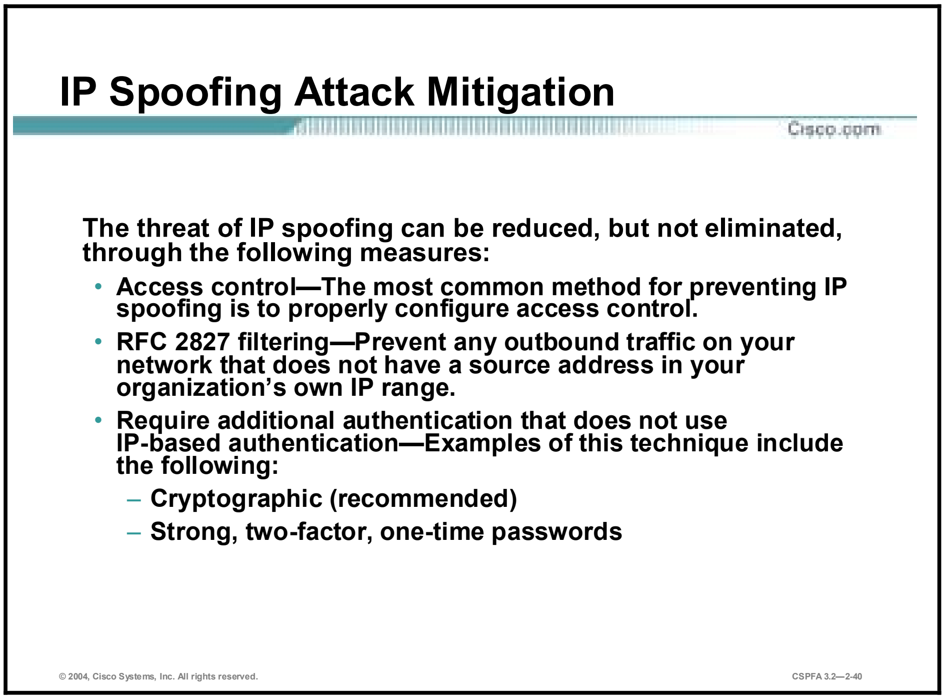
CSPFA 3.2—2-39

An IP spoofing attack occurs when an attacker outside your network pretends to be a trusted computer, either by using an IP address that is within the range of IP addresses for your network or by using an authorized external IP address that you trust and to which you wish to provide access to specified resources on your network.

Normally, an IP spoofing attack is limited to the injection of data or commands into an existing stream of data passed between a client and server application or a peer-to-peer network connection. To enable bidirectional communication, the attacker must change all routing tables to point to the spoofed IP address. Another approach the attacker could take is simply not to worry about receiving any response from the applications. For example, if an attacker is attempting to get a system to mail him or her a sensitive file, application responses are unimportant.

However, if an attacker manages to change the routing tables to point to the spoofed IP address, he or she can receive all the network packets that are addressed to the spoofed address and reply just as any trusted user can. Like packet sniffers, IP spoofing use is not restricted to people who are external to the network.

Although this use is not as common, IP spoofing can also provide access to user accounts and passwords, and it can also be used in other ways. For example, an attacker can emulate one of your internal users in ways that prove embarrassing for your organization; the attacker could send e-mail messages to business partners that appear to have originated from someone within your organization. Such attacks are easier when an attacker has a user account and password, but they are possible when simple spoofing attacks are combined with knowledge of messaging protocols.



IP Spoofing Attack Mitigation

The threat of IP spoofing can be reduced, but not eliminated, through the following measures:

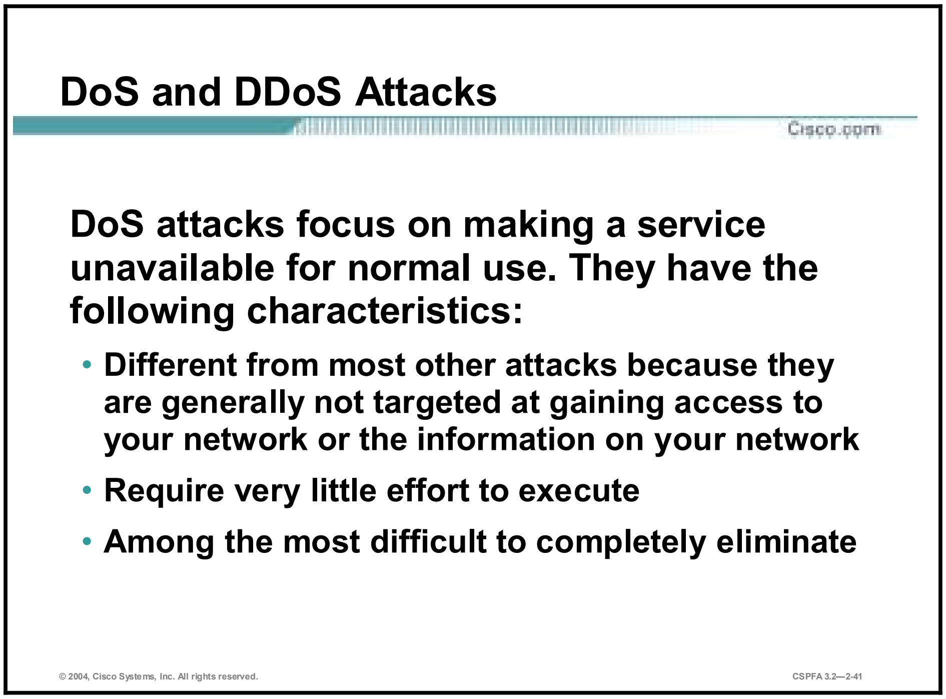
- **Access control—The most common method for preventing IP spoofing is to properly configure access control.**
- **RFC 2827 filtering—Prevent any outbound traffic on your network that does not have a source address in your organization’s own IP range.**
- **Require additional authentication that does not use IP-based authentication—Examples of this technique include the following:**
  - **Cryptographic (recommended)**
  - **Strong, two-factor, one-time passwords**

CSPFA 3.2—2-40

The threat of IP spoofing can be reduced, but not eliminated, through the following measures:

- Access control—The most common method for preventing IP spoofing is to properly configure access control. To reduce the effectiveness of IP spoofing, configure access control to deny any traffic from the external network that has a source address that should reside on the internal network. Note that this helps prevent spoofing attacks only if the internal addresses are the only trusted addresses. If some external addresses are trusted, this method is not effective.
- RFC 2827 filtering—You can prevent users of your network from spoofing other networks (and be a good Internet citizen at the same time) by preventing any outbound traffic on your network that does not have a source address in your organization's own IP range.

  This filtering denies any traffic that does not have the source address that was expected on a particular interface. For example, if an ISP is providing a connection to the IP address 15.1.1.0/24, the ISP could filter traffic so that only traffic sourced from address 15.1.1.0/24 can enter the ISP router from that interface. Note that unless all ISPs implement this type of filtering, its effectiveness is significantly reduced.
- Additional authentication—The most effective method for mitigating the threat of IP spoofing is the same as the most effective method for mitigating the threat of packet sniffers: namely, eliminating its effectiveness. IP spoofing can function correctly only when devices use IP address-based authentication; therefore, if you use additional authentication methods, IP spoofing attacks are irrelevant. Cryptographic authentication is the best form of additional authentication, but when that is not possible, strong two-factor authentication using OTPs can also be effective.



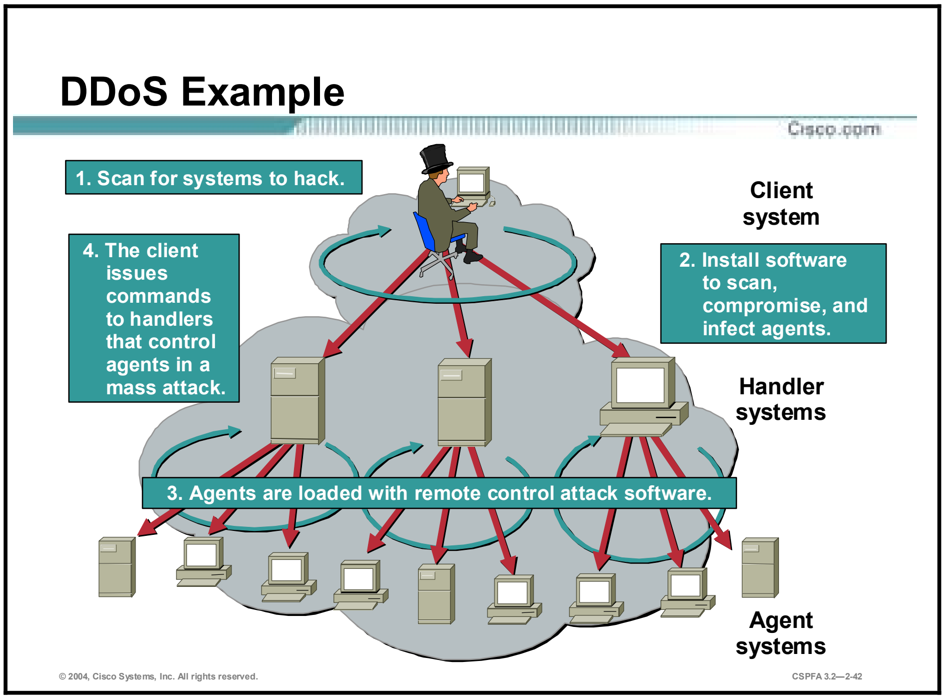
DoS and DDoS Attacks

DoS attacks focus on making a service unavailable for normal use. They have the following characteristics:

- **Different from most other attacks because they are generally not targeted at gaining access to your network or the information on your network**
- **Require very little effort to execute**
- **Among the most difficult to completely eliminate**

CSPFA 3.2—2-41

DoS attacks are different from most other attacks because they are not targeted at gaining access to your network or the information on your network. These attacks focus on making a service unavailable for normal use, which is typically accomplished by exhausting some resource limitation on the network or within an operating system or application. These attacks require little effort to execute because they typically take advantage of protocol weaknesses or because the attacks are carried out using traffic that would normally be allowed into a network. DoS attacks are among the most difficult to completely eliminate because of the way they use protocol weaknesses and “native” traffic to attack a network.



DDoS Example

1. Scan for systems to hack.

Client system

2. Install software to scan, compromise, and infect agents.

Handler systems

3. Agents are loaded with remote control attack software.

4. The client issues commands to handlers that control agents in a mass attack.

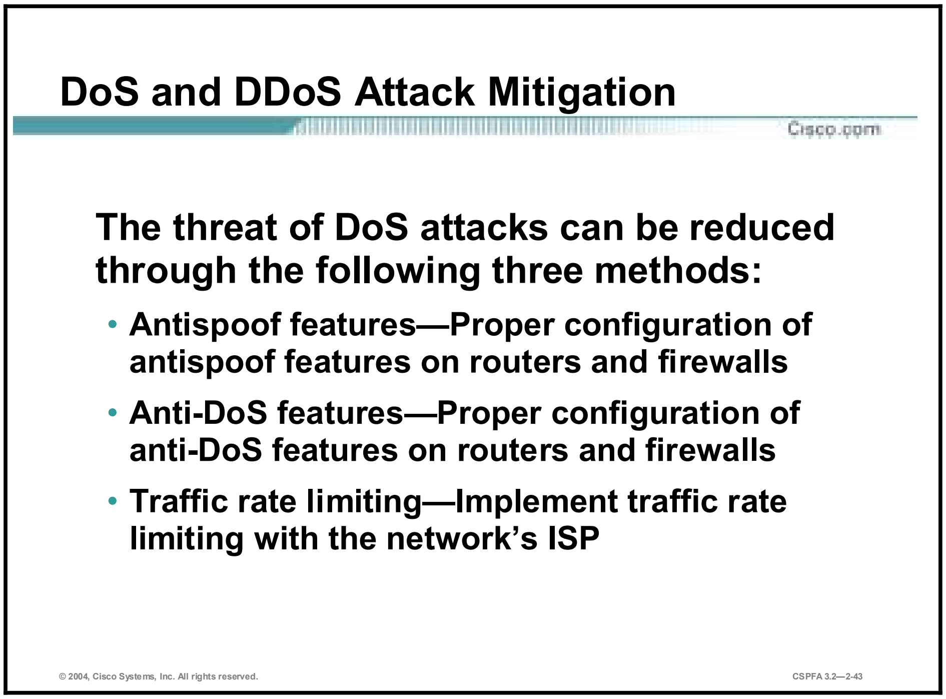
Agent systems

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—2-42

DDoS attacks are the “next generation” of DoS attacks on the Internet. This type of attack is not new—UDP and TCP SYN flooding, Internet Control Message Protocol (ICMP) echo request floods, and ICMP directed broadcasts (also known as smurf attacks) are similar—but the scope certainly is new. Victims of DDoS attacks experience packet flooding from many different sources, possibly spoofed IP source addresses, that bring their network connectivity to a grinding halt. In the past, the typical DoS attack involved a single attacker’s attempt to flood a target host with packets. With DDoS tools, an attacker can conduct the same attack using thousands of systems.

In the figure, the hacker uses a terminal to scan for systems to hack. When the handler systems are accessed, the hacker then installs software on them to scan for, compromise, and infect agent systems. When the agent systems are accessed, the hacker then loads remote control attack software to carry out the DoS attack.



DoS and DDoS Attack Mitigation

The threat of DoS attacks can be reduced through the following three methods:

- **Antispoof features—Proper configuration of antispoof features on routers and firewalls**
- **Anti-DoS features—Proper configuration of anti-DoS features on routers and firewalls**
- **Traffic rate limiting—Implement traffic rate limiting with the network's ISP**

CSPFA 3.2—2-43

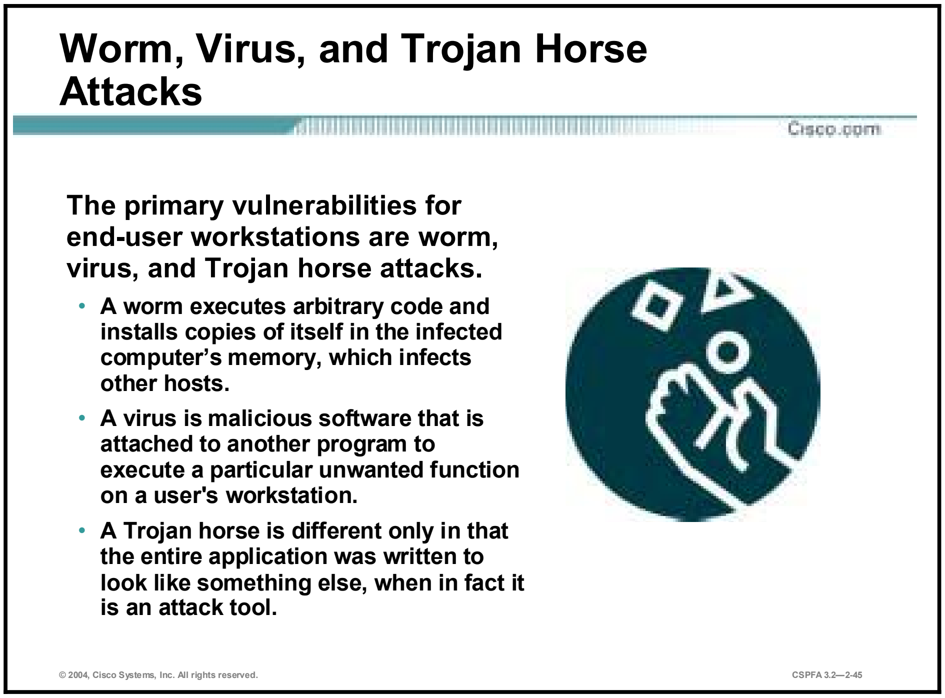
When they involve specific network server applications, such as an HTTP server or an FTP server, these attacks can focus on acquiring and keeping open all the available connections supported by that server, effectively locking out valid users of the server or service. DoS attacks can also be implemented using common Internet protocols, such as TCP and ICMP. While most DoS attacks exploit a weakness in the overall architecture of the system being attacked rather than a software bug or security hole, some attacks compromise the performance of your network by flooding the network with undesired, and often useless, network packets and by providing false information about the status of network resources.

The threat of DoS attacks can be reduced through the following three methods:

- Antispoof features—Proper configuration of antispoof features on your routers and firewalls can reduce your risk. This configuration includes RFC 2827 filtering at a minimum. If hackers cannot mask their identities, they might not attack.
- Anti-DoS features—Proper configuration of anti-DoS features on routers and firewalls can help limit the effectiveness of an attack. These features often involve limits on the amount of half-open connections that a system allows at any given time.
- Traffic rate limiting—An organization can implement traffic rate limiting with its ISP. This type of filtering limits the amount of nonessential traffic that crosses network segments at a certain rate. A common example is to limit the amount of ICMP traffic allowed into a network because it is used only for diagnostic purposes. ICMP-based DDoS attacks are common.

Worm, Virus, and Trojan Horse Attacks and Mitigation

This topic describes worm, virus, and Trojan horse attacks and their mitigation.



Worm, Virus, and Trojan Horse Attacks

Cisco.com

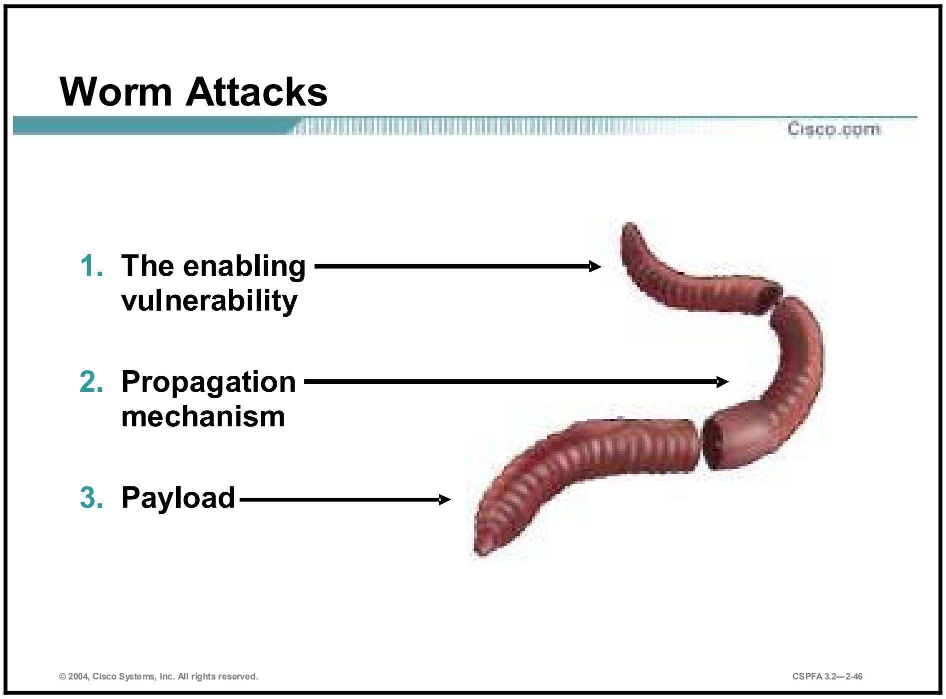
The primary vulnerabilities for end-user workstations are worm, virus, and Trojan horse attacks.

- **A worm executes arbitrary code and installs copies of itself in the infected computer's memory, which infects other hosts.**
- **A virus is malicious software that is attached to another program to execute a particular unwanted function on a user's workstation.**
- **A Trojan horse is different only in that the entire application was written to look like something else, when in fact it is an attack tool.**

CSPFA 3.2—2-45

The primary vulnerabilities for end-user workstations are worm, virus, and Trojan horse attacks.

- A worm executes arbitrary code and installs copies of itself in the infected computer's memory, which infects other hosts.
- A virus is malicious software that is attached to another program to execute a particular unwanted function on a user's workstation.
- A Trojan horse is different only in that the entire application was written to look like something else, when in fact it is an attack tool.



Worm Attacks

Cisco.com

1. The enabling vulnerability
2. Propagation mechanism
3. Payload

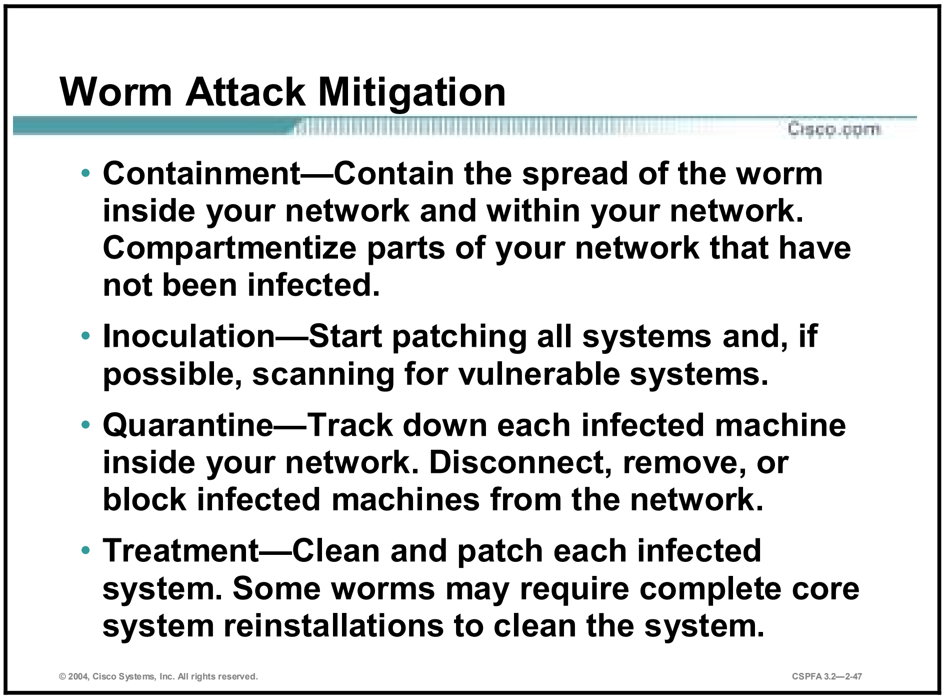
© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—2-46

The anatomy of a worm attack is as follows:

- The enabling vulnerability—A worm installs itself using an exploit vector on a vulnerable system.
- Propagation mechanism—After gaining access to devices, a worm replicates and selects new targets.
- Payload—Once the device is infected with a worm, the attacker has access to the host—often as a privileged user. Attackers could use a local exploit to escalate their privilege level to administrator.

Typically, worms are self-contained programs that attack a system and try to exploit a vulnerability in the target. Upon successful exploitation of the vulnerability, the worm copies its program from the attacking host to the newly exploited system to begin the cycle again. A virus normally requires a vector to carry the virus code from one system to another. The vector can be a word-processing document, an e-mail message, or an executable program. The key element that distinguishes a computer worm from a computer virus is that human interaction is required to facilitate the spread of a virus.



Worm Attack Mitigation

Cisco.com

- **Containment—Contain the spread of the worm inside your network and within your network. Compartmentize parts of your network that have not been infected.**
- **Inoculation—Start patching all systems and, if possible, scanning for vulnerable systems.**
- **Quarantine—Track down each infected machine inside your network. Disconnect, remove, or block infected machines from the network.**
- **Treatment—Clean and patch each infected system. Some worms may require complete core system reinstallations to clean the system.**

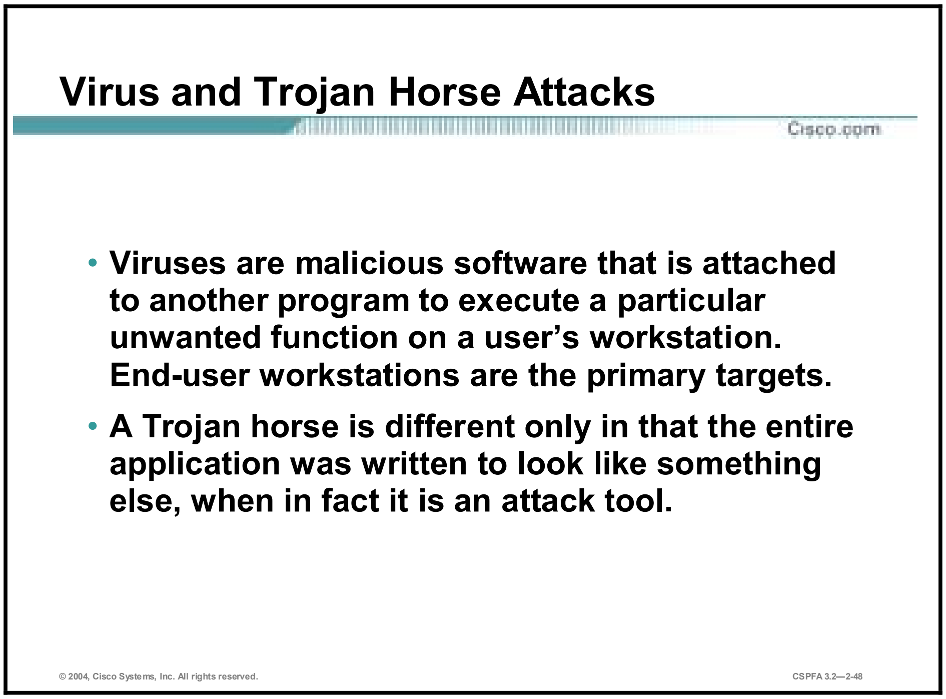
CSPFA 3.2—2-47

Worm attack mitigation requires diligence on the part of system and network administration staff. Coordination between system administration, network engineering, and security operations personnel is critical in responding effectively to a worm incident. The following are the recommended steps for worm attack mitigation:

- Containment
- Inoculation
- Quarantine
- Treatment

Typical incident response methodologies can be subdivided into six major categories. The following categories are based on the network service provider security (NSP-SEC) incident response methodology:

- Preparation—Acquire the resources to respond.
- Identification—Identify the worm.
- Classification—Classify the type of worm.
- Traceback—Trace the worm back to its origin.
- Reaction—Isolate and repair the affected systems.
- Post mortem—Document and analyze the process used for the future.



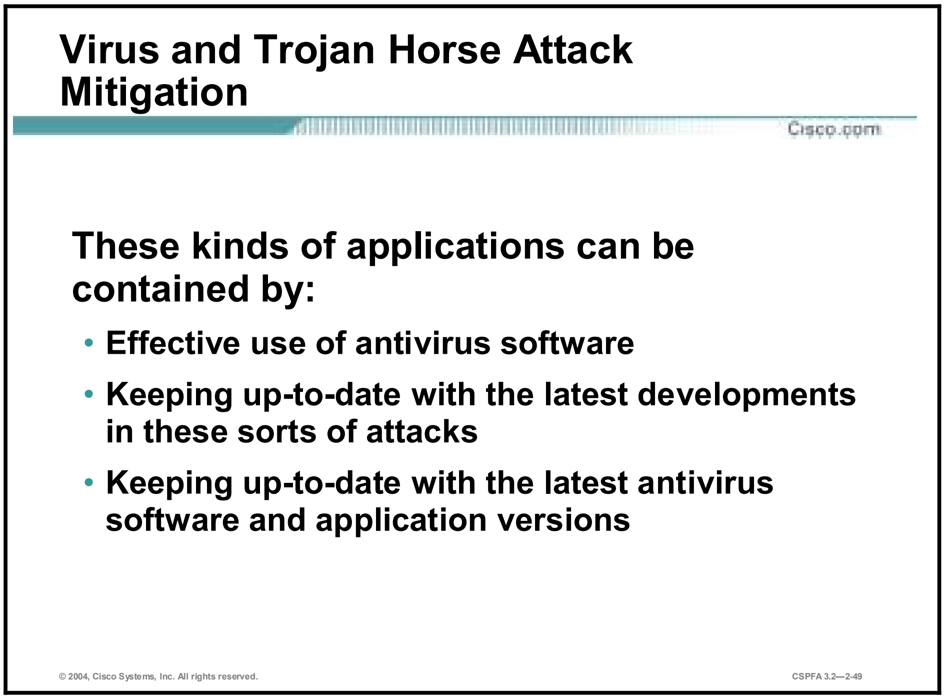
Virus and Trojan Horse Attacks

- **Viruses are malicious software that is attached to another program to execute a particular unwanted function on a user's workstation. End-user workstations are the primary targets.**
- **A Trojan horse is different only in that the entire application was written to look like something else, when in fact it is an attack tool.**

CSPFA 3.2—2-48

The primary vulnerabilities for end-user workstations are viruses and Trojan horse attacks. Viruses are malicious software that is attached to another program to execute a particular unwanted function on a user's workstation. An example of a virus is a program that is attached to command.com (the primary interpreter for Windows systems) that deletes certain files and infects any other versions of command.com that it can find.

A Trojan horse is different only in that the entire application was written to look like something else, when in fact it is an attack tool. An example of a Trojan horse is a software application that runs a simple game on the user's workstation. While the user is occupied with the game, the Trojan horse mails a copy of itself to every user in the user's address book. The other users receive the game and then play it, thus spreading the Trojan horse.



Virus and Trojan Horse Attack Mitigation

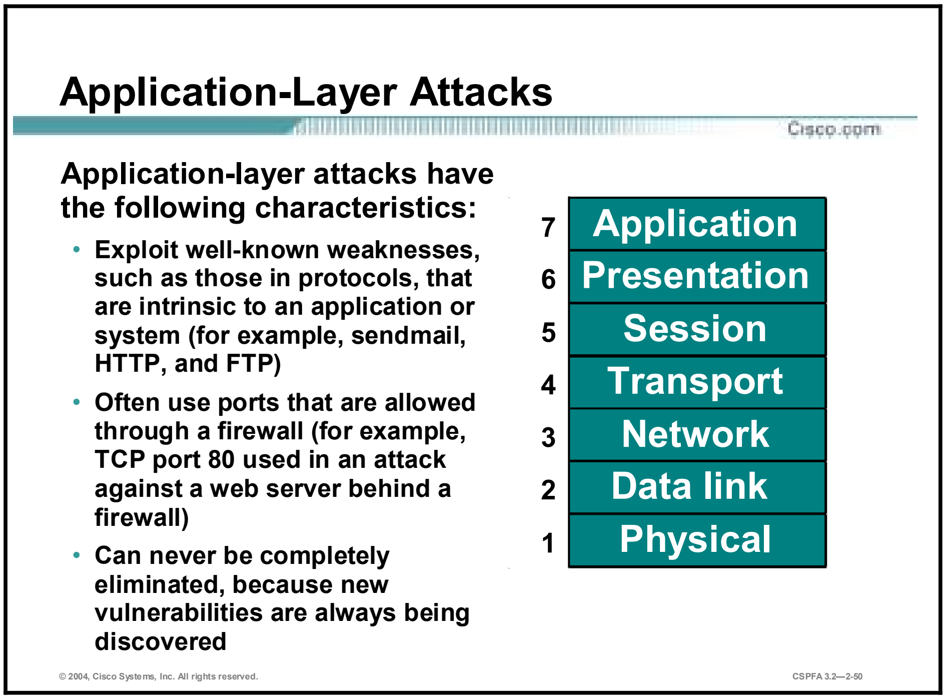
Cisco.com

These kinds of applications can be contained by:

- Effective use of antivirus software
- Keeping up-to-date with the latest developments in these sorts of attacks
- Keeping up-to-date with the latest antivirus software and application versions

CSPFA 3.2—2-49

These kinds of applications can be contained through the effective use of antivirus software at the user level and potentially at the network level. Antivirus software can detect most viruses and many Trojan horse applications and prevent them from spreading in the network. Keeping up-to-date with the latest developments in these sorts of attacks can also lead to a more effective posture against these attacks. As new virus or Trojan applications are released, enterprises need to keep up-to-date with the latest antivirus software and application versions.



Application-Layer Attacks

Cisco.com

Application-layer attacks have the following characteristics:

- Exploit well-known weaknesses, such as those in protocols, that are intrinsic to an application or system (for example, sendmail, HTTP, and FTP)
- Often use ports that are allowed through a firewall (for example, TCP port 80 used in an attack against a web server behind a firewall)
- Can never be completely eliminated, because new vulnerabilities are always being discovered

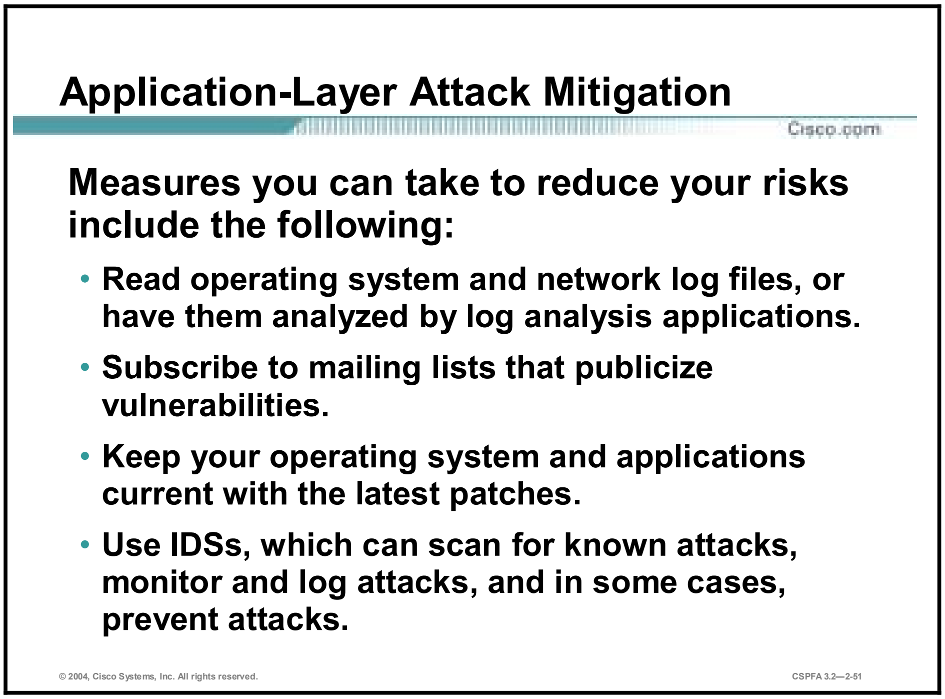
7 Application
6 Presentation
5 Session
4 Transport
3 Network
2 Data link
1 Physical

CSPFA 3.2—2-50

Application-layer attacks can be implemented using several different methods:

- One of the most common methods is exploiting well-known weaknesses in software commonly found on servers, such as sendmail, PostScript, and FTP. By exploiting these weaknesses, attackers can gain access to a computer with the permissions of the account running the application, which is usually a privileged, system-level account.
- Trojan horse program attacks are implemented using programs that an attacker substitutes for common programs. These programs may provide all the functionality that the normal program provides, but also include other features that are known to the attacker, such as monitoring login attempts to capture user account and password information. These programs can capture sensitive information and distribute it back to the attacker. They can also modify application functionality, such as applying a blind carbon copy to all e-mail messages so that the attacker can read all of your organization's e-mail.

  One of the oldest forms of application-layer attacks is a Trojan horse program that displays a screen, banner, or prompt that the user believes is the valid login sequence. The program then captures the information that the user enters and stores or e-mails it to the attacker. Next, the program either forwards the information to the normal login process (normally impossible on modern systems) or simply sends an expected error to the user (for example, Bad Username/Password Combination), exits, and starts the normal login sequence. The user, believing that he or she has incorrectly entered the password (a common mistake experienced by everyone), re-enters the information and is allowed access.
- One of the newest forms of application-layer attacks exploits the openness of several new technologies: the HTML specification, web browser functionality, and HTTP. These attacks, which include Java applets and ActiveX controls, involve passing harmful programs across the network and loading them through a user's browser.



Application-Layer Attack Mitigation

Measures you can take to reduce your risks include the following:

- **Read operating system and network log files, or have them analyzed by log analysis applications.**
- **Subscribe to mailing lists that publicize vulnerabilities.**
- **Keep your operating system and applications current with the latest patches.**
- **Use IDSs, which can scan for known attacks, monitor and log attacks, and in some cases, prevent attacks.**

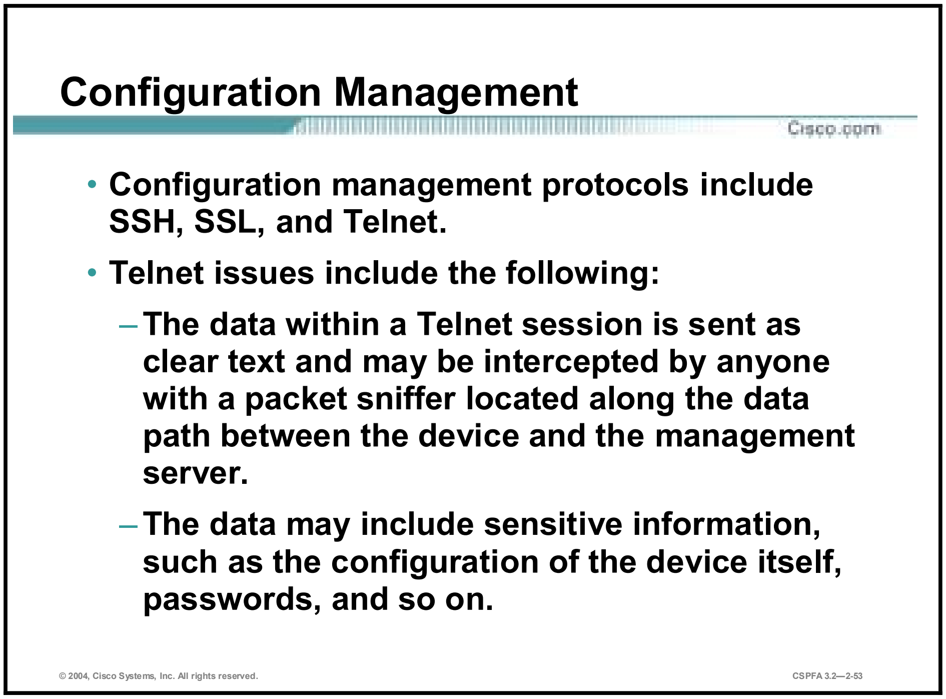
CSPFA 3.2—2-51

The following are some measures you can take to reduce your risks for application-layer attacks:

- Read operating system and network log files or have them analyzed—It is important to review all logs and take action accordingly.
- Subscribe to mailing lists that publicize vulnerabilities—Most application and operating system vulnerabilities are published on the Web by various sources.
- Keep your operating system and applications current with the latest patches—Always test patches and fixes in a nonproduction environment. This practice prevents downtime and keeps errors from being generated unnecessarily.
- Use IDSs to scan for known attacks, monitor and log attacks, and in some cases, prevent attacks—The use of IDSs can be essential to identifying security threats and mitigating some of those threats. In most cases, it can be done automatically.

Management Protocols and Functions

The protocols used to manage your network can in themselves be a source of vulnerability. This topic examines common management protocols and how they can be exploited.



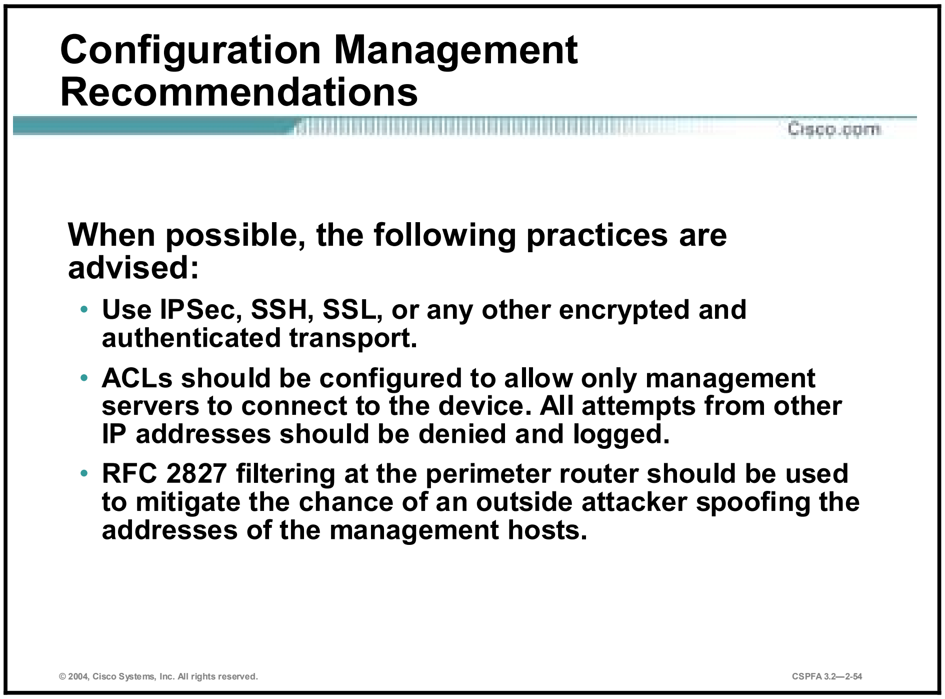
Configuration Management

Cisco.com

- Configuration management protocols include SSH, SSL, and Telnet.
- Telnet issues include the following:
  - The data within a Telnet session is sent as clear text and may be intercepted by anyone with a packet sniffer located along the data path between the device and the management server.
  - The data may include sensitive information, such as the configuration of the device itself, passwords, and so on.

CSPFA 3.2—2-53

If the managed device does not support any of the recommended protocols, such as SSH and SSL, Telnet may have to be used (although this protocol is not highly recommended). The network administrator should recognize that the data within a Telnet session is sent as clear text and may be intercepted by anyone with a packet sniffer located along the data path between the managed device and the management server. The clear text may include important information, such as the configuration of the device itself, passwords, and other sensitive data.



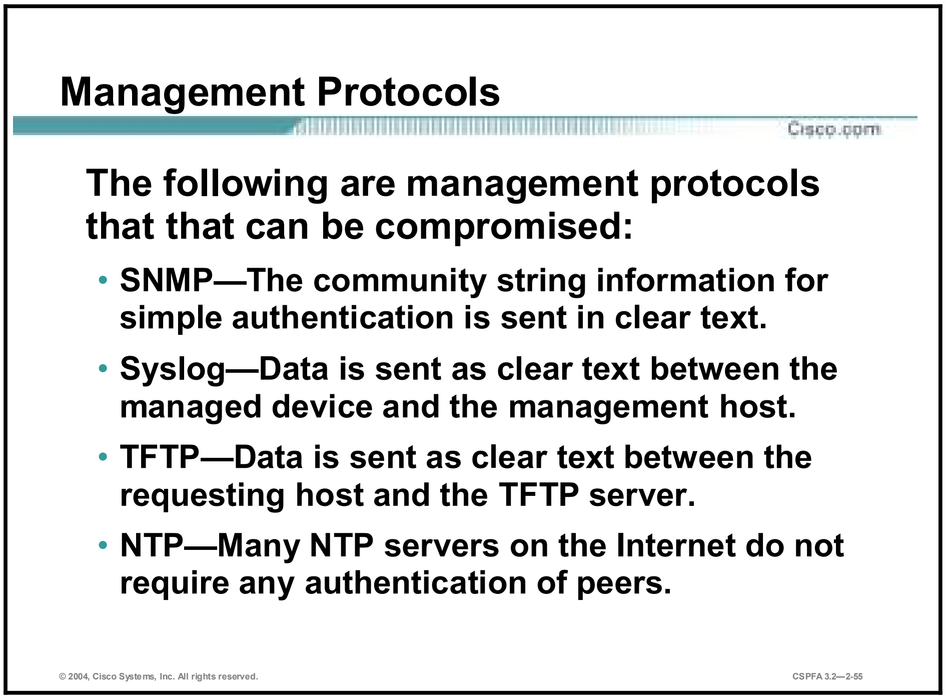
Configuration Management Recommendations

When possible, the following practices are advised:

- **Use IPSec, SSH, SSL, or any other encrypted and authenticated transport.**
- **ACLs should be configured to allow only management servers to connect to the device. All attempts from other IP addresses should be denied and logged.**
- **RFC 2827 filtering at the perimeter router should be used to mitigate the chance of an outside attacker spoofing the addresses of the management hosts.**

CSPFA 3.2—2-54

Regardless of whether SSH, SSL, or Telnet is used for remote access to the managed device, access control lists (ACLs) should be configured to allow only management servers to connect to the device. All attempts from other IP addresses should be denied and logged. RFC 2827 filtering at the ingress router should also be implemented to reduce the chance of an attacker from outside the network spoofing the addresses of the management hosts.



Management Protocols

The following are management protocols that that can be compromised:

- **SNMP—The community string information for simple authentication is sent in clear text.**
- **Syslog—Data is sent as clear text between the managed device and the management host.**
- **TFTP—Data is sent as clear text between the requesting host and the TFTP server.**
- **NTP—Many NTP servers on the Internet do not require any authentication of peers.**

CSPFA 3.2—2-55

Simple Network Management Protocol (SNMP) is a network management protocol that can be used to retrieve information from a network device (commonly referred to as read-only access) or to remotely configure parameters on the device (commonly referred to as read-write access). SNMP uses passwords, called community strings, within each message as a very simple form of security. Unfortunately, most implementations of SNMP on networking devices today send the community string in clear text along with the message. Therefore, SNMP messages may be intercepted by anyone with a packet sniffer located along the data path between the device and the management server, and the community string may be compromised.

Syslog, which is information generated by a device that has been configured for logging, is sent as clear text between the managed device and the management host. Syslog has no packet-level integrity checking to ensure that the packet contents have not been altered in transit. An attacker may alter Syslog data in order to confuse a network administrator during an attack.

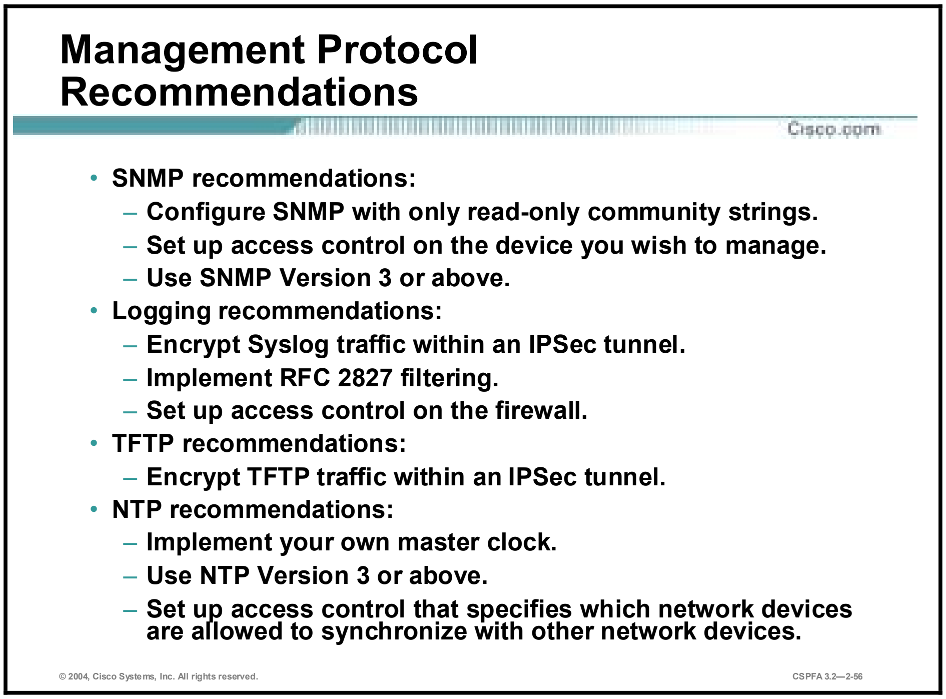
Trivial File Transfer Protocol (TFTP) is used for transferring configuration or system files across the network. TFTP uses UDP for the data stream between the requesting host and the TFTP server.

As with other management protocols that send data in clear text, the network administrator should recognize that the data within a TFTP session might be intercepted by anyone with a packet sniffer located along the data path between the device and the management server. Where possible, TFTP traffic should be encrypted within an IPSec tunnel in order to reduce the chance of its being intercepted.

Network Time Protocol (NTP) is used to synchronize the clocks of various devices across a network. Synchronization of the clocks within a network is critical for digital certificates and for correct interpretation of events within Syslog data.

A secure method of providing clocking for the network is for network administrators to implement their own master clocks for private networks synchronized to Coordinated Universal Time (UTC) via satellite or radio. However, clock sources are available for synchronization via the Internet, for network administrators who do not wish to implement their own master clocks because of cost or other reasons.

An attacker could attempt a DoS attack on a network by sending bogus NTP data across the Internet in an attempt to change the clocks on network devices in such a manner that digital certificates are considered invalid. Further, an attacker could attempt to confuse a network administrator during an attack by disrupting the clocks on network devices. This scenario would make it difficult for the network administrator to determine the order of Syslog events on multiple devices.



Management Protocol Recommendations

- SNMP recommendations:
  - Configure SNMP with only read-only community strings.
  - Set up access control on the device you wish to manage.
  - Use SNMP Version 3 or above.
- Logging recommendations:
  - Encrypt Syslog traffic within an IPSec tunnel.
  - Implement RFC 2827 filtering.
  - Set up access control on the firewall.
- TFTP recommendations:
  - Encrypt TFTP traffic within an IPSec tunnel.
- NTP recommendations:
  - Implement your own master clock.
  - Use NTP Version 3 or above.
  - Set up access control that specifies which network devices are allowed to synchronize with other network devices.

CSPFA 3.2—2-56

The following are SNMP recommendations:

- Configure SNMP with only read-only community strings.
- Set up access control on the device you wish to manage via SNMP to allow access by only the appropriate management hosts.
- Use SNMP Version 3 or above.

When possible, the following practices are advised:

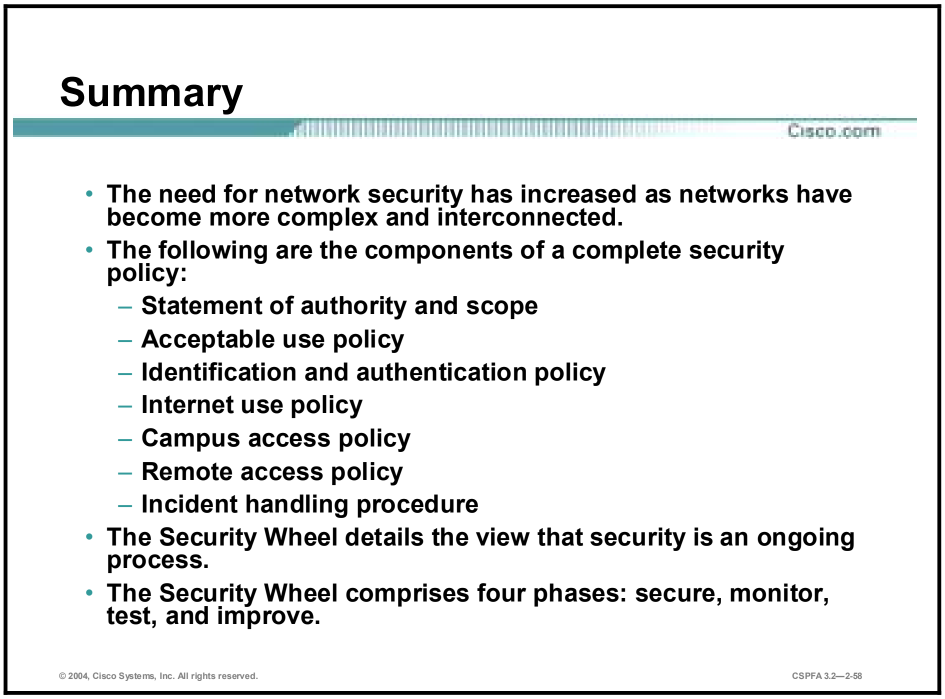
- Encrypt Syslog traffic within an IPSec tunnel.
- When allowing Syslog access from devices on the outside of a firewall, you should implement RFC 2827 filtering at the perimeter router.
- ACLs should also be implemented on the firewall in order to allow Syslog data from only the managed devices themselves to reach the management hosts.
- When possible, TFTP traffic should be encrypted within an IPSec tunnel in order to reduce the chance of its being intercepted.

The following are NTP recommendations:

- Implement your own master clock for private network synchronization.
- Use NTP Version 3 or above because these versions support a cryptographic authentication mechanism between peers.
- Use ACLs that specify which network devices are allowed to synchronize with other network devices.

Summary

This topic summarizes what you learned in this lesson.

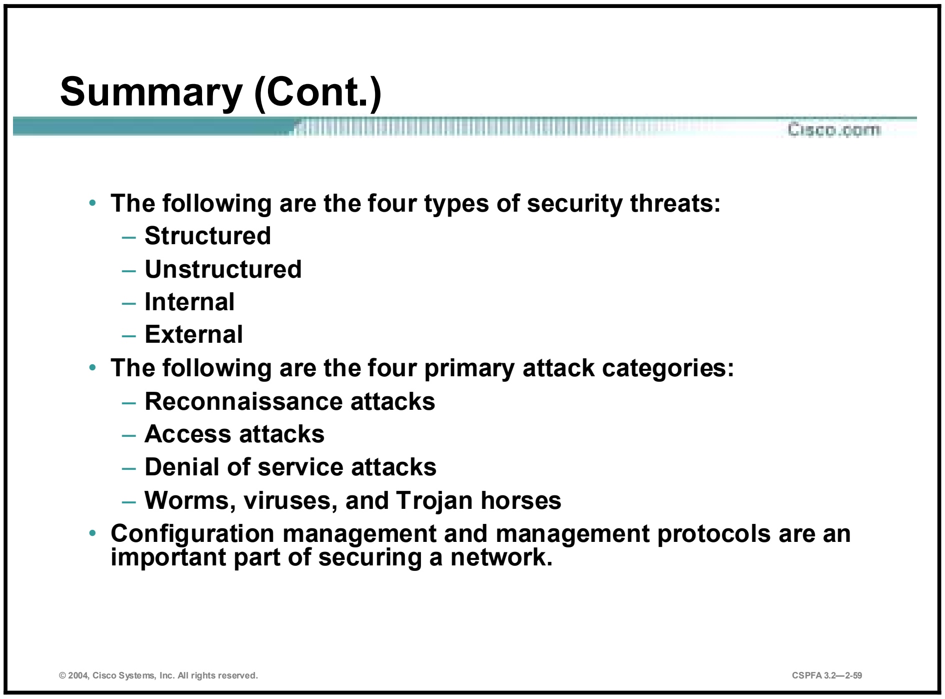


Summary

Cisco.com

- The need for network security has increased as networks have become more complex and interconnected.
- The following are the components of a complete security policy:
  - Statement of authority and scope
  - Acceptable use policy
  - Identification and authentication policy
  - Internet use policy
  - Campus access policy
  - Remote access policy
  - Incident handling procedure
- The Security Wheel details the view that security is an ongoing process.
- The Security Wheel comprises four phases: secure, monitor, test, and improve.

CSPFA 3.2—2-58



Summary (Cont.)

Cisco.com

- The following are the four types of security threats:
  - Structured
  - Unstructured
  - Internal
  - External
- The following are the four primary attack categories:
  - Reconnaissance attacks
  - Access attacks
  - Denial of service attacks
  - Worms, viruses, and Trojan horses
- Configuration management and management protocols are an important part of securing a network.

CSPFA 3.2—2-59

3

Cisco PIX Firewall **Tech**nology and Features

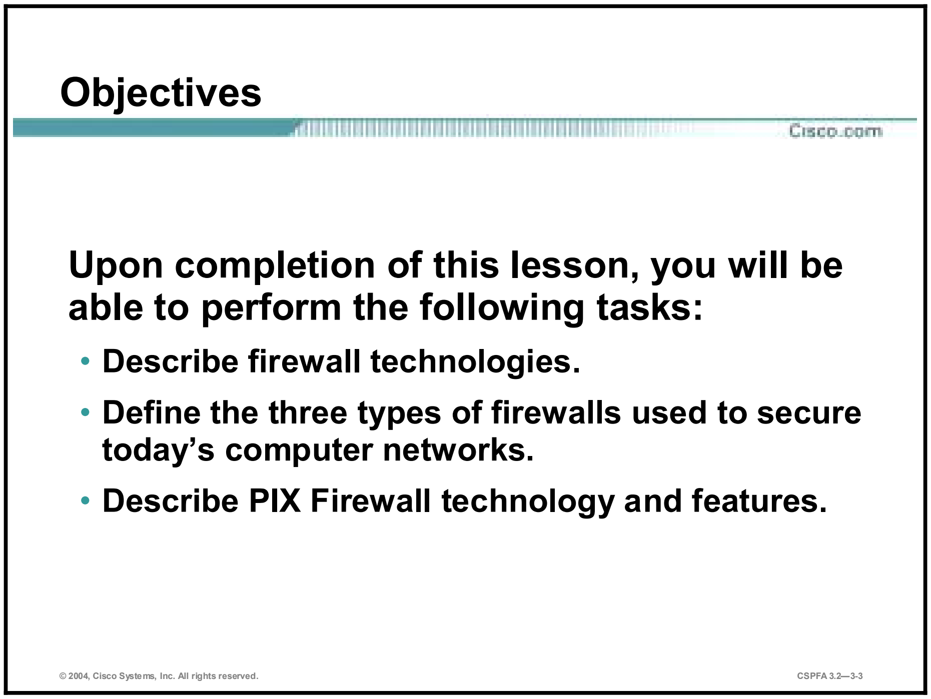
Overview

This lesson includes the following topics:

- Objectives
- Firewalls
- PIX Firewall overview
- Summary

Objectives

This topic lists the lesson's objectives.



Objectives

Upon completion of this lesson, you will be able to perform the following tasks:

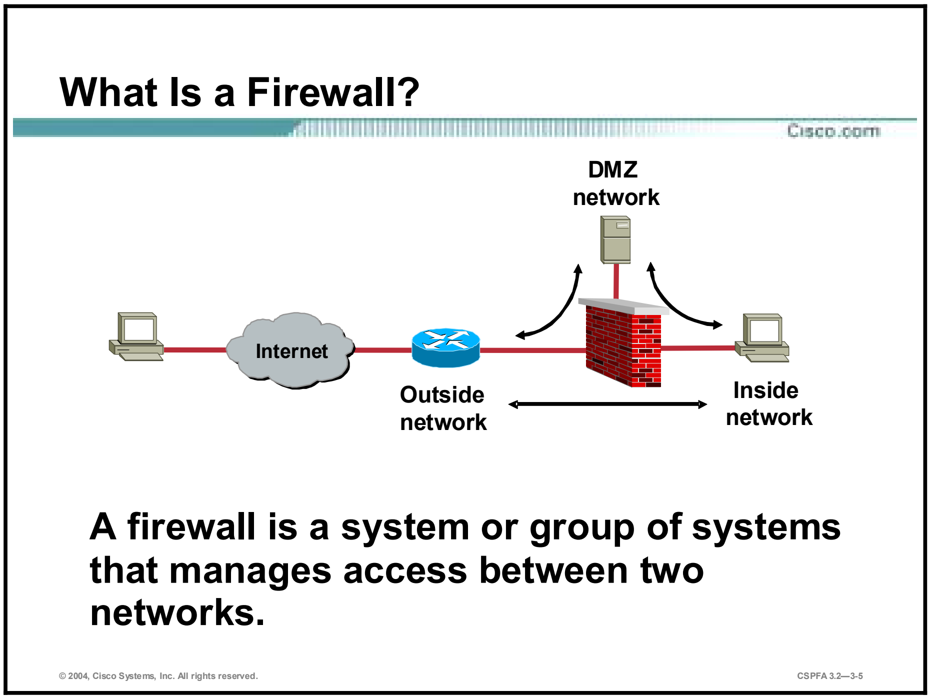
- **Describe firewall technologies.**
- **Define the three types of firewalls used to secure today's computer networks.**
- **Describe PIX Firewall technology and features.**

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—3-3

Firewalls

This topic provides an explanation of a firewall.

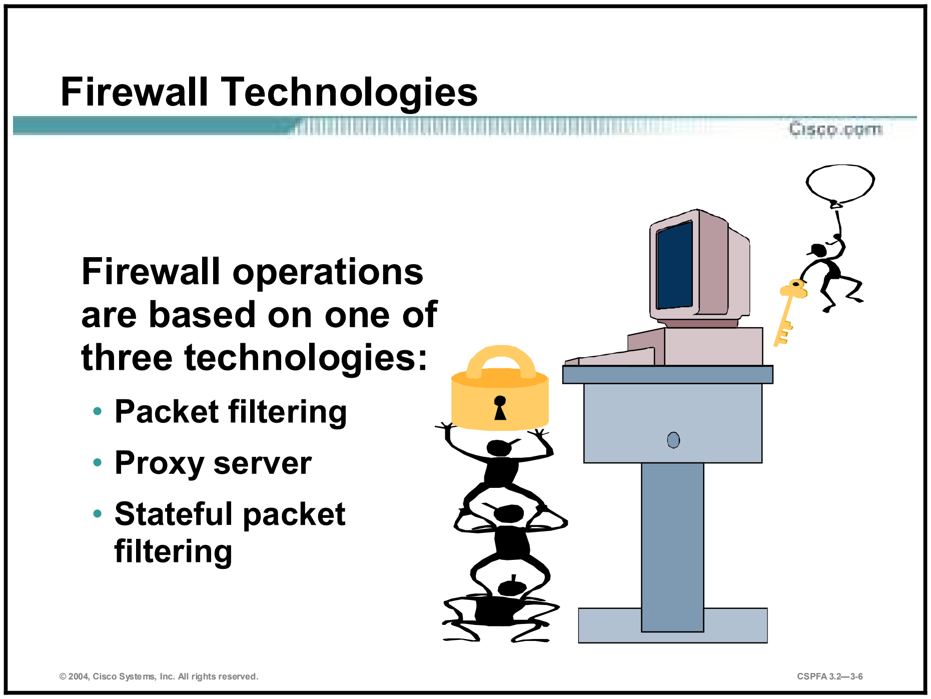


What Is a Firewall?

A firewall is a system or group of systems that manages access between two networks.

By conventional definition, a firewall is a partition made of fireproof material designed to prevent the spread of fire from one part of a building to another. It can also be used to isolate one compartment from another.

When applying the term firewall to a computer network, a firewall is a system or group of systems that enforces an access control policy between two or more networks.



Firewall Technologies

Cisco.com

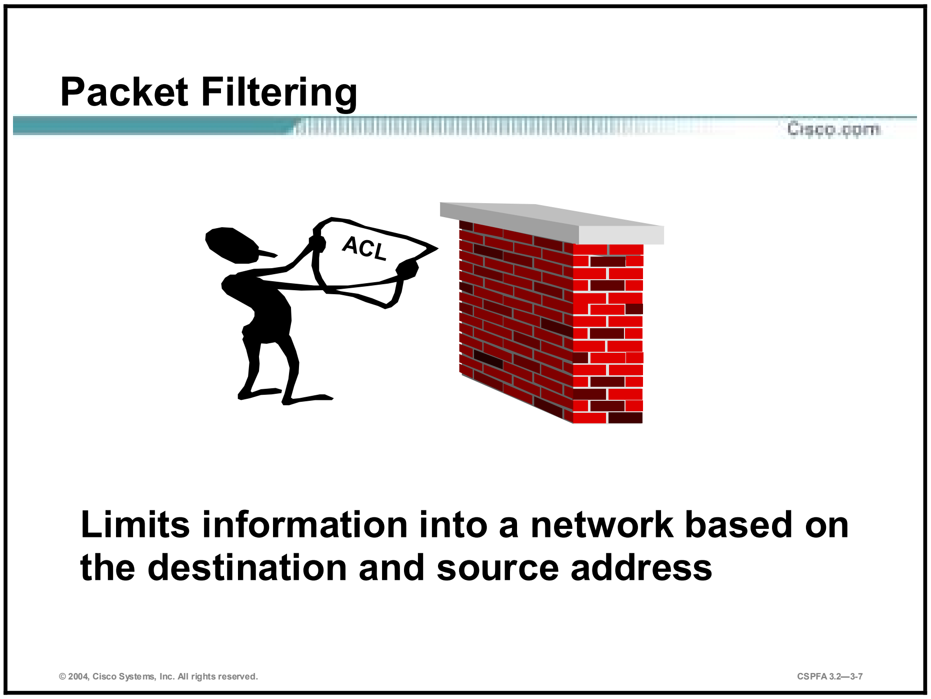
Firewall operations are based on one of three technologies:

- **Packet filtering**
- **Proxy server**
- **Stateful packet filtering**

CSPFA 3.2—3-6

Firewall operations are based on one of three technologies:

- Packet filtering—Limits information into a network based on static packet header information.
- Proxy server—Requests connections between a client on the inside of the firewall and the Internet.
- Stateful packet filtering—Combines the best of packet filtering and proxy server technologies.



Packet Filtering

Cisco.com

Limits information into a network based on the destination and source address

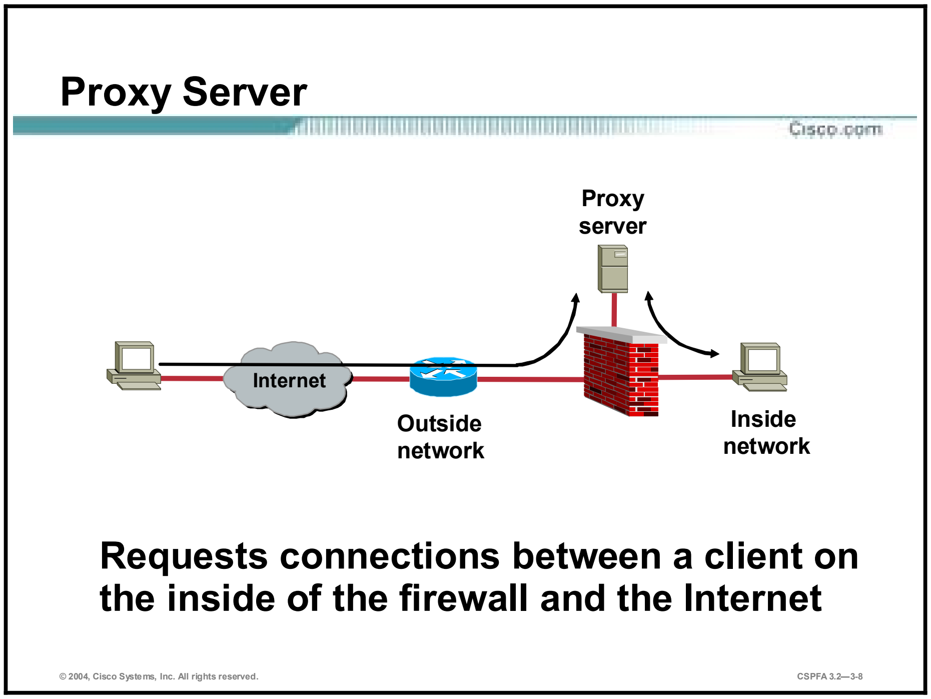
CSPFA 3.2—3-7

A firewall can use packet filtering to limit information entering a network, or information moving from one segment of a network to another. Packet filtering uses access control lists (ACLs), which allow a firewall to accept or deny access based on packet types and other variables.

This method is effective when a protected network receives a packet from an unprotected network. Any packet that is sent to the protected network and does not fit the criteria defined by the ACLs is dropped.

However, there are problems with packet filtering:

- Arbitrary packets can be sent that fit the ACL criteria and, therefore, pass through the filter.
- Packets can pass through the filter by being fragmented.
- Complex ACLs are difficult to implement and maintain correctly.
- Some services cannot be filtered.



Proxy Server

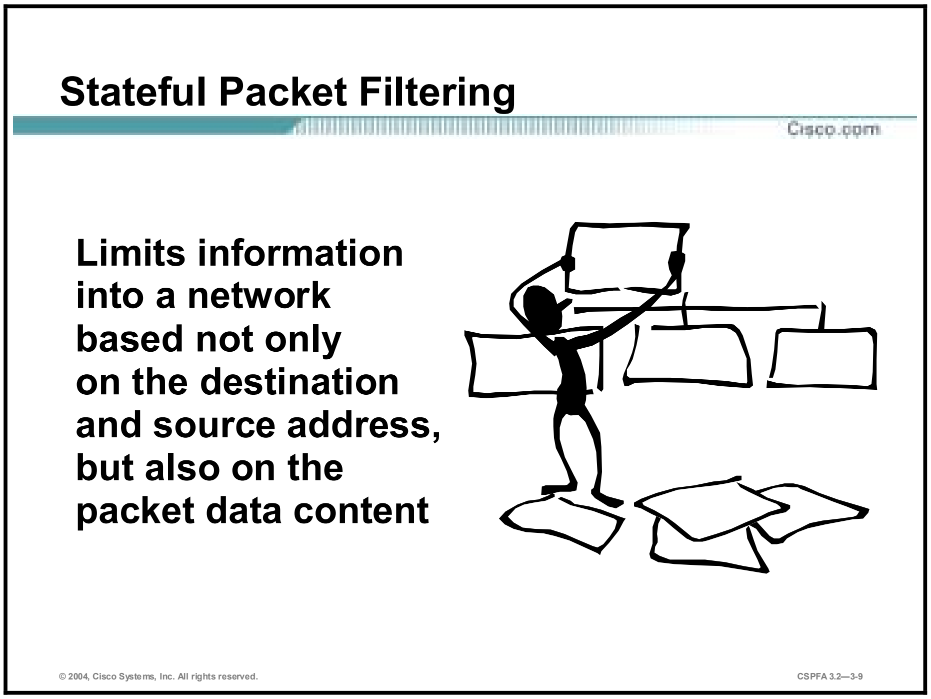
Requests connections between a client on the inside of the firewall and the Internet

CSPFA 3.2—3-8

A proxy server is a firewall device that examines packets at higher layers of the Open Systems Interconnection (OSI) model. This device hides valuable data by requiring users to communicate with a secure system by means of a proxy. Users gain access to the network by going through a process that establishes session state, user authentication, and authorized policy. This means that users connect to outside services via application programs (proxies) running on the gateway connecting to the outside unprotected zone.

However, there are problems with the proxy server because it:

- Creates a single point of failure, which means that if the entrance to the network is compromised, then the entire network is compromised.
- Is difficult to add new services to the firewall.
- Performs more slowly under stress.



Stateful Packet Filtering

Cisco.com

Limits information into a network based not only on the destination and source address, but also on the packet data content

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—3-9

Stateful packet filtering is the method used by the Cisco PIX Firewall. This technology maintains complete session state. Each time a Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) connection is established for inbound or outbound connections, the information is logged in a stateful session flow table.

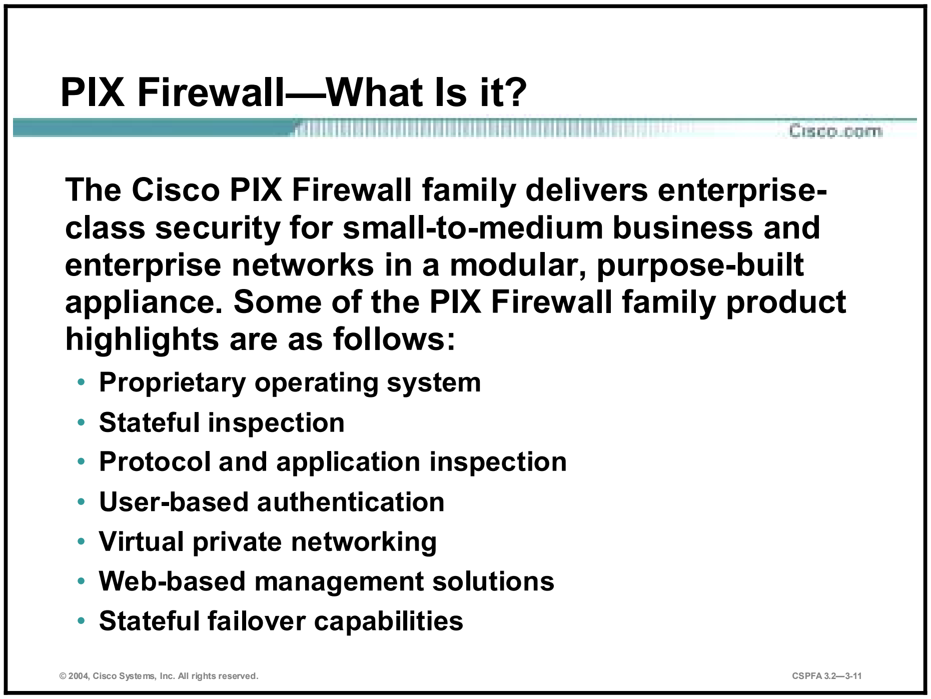
The stateful session flow table contains the source and destination addresses, port numbers, TCP sequencing information, and additional flags for each TCP or UDP connection associated with that particular session. This information creates a connection object and, consequently, all inbound and outbound packets are compared against session flows in the stateful session flow table. Data is permitted through the firewall only if an appropriate connection exists to validate its passage.

This method is effective because

- It works on packets and connections.
- It operates at a higher performance level than packet filtering or using a proxy server.
- It records data in a table for every connection or connectionless transaction. This table serves as a reference point to determine if packets belong to an existing connection or are from an unauthorized source.

PIX Firewall Overview

This topic discusses the basic concepts of the PIX Firewall.



PIX Firewall—What Is it?

Cisco.com

The Cisco PIX Firewall family delivers enterprise-class security for small-to-medium business and enterprise networks in a modular, purpose-built appliance. Some of the PIX Firewall family product highlights are as follows:

- **Proprietary operating system**
- **Stateful inspection**
- **Protocol and application inspection**
- **User-based authentication**
- **Virtual private networking**
- **Web-based management solutions**
- **Stateful failover capabilities**

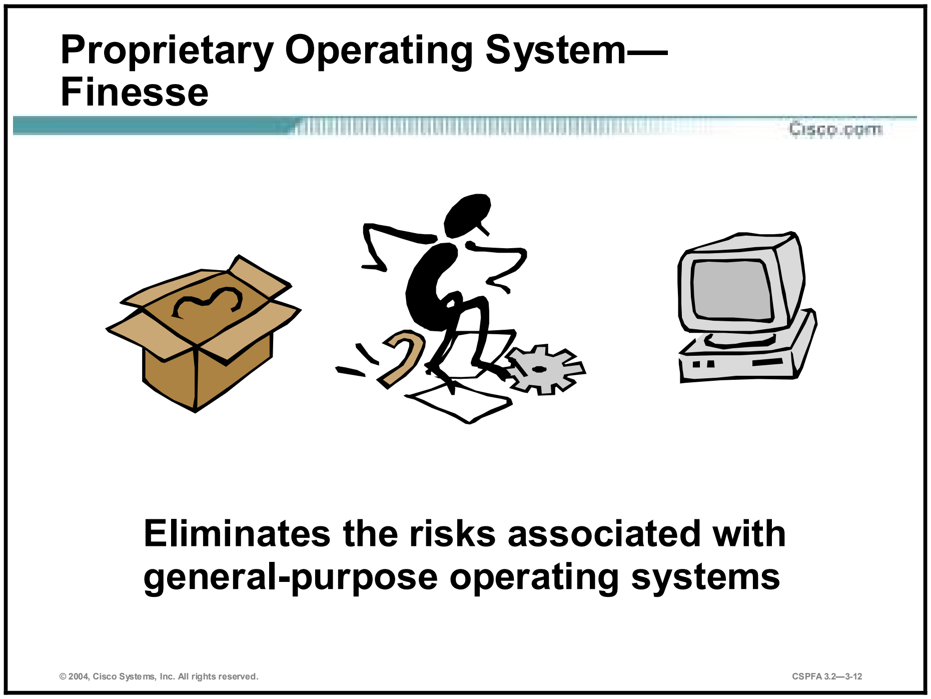
CSPFA 3.2—3-11

The Private Internet Exchange (PIX) Firewall is a key element in the overall Cisco end-to-end security solution. The world-leading Cisco PIX Firewall Series provides enterprise-class, integrated network security services including stateful inspection firewalling, protocol and application inspection, virtual private network (VPN), in-line intrusion protection, and rich multimedia and voice security—in cost-effective, easy-to-deploy solutions. Ranging from compact, "plug-and-play" desktop firewalls for small offices to carrier-class gigabit firewalls for the most demanding enterprise and service-provider environments, Cisco PIX Firewalls provide robust security, performance, and reliability for network environments of all sizes.

Some of the PIX Firewall product highlights are as follows:

- Security, performance and reliability in purpose-built security appliances
- State-of-the-art stateful inspection firewalling using patented Adaptive Security Algorithm (ASA)
- Integrated protocol and application inspection engines that examine packet streams at Layers 4 through 7
- User-based authentication of inbound and outbound connections
- Robust VPN for secure site-to-site and remote access connections
- Simple, web-based management with PIX Device Manager (PDM)
- Stateful failover capabilities that ensure resilient network protection
- Dynamic and static Network Address Translation (NAT) and Port Address Translation (PAT)

- Integrated Intrusion Protection guards against popular Internet threats such as denial-of-service (DoS) attacks
- Robust remote manageability using CiscoWorks Firewall Management Center, Telnet/Secure Shell (SSH), Simple Network Management Protocol (SNMP) and Syslog

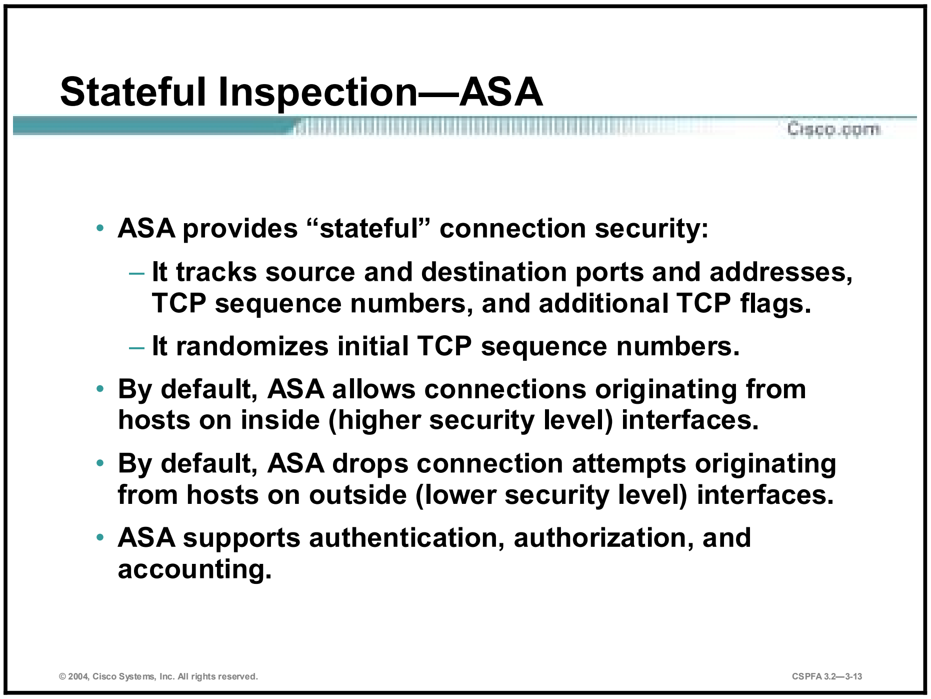


Proprietary Operating System—Finesse

Eliminates the risks associated with general-purpose operating systems

CSPFA 3.2—3-12

Finesse, a Cisco proprietary operating system, is a non-UNIX, non-Windows NT, Cisco IOS software-like operating system. Use of Finesse eliminates the risks associated with the general-purpose operating systems. It enables the PIX Firewall to deliver outstanding performance with up to 500,000 simultaneous connections—dramatically greater than any UNIX-based firewall.



Stateful Inspection—ASA

- ASA provides "stateful" connection security:
  - It tracks source and destination ports and addresses, TCP sequence numbers, and additional TCP flags.
  - It randomizes initial TCP sequence numbers.
- By default, ASA allows connections originating from hosts on inside (higher security level) interfaces.
- By default, ASA drops connection attempts originating from hosts on outside (lower security level) interfaces.
- ASA supports authentication, authorization, and accounting.

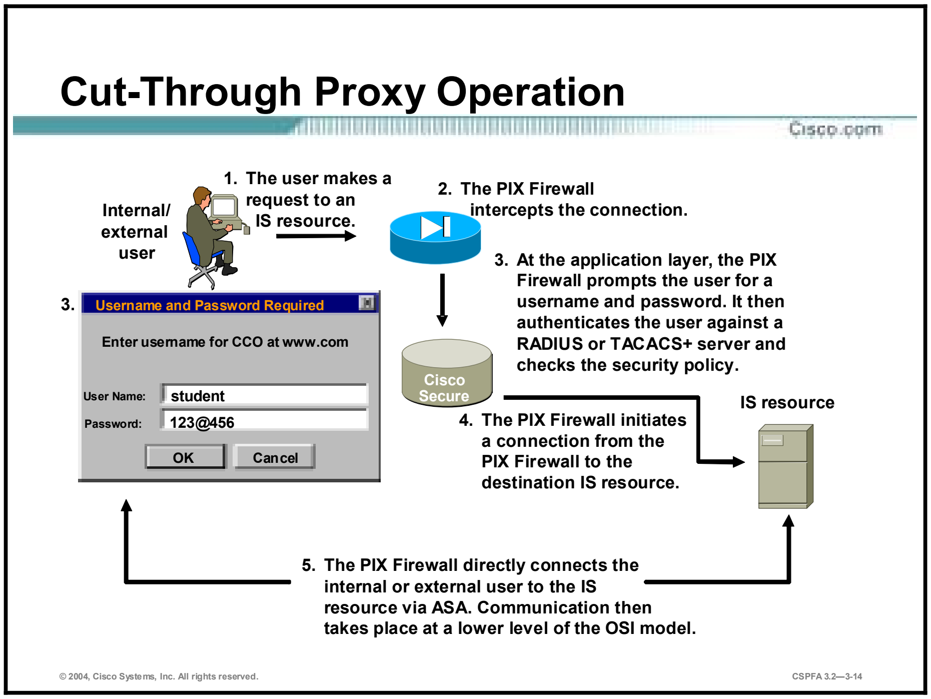
CSPFA 3.2—3-13

The heart of the PIX Firewall is the Adaptive Security Algorithm (ASA). The ASA maintains the secure perimeters between the networks controlled by the firewall. The stateful, connection-oriented ASA design creates session flows based on source and destinations addresses. It randomizes TCP sequence numbers, port numbers, and additional TCP flags before completion of the connection. This function is always in operation, monitoring return packets to ensure they are valid, and allows one-way (inside to outside) connections without an explicit configuration for each internal system and application. The randomizing of the TCP sequence numbers is to minimize the risk of a TCP sequence number attack. Because of the ASA, the PIX Firewall is less complex and more robust than a packet filtering-designed firewall.

Stateful packet filtering is a secure method of analyzing data packets that places extensive information about a data packet into a table. Each time a TCP connection is established for inbound or outbound connections through the PIX Firewall, the information about the connection is logged in a stateful session flow table. For a session to be established, information about the connection must match information stored in the table. With this methodology, the stateful filters work on the connections and not the packets, making it a more stringent security method with its sessions immune to hijacking.

Like a fingerprint, stateful packet filtering

- Obtains the session identifying parameters, IP addresses, and ports for each TCP connection.
- Logs the data in a stateful session flow table and creates a session object.
- Compares the inbound and outbound packets against session flows in the connection table.
- Allows data packets to flow through the PIX Firewall only if an appropriate connection exists to validate their passage.
- Temporarily sets up a connection object until the connection is terminated.



Cut-Through Proxy Operation

1. The user makes a request to an IS resource.

2. The PIX Firewall intercepts the connection.

3. At the application layer, the PIX Firewall prompts the user for a username and password. It then authenticates the user against a RADIUS or TACACS+ server and checks the security policy.

3. Username and Password Required

Enter username for CCO at www.com

User Name: student

Password: 123@456

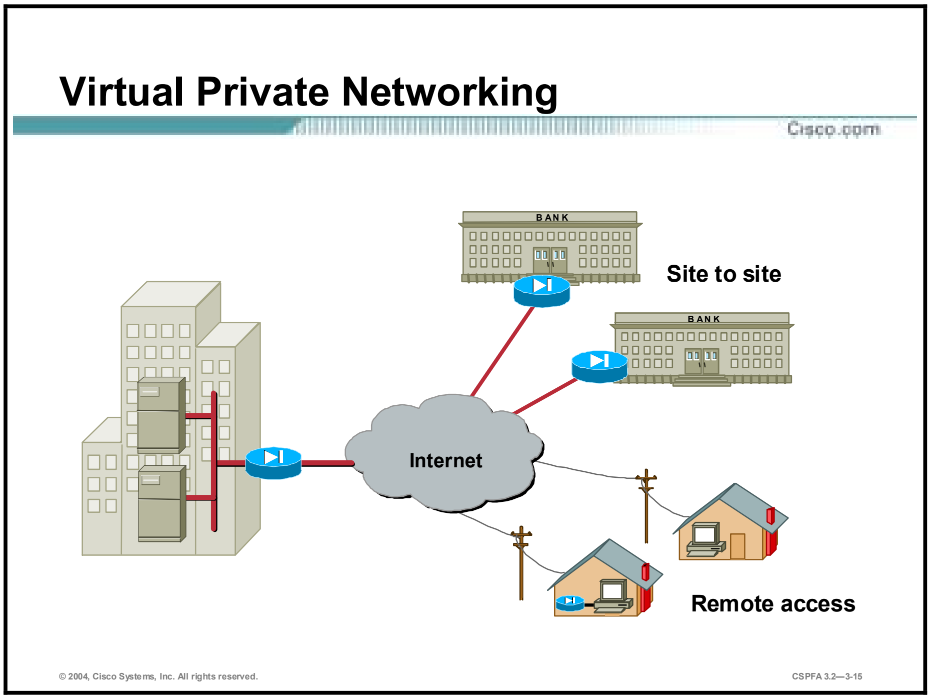
4. The PIX Firewall initiates a connection from the PIX Firewall to the destination IS resource.

5. The PIX Firewall directly connects the internal or external user to the IS resource via ASA. Communication then takes place at a lower level of the OSI model.

CSPFA 3.2—3-14

Cut-through proxy is a method of transparently verifying the identity of the users at the firewall, and permitting or denying access to any TCP- or UDP-based applications. This is also known as user-based authentication of inbound or outbound connections. Unlike a proxy server that analyzes every packet at the application layer of the OSI model, the PIX Firewall first challenges a user at the application layer. After the user is authenticated and the policy is checked, the PIX Firewall shifts the session flow to a lower layer of the OSI model for dramatically faster performance. This allows security policies to be enforced on a per-user-identification basis.

Connections must be authenticated with a user identification and password before they can be established. The user identification and password is entered via an initial HTTP, HTTPS, Telnet, or FTP connection. This method eliminates the price performance impact that UNIX system-based firewalls impose in similar configurations, and allows a finer level of administrative control over connections. The cut-through proxy method of the PIX Firewall also leverages the authentication and authorization services of the Cisco Secure Access Control Server (Cisco Secure ACS). The PIX Firewall is interoperable and scalable with IPSec, which includes an umbrella of security and authentication protocols, such as Internet Key Exchange (IKE) and Public Key Infrastructure (PKI). The PIX Firewall offers an IPSec-based VPN. Remote clients can securely access corporate networks through their Internet service providers (ISPs).

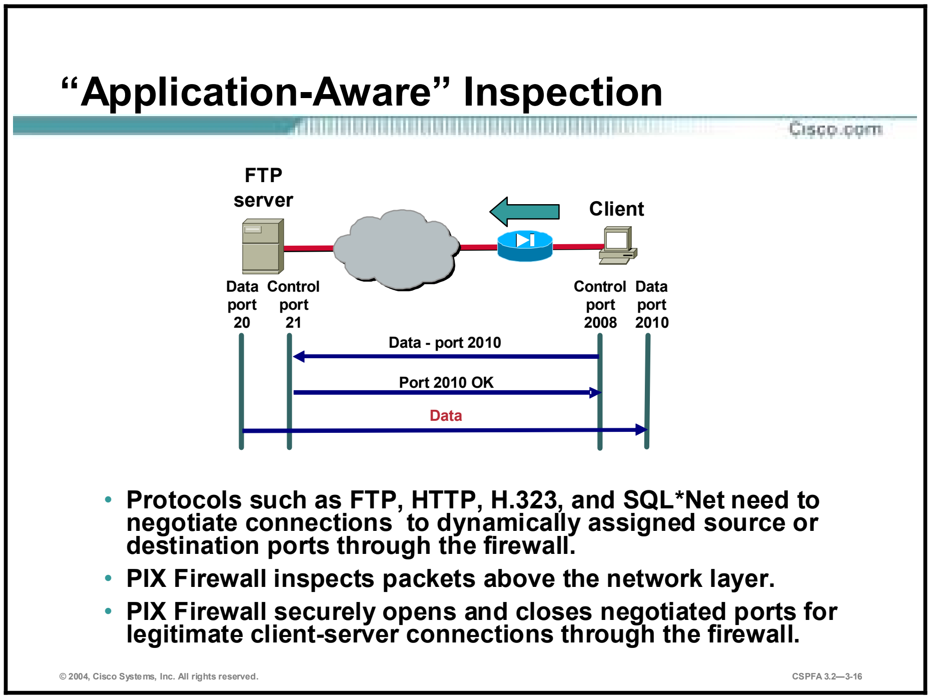


Virtual Private Networking

CSPFA 3.2—3-15

A virtual private network (VPN) is a service offering secure, reliable connectivity over a shared, public network infrastructure such as the Internet. Because the infrastructure is shared, connectivity can be provided at lower cost than existing dedicated private networks. The PIX Firewall enables VPNs for both site-to-site and remote access networks. PIX Firewall supports the following configurations:

- PIX Firewall to PIX Firewall secure VPN gateway—Two or more PIX Firewalls can enable a VPN, which secures traffic from devices behind the PIX Firewalls. The secure VPN gateway topology prevents the user from having to implement VPN devices or software inside the network, making the secure gateway transparent to users.
- PIX Firewall to Cisco IOS router secure VPN gateway The PIX Firewall and Cisco router, running Cisco VPN software, can interoperate to create a secure VPN gateway between networks.
- Cisco VPN Client to PIX Firewall via dialup The PIX Firewall can become a VPN endpoint for the VPN Client over a dialup network. The dialup network can consist of ISDN, public switched telephone network (PSTN) (analog modem), or digital subscriber line (DSL) communication channels.
- Cisco VPN Client to PIX Firewall via network—The PIX Firewall can become a VPN endpoint for the VPN Client over an IP network.
- Other vendor products to PIX Firewall—Products from other vendors can connect to the PIX Firewall if they conform to open VPN standards.



"Application-Aware" Inspection

- Protocols such as FTP, HTTP, H.323, and SQL*Net need to negotiate connections to dynamically assigned source or destination ports through the firewall.
- PIX Firewall inspects packets above the network layer.
- PIX Firewall securely opens and closes negotiated ports for legitimate client-server connections through the firewall.

CSPFA 3.2—3-16

Today many corporations use the Internet for business transactions. For the corporations to keep their internal networks secure from potential threats from the Internet, they can implement firewalls on their internal network. Even though these firewalls help protect a corporation's internal networks from external threats, firewalls have caused problems as well. For example, some of the protocols and applications that the corporations use to communicate are not allowed through the firewalls. Specifically, protocols need to negotiate FTP, HTTP, H.323, and SQL*Net connections to dynamically assigned source or destination ports, or IP addresses, through the firewall.

A good firewall has to inspect packets above the network layer and do the following as required by the protocol or application:

- Securely open and close negotiated ports or IP addresses for legitimate client-server connections through the firewall.
- Use NAT-relevant instances of an IP address inside a packet.
- Use PAT-relevant instances of ports inside a packet.
- Inspect packets for signs of malicious application misuse.

You can configure the Cisco PIX Firewall to allow the required protocols or applications through. This enables a corporation's internal networks to remain secure while still being able to continue day-to-day business over the Internet.

Web-Based Management Solutions

PIX Device Manager

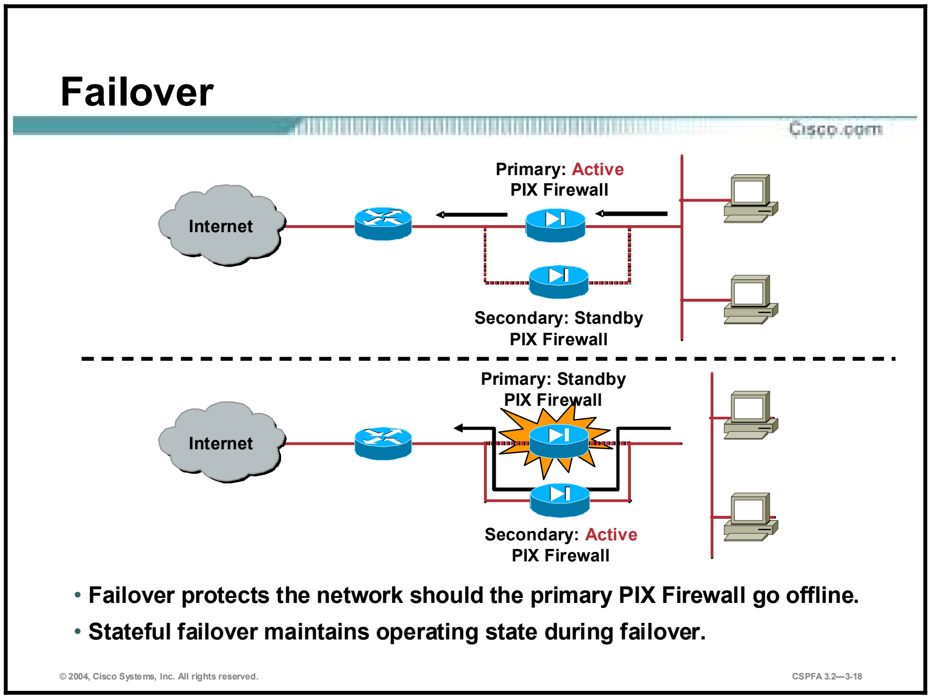
Firewall Management Center

CSPFA 3.2—3-17

PIX Device Manager and Firewall Management Center are browser-based configuration tools designed to help you set up, configure, and monitor your Cisco PIX Firewall graphically, without requiring an extensive knowledge of the PIX Firewall command line interface (CLI).

PDM monitors and configures a *single* PIX Firewall. You can use PDM to create a new configuration. You can use PDM to monitor and maintain current PIX Firewalls. You can point your browser to more than one PIX Firewall and administer several PIX Firewalls from a single workstation.

CiscoWorks 2000 Management Center for Firewalls (Firewall MC) is a web-based interface for configuring and managing *multiple* Cisco PIX Firewalls. Firewall MC has a look and feel similar to Cisco PIX Device Manager (PDM); however, with Firewall MC, you can configure multiple firewalls instead of configuring only one at a time. Firewall MC centralizes and accelerates the deployment and management of multiple PIX Firewalls.



Failover

- **Failover protects the network should the primary PIX Firewall go offline.**
- **Stateful failover maintains operating state during failover.**

CSPFA 3.2—3-18

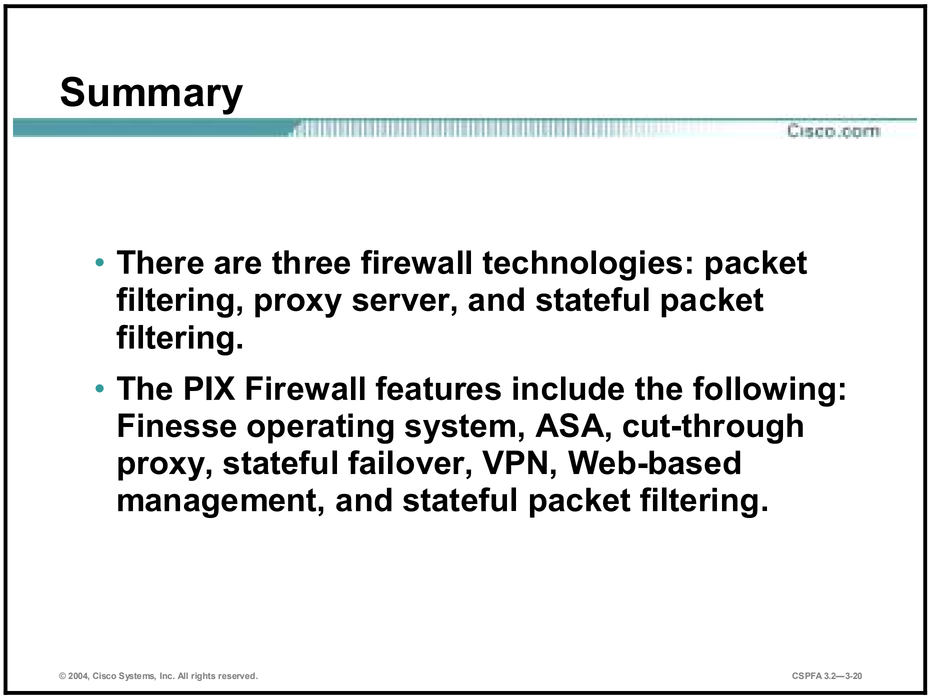
Failover provides a mechanism for the PIX Firewall to be redundant by allowing two identical firewalls, hardware and software, to serve the same functionality. The active firewall performs normal security functions, while the standby firewall monitors, ready to take control should the active firewall fail.

The PIX Firewall can use a serial cable for short-distance failover or an Ethernet cable for long-distance (LAN-based) failover. In both of these scenarios, the PIX Firewall can be configured for stateful failover so that active connections remain when failover occurs. When failover occurs, Syslog messages are generated indicating the cause of failure.

Note	PIX Firewall models that support failover include legacy models 515 and 520, which are not featured in this course, as well as current models 515E, 525, and 535.

Summary

This topic summarizes what you learned in this lesson.



Summary

- There are three firewall technologies: packet filtering, proxy server, and stateful packet filtering.
- The PIX Firewall features include the following: Finesse operating system, ASA, cut-through proxy, stateful failover, VPN, Web-based management, and stateful packet filtering.

Cisco.com

CSPFA 3.2—3-20

4

Cisco PIX Firewall Family

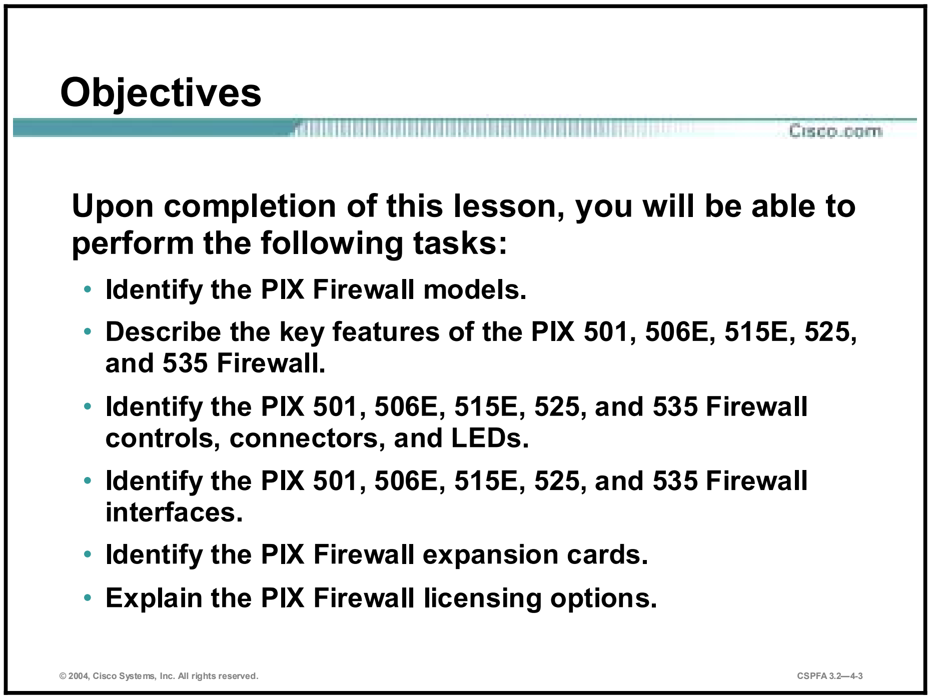
Overview

This lesson includes the following topics:

- Objectives
- PIX Firewall models
- PIX Firewall licensing
- Firewall Services Module
- Summary

Objectives

This topic lists the lesson's objectives:

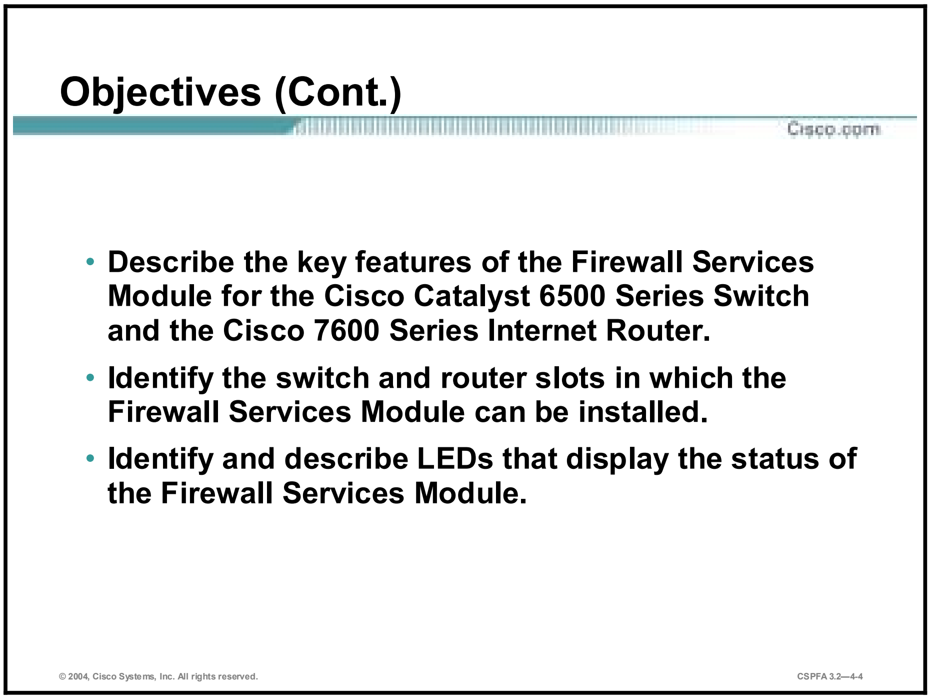


Objectives

Upon completion of this lesson, you will be able to perform the following tasks:

- **Identify the PIX Firewall models.**
- **Describe the key features of the PIX 501, 506E, 515E, 525, and 535 Firewall.**
- **Identify the PIX 501, 506E, 515E, 525, and 535 Firewall controls, connectors, and LEDs.**
- **Identify the PIX 501, 506E, 515E, 525, and 535 Firewall interfaces.**
- **Identify the PIX Firewall expansion cards.**
- **Explain the PIX Firewall licensing options.**

CSPFA 3.2—4-3



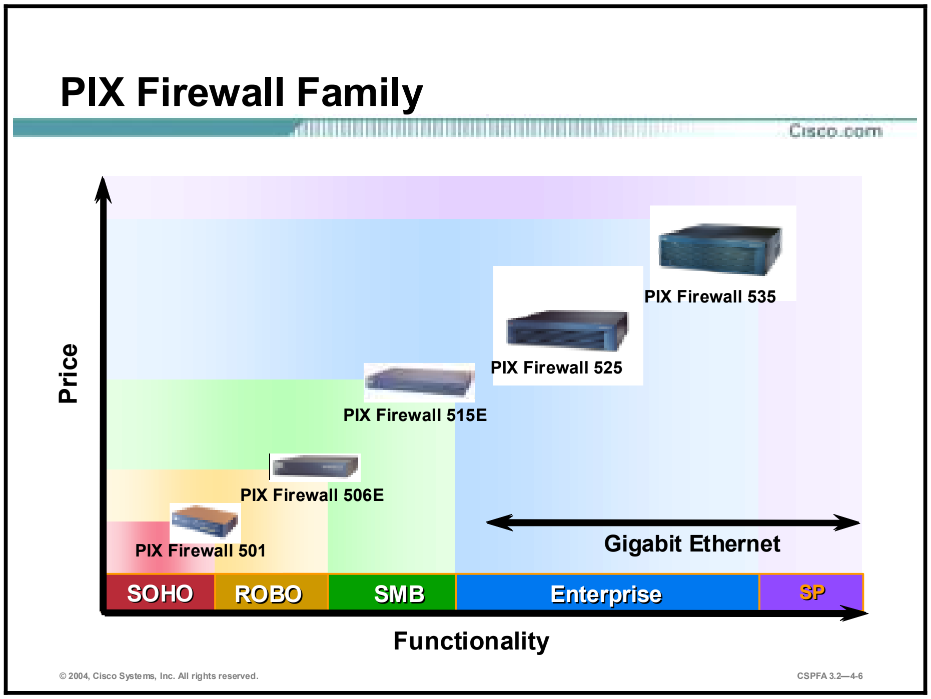
Objectives (Cont.)

- Describe the key features of the Firewall Services Module for the Cisco Catalyst 6500 Series Switch and the Cisco 7600 Series Internet Router.
- Identify the switch and router slots in which the Firewall Services Module can be installed.
- Identify and describe LEDs that display the status of the Firewall Services Module.

CSPFA 3.2—4-4

PIX Firewall Models

This topic describes the PIX Firewall 500 models.



PIX Firewall Family

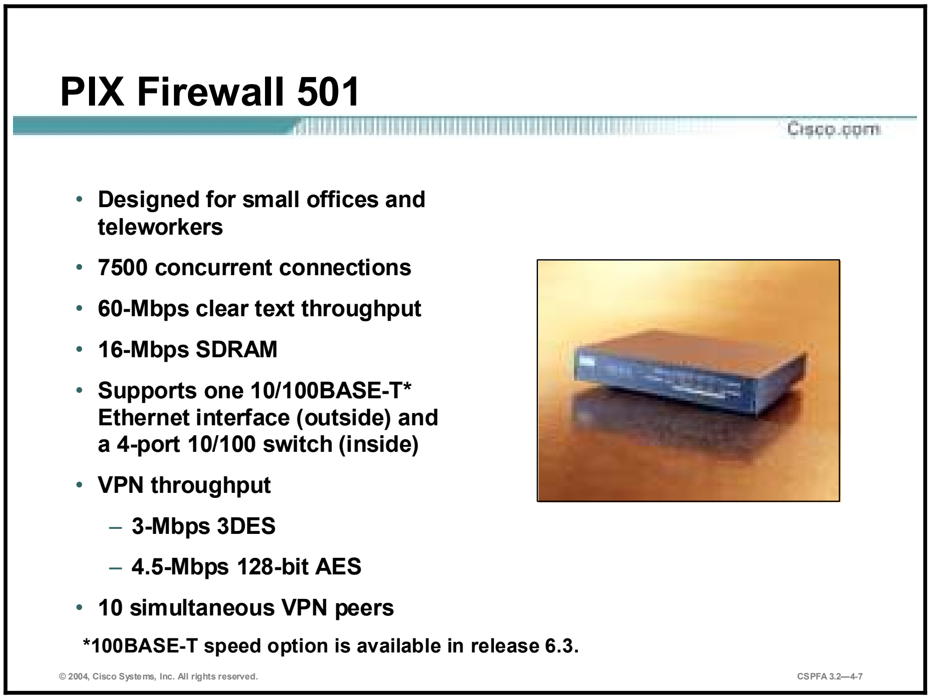
© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—4-6

The Cisco PIX Firewall 500 series scales to meet a range of requirements and network sizes, and currently consists of five models: the PIX Firewall 501, 506E, 515E, 525, and 535. The PIX Firewall 501 has an integrated 10/100BASE-T port (100BASE-T option available in release 6.3) and an integrated four-port 10/100 switch. The PIX Firewall 506E has dual integrated 10/100BASE-T ports (100BASE-T option available in release 6.3 for 506E only). The PIX Firewall 515E supports single-port or four-port 10/100 Ethernet cards. The PIX Firewall 525 supports single-port or four-port 10/100 Fast Ethernet and Gigabit Ethernet. The PIX Firewall 535 supports Fast Ethernet and Gigabit Ethernet. The PIX Firewall 515E, 525, and 535 models come with an integrated virtual private network (VPN) Accelerator (VAC) card.

The PIX Firewall is secure right out of the box. After a few installation procedures and an initial configuration of six general commands, your PIX Firewall is operational and protecting your network. These PIX Firewall commands enable connections from the inside interface access to the outside interface, and block all connections from the outside interface to the inside interface.

Note Prior to PIX Firewall Software Release 6.3, the PIX Firewall 501 outside interface and 506E outside and inside interfaces operated at 10BASE-T. With the upgrade to software release 6.3, the PIX Firewall 501 outside interface and 506E outside and inside interfaces can operate at 10/100BASE-T. To enable the speed change on the interface requires a software upgrade only.



PIX Firewall 501

- Designed for small offices and teleworkers
- 7500 concurrent connections
- 60-Mbps clear text throughput
- 16-Mbps SDRAM
- Supports one 10/100BASE-T* Ethernet interface (outside) and a 4-port 10/100 switch (inside)
- VPN throughput
  - 3-Mbps 3DES
  - 4.5-Mbps 128-bit AES
- 10 simultaneous VPN peers

*100BASE-T speed option is available in release 6.3.

CSPFA 3.2—4-7

The PIX Firewall 501 measures only 1.0 x 6.25 x 5.5 inches and weighs only 0.75 pounds, yet it delivers enterprise-class security for small offices and teleworkers. Ideal for securing high-speed, “always on” broadband environments, the PIX Firewall 501 provides small office networking features and powerful remote management capabilities in a compact, all-in-one solution.

The PIX Firewall 501 provides a convenient way for multiple computers to share a single broadband connection. In addition to its RS-232 (RJ-45) 9600 baud console port and its integrated 10/100BASE-T port (100BASE-T option available in release 6.3) for the outside interface, it features an integrated auto-sensing, auto-MDIX (medium-dependent interface crossover) four-port 10/100 switch for the inside interface. Auto-MDIX support eliminates the need to use crossover cables with devices connected to the switch.

The PIX Firewall 501 can also secure all network communications from remote offices to corporate networks across the Internet using its standards-based Internet Key Exchange (IKE)/IPSec VPN capabilities. Users can also enjoy plug-and-play networking by taking advantage of the built-in Dynamic Host Configuration Protocol (DHCP) server within the PIX Firewall, which automatically assigns network addresses to the computers when they are powered on.

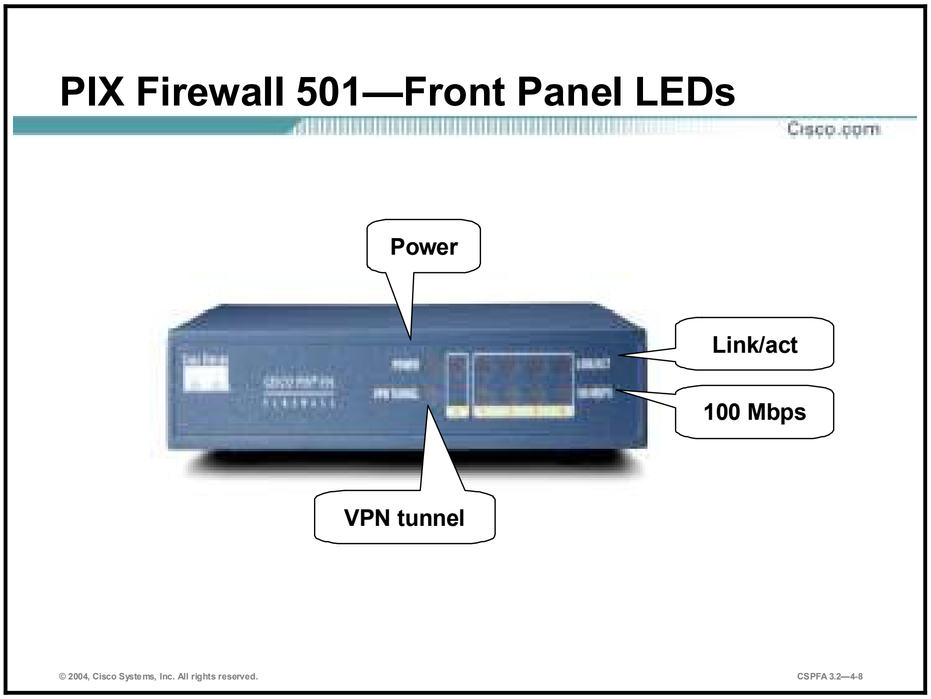
With PIX Firewall 501 Software Release 6.3, there are several product licensing options available. Choose an appropriate user license. Each user license supports a maximum number of concurrent source IP addresses from the internal network to traverse through the PIX Firewall 501. One can choose between a 10-user, 50-user, or unlimited user license. For VPN encryption, there are two options: Data Encryption Standard (DES), which supports 56-bit DES encryption, or Triple-DES (3DES), which supports both 168-bit 3DES and up to 256-bit Advanced Encryption Standard (AES) encryption. There is more on software licensing later in this lesson.

The PIX Firewall 501 comes with an integrated security lock slot for improved physical security and contains 8 MB of Flash memory.

Note The cable lock for the security lock slot is not provided with the firewall.

Note The Cisco PIX Firewall 501 requires software version 6.1(1) or higher.

Note Prior to release 6.3, the outside interface was a half-duplex 10BASE-T Ethernet interface. With release 6.3, the outside interface can be configured for half- or full-duplex and 10 or 100BASE-T. To enable this feature requires a software upgrade to release 6.3 only.



PIX Firewall 501—Front Panel LEDs

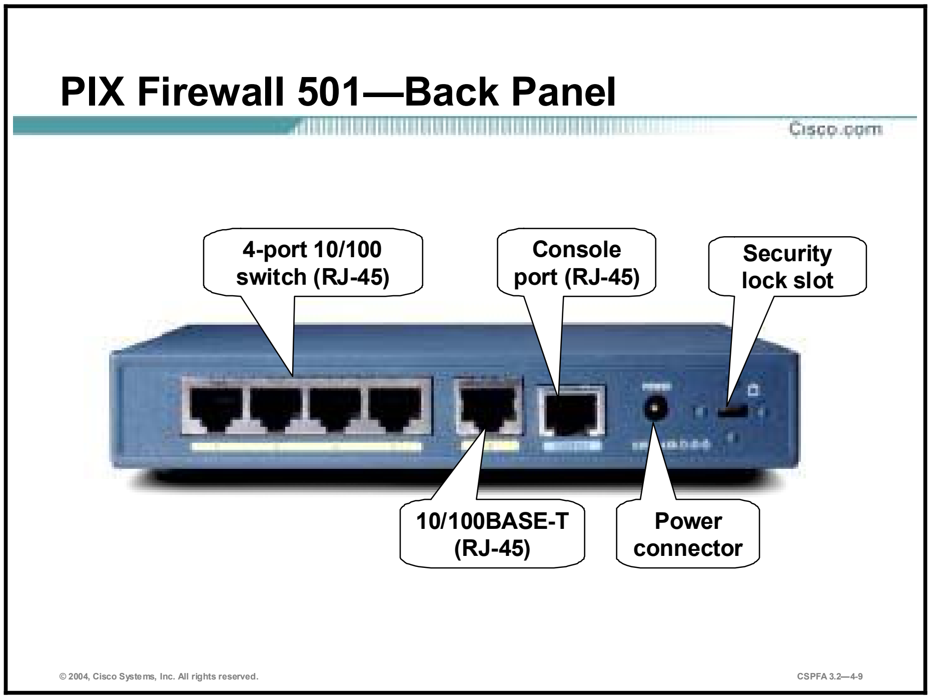
CSPFA 3.2—4-8

The following are the LEDs:

- Power—When the light is green, the device is powered on.
- Link/Act—When the light is flashing green, network activity (such as Internet access), is present. When the light is green, the correct cable is in use and the connected equipment has power and is operational. When the light is off, no link is established.
- VPN Tunnel—When the light is green, one or more IKE/IPSec VPN tunnels are established. When the light is off, one or more IKE/IPSec VPN tunnels are disabled. If the standard configuration has not been modified to support VPN tunnels, the LED does not light up because it is disabled by default.

Note The VPN Tunnel LED does not light up when Point-to-Point Tunneling Protocol/Layer 2 Tunneling Protocol (PPTP/L2TP) tunnels are established.

- 100 MBPS—When the light is green, the interface is enabled at 100 Mbps (autonegotiated). When the light is off, the interface is enabled at 10 Mbps.



PIX Firewall 501—Back Panel

Cisco.com

4-port 10/100 switch (RJ-45)

Console port (RJ-45)

Security lock slot

10/100BASE-T (RJ-45)

Power connector

© 2004, Cisco Systems, Inc. All rights reserved.

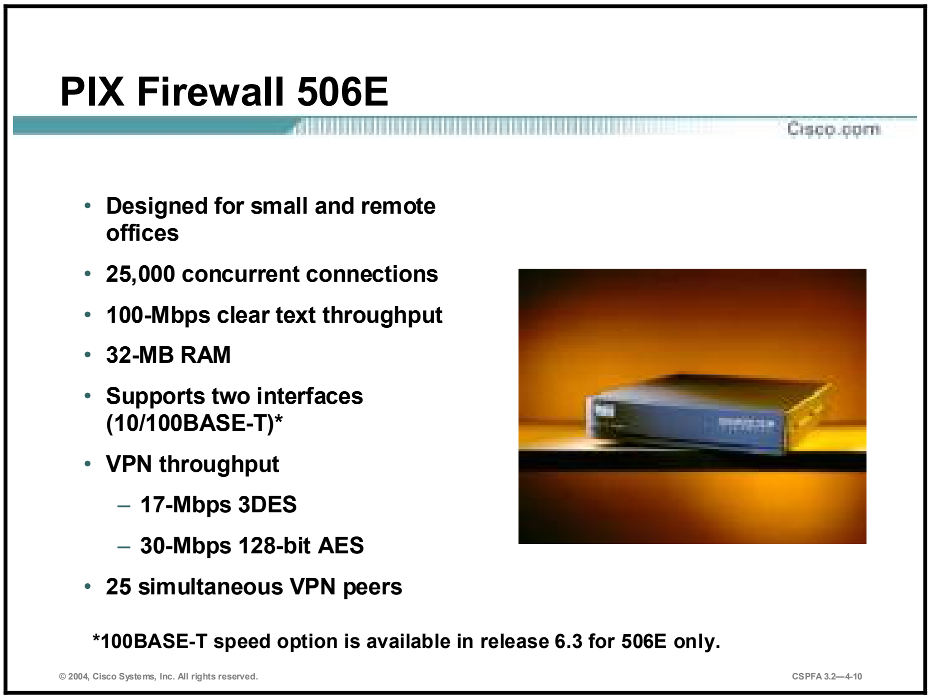
CSPFA 3.2—4-9

This figure shows the back panels of the PIX Firewall 501. The following are the PIX Firewall 501 features:

- 10/100 switch ports—Ports in the autosensing, auto-MDIX switch used for the inside interface. Connect your PC or other network devices to one of the four switched ports, which are numbered 1 through 4.
- 10/100BASE-T port (100BASE-T option available in release 6.3)—Port 0, a half- or full-duplex Ethernet port for the public network. The PIX Firewall 501 comes with a yellow Ethernet cable (72-1482-01) and an orange Ethernet cable (72-3515-01). Use the yellow cable to connect the device to a switch or hub. Use the orange cable to connect the device to a DSL modem, cable modem, or router.
- Console port—RS-232 (RJ-45) 9600-baud port used to connect a computer to the PIX Firewall for console operations.
- Power connector—Used to attach the power supply cable to the PIX Firewall. The PIX Firewall 501 does not have a power switch.
- Security lock slot—A slot that accepts standard desktop cable locks to provide physical security for small portable equipment, such as laptop computers.

Note When installing the PIX Firewall 501, place the chassis on a flat, stable surface. The chassis is not rack mountable.

Note Prior to release 6.3, the outside interface was a half-duplex 10BASE-T Ethernet interface. With release 6.3, the outside interface can be configured for half- or full-duplex and 10 or 100BASE-T. To enable this feature requires a software upgrade to release 6.3 only.



PIX Firewall 506E

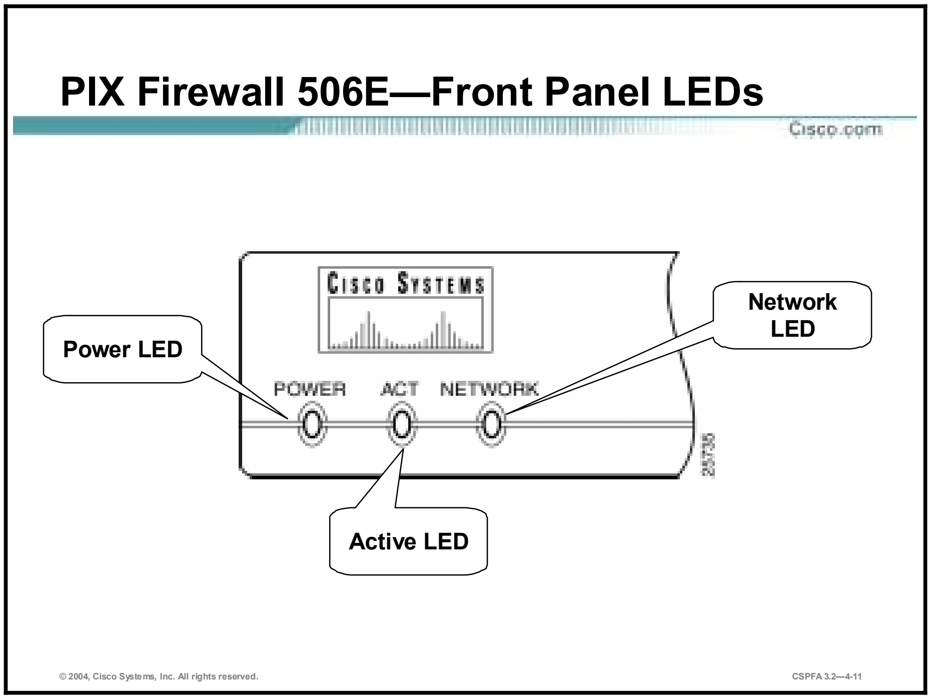
- Designed for small and remote offices
- 25,000 concurrent connections
- 100-Mbps clear text throughput
- 32-MB RAM
- Supports two interfaces (10/100BASE-T)*
- VPN throughput
  - 17-Mbps 3DES
  - 30-Mbps 128-bit AES
- 25 simultaneous VPN peers

*100BASE-T speed option is available in release 6.3 for 506E only.

CSPFA 3.2—4-10

The PIX Firewall 506E is designed for companies that are leveraging the cost advantages of the Internet and allowing employees to work remotely. It delivers full firewall protection, as well as IPSec VPN capabilities. The PIX Firewall 506E can connect with up to 25 VPN peers simultaneously, and provides users with a complete implementation of IPSec standards. It comes with 8 MB of Flash memory and 2 integrated 10/100BASE-T* ports, is compact in size (8 x 12 x 1.7 inches), and uses TFTP for image download and upgrade. PIX Firewall 506E license is provided in a single, unlimited user license. With PIX Firewall Software Release 6.3, there are two VPN encryption options: DES which supports 56-bit DES encryption or 3DES which supports both 168-bit 3DES and up to 256-bit AES encryption. There is more on software licensing later in this lesson.

Note 100BASE-T port speed is available starting in software release 6.3. Prior to release 6.3, the PIX 506E supported a port speed of 10BASE-T only. The 100BASE-T performance upgrade is software based. There is no PIX Firewall hardware upgrade necessary.

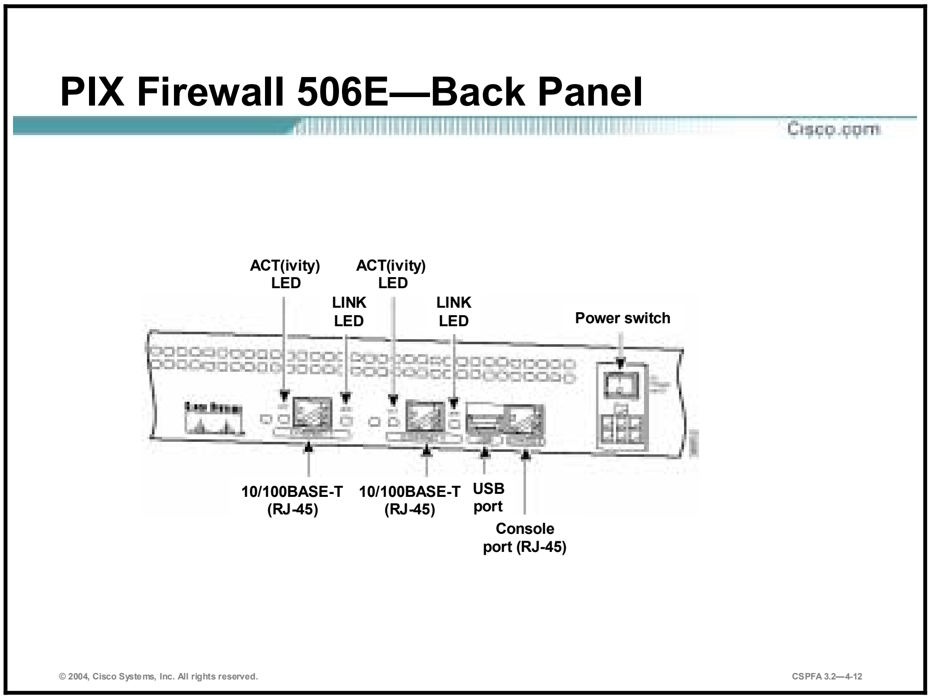


PIX Firewall 506E—Front Panel LEDs

CSPFA 3.2—4-11

The following are the LEDs:

- POWER—When the PIX Firewall has power, the light shines.
- ACT—When the software image has been loaded on the PIX Firewall 506E, the light shines.
- NETWORK—When at least one network interface is passing traffic, the light shines.



PIX Firewall 506E—Back Panel

Cisco.com

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—4-12

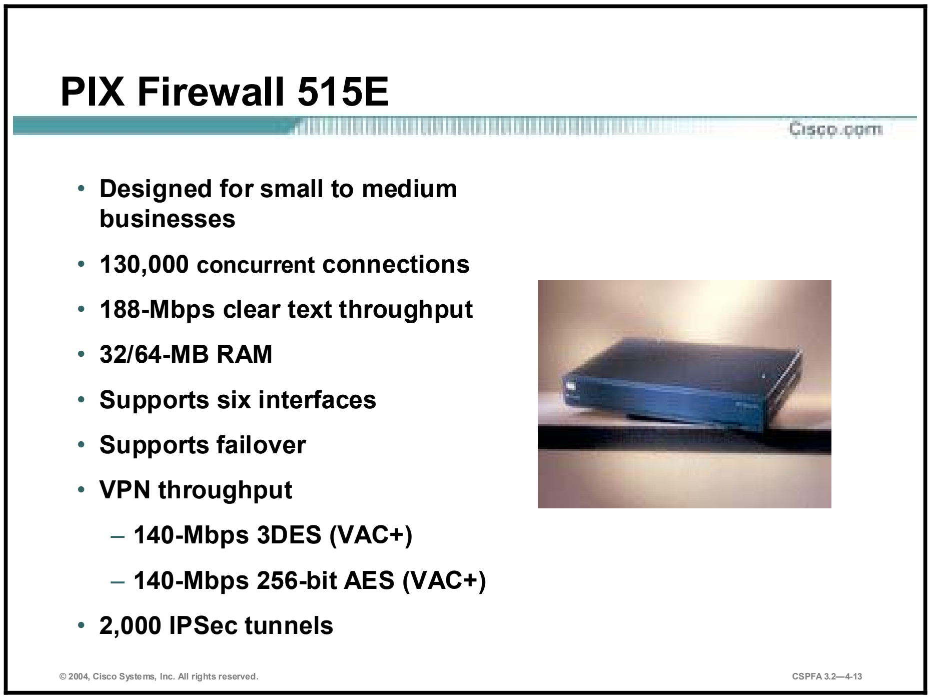
On the PIX Firewall 506E, Ethernet 1 connects the inside network and Ethernet 0 is for the outside network. Use the console port to connect a computer to enter configuration commands. The USB port to the left of the console port is not used.

The power connection is directly beneath the power switch. The PIX Firewall 506E uses an external AC to DC power supply.

The LEDs display the following transmission states:

- ACT—Shows network activity.
- LINK—Shows that data is passing on the network to which the connector is attached.

Note 100BASE-T port speed is available starting in software release 6.3. Prior to release 6.3, the PIX 506E supported a port speed of 10BASE-T only. The 100BASE-T performance upgrade is software based. There is no PIX Firewall hardware upgrade necessary.



PIX Firewall 515E

- Designed for small to medium businesses
- 130,000 concurrent connections
- 188-Mbps clear text throughput
- 32/64-MB RAM
- Supports six interfaces
- Supports failover
- VPN throughput
  - 140-Mbps 3DES (VAC+)
  - 140-Mbps 256-bit AES (VAC+)
- 2,000 IPSec tunnels

CSPFA 3.2—4-13

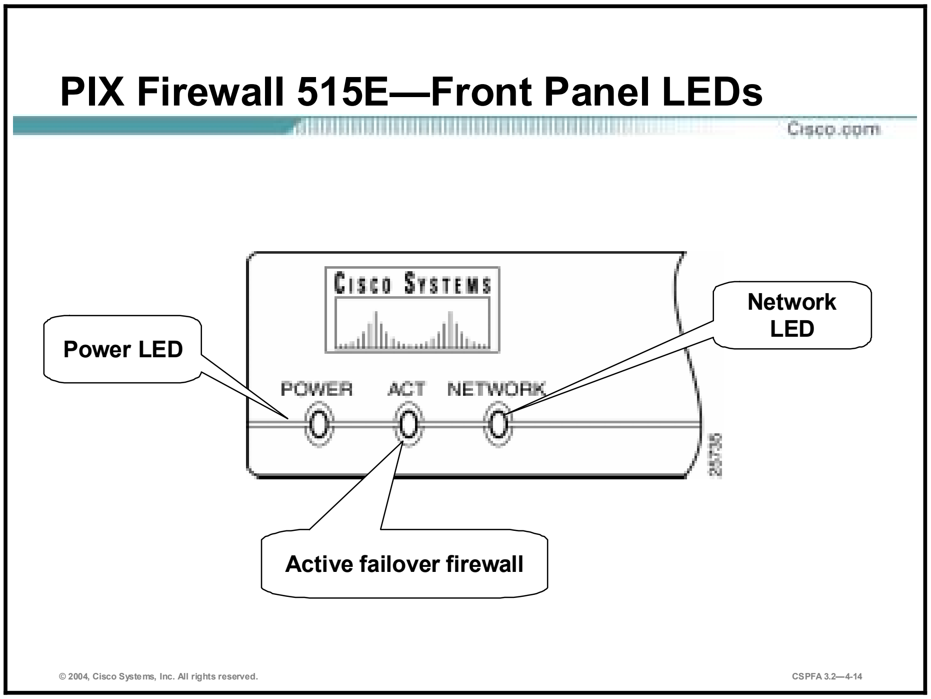
The PIX Firewall 515E is designed for small- and medium-sized businesses. It delivers full firewall protection, as well as IPSec VPN capabilities with complete implementation of IPSec standards. You can create and terminate VPN tunnels between two PIX Firewalls, between a PIX Firewall and any Cisco VPN-enabled router, and between a PIX Firewall and the Cisco (VPN) Client. The PIX Firewall 515E is also ideal for remote sites that require only two-way communication with their corporate network.

The PIX Firewall 515E supports up to six 10/100 Ethernet ports. This allows for more robust traffic configurations as well as establishing a protected DMZ for hosting a web site or performing URL filtering and virus detection. With the restricted license, it supports three interfaces; with the unrestricted license (UR), it supports six interfaces.

This model also features integrated hardware-based IPSec acceleration, delivering VPN performance of up to 140 Mbps while freeing system resources for other mission-critical security functions. IPSec acceleration is provided by an integrated PIX Firewall VPN Accelerator Plus card (VAC+), or the PIX Firewall VAC. There is more information on the VAC and VAC+ cards later in this lesson.

The PIX Firewall 515E is rack-mountable, comes with 16 MB of Flash memory, and uses TFTP for image download and upgrade.

Note When a PIX Firewall 515E is ordered, the order automatically includes a VAC+ unless specified otherwise.

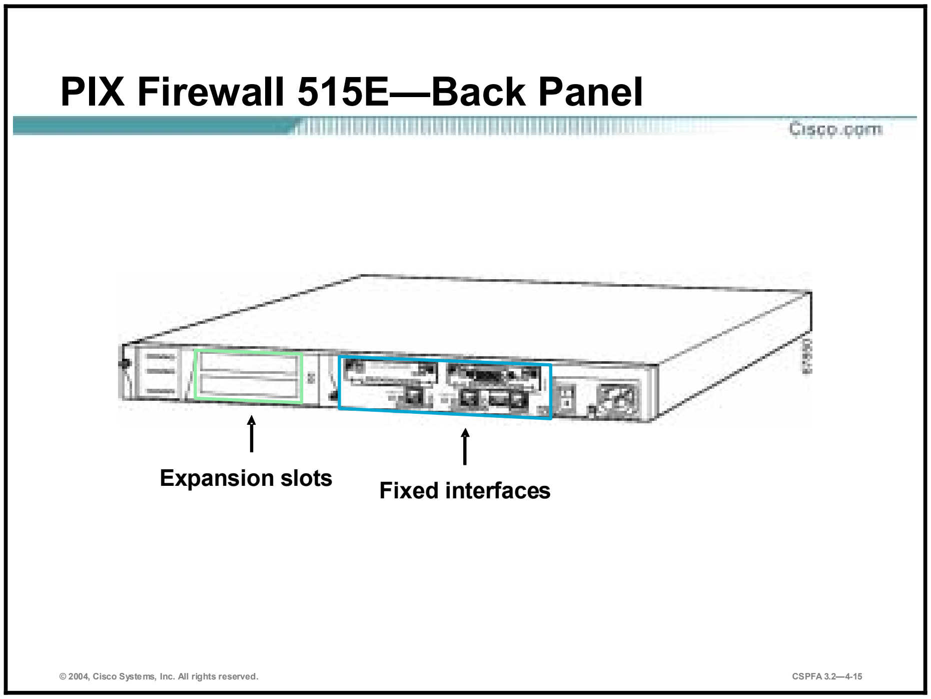


PIX Firewall 515E—Front Panel LEDs

CSPFA 3.2—4-14

The following are the LEDs:

- Power—When the PIX Firewall has power, the light shines.
- ACT—When the PIX Firewall is used in a standalone configuration, the light shines. When the PIX Firewall is configured for failover operations, the light shines on the active PIX Firewall.
- Network—The light shines when at least one network interface is passing traffic.

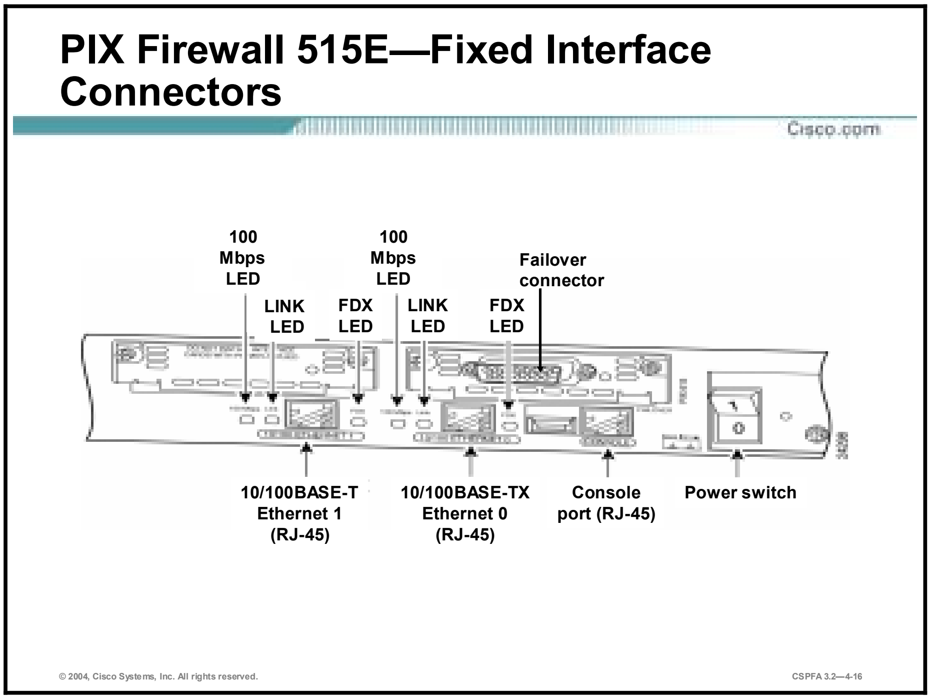


PIX Firewall 515E—Back Panel

Cisco.com

CSPFA 3.2—4-15

The PIX Firewall 515E back plane can be logically divided into two sections, fixed interfaces and expansion slots. The fixed interfaces provide two 10/100BASE-TX Ethernet ports, a console port and a failover connector. The expansion slots provide two 32-bit/33-MHz PCI slots. These PCI slots support easy installation of additional network interfaces and VAC or VAC+. There is more on the fixed interfaces and expansion slot option cards later in this lesson.



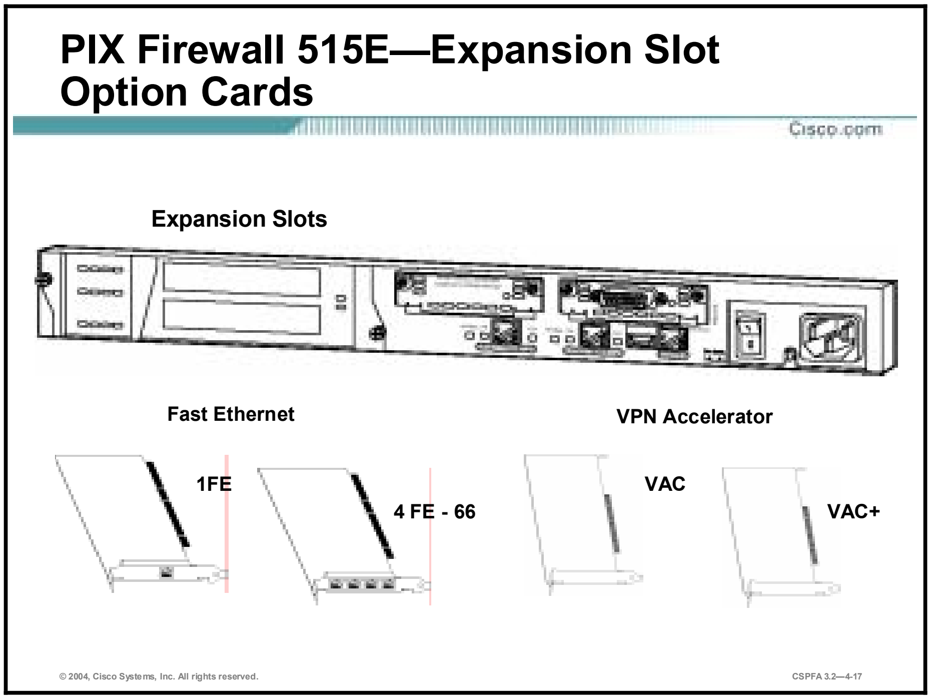
PIX Firewall 515E—Fixed Interface Connectors

CSPFA 3.2—4-16

This figure shows the fixed interfaces of the PIX Firewall 515E. The following lists the PIX Firewall 515E features:

- Ethernet connections—With software versions 5.2 and higher, any port, whether fixed or a PCI expansion port, and any interface type, can be assigned to be the inside or outside network port.
- Console port—Used to connect a computer to the PIX Firewall for console operations.
- Failover connection—Used to attach a failover cable between two PIX Firewalls.
- 100 Mbps LED—100 Mbps, 100BASE-TX communication for the respective connector. If the light is off, the PIX Firewall 515E uses 10 Mbps data exchange.
- LINK LED—Indicates that data is passing on the network to which the connector is attached.
- FDX LED—Indicates that the connection uses full duplex data exchange (data can be transmitted and received simultaneously). If the light is off, half duplex is in effect.
- Power switch—Controls the power to the PIX Firewall.

Note The USB port to the left of the console port and the detachable plate above the Ethernet 1 connector are for future PIX Firewall enhancements.



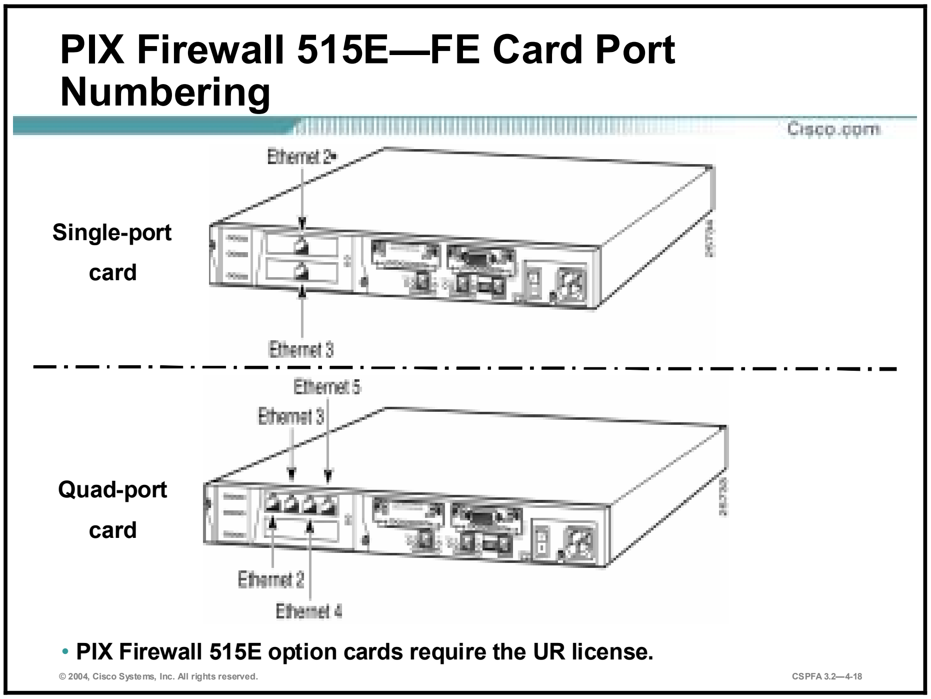
PIX Firewall 515E—Expansion Slot Option Cards

Cisco.com

CSPFA 3.2—4-17

The two expansion slots support Fast Ethernet expansion option cards and Hardware VPN Accelerator cards. The features of both cards are as follows:

- Fast Ethernet expansion option cards—Support the easy installation of additional network interfaces. The Fast Ethernet expansion option cards include single-port and four-port Fast Internet cards. Four-port Fast Ethernet interface card delivers increased port density per PCI slot. With the restricted license, the PIX Firewall 515E supports one additional expansion network port. With the restricted license, the PIX Firewall 515E supports up to four additional expansion network ports.
- Hardware VPN Acceleration option cards—Deliver high performance VPN services via support of VAC and VAC+. The hardware-based VAC and VAC+ cards handle the voluminous mathematical functions required for IPSec. Offloading encryption functions to the VAC and VAC+ cards improves IPSec encryption processing. The VAC card provides 56-bit DES and 168-bit 3DES encryption. The VAC card has a 32-bit, 33-MHz PCI interface. The VAC+ card, in addition to supporting DES and 3DES, also provides 128-, 192-, and 256-bit AES encryption. The VAC+ card has a 64-bit, 66-MHz PCI interface. The VAC+ card is supported in Cisco PIX Software Release 6.3(1) or later. VAC and VAC+ cards are limited to one per 515E, 525, and 535 chassis.



PIX Firewall 515E—FE Card Port Numbering

Single-port card

Quad-port card

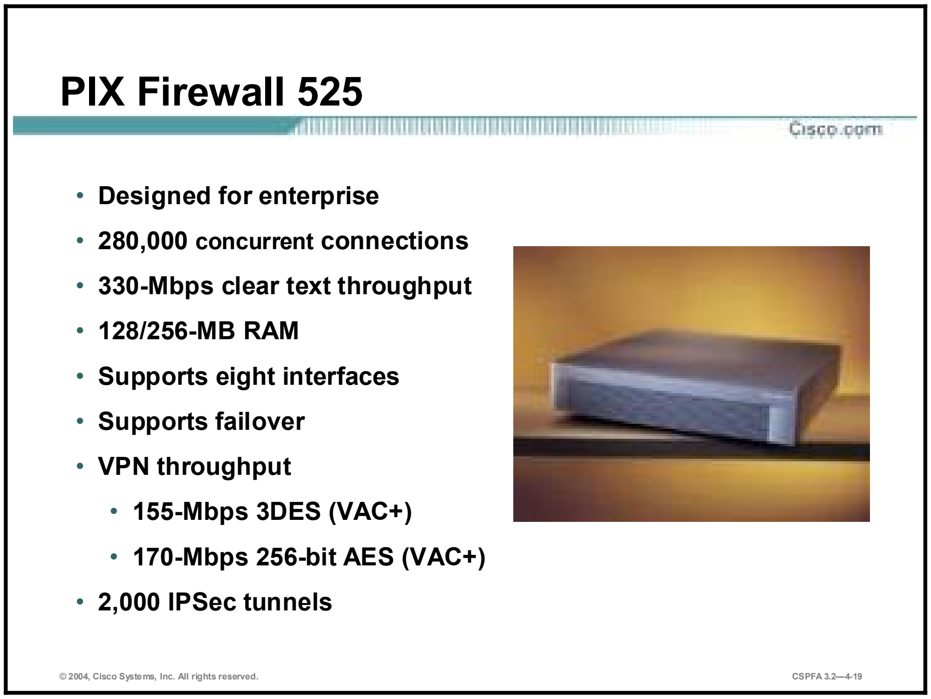
- PIX Firewall 515E option cards require the UR license.

CSPFA 3.2—4-18

If your PIX Firewall has one or two single-port Ethernet cards installed in the auxiliary assembly on the left of the PIX Firewall at the rear, the cards are numbered top to bottom so that the top card is Ethernet 2 and the bottom card is Ethernet 3.

The quad port card is a four-port Ethernet card. When you connect the perimeter network cables to this card, you begin with the far left connector and move to the right. For example, Ethernet 2 will go in the far left connector, Ethernet 3 in the second connector from the left, and so on.

Note The maximum number of interfaces allowed is six. Additional interfaces are not recognized.



PIX Firewall 525

Cisco.com

- Designed for enterprise
- 280,000 concurrent connections
- 330-Mbps clear text throughput
- 128/256-MB RAM
- Supports eight interfaces
- Supports failover
- VPN throughput
  - 155-Mbps 3DES (VAC+)
  - 170-Mbps 256-bit AES (VAC+)
- 2,000 IPSec tunnels

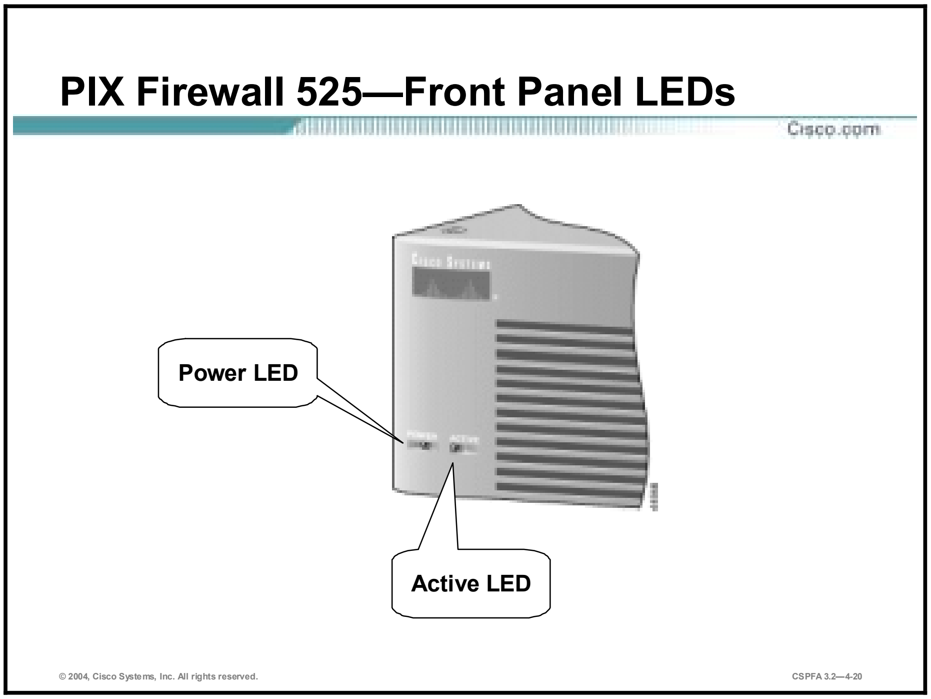
CSPFA 3.2—4-19

The PIX Firewall 525 is intended for enterprise and service provider use. Ideal for protecting the enterprise headquarters' perimeter, the PIX Firewall 525 delivers full firewall protection, as well as IPSec VPN capabilities.

The PIX Firewall 525 supports a broad range of network interface cards. Standard cards include single-port or four-port 10/100 Fast Ethernet and Gigabit Ethernet (with UR license). With the restricted license, it supports six interfaces; with the UR license, it supports eight interfaces.

The PIX Firewall 525 also offers multiple power supply options. You can choose between AC and a 48 DC power supply. Either option can be paired with a second power supply for redundancy and high availability.

Note	Currently, a VAC+ card is included with every PIX Firewall 525 ordered unless otherwise specified.



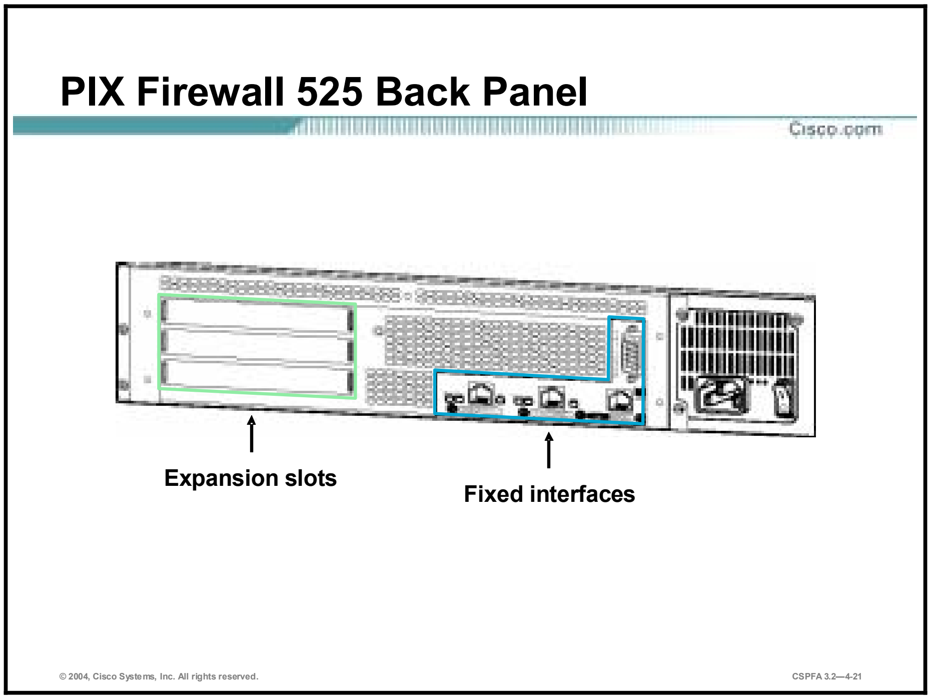
PIX Firewall 525—Front Panel LEDs

Cisco.com

CSPFA 3.2—4-20

There are two LEDs on the front panel of the PIX Firewall 525. The LEDs function as follows:

- POWER—On when the firewall has power.
- ACT—On when the firewall is the active failover firewall. If failover is present, the light is on when the firewall is the active firewall, and off when the firewall is in standby mode.



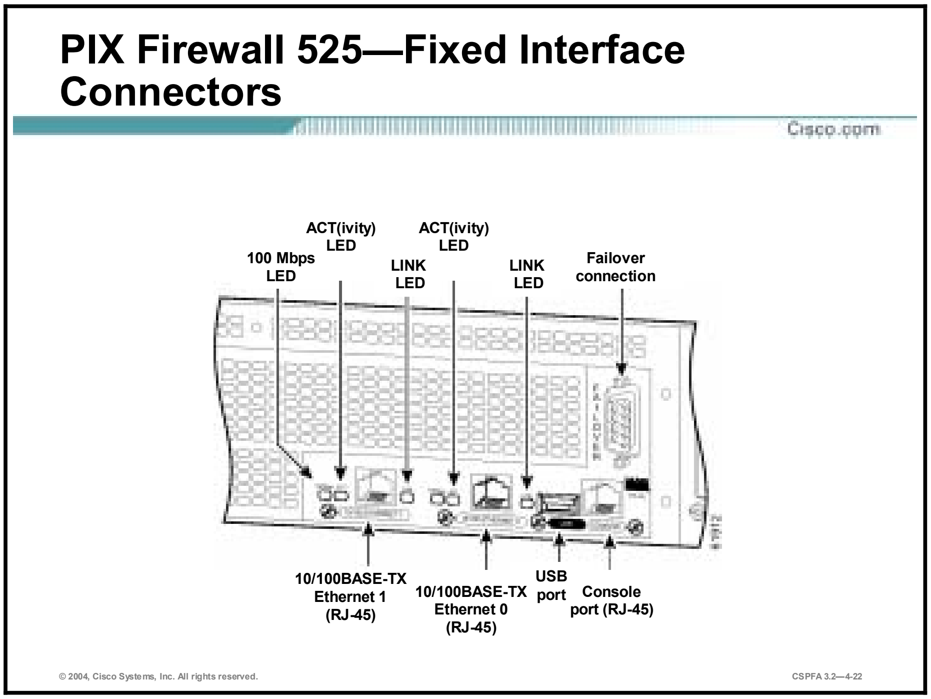
PIX Firewall 525 Back Panel

Expansion slots

Fixed interfaces

CSPFA 3.2—4-21

The PIX Firewall 525 back plane can be logically divided into two sections, fixed interfaces and expansion slots. The fixed interfaces provide two 10/100BASE-TX Ethernet ports, a console port and a failover connector. The expansion slots provide three 32-bit/33-MHz PCI slots. These PCI slots support easy installation of additional network interfaces and VAC or VAC+. There is more on the fixed interfaces and expansion slot option cards later in this lesson.



PIX Firewall 525—Fixed Interface Connectors

CSPFA 3.2—4-22

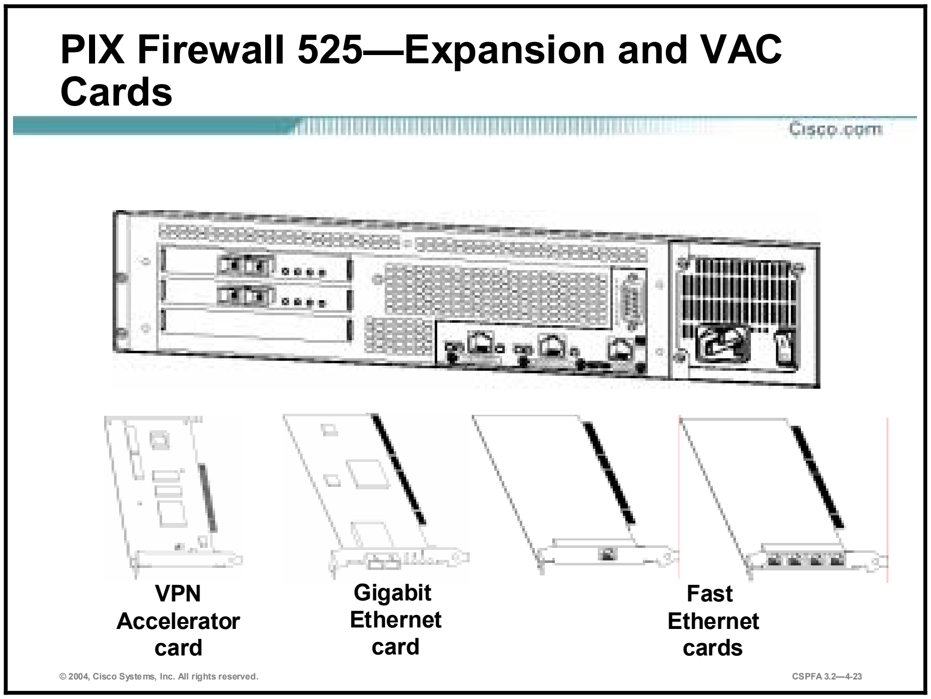
On the back of the PIX Firewall 525, there are three LEDs for each RJ-45 interface port and three types of fixed interface connectors. The LEDs display the following transmission states:

- 100 Mbps—100-Mbps, 100BASE-TX communication. If the light is off during network activity, that port is using 10 Mbps data exchange.
- ACT—Shows network activity.
- LINK—Shows that data is passing through that interface.

The following are fixed connectors on the back of the PIX Firewall 525:

- RJ-45—Network and console connectors.
- DB-15—Failover cable connector.
- USB—Not used at the present time.

The inside, outside, or perimeter network connections can be made to any available interface port on the PIX Firewall 525. If you are only using the Ethernet 0 and Ethernet 1 ports, connect the inside network cable to the interface connector marked Ethernet 0 or Ethernet 1. Connect the outside network cable to the remaining Ethernet port.



PIX Firewall 525—Expansion and VAC Cards

Cisco.com

VPN Accelerator card

Gigabit Ethernet card

Fast Ethernet cards

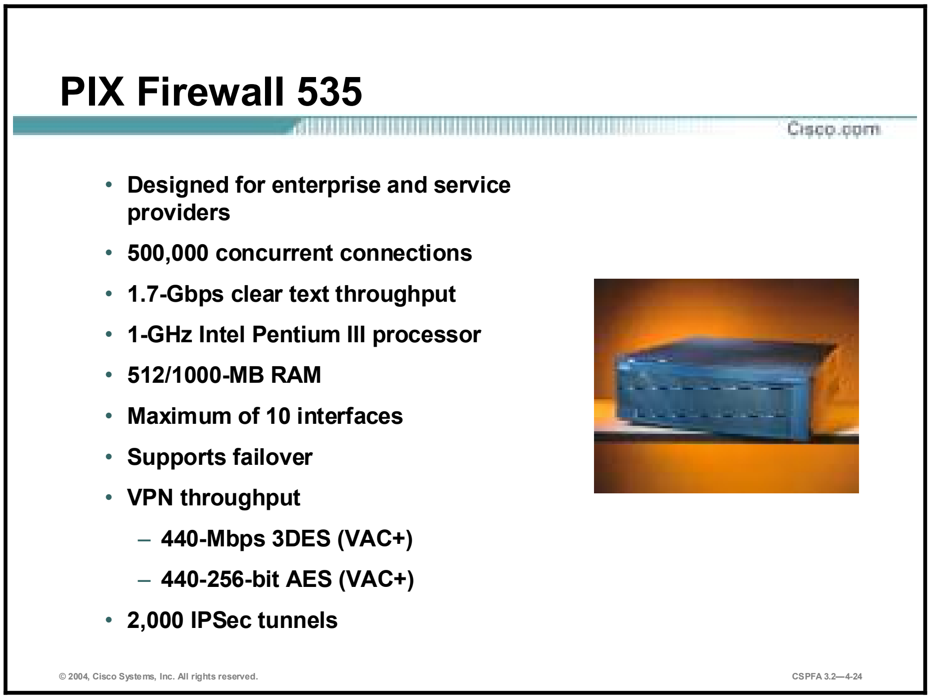
© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—4-23

The PIX Firewall 525 supports easy installation of additional network interfaces via three PCI expansion slots. It supports expansions cards including single-port Fast Ethernet card, four-port Fast Ethernet card, single-port Gigabit Ethernet card and hardware VPN Accelerator cards, VAC and VAC+ cards.

A maximum of six interfaces are supported with a Restricted license, and a maximum of eight interfaces are possible with the Unrestricted license. Currently, a VAC+ card is included with every PIX 525 ordered unless otherwise specified.

When connecting the network cables to the expansion interface ports, use the following guidelines: the first expansion port number, at the top left, is interface 2. Starting from that port and going from left to right and top to bottom, the next port is interface 3, the next is interface 4, and so on.



PIX Firewall 535

Cisco.com

- Designed for enterprise and service providers
- 500,000 concurrent connections
- 1.7-Gbps clear text throughput
- 1-GHz Intel Pentium III processor
- 512/1000-MB RAM
- Maximum of 10 interfaces
- Supports failover
- VPN throughput
  - 440-Mbps 3DES (VAC+)
  - 440-256-bit AES (VAC+)
- 2,000 IPSec tunnels

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—4-24

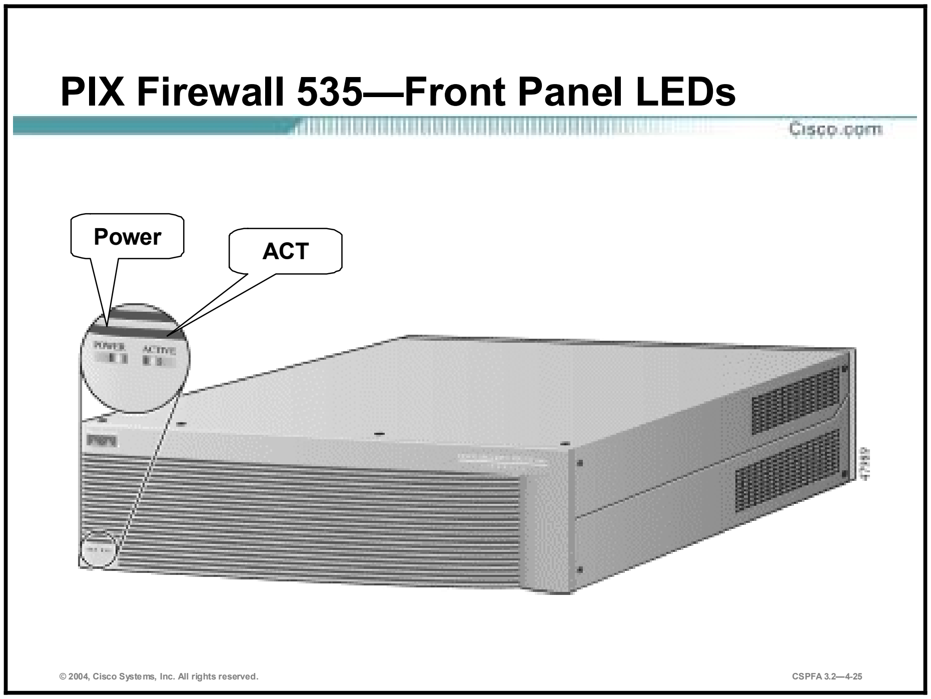
The PIX Firewall 535 is intended for enterprise and service provider use. It has a throughput of 1.7 Gbps with the ability to handle up to 500,000 concurrent connections. Supporting both site-to-site and remote access VPN applications via 56-bit DES, 168-bit 3DES, or AES, the PIX Firewall 535 can deliver 440 Mbps of 3DES throughput with a VAC+ card and 2,000 IPSec tunnels.

The PIX Firewall 535 supports both Fast Ethernet and Gigabit Ethernet interfaces, and comes with an integrated VPN Accelerator card.

Note If, after configuring a PIX Firewall for Gigabit Ethernet cards, you replace the cards with 10/100 Ethernet cards, the order of the cards in the configuration changes from what you originally configured. For example, if you configure ethernet0 for a Gigabit Ethernet card assigned to the inside interface and replace this card with a 10/100 Ethernet card, the card may no longer appear as ethernet0.

The PIX Firewall 535 comes with 16 MB of Flash memory and supports the PIX Firewall software version 5.3 or later.

Note Currently, a VAC+ card is included with every PIX 535 ordered unless otherwise specified.



PIX Firewall 535—Front Panel LEDs

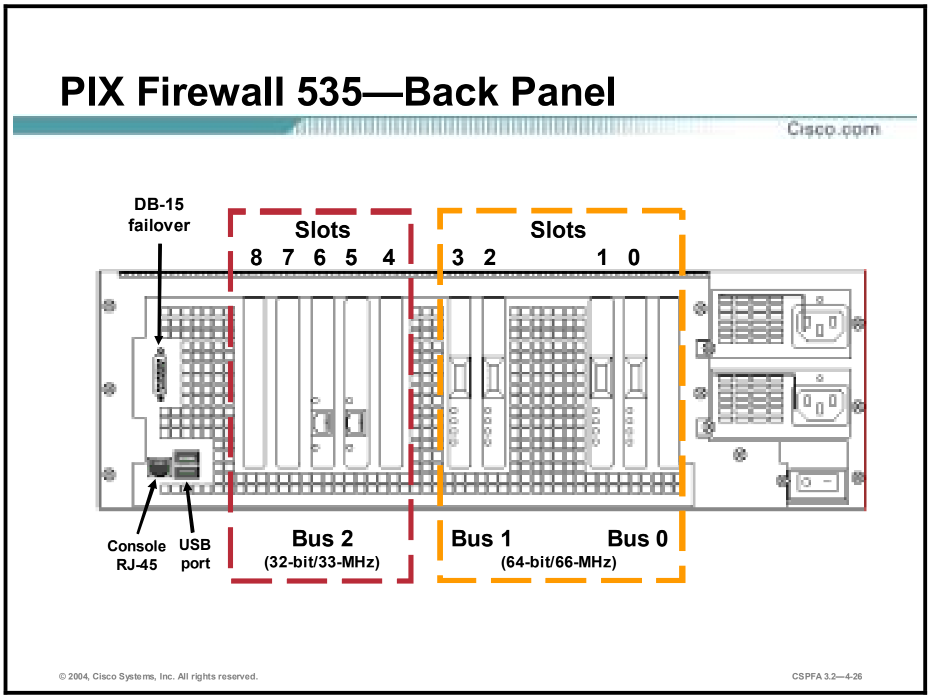
Cisco.com

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—4-25

There are two LEDs on the front panel of the PIX Firewall 535. The LEDs function as follows:

- Power—On when the PIX Firewall has power.
- ACT—On when the PIX Firewall is the active failover firewall. If failover is present, the light is on when the PIX Firewall is the active firewall and off when the PIX Firewall is in standby mode.



PIX Firewall 535—Back Panel

CSPFA 3.2—4-26

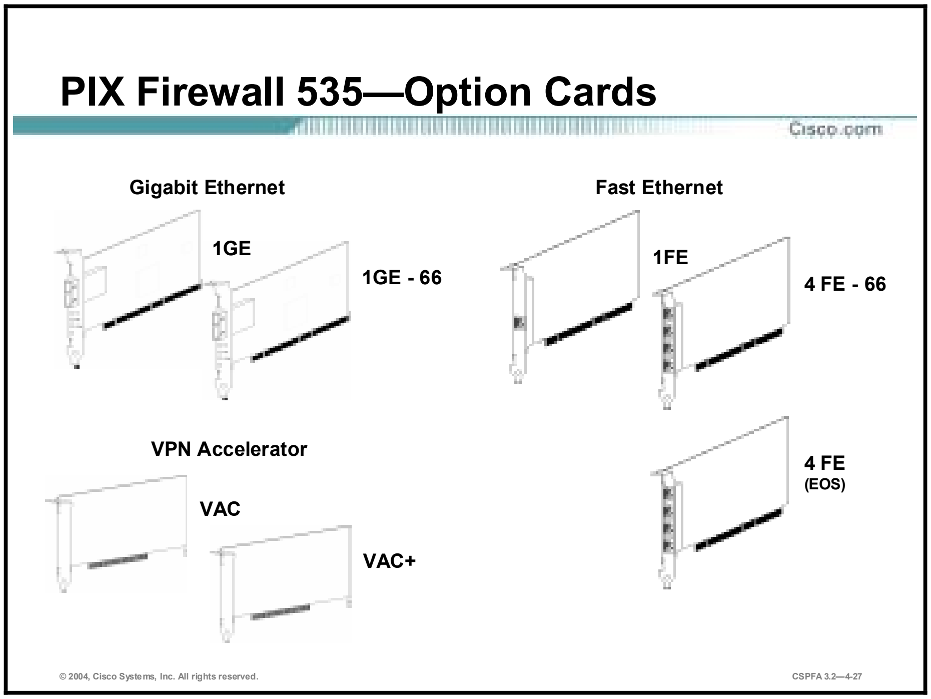
There are three separate buses for the nine interface slots in the PIX Firewall 535. The figure is a reference for the interface slot configuration on the PIX Firewall 535.

The slots and buses are configured as follows:

- Slots 0 and 1—64-bit, 66-MHz bus 0
- Slots 2 and 3—64-bit, 66-MHz bus 1
- Slots 4 through 8—32-bit, 33-MHz bus 2

The PIX Firewall 535 expansion port function is dependent on three things: PIX Firewall software license, expansion card PCI bus interface, and PIX Firewall 535 slot bus speed. A description of each item is as follows:

- Software license—A maximum of eight interfaces are supported with a Restricted license, and a maximum of ten interfaces are possible with the Unrestricted license.
- Expansion Card PCI bus interface—An expansion card ships with either a 32-bit, 33-MHz or 64-bit, 66-MHz PCI interface. The 64-bit, 66-MHz PCI bus interface card delivers potentially higher throughput. The 64-bit, 66-MHz PCI bus interface cards are backward compatible with the PIX Firewall 32-bit, 33-MHz expansion slots.
- Slot bus speed—A PIX Firewall 535 has two different bus configurations. Bus 0 and bus 1 are configured for 64-bit, 66-MHz. Bus 2 is configured for 32-bit, 33-MHz.



PIX Firewall 535—Option Cards

Cisco.com

Gigabit Ethernet

1GE

1GE - 66

Fast Ethernet

1FE

4 FE - 66

VPN Accelerator

VAC

VAC+

4 FE
(EOS)

CSPFA 3.2—4-27

There are three separate types of options cards available for the PIX Firewall 535, Gigabit Ethernet (1GE), single- (1FE) and four-port (4FE) Fast Ethernet, and VPN Accelerator cards (VAC and VAC+). Notice that for most card types, there is a 33-MHz and a 66-MHz version. For example, the 1GE card has a 33-MHz PCI interface. The 1GE-66 card has a 66-MHz PCI interface. There are also the nine interface slots and three buses in the PIX 535.

The slots and buses are configured as follows:

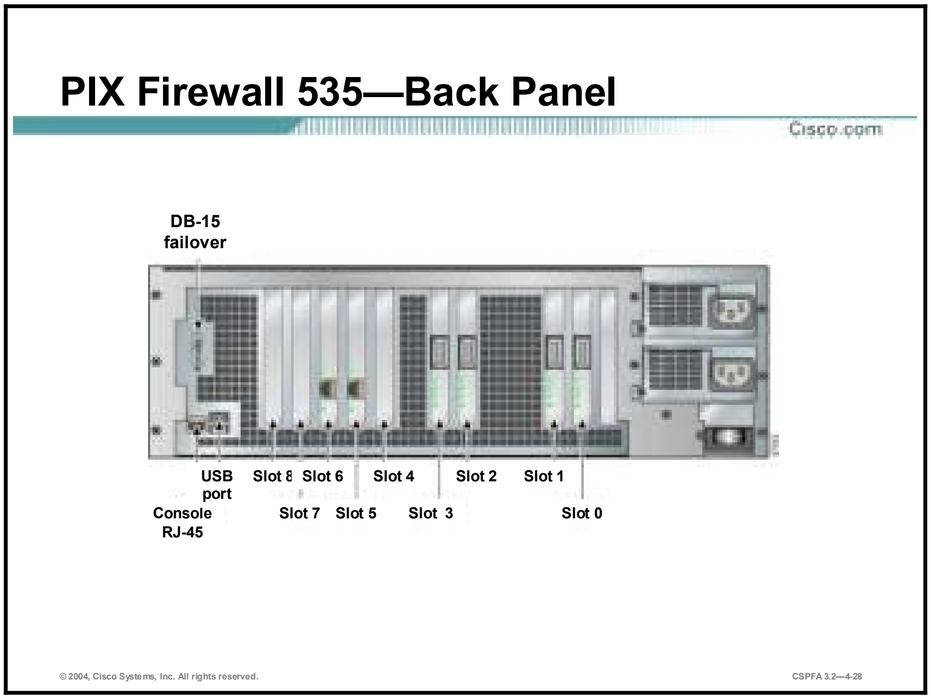
- Slots 0 and 1–64-bit/66-MHz bus 0
- Slots 2 and 3–64-bit/66-MHz bus 1
- Slots 4 to 8–32-bit/33-MHz bus 2

For optimum performance and throughput for the interface circuit boards, use the following guidelines:

- A total of eight interfaces are configurable on the PIX 535 with the restricted license, and a total of ten are configurable with the unrestricted license.
- For best performance, the 1GE-66, 4FE-66, and VAC+ (66 MHz) circuit boards should be installed in a 64-bit/66-MHz card slot. Performance will be degraded if this recommendation is not followed.
- The 1GE, 1FE, 4FE, and VAC (33 MHz) circuit boards should be installed in the 32-bit/33-MHz card slots.

Note The 1GE circuit board is not recommended for use in the PIX 535, because it can severely degrade performance. It is capable of only half the throughput of the 1GE-66 circuit board. If this circuit board is detected in the PIX 535, a warning about degraded performance will be issued.

- The 4FE card (end of sale—July 2003) can be installed only in a 32-bit/33-MHz card slot and must never be installed in a 64-bit/66-MHz card slot. Installation of this circuit board in a 64-bit/66-MHz card slot can cause the system to hang at boot time.
- The 1FE circuit board (33 MHz) can be installed in any bus or slot (32-bit/33-MHz or 64-bit/66-MHz). Up to nine 1FE circuit boards or up to two 4FE circuit boards can be installed. The 1FE circuit boards should be installed in the 32-bit/33-MHz card slots first.
- Do not mix the 1FE circuit boards with the 1GE-66 circuit boards on the same 64-bit/66-MHz bus (Bus 0 or bus 1). The overall speed of the bus is reduced by the lower-speed circuit board.
- If stateful failover is enabled for 1GE-66 traffic, the failover link must be PIX-1GE-66. The amount of stateful failover information is proportional to the amount of traffic flowing through the PIX Firewall and, if it is not configured properly, loss of state information or 256-byte block depletion can occur.



PIX Firewall 535—Back Panel

CSPFA 3.2—4-28

Depending upon the type of interface, there are four possible LEDs for each network interface port. The LEDs for the network interface ports display the following transmission states:

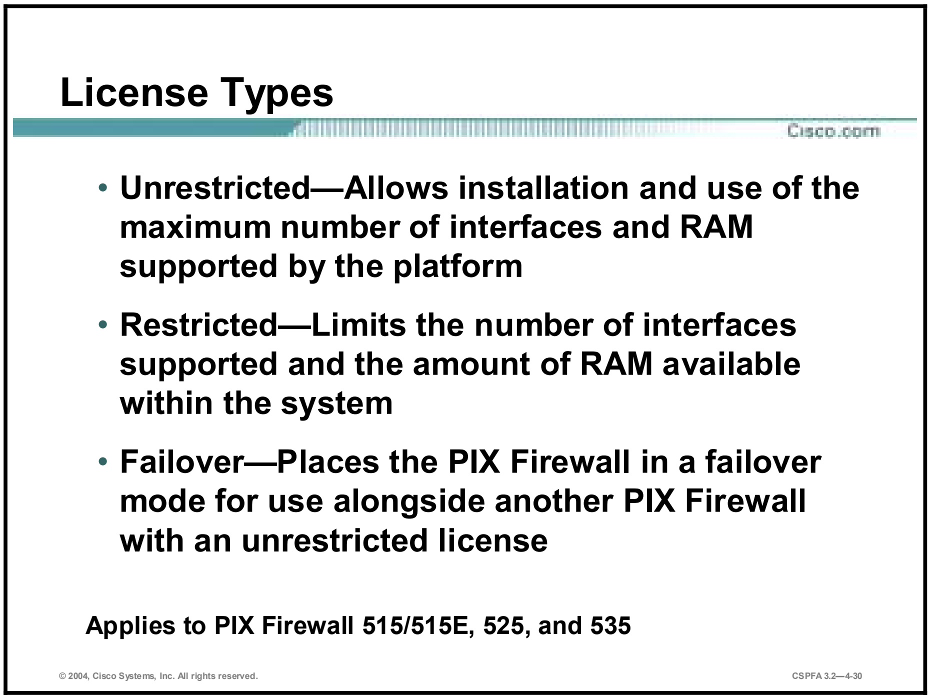
- 100 Mbps—100-Mbps 100BASE-TX communication. If the light is off during network activity, that port is using 10-Mbps data exchange.
- ACT—Shows network activity.
- LINK—Shows that data is passing through that interface.
- FDX—Shows that the connection uses full-duplex data exchange where data can be transmitted and received simultaneously. If this light is off, half-duplex is in effect.

When connecting the inside, outside, or perimeter network cables to the interface ports on the PIX Firewall 535, starting from the right and moving left, the connectors are Ethernet 0, Ethernet 1, Ethernet 2, and so forth.

Note The PIX Firewall 535 is equipped with hot-swappable power supplies. Should a power supply fail, you can remove the power supply without powering off the PIX Firewall 535.

PIX Firewall Licensing

This topic explains the licensing options for the PIX Firewall.



License Types

- **Unrestricted—Allows installation and use of the maximum number of interfaces and RAM supported by the platform**
- **Restricted—Limits the number of interfaces supported and the amount of RAM available within the system**
- **Failover—Places the PIX Firewall in a failover mode for use alongside another PIX Firewall with an unrestricted license**

Applies to PIX Firewall 515/515E, 525, and 535

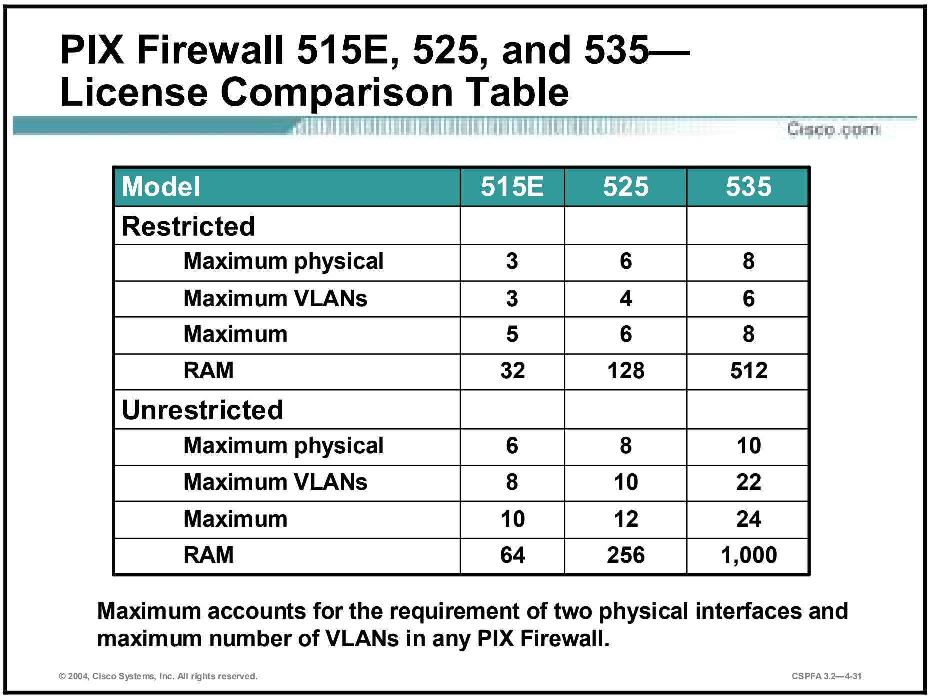
CSPFA 3.2—4-30

Current Cisco PIX Firewall licensing is based on a “feature-based” license key system. The PIX Firewall license determines the level of service it provides, its functions in a network, and the maximum number of interfaces and memory it can support. For the PIX Firewall family, the following licensing is available:

- PIX Firewall 501—Provided with a 10-user, 50-user, or unlimited user licenses in PIX Firewall Software Release 6.3. Each license allows up to a specified number of concurrent source IP addresses from your internal network to traverse the firewall. For instance, the 50-user license allows up to 50 concurrent source IP addresses from your internal network to traverse the firewall. If a PIX Firewall 501 requires more concurrent users to traverse the Firewall, the following upgrade user licenses are available: 10-user to 50-user, 10-user to unlimited, and 50-user to unlimited licenses.
- PIX Firewall 506E—Provided in a single, unlimited-user license.
- PIX Firewall 515E, 525, and 535 models—Available with the following basic license types:
  - Unrestricted—PIX Firewall platforms in an UR license mode allow installation and use of the maximum number of interfaces and RAM supported by the platform. The Unrestricted license supports failover.
  - Restricted—PIX Firewall platforms in a restricted (R) license mode limit the number of interfaces supported and the amount of RAM available within the system. A restricted licensed firewall does not support a redundant system for failover configurations.
  - Failover—The failover (FO) software license places the PIX Firewall in a failover mode for use alongside another PIX Firewall with a UR license.

Cisco supplies an activation key with a license. The activation key is based on the type of license and the serial number of the PIX Firewall. To enable the license features, enter the activation key into the PIX Firewall configuration and then reboot the PIX Firewall. Upon reboot, the new license features should take effect.

Note An activation key is "tied" to a specific PIX Firewall, such as PIX Firewall-serial number 12345678. An activation key is not specific to a particular PIX Firewall software version.



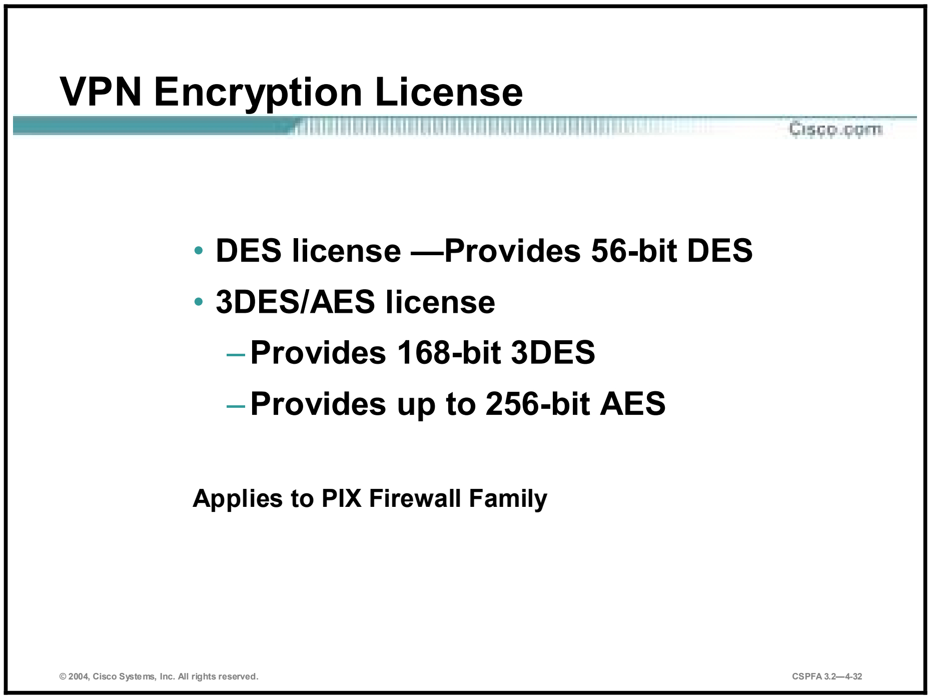
PIX Firewall 515E, 525, and 535—License Comparison Table

Model	515E	525	535
Restricted			
Maximum physical	3	6	8
Maximum VLANs	3	4	6
Maximum	5	6	8
RAM	32	128	512
Unrestricted			
Maximum physical	6	8	10
Maximum VLANs	8	10	22
Maximum	10	12	24
RAM	64	256	1,000

Maximum accounts for the requirement of two physical interfaces and maximum number of VLANs in any PIX Firewall.

CSPFA 3.2—4-31

The table in the figure compares the restricted and unrestricted licenses of the PIX Firewall 515E, 525, and 535 models. Across the top of the chart is the PIX Firewall model, 515E, 525, and 535. Down the left side are the restricted and unrestricted features. Each PIX Firewall column compares the listed features available with each license. For each license type, the table provides the maximum number of physical interfaces, the maximum number of VLANS, the maximum number of interfaces, and the RAM size.



VPN Encryption License

- DES license —Provides 56-bit DES
- 3DES/AES license
  - Provides 168-bit 3DES
  - Provides up to 256-bit AES

Applies to PIX Firewall Family

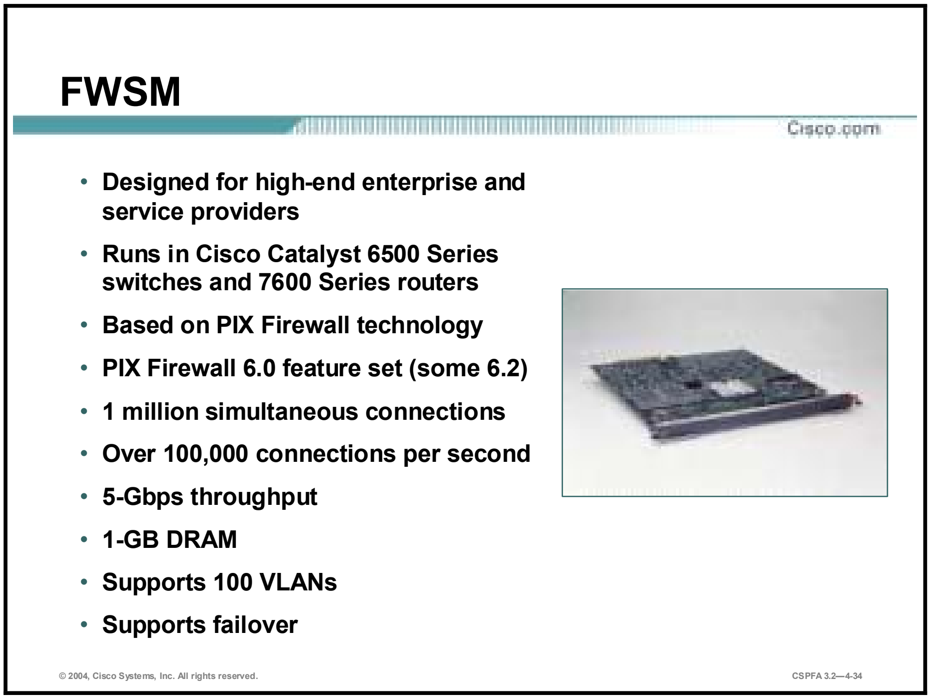
CSPFA 3.2—4-32

In addition to upgrading the PIX Firewall license, you may wish to add data encryption services, or increase the level of data encryption your PIX Firewall can provide. You can fill out an online form at the PIX Firewall Software page on Cisco.com to obtain a free 56-bit DES key. There is a separate form to install or upgrade to 168-bit 3DES encryption. For failover configurations, the UR and FO firewalls each require their own unique corresponding DES or 3DES license for failover functionality.

Adding cryptographic services and upgrading your PIX Firewall license both require obtaining and installing an activation key. Log on to Cisco.com for current information on obtaining activation keys.

Firewall Services Module

This topic describes the Firewall Services Module (FWSM) for the Cisco Catalyst 6500 Switch and Cisco 7600 Series Internet Router.



FWSM

- Designed for high-end enterprise and service providers
- Runs in Cisco Catalyst 6500 Series switches and 7600 Series routers
- Based on PIX Firewall technology
- PIX Firewall 6.0 feature set (some 6.2)
- 1 million simultaneous connections
- Over 100,000 connections per second
- 5-Gbps throughput
- 1-GB DRAM
- Supports 100 VLANs
- Supports failover

CSPFA 3.2—4-34

The Firewall Services Module (FWSM) is a multigigabit integrated firewall module for the Cisco Catalyst 6500 Series switch and the Cisco 7600 Series Internet router. It is fabric-enabled and capable of interacting with the bus and the switch fabric. Based on PIX Firewall technology, FWSM provides stateful firewall functionality in these switches and routers.

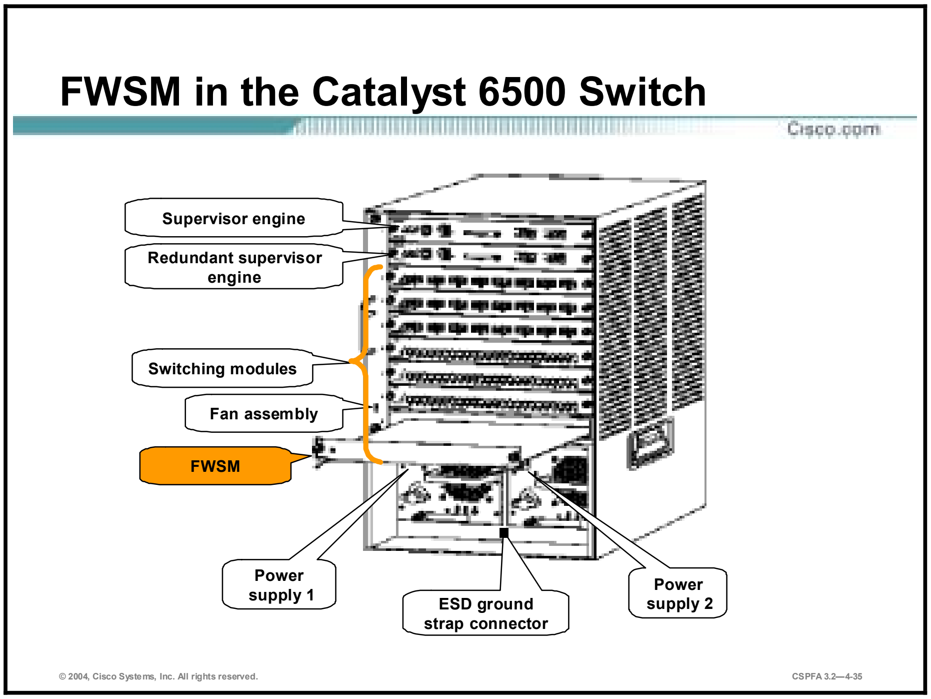
The following are the key features of FWSM:

- High-performance, 5-Gbps throughput, full-duplex firewall functionality
- Includes entire PIX Firewall 6.0 software feature set and the following PIX Firewall Software Release 6.2 features:
  - Command authorization
  - Object grouping
  - Internet Locator Service (ILS)/NetMeeting fix-up
  - URL filtering enhancement
- 3 million pps throughput
- Support for 100 VLANs
- 1 million concurrent connections
- LAN failover—Active or standby, and inter chassis or intra chassis
- Dynamic routing with Open Shortest Path First (OSPF) and passive Routing Information Protocol (RIP)
- Supports multiple modules per chassis

The following table shows the major differences between the PIX Firewall and FWSM.

	FWSM	PIX Firewall
Interfaces supported	100 (via VLANs)	10
Failover license	Not required	Required
VPN functionality	Not present	Present
Routing	Supports dynamic routing via OSPF and passive RIP	Supports static routing and passive RIP

The FWSM comes with 128 MB of CompactFlash memory and 1 GB of DRAM memory. Memory is not field upgradable.



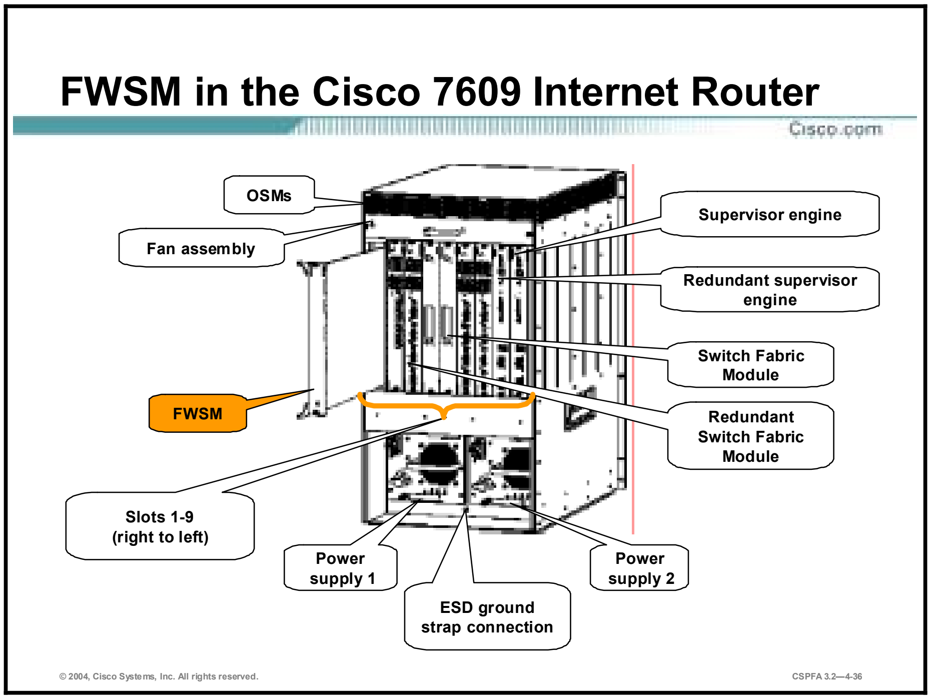
FWSM in the Catalyst 6500 Switch

CSPFA 3.2—4-35

The figure shows the FWSM installed in slot 9 of a Catalyst 6500 9-slot chassis. Slot 1 is reserved for the Supervisor Engine. The supervisor engine is the control module that defines and drives all operational capabilities of the switch. Slot 2 can contain an additional redundant Supervisor Engine in the event the supervisor engine in slot 1 fails. If a redundant Supervisor Engine is not used, slot 2 is available for switching modules.

A Switch Fabric Module can be installed to work with the Supervisor Engine 2 to deliver an increase in available system bandwidth. Using a Switch Fabric Module restricts the slot in which the FWSM can be installed by taking up an available slot. If you install a Switch Fabric Module, it must be installed in slot 5 of the Catalyst 6500 Switch. For redundancy, you can install an additional Switch Fabric Module in slot 6. If you do not install a redundant Switch Fabric Module, slot 6 is available for other supported modules, such as the FWSM. If you do not install any Switch Fabric Modules, slot 5 and slot 6 are available for the FWSM.

Note Detailed instructions on installing a Catalyst 6500 Series switch are provided in *Cisco Advanced Catalyst 6500 Switched Networks* (CACSN) v1.1. The course introduces the details of the Cisco Catalyst 6500 Series switches, including a detailed overview of many of the modules and components.



FWSM in the Cisco 7609 Internet Router

Cisco.com
OSMs
Fan assembly
FWSM
Slots 1-9 (right to left)
Power supply 1
ESD ground strap connection
Power supply 2
Supervisor engine
Redundant supervisor engine
Switch Fabric Module
Redundant Switch Fabric Module

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—4-36

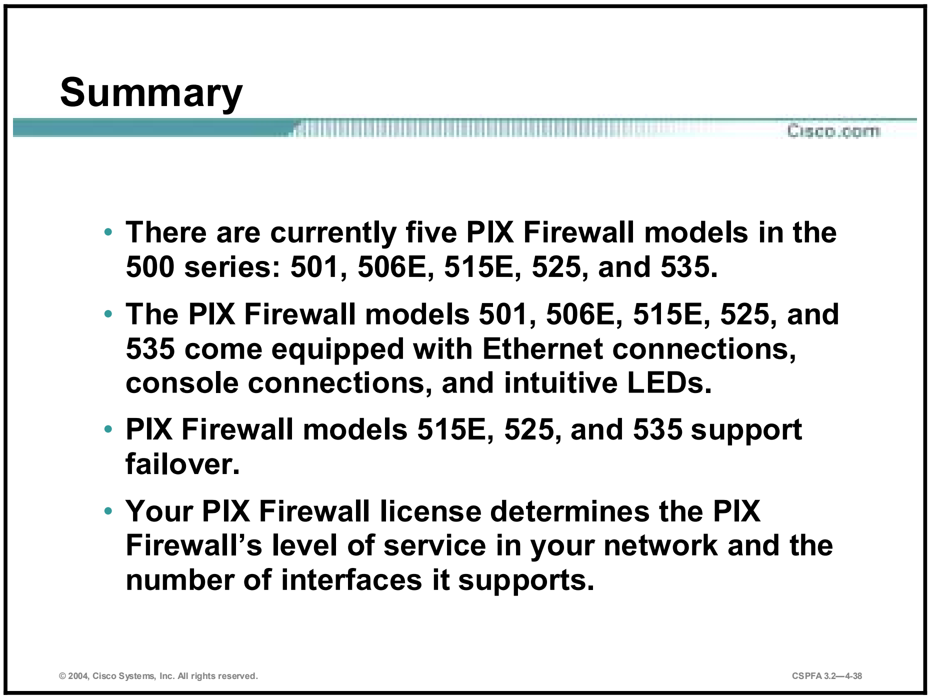
The figure shows the FWSM installed in slot 9 of a Cisco 7609 Internet Router. As with the Catalyst 6500 switch, the Supervisor Engine must be installed in slot 1. A redundant Supervisor Engine can be installed in slot 2. If a redundant Supervisor Engine 2 module is not installed, slot 2 is available for Optical Services Modules (OSMs) or other supported modules, such as the FWSM.

A Switch Fabric Module can be installed to work with the Supervisor Engine 2 module to deliver an increase in available system bandwidth The use of a Switch Fabric Module restricts the slot in which the FWSM can be installed by taking up a slot. If a Switch Fabric Module is installed, it must be installed in slot 5 of the Cisco 7606 or the 7609 Internet Router. For redundancy, you can install an additional Switch Fabric Module in slot 6. If you do not use a redundant Switch Fabric Module, slot 6 is available for OSMs or other supported modules, such as the FWSM. If you do not install any Switch Fabric Modules, slot 5 and slot 6 are available for OSMs or the FWSM.

Note The OSMs provide high-speed, high-density WAN connectivity. These modules allow service providers to increase network bandwidth and performance while offering a wide range of IP services.

Summary

This topic summarizes what you learned in this lesson.

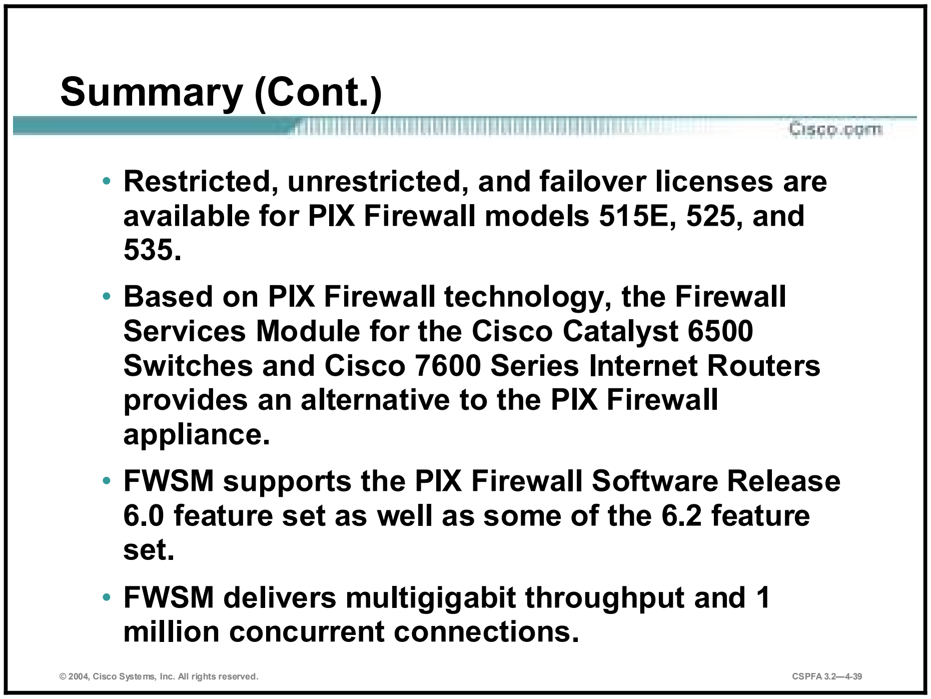


Summary

Cisco.com

- There are currently five PIX Firewall models in the 500 series: 501, 506E, 515E, 525, and 535.
- The PIX Firewall models 501, 506E, 515E, 525, and 535 come equipped with Ethernet connections, console connections, and intuitive LEDs.
- PIX Firewall models 515E, 525, and 535 support failover.
- Your PIX Firewall license determines the PIX Firewall’s level of service in your network and the number of interfaces it supports.

CSPFA 3.2—4-38



Summary (Cont.)

Cisco.com

- Restricted, unrestricted, and failover licenses are available for PIX Firewall models 515E, 525, and 535.
- Based on PIX Firewall technology, the Firewall Services Module for the Cisco Catalyst 6500 Switches and Cisco 7600 Series Internet Routers provides an alternative to the PIX Firewall appliance.
- FWSM supports the PIX Firewall Software Release 6.0 feature set as well as some of the 6.2 feature set.
- FWSM delivers multigigabit throughput and 1 million concurrent connections.

CSPFA 3.2—4-39

5

Getting Started with the Cisco PIX Firewall

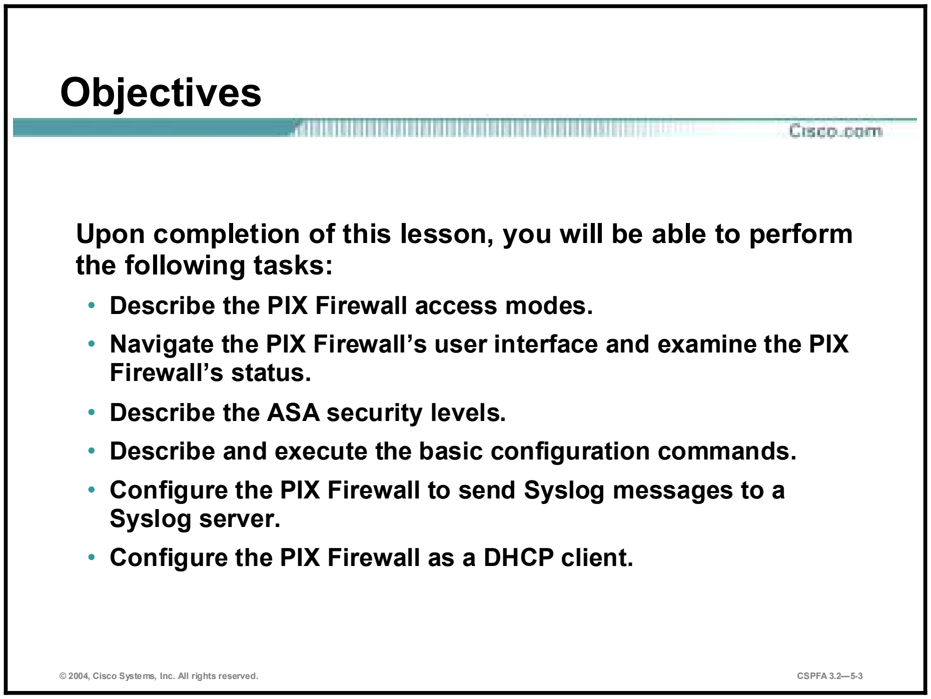
Overview

This lesson includes the following topics:

- Objectives
- User interface
- Configuring the PIX Firewall
- ASA security levels
- Basic PIX Firewall configuration
- Examining the PIX Firewall status
- Time setting and NTP support
- Syslog configuration
- Summary
- Lab exercise

Objectives

This topic lists the lesson's objectives.



Objectives

Cisco.com

Upon completion of this lesson, you will be able to perform the following tasks:

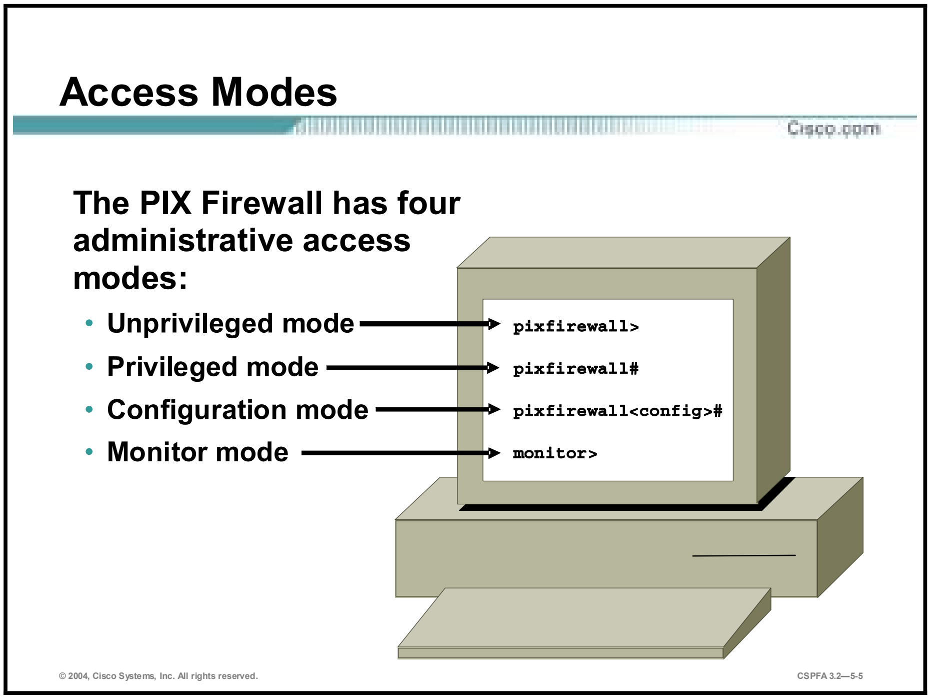
- **Describe the PIX Firewall access modes.**
- **Navigate the PIX Firewall's user interface and examine the PIX Firewall's status.**
- **Describe the ASA security levels.**
- **Describe and execute the basic configuration commands.**
- **Configure the PIX Firewall to send Syslog messages to a Syslog server.**
- **Configure the PIX Firewall as a DHCP client.**

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—5-3

User Interface

This topic references access modes and commands associated with the operation of the Cisco PIX Firewall.



Access Modes

The PIX Firewall has four administrative access modes:

- Unprivileged mode
- Privileged mode
- Configuration mode
- Monitor mode

```
pixfirewall>
pixfirewall#
pixfirewall<config>#
monitor>
```

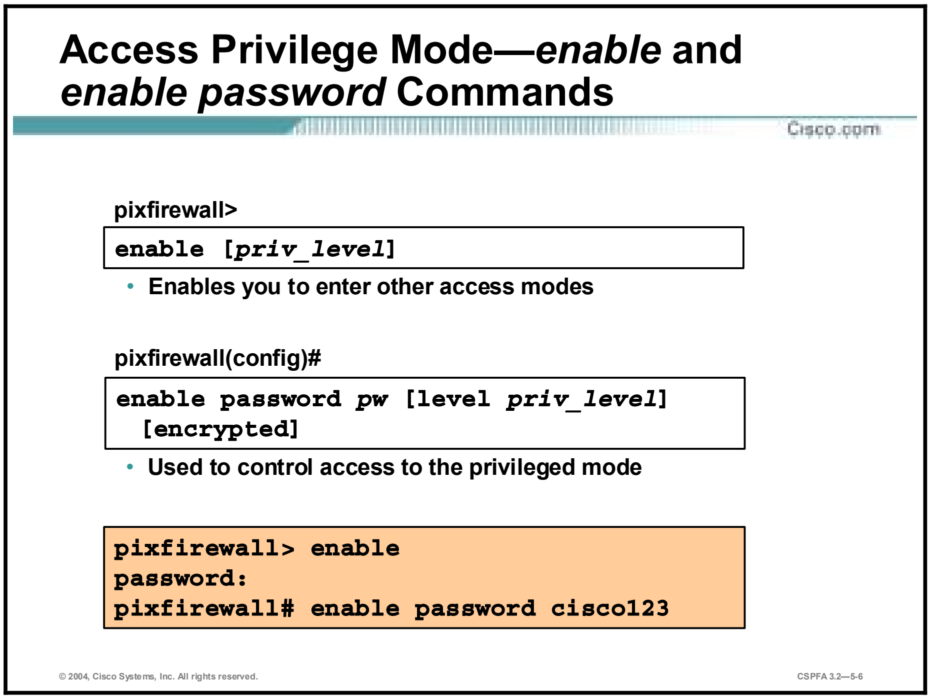
The PIX Firewall contains a command set based on the Cisco IOS, and provides four administrative access modes:

- Unprivileged mode—This mode is available when you first access the PIX Firewall. The > prompt is displayed. This mode provides a restricted, limited, view of PIX Firewall settings.
- Privileged mode—This mode displays the # prompt and enables you to change the current settings. Any unprivileged command also works in privileged mode.
- Configuration mode—This mode displays the (config)# prompt and enables you to change system configurations. All privileged, unprivileged, and configuration commands work in this mode.
- Monitor mode—This is a special mode that enables you to update the image over the network or to perform password recovery. While in the monitor mode, you can enter commands specifying the location of the TFTP server and the PIX Firewall software image or password recovery binary file to download.

Within each access mode, you can abbreviate most commands down to the fewest unique characters for a command. For example, you can enter **write t** to view the configuration instead of entering the full command **write terminal**. You can enter **en** instead of **enable** to start privileged mode, and **con t** instead of **configuration terminal** to start configuration mode.

Help information is available from the PIX Firewall command line by entering **help** or **?** to list all commands. If you enter **help** or **?** after a command (for example, **route?**), the command syntax is listed. The number of commands listed when you use the question mark or help command differs by access mode so that unprivileged mode offers the least commands and configuration mode offers the greatest number of commands. In addition, you can enter any command by itself on the command line and then press **Enter** to view the command syntax.

Note You can create your configuration on a text editor and then cut and paste it into the configuration. You can paste the configuration in a line at a time, or the entire configuration at once. Always check your configuration after pasting large blocks of text to be sure everything has been copied.



Access Privilege Mode—*enable* and *enable password* Commands

pixfirewall>

```
enable [priv_level]
```

- Enables you to enter other access modes

pixfirewall(config)#

```
enable password pw [level priv_level]
  [encrypted]
```

- Used to control access to the privileged mode

```
pixfirewall> enable
password:
pixfirewall# enable password cisco123
```

CSPFA 3.2—5-6

Upon first accessing a PIX Firewall, the administrator is presented with **pixfirewall>** prompt. This is the unprivileged mode. This mode enables one to view restricted settings. In a previously configured PIX Firewall, pixfirewall > may be replaced with a network specific hostname prompt such as Paris >, London >, etc. To get started with the PIX Firewall, the first command you need to know is the **enable** command. It provides entrance to the privileged access modes. After you enter **enable**, the PIX Firewall prompts you for your privileged mode password. By default, a password is not required, so you can press **Enter** at the password prompt, or you can create a password of your choice. After you are in privileged mode, notice that the prompt has changed to #.

The **enable password** command sets the privileged mode password. The password is case-sensitive and can be from 3 to 16 alphanumeric characters long. Any character can be used except the question mark, space, and colon.

If you create a password, write it down and store it in a manner consistent with your site's security policy. After you create this password, you cannot view it again because it is stored as an MD5 hash. The **show enable password** command lists the encrypted form of the password. After passwords are encrypted, they cannot be reversed back to plain text.

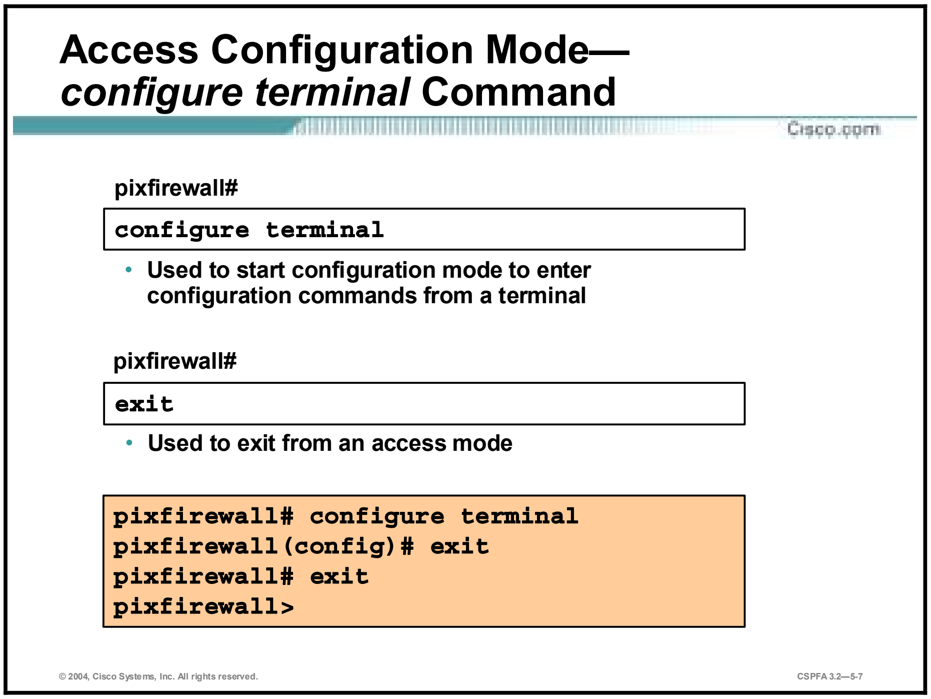
The syntax for the **enable** commands is as follows:

enable [*priv_level*]

enable password ***pw*** **[level** ***priv_level*****] [encrypted]**

priv_level	The privilege level, from 0 to 15.
pw	Specifies a case-sensitive password of 3 to 16 alphanumeric characters.
encrypted	Specifies that the password you entered is already encrypted.

Note An empty password is also changed into an encrypted string.



Access Configuration Mode—*configure terminal* Command

Cisco.com

pixfirewall#

```
configure terminal
```

- Used to start configuration mode to enter configuration commands from a terminal

pixfirewall#

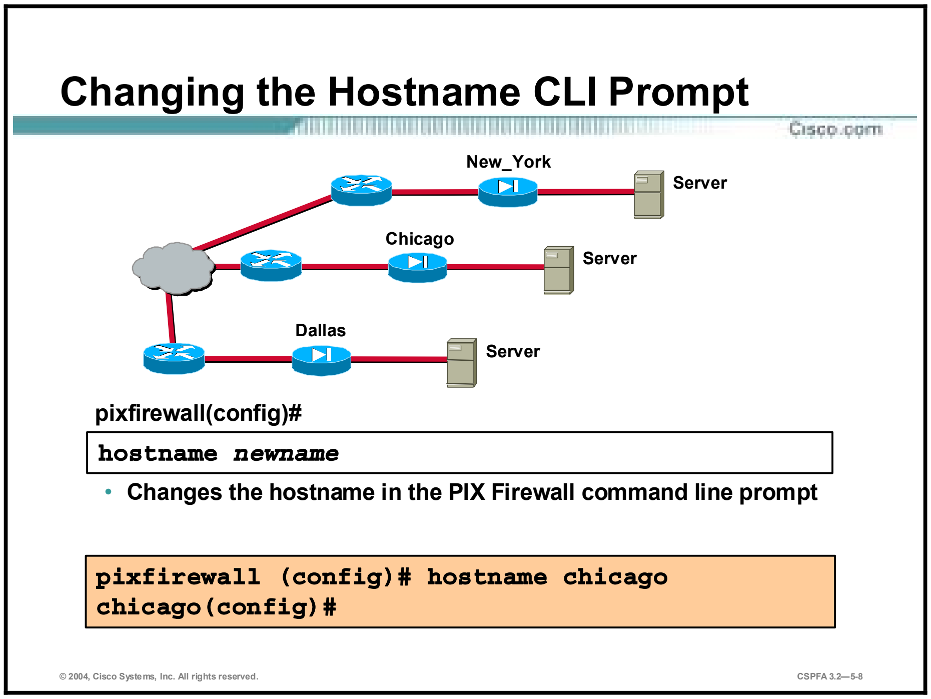
```
exit
```

- Used to exit from an access mode

```
pixfirewall# configure terminal
pixfirewall(config)# exit
pixfirewall# exit
pixfirewall>
```

CSPFA 3.2—5-7

Use the **configure terminal** command to move from privileged mode to configuration mode. As soon as you enter the command, the prompt changes to (config)#. Configuration mode enables one to change system configurations. There is more on the PIX Firewall basic commands later in this lesson. Use the **exit** or **quit** command to exit and return to the previous mode.



Changing the Hostname CLI Prompt

pixfirewall(config)#

```
hostname newname
```

- Changes the hostname in the PIX Firewall command line prompt

```
pixfirewall (config)# hostname chicago
chicago(config)#
```

In the example in the figure above, notice the PIX Firewall default hostname label is pixfirewall. In a network of multiple PIX Firewalls, it may be advantageous to assign a unique hostname label to each PIX Firewall. To accomplish this, use the **hostname** command. The **hostname** command changes the hostname label on the prompts. The hostname can be up to 16 alphanumeric characters, and upper- and lower-case. The default hostname is pixfirewall.

The syntax for the **hostname** command is as follows:

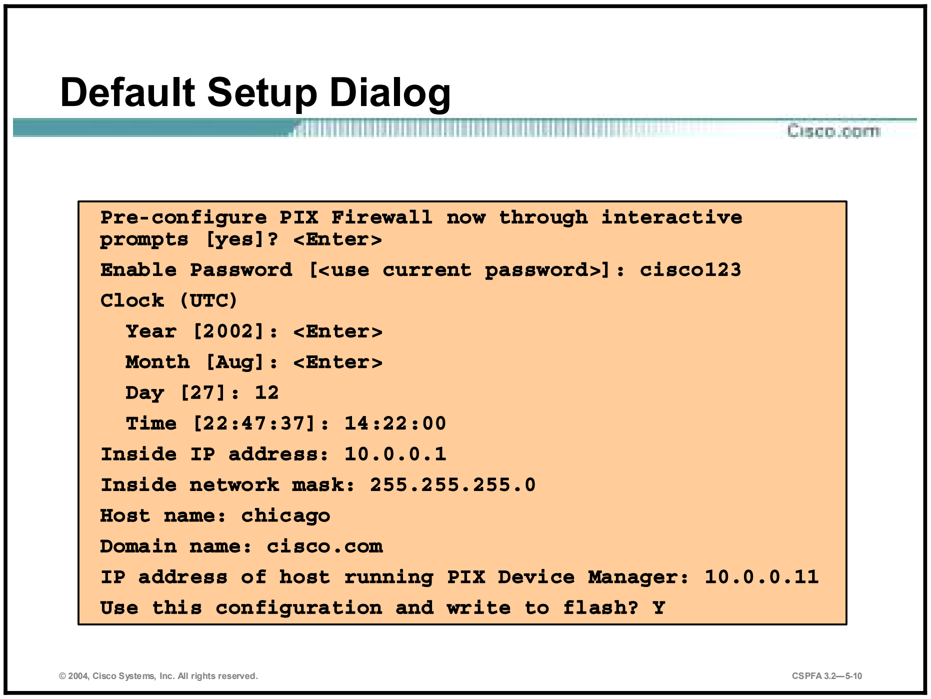
hostname ***newname***

newname	New hostname for the PIX Firewall prompt.

In the figure, the default hostname label of pixfirewall is changed to “chicago” using the **hostname** command.

Configuring the PIX Firewall

This topic explains some basic commands that are useful for configuring the PIX Firewall.



Default Setup Dialog

```
Pre-configure PIX Firewall now through interactive
prompts [yes]? <Enter>
Enable Password [<use current password>]: cisco123
Clock (UTC)
  Year [2002]: <Enter>
  Month [Aug]: <Enter>
  Day [27]: 12
  Time [22:47:37]: 14:22:00
Inside IP address: 10.0.0.1
Inside network mask: 255.255.255.0
Host name: chicago
Domain name: cisco.com
IP address of host running PIX Device Manager: 10.0.0.11
Use this configuration and write to flash? Y
```

CSPFA 3.2—5-10

When a non-configured PIX Firewall boots up, it prompts you to pre-configure it through interactive prompts. If you press **Enter** to accept the default answer of yes, you are presented with a series of prompts that lead you through the basic configuration steps. The figure shows an example of how to respond to the prompts.

The setup dialog was designed to pre-configure the PIX Firewall to interact with the PIX Device Manager (PDM). The PIX Firewall requires some pre-configuration before PDM can connect to it. PDM is a GUI that can be used to configure and monitor the PIX Firewall. It is discussed in another lesson.

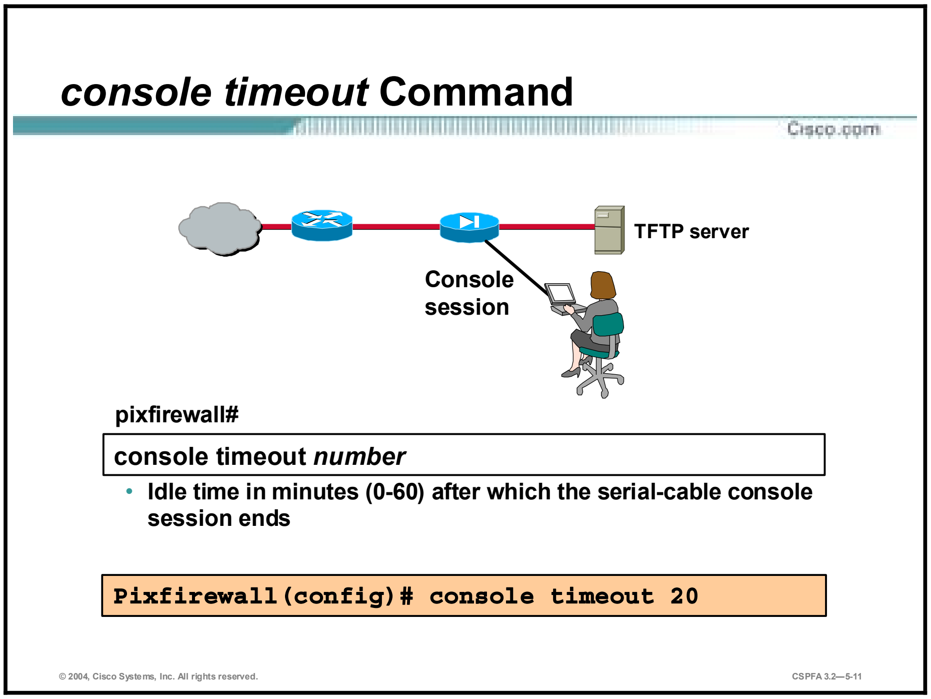
The setup dialog can be also be accessed by entering the **setup** command. The following are the prompts found in the setup dialog:

- Enable Password—Specifies an enable password for this PIX Firewall.
- Clock (UTC)—Sets the PIX Firewall clock to Universal Coordinated Time (also known as Greenwich Mean Time).
- Year—Specifies the current year, or defaults to the year stored in the host computer.
- Month—Specifies the current month, or defaults to the month stored in the host computer.
- Day—Specifies the current day, or defaults to the day stored in the host computer.
- Time—Specifies the current time in hh:mm:ss format, or defaults to the time stored in the host computer.
- Inside IP address—Network interface IP address of the PIX Firewall.
- Inside network mask—A network mask that applies to the inside IP address.

- Host name—The host name you want to display in the PIX Firewall CLI prompt.
- Domain name—The DNS domain name of the network on which the PIX Firewall runs (for example, example.com).
- IP address of host running PIX Device Manager—IP address on which PDM connects to the PIX Firewall.

At the end of the setup dialog, you are asked if you want to write the configuration to Flash memory. If you answer yes, the configuration you just entered is saved to Flash memory. If you answer no, the setup dialog repeats using the values already entered as the defaults for the questions.

Note You can escape the setup dialog by pressing **Control-Z**.



console timeout Command

TFTP server

Console session

pixfirewall#

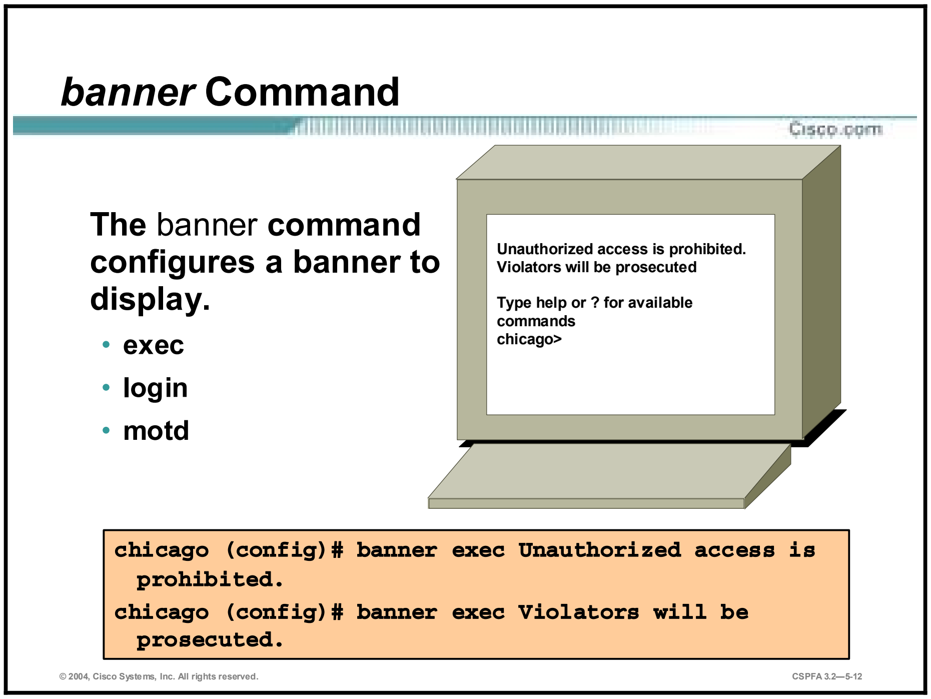
```
console timeout number
```

- Idle time in minutes (0-60) after which the serial-cable console session ends

```
Pixfirewall(config)# console timeout 20
```

CSPFA 3.2—5-11

By default, there is no timeout value for console session users. If a console user walks away from an open session, the session remains open. For security reasons, it may be advantageous to configure an idle timeout value in the PIX Firewall. If there is no activity for a pre-defined time, the PIX Firewall will end the console session. The **console timeout** command sets the timeout value for any authenticated, privileged mode, or configuration mode user session when accessing the firewall console through a serial cable. The default value is zero, no timeout. This may present a security risk. By setting the number to a non-zero number, the user is logged out after the specified period of inactivity. This timeout does not alter the Telnet or SSH timeouts; these access methods maintain their own timeout values.



banner Command

Cisco.com

The banner command configures a banner to display.

- exec
- login
- motd

```
Unauthorized access is prohibited.
Violators will be prosecuted

Type help or ? for available
commands
chicago>
```

```
chicago (config)# banner exec Unauthorized access is
  prohibited.
chicago (config)# banner exec Violators will be
  prosecuted.
```

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—5-12

The **banner** command enables the administrator to define messages in the PIX Firewall. There are three types of banner commands, exec, login, and message of the day (motd). The usage of each banner type is as follows:

- **Exec—**Configures the system to display a banner before displaying the privilege mode prompt.
- **Login—**Configures the system to display a banner before the password login prompt when accessing the firewall using telnet.
- **Motd**—Configures the system to display a message-of-the-day banner.

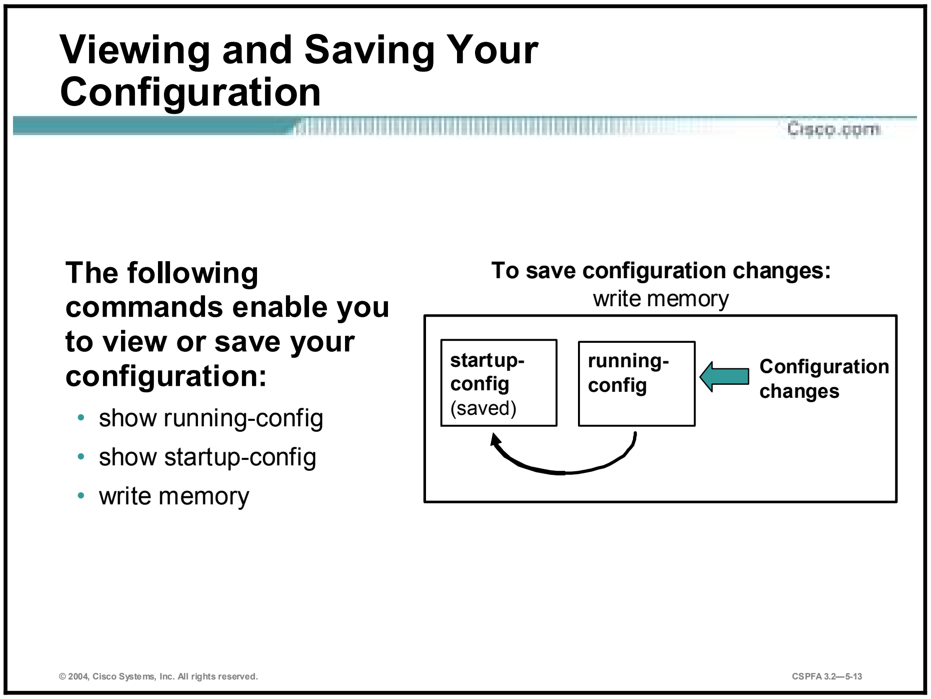
The **banner** command configures a banner to display for the option specified. The *text* string consists of all characters following the first white space (space) until the end of the line (carriage return or LF). Spaces in the text are preserved. However, tabs cannot be entered through the CLI. Multiple lines in a banner are handled by entering a new banner command for each line you wish to add. Each line is then appended to the end of the existing banner.

In the figure, the administrator wants to add a legal statement to the login process. The **banner** command enables the administrator is preface all console sessions with the statement, “Unauthorized access is not permitted. Violators will be prosecuted.”

To replace a banner, use the **no banner** command before adding the new lines. The **no banner {exec |login | motd**} command removes all the lines for the banner option specified. The **no banner** command does not selectively delete text strings, so any *text* entered at the end of the **no banner** command is ignored.

The **clear banner** command removes all the banners.

The **show banner {motd | exec | login**} command displays the specified banner option and all the lines configured for it. If a banner option is not specified, then all the banners are displayed.



Viewing and Saving Your Configuration

Cisco.com

The following commands enable you to view or save your configuration:

- show running-config
- show startup-config
- write memory

To save configuration changes:
write memory

startup-config (saved)

running-config

Configuration changes

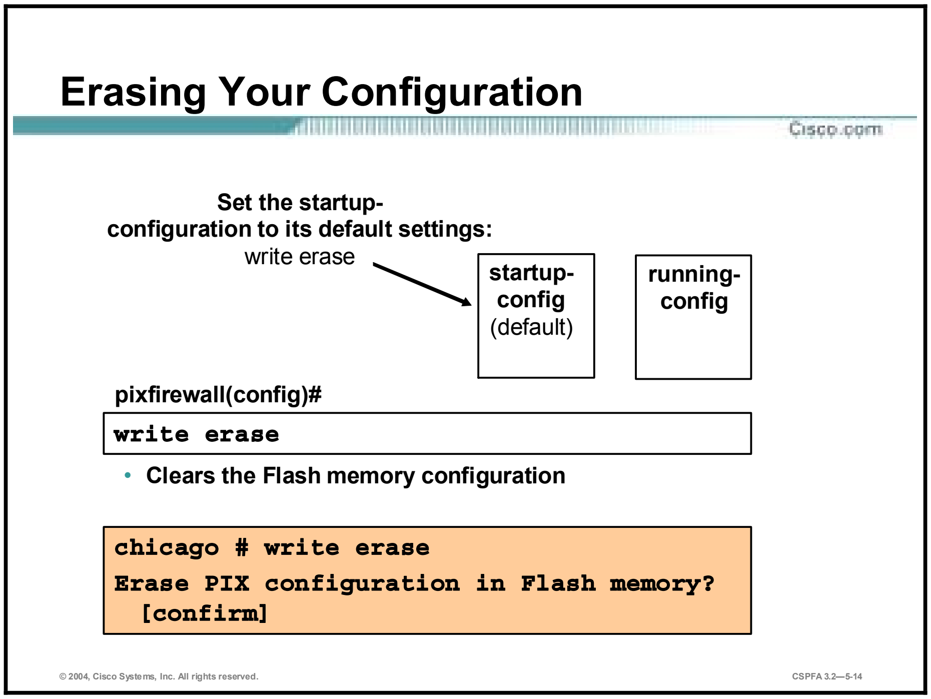
© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—5-13

There are two configuration memories in the PIX, running-configuration and startup-configuration. The **show running-config** command displays the current configuration in the PIX Firewall's RAM on the terminal. Any changes made to the PIX Firewall's configuration are written into the running-configuration. This is volatile RAM. If the PIX Firewall looses power, or is re-booted, any changes to the running-configuration that were not previously saved are lost. You can also display the current running-configuration with the **write terminal** command.

The **write memory** command saves the current running-configuration to Flash memory, startup-configuration. It is the same as answering yes to the setup dialog prompt that asks if you wish to save the current configuration to Flash memory. When the configuration is written to flash memory, you can view it with either the **show startup-config** or **show configure** command.

Another useful command is **show history**, which displays previously entered commands. You can examine commands individually with the up and down arrows or by entering **Ctrl+p** to view previously entered lines or **Ctrl+n** to view the next line.



Erasing Your Configuration

Cisco.com

Set the startup-configuration to its default settings:

write erase

startup-config (default)
running-config

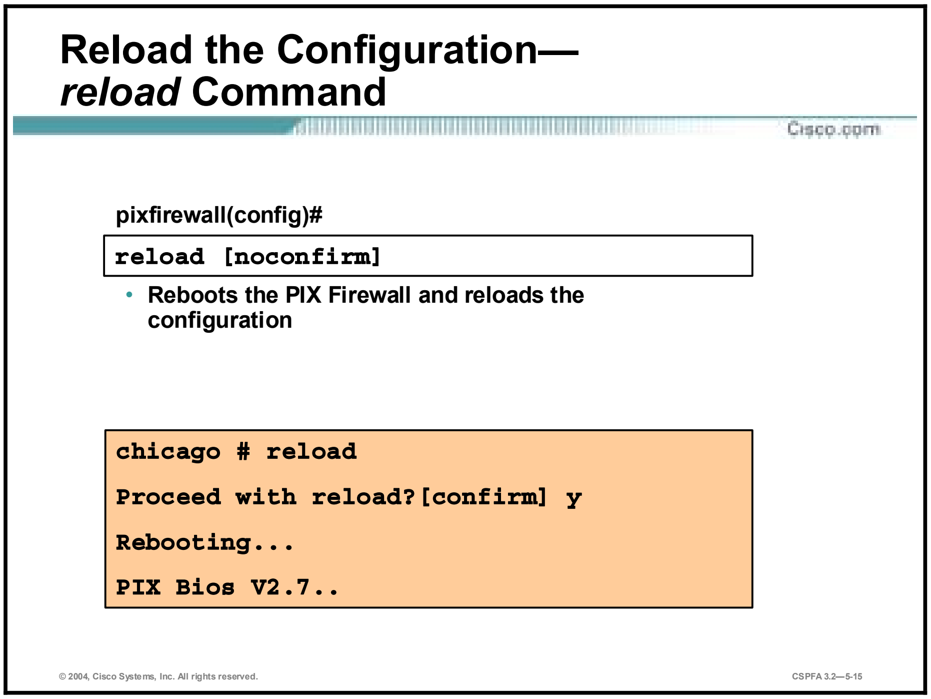
pixfirewall(config)#

```
write erase
```

- Clears the Flash memory configuration

```
chicago # write erase
Erase PIX configuration in Flash memory?
  [confirm]
```

The **write erase** command clears the startup-configuration. When you issue this command, you are prompted to confirm if you want to erase the startup configuration. If you enter **yes**, the startup configuration is erased. At this point, one can power cycle, or re-boot the PIX Firewall. The PIX Firewall reverts to the default configuration. Or, one can copy the running configuration to flash by issuing the **write memory** command.



Reload the Configuration— *reload* Command

Cisco.com

pixfirewall(config)#

```
reload [noconfirm]
```

- Reboots the PIX Firewall and reloads the configuration

```
chicago # reload
Proceed with reload?[confirm] y
Rebooting...
PIX Bios V2.7..
```

CSPFA 3.2—5-15

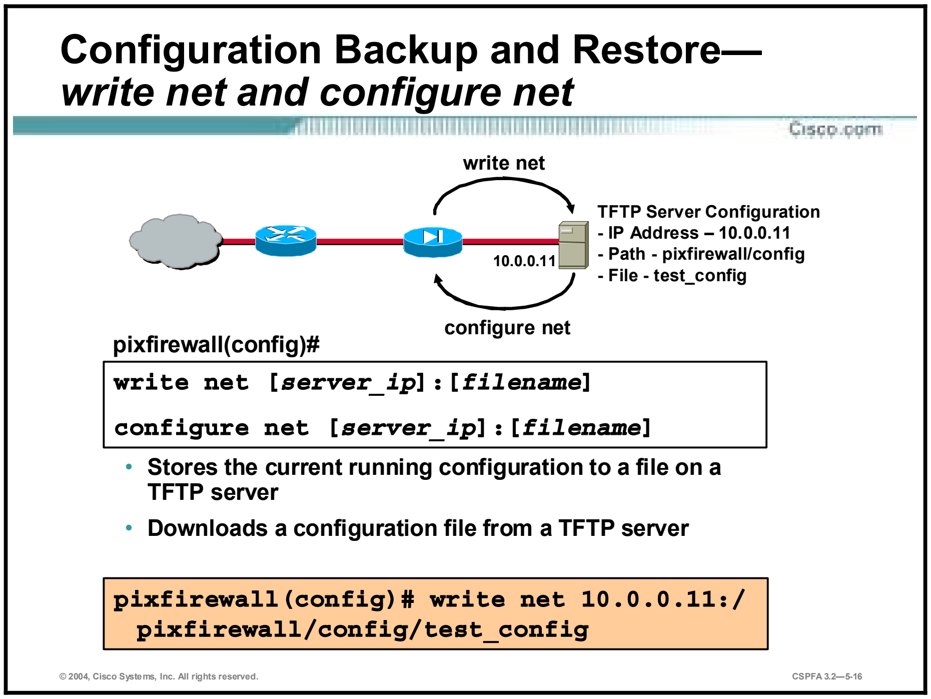
The **reload** command reboots the PIX Firewall and reloads the configuration from Flash memory. You are prompted for confirmation before the reload process begins with the prompt "Proceed with reload?" Any response other than **n** causes the reboot to occur.

Configuration changes not written to Flash memory are lost after reload. Before re-booting, store the current configuration in Flash memory with the **write memory** command.

The **noconfirm** option permits the PIX Firewall to reload without user confirmation. The PIX Firewall does not accept abbreviations to the keyword **noconfirm**.

If you wish to return the PIX Firewall back to the factory default configuration, perform a **write erase** and a **reload**. The **write erase** clears the startup configuration and reverts to the factory default parameters. **Reload** command re-boots the PIX Firewall using the startup configuration, in this case the factory default configuration.

An administrator can backup or restore a PIX Firewall configuration. The **write net** command stores the current configuration into a file on a TFTP server elsewhere in the network. The **configure net** command restores the configuration from the server to the PIX firewall. To complete the backup or restore, the administrator must supply information about the TFTP server such as the IP address and the file pathname. There is more information on configuring the write net and configure net later in this lesson.



Configuration Backup and Restore—*write net and configure net*

TFTP Server Configuration
- IP Address – 10.0.0.11
- Path - pixfirewall/config
- File - test_config

pixfirewall(config)#

```
write net [server_ip]:[filename]

configure net [server_ip]:[filename]
```

- Stores the current running configuration to a file on a TFTP server
- Downloads a configuration file from a TFTP server

```
pixfirewall(config)# write net 10.0.0.11:/
  pixfirewall/config/test_config
```

CSPFA 3.2—5-16

The **write net** command enables you to store the current configuration to a file on a TFTP server elsewhere in the network. The **configure net** command merges the current running configuration with the TFTP configuration stored at the IP address you specify and from the file you name. To use the **configure net** and **write net** commands, one must specify both the server IP address and full path of the **tftp-server**.

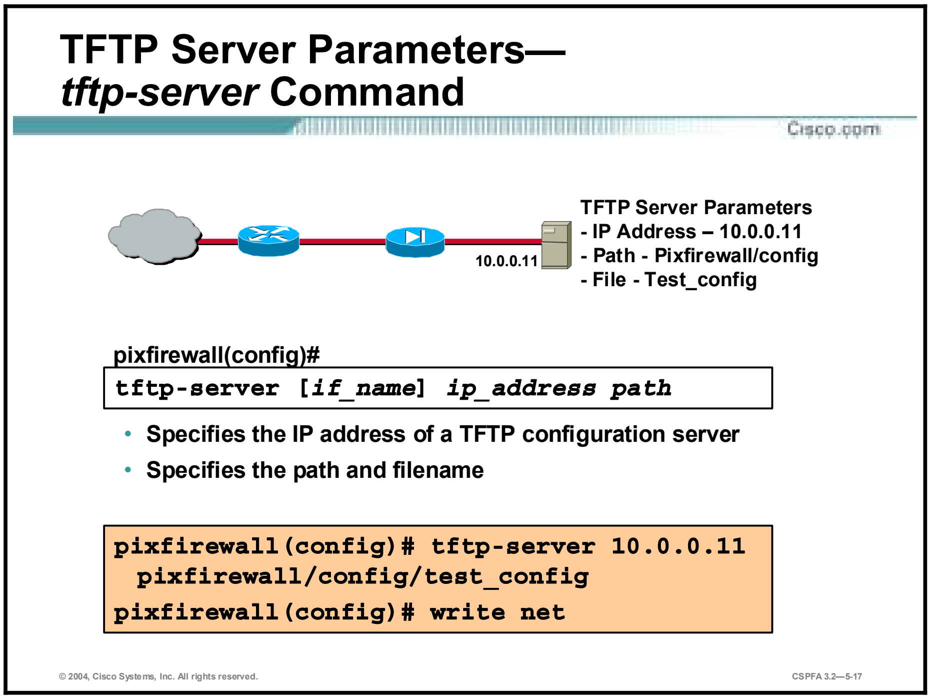
If you have an existing PIX Firewall configuration on a TFTP server and store a shorter configuration with the same filename on the TFTP server, some TFTP servers will leave some of the original configuration after the first "end" mark. This does not affect the PIX Firewall because the **configure net** command stops reading when it reaches the first "end" mark. However, this may cause confusion if you view the configuration and see extra text at the end of the configuration. This does not occur if you are using Cisco TFTP Server version 1.1 for Windows NT.

The example in the figure specifies the TFTP server address as 10.0.0.11 and the path to the file test_config as pixfirewall/config. Because the interface where the TFTP server resides is not specified, the inside interface is assumed. The **write net** command tells the PIX Firewall to store the configuration in the test_config file.

The syntax for the **write net** and **configure net** commands is as follows:

write net [*server_ip*]:[*filename*]

configure net [*server_ip*]:[*filename*]



TFTP Server Parameters—*tftp-server* Command

Cisco.com
10.0.0.11

TFTP Server Parameters
- IP Address – 10.0.0.11
- Path - Pixfirewall/config
- File - Test_config

pixfirewall(config)#

```
tftp-server [if_name] ip_address path
```

- Specifies the IP address of a TFTP configuration server
- Specifies the path and filename

```
pixfirewall(config)# tftp-server 10.0.0.11
  pixfirewall/config/test_config
pixfirewall(config)# write net
```

CSPFA 3.2—5-17

Rather than write the full server IP address and file pathname every time the configuration is backed-up or restored, the PIX enables the administrator to “split” the command into two commands, the **write net** or **config net** commands and the **tftp-server** command. The **write net** and **config net** commands backup the current configuration, and restores a configuration from the tftp server, respectively. The **tftp-server** command defines the IP address and the file path name of the tftp server. The **write net** and **config net** command relies on the server IP address and file pathname specified in the **tftp-server** command. The information you specify in the **tftp-server** is appended to the **config net** and **write net** commands. The more you specify of a file and path name with the **tftp-server** command, the less you need to specify with the **config net** and **write net** commands. If you specify the IP address and full path and filename in the **tftp-server** command, the **config net** and **write net** commands can be represented with a colon (:), write net : or config-net :.

The **no tftp-server** command disables access to the server, and the **clear tftp-server** command removes the **tftp-server** command from your configuration. The **show tftp-server** command lists the **tftp-server** command statements in the current configuration.

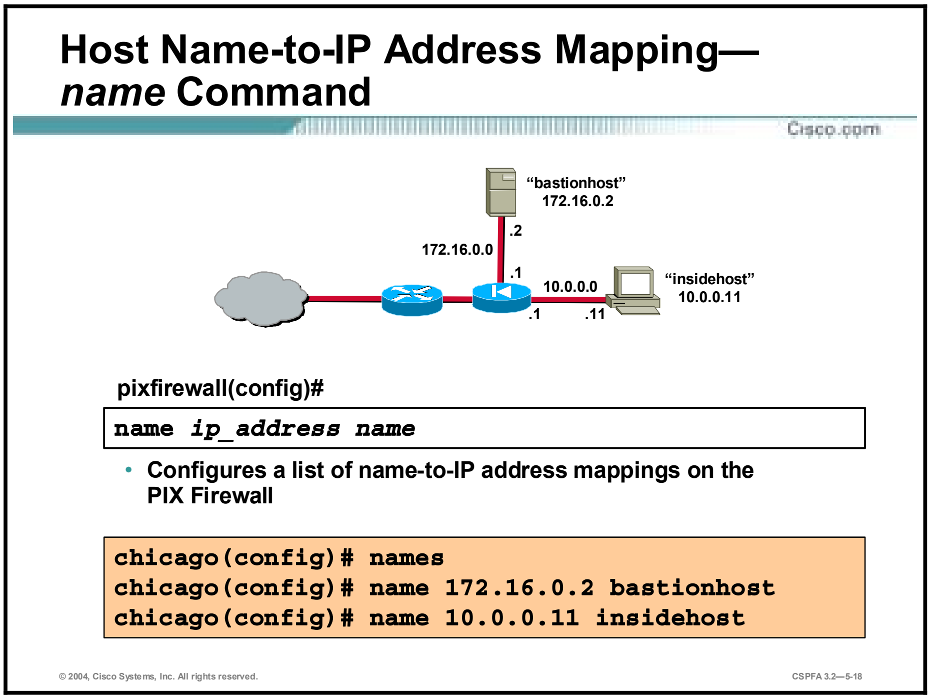
The syntax for the **tftp-server** command is as follows:

tftp-server [*if_name*] *ip_address path*

if_name	Interface name on which the TFTP server resides. If not specified, an internal interface is assumed. If you specify the outside interface, a warning message informs you that the outside interface is insecure.
ip_address	The IP address or network of the TFTP server.
path	The path and filename of the configuration file. The format for path differs by the type of operating system on the server. The contents of path are passed directly to the server without interpretation or checking. The configuration file must exist on the TFTP server. Many TFTP servers require the configuration file to be world-writable to write to it and world-readable to read from it.

Note The PIX Firewall supports only one TFTP server.

Note If you erase the configuration, you must re-enable and set an IP address on the interface connected to the TFTP server before the PIX Firewall can read a new configuration from the TFTP server.



Host Name-to-IP Address Mapping—*name* Command

pixfirewall(config)#

```
name ip_address name
```

- Configures a list of name-to-IP address mappings on the PIX Firewall

```
chicago(config)# names
chicago(config)# name 172.16.0.2 bastionhost
chicago(config)# name 10.0.0.11 insidehost
```

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—5-18

The use of the **name** command enables you to configure a list of name-to-IP address mappings on the PIX Firewall. This allows the use of names in the configuration instead of IP addresses. In the figure above, the server and PC's IP addresses are mapped to names, bastionhost, and insidehost. Bastionhost and insidehost can be used in place of an IP address in any PIX Firewall command reference, for instance the command ping insidehost.

The syntax for the **name** command is as follows:

name *ip_address name*

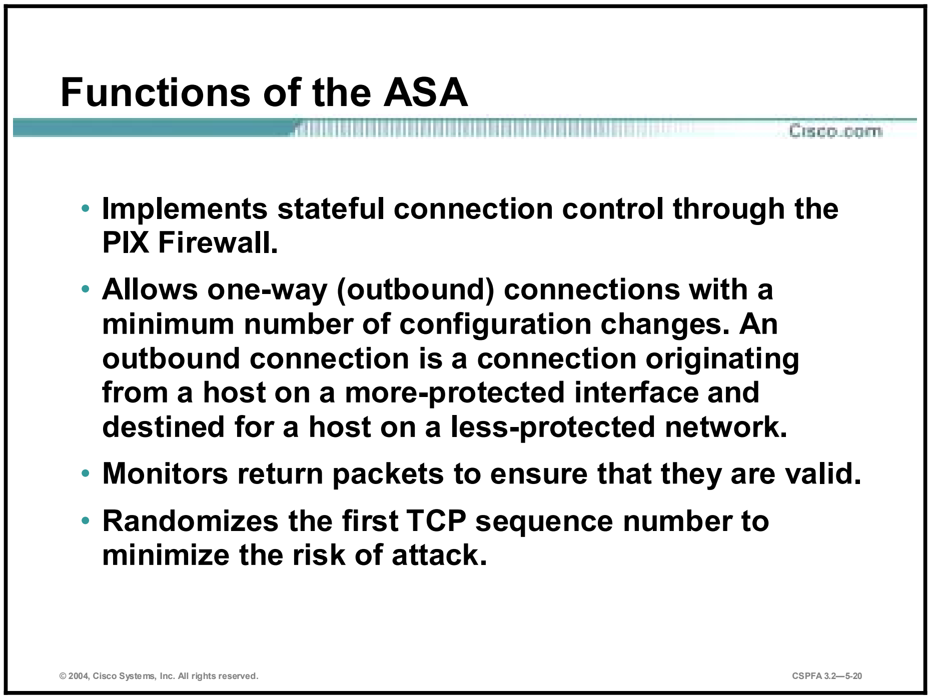
ip_address	The IP address of the host being named.
name	The name assigned to the IP address.

Allowable characters for the name are a to z, A to Z, 0 to 9, a dash (-), and an underscore (_). The name cannot start with a number. If the name is over 16 characters long, the **name** command fails. After the name is defined, it can be used in any PIX Firewall command reference in place of an IP address. The **names** command enables the use of the **name** command. The **clear names** command clears the list of names from the PIX Firewall configuration. The **no names** command disables the use of the text names, but does not remove them from the configuration. The **show names** command lists the **name** command statements in the configuration.

Note Most commands can be removed or disabled by placing the word **no** in front of the command. For example, the **no** form of the **names** command shown previously disables the use of names.

ASA Security Levels

This topic discusses the Adaptive Security Algorithm (ASA) and the ASA security levels.



Functions of the ASA

Cisco.com

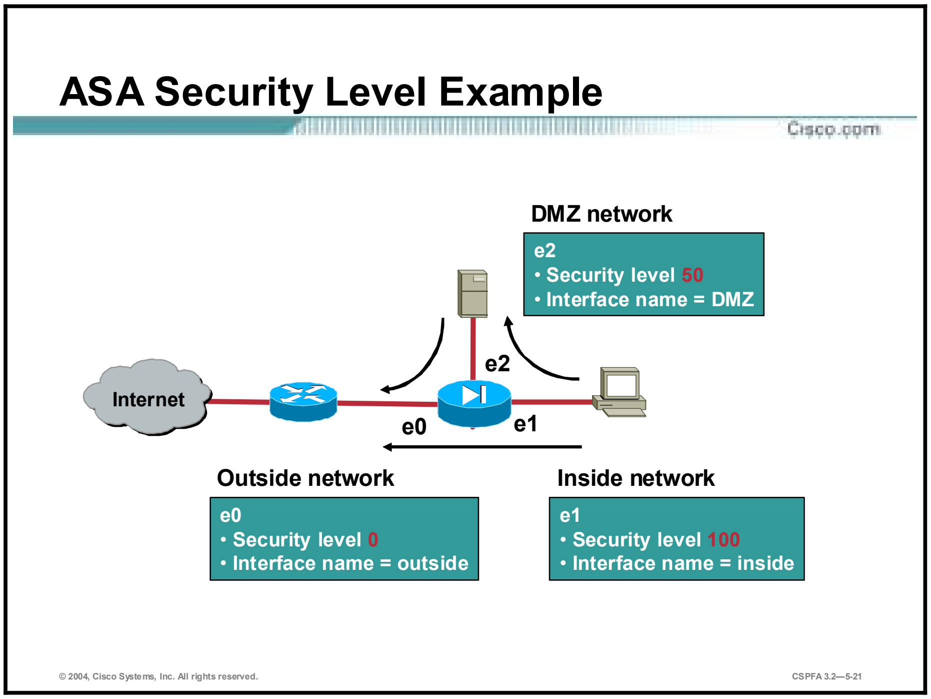
- Implements stateful connection control through the PIX Firewall.
- Allows one-way (outbound) connections with a minimum number of configuration changes. An outbound connection is a connection originating from a host on a more-protected interface and destined for a host on a less-protected network.
- Monitors return packets to ensure that they are valid.
- Randomizes the first TCP sequence number to minimize the risk of attack.

CSPFA 3.2—5-20

The ASA is a stateful approach to security. Every inbound packet (the packet originating from a host on a less-protected network and destined for a host on a more-protected network) is checked against the ASA and against connection state information in the PIX Firewall's memory. Knowledge of the ASA is fundamental to implementing Internet access security because it performs the following tasks:

- Implements stateful connection control through the PIX Firewall.
- Allows one-way (outbound) connections with a minimum number of configuration changes. An outbound connection is a connection originating from a host on a more-protected interface and destined for a host on a less-protected network.
- Monitors return packets to ensure they are valid.
- Randomizes the first TCP sequence number to minimize the risk of attack.

ASA maintains the secure perimeters between the networks controlled by the PIX Firewall. The stateful connection-oriented ASA design creates session flows based on source destination addresses as well as TCP and UDP port numbers. ASA randomizes TCP sequence numbers before the completion of the connection. This function is always running, monitoring return packets to ensure that they are valid.



ASA Security Level Example

Cisco.com

DMZ network

e2
- Security level 50
- Interface name = DMZ

Outside network

e0
- Security level 0
- Interface name = outside

Inside network

e1
- Security level 100
- Interface name = inside

CSPFA 3.2—5-21

The security level designates whether an interface is trusted (and more protected) or untrusted (and less protected) relative to another interface. An interface is considered trusted (and more protected) in relation to another interface if its security level is higher than the other interface's security level, and is considered untrusted (and less protected) in relation to another interface if its security level is lower than the other interface's security level.

The primary rule for security levels is that an interface with a higher security level can access an interface with a lower security level. Conversely, an interface with a lower security level cannot access an interface with a higher security level without an access control list (ACL), which is discussed later in the lesson. Security levels range from 0 to 100, and the following are more specific rules for these security levels:

- Security level 100—This is the highest security level for the inside interface of the PIX Firewall. This is the default setting for the PIX Firewall and cannot be changed. Because 100 is the most trusted interface security level, your corporate network should be set up behind it. This is so that no one else can access it unless they are specifically given permission, and so that every device behind this interface can have access outside the corporate network.
- Security level 0—This is the lowest security level for the outside interface of the PIX Firewall. This is the default setting for the PIX Firewall and cannot be changed. Because 0 is the least-trusted interface security level, you should set your most untrusted network behind this interface so that it does not have access to other interfaces unless it is specifically given permission. This interface is usually used for your Internet connection.
- Security levels 1–99—These are the security levels that you can assign to the perimeter interfaces connected to the PIX Firewall. You assign the security levels based on the type of access you want each device to have.

The following are examples of different interface connections between the PIX Firewall and other perimeter devices:

- More secure interface (the higher security level) to a less secure interface (the lower security level)—Traffic originating from the inside interface of the PIX Firewall with a security level of 100 to the outside interface of the PIX Firewall with a security level of 0 follows this rule: allow all IP-based traffic unless restricted by ACLs, authentication, or authorization.
- Less secure interface (lower security level) to a more secure interface (higher security level)—Traffic originating from the outside interface of the PIX Firewall with a security level of 0 to the inside interface of the PIX Firewall with a security level of 100 follows this rule: drop all packets unless specifically allowed by an **access list** command. Further restrict the traffic if authentication and authorization is used.
- Same secure interface to a same secure interface—No traffic flows between two interfaces with the same security level.

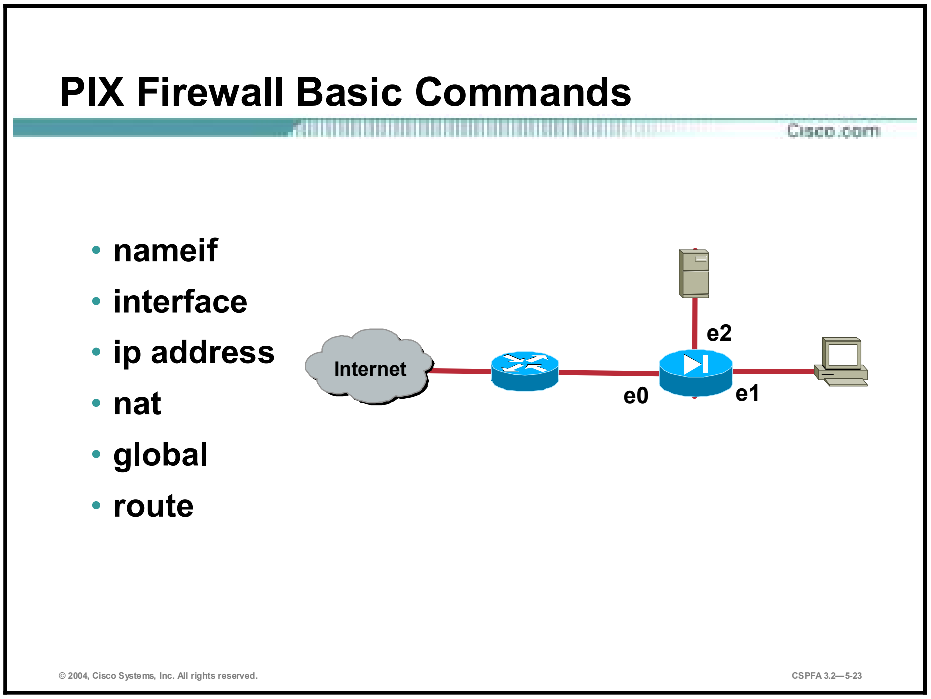
The following table explains the diagram in the previous figure:

Interface Pair	Relative Interface Relationship for Ethernet 2 (DMZ) Interface	Configuration Guidelines
Outside security 0 to DMZ security 50	DMZ is considered trusted	Statics and ACLs must be configured to enable sessions originated from the outside interface to the DMZ interface.
Inside security 100 to DMZ security 50	DMZ is considered untrusted	Globals and NAT are configured to enable sessions originated from the inside interface to the DMZ interface. Statics may be configured for the DMZ interface to ensure service hosts have the same source address.

Note The PIX Firewall can support up to ten interfaces depending on the model and license.

Basic PIX Firewall Configuration

This topic contains the basic commands needed to make the PIX Firewall operational.



PIX Firewall Basic Commands

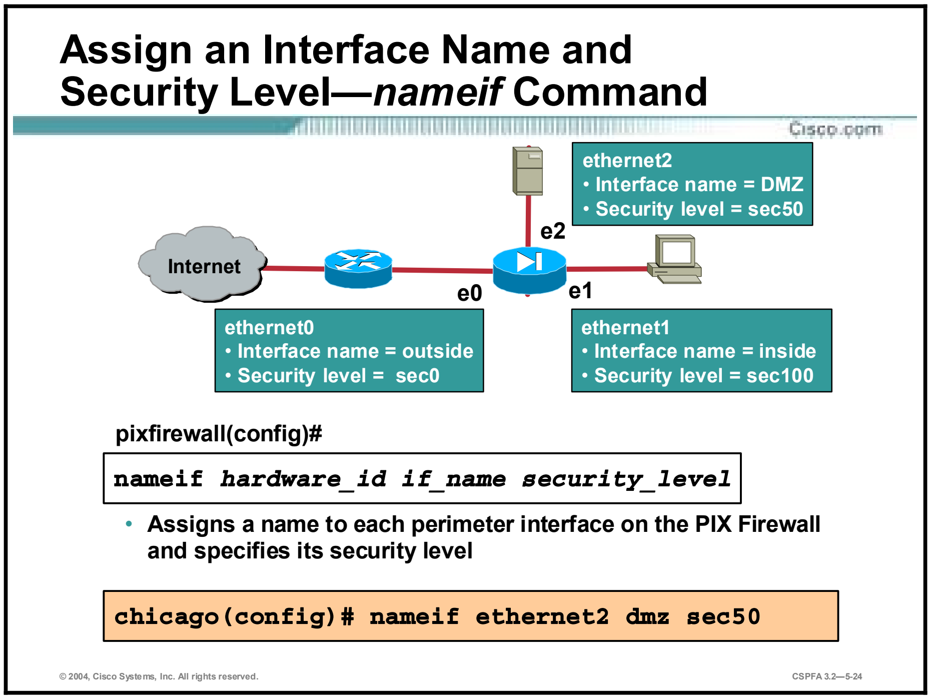
Cisco.com

- nameif
- interface
- ip address
- nat
- global
- route

CSPFA 3.2—5-23

The following are some of the primary configuration commands for the PIX Firewall.

- **nameif**—Assigns a name to each perimeter interface and specifies its security level.
- **interface**—Configures the type and capability of each perimeter interface.
- **ip address**—Assigns an IP address to each interface.
- **nat**—Shields IP addresses on the inside network from the outside network.
- **global**—Creates a pool of one or more IP addresses for use in NAT and PAT.
- **route**—Defines a static or default route for an interface.



Assign an Interface Name and Security Level—*nameif* Command

ethernet2
- Interface name = DMZ
- Security level = sec50

ethernet0
- Interface name = outside
- Security level = sec0

ethernet1
- Interface name = inside
- Security level = sec100

pixfirewall(config)#

```
nameif hardware_id if_name security_level
```

- Assigns a name to each perimeter interface on the PIX Firewall and specifies its security level

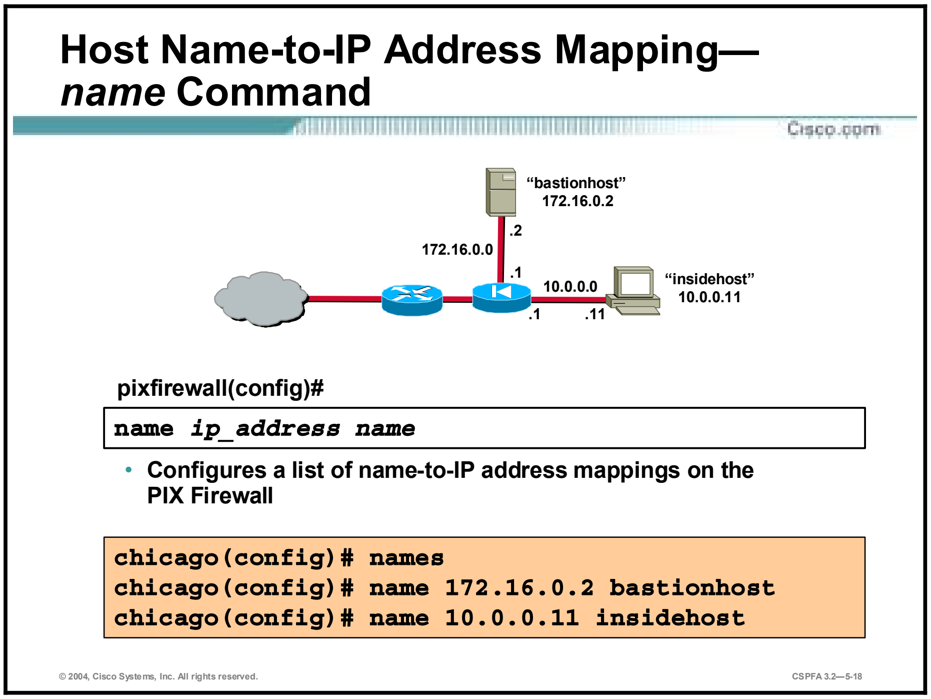
```
chicago(config)# nameif ethernet2 dmz sec50
```

CSPFA 3.2—5-24

The command **nameif** assigns a name to each interface on the PIX Firewall and specifies its security level (except for the inside and outside PIX Firewall interfaces, which are named by default). The first two interfaces have the default names **inside** and **outsid**e. The **inside** interface has a default security level of 100; the **outside** interface has a default security level of 0. In the figure, interface Ethernet 2 was assigned a name of DMZ with a security level of 50. The syntax for the **nameif** command is as follows:

nameif *hardware_id if_name security_level*

hardware_id	The hardware name for the network interface that specifies the slot location of the interface on the PIX Firewall motherboard. For more information on PIX Firewall hardware configuration, refer to the *Cisco PIX Firewall Hardware Installation Guide*. A logical choice for an Ethernet interface is **ethernet*n*.** These names can also be abbreviated with any leading characters in the name, for example, **ether1** or **e2**.
if_name	Describes the perimeter interface. This name is assigned by you, and must be used in all future configuration references to the perimeter interface.
security_level	Indicates the security level for the perimeter interface. Enter a security level of sec1–sec99.



Host Name-to-IP Address Mapping—*name* Command

Cisco.com

pixfirewall(config)#

```
name ip_address name
```

- Configures a list of name-to-IP address mappings on the PIX Firewall

```
chicago(config)# names
chicago(config)# name 172.16.0.2 bastionhost
chicago(config)# name 10.0.0.11 insidehost
```

CSPFA 3.2—5-18

The **interface** command identifies hardware, sets its hardware speed, and enables the interface. The shutdown option disables an interface. When you first install the PIX Firewall, all interfaces are shut down by default. You must explicitly enable them by entering the **interface** command without the shutdown option. In the figure, interfaces Ethernet 0, Ethernet 1, and Ethernet 2 are set for 100 Mbps full-duplex communications.

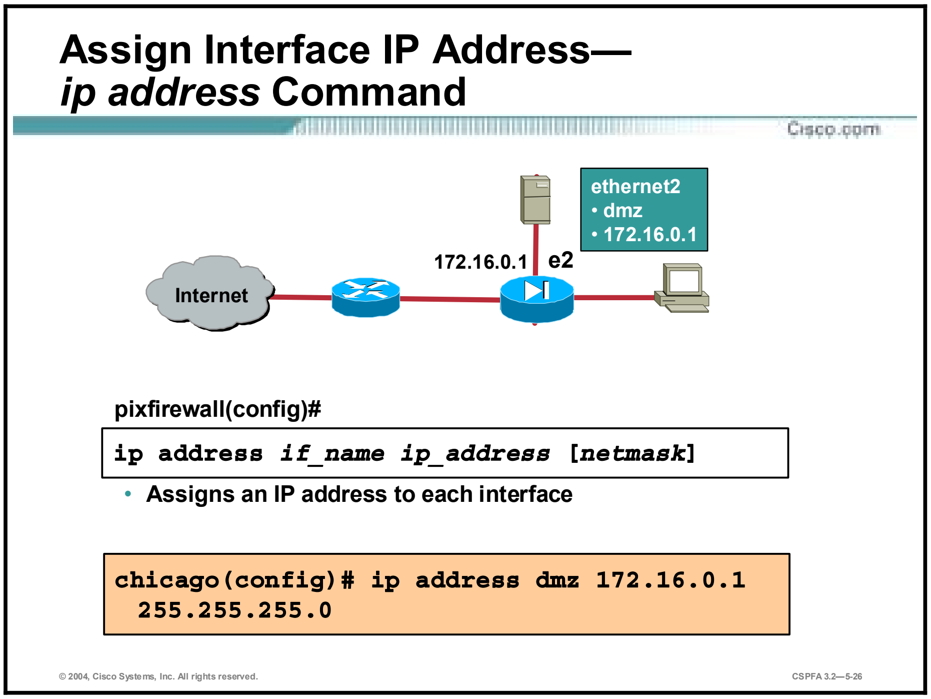
The syntax for the **interface** command is as follows:

interface ***hardware_id*** **[*hardware_speed*] [shutdown]**

hardware_id	Specifies an interface and its slot location on the PIX Firewall. This is the same variable that was used during the **nameif** command.
hardware_speed	Determines the connection speed. Possible Ethernet values are as follow: ■ **10baset**—Set for 10 Mbps Ethernet half-duplex communication. ■ **10full**—Set for 10 Mbps Ethernet full-duplex communication. ■ **100basetx**—Set for 100 Mbps Ethernet half-duplex communication. ■ **100full**—Set for 100 Mbps Ethernet full-duplex communication. ■ **1000sxfull**—Set for 1000 Mbps Gigabit Ethernet full-duplex operation. ■ **1000basesx**—Set for 1000 Mbps Gigabit Ethernet half-duplex operation. ■ **1000auto**—Set for 1000 Mbps Gigabit Ethernet to auto-negotiate full- or half-duplex. It is recommended that you do not use this option to maintain compatibility with switches and other devices in your network.

	■ **aui**—Set for 10 Mbps Ethernet half-duplex communication with an AUI cable interface. ■ **auto**—Set Ethernet speed automatically. The auto keyword can only be used with the Intel 10/100 automatic speed sensing network interface card. ■ **bnc**—Set for 10 Mbps Ethernet half-duplex communication with a BNC cable interface. Possible Token Ring values are as follow: ■ **4mbps**—4 Mbps data transfer speed. You can specify this as 4. ■ **16mbps**—(Default.) 16 Mbps data transfer speed. You can specify this as 16.
shutdown	Administratively shuts down the interface.

Although the hardware speed is set to automatic speed sensing by default, it is recommended that you specify the speed of the network interfaces. This enables the PIX Firewall to operate in network environments that may include switches or other devices that do not handle auto sensing correctly.



Assign Interface IP Address—*ip address* Command

pixfirewall(config)#

```
ip address if_name ip_address [netmask]
```

- Assigns an IP address to each interface

```
chicago(config)# ip address dmz 172.16.0.1
  255.255.255.0
```

CSPFA 3.2—5-26

Each interface on the PIX Firewall must be configured with an IP address. Use the **ip address** command for this purpose. If you make a mistake while entering this command, re-enter it with the correct information. The **clear ip** command resets all interface IP addresses to no IP address. In the figure, the dmz interface is configured with an IP address of 172.16.0.1 and a mask of 255.255.255.0.

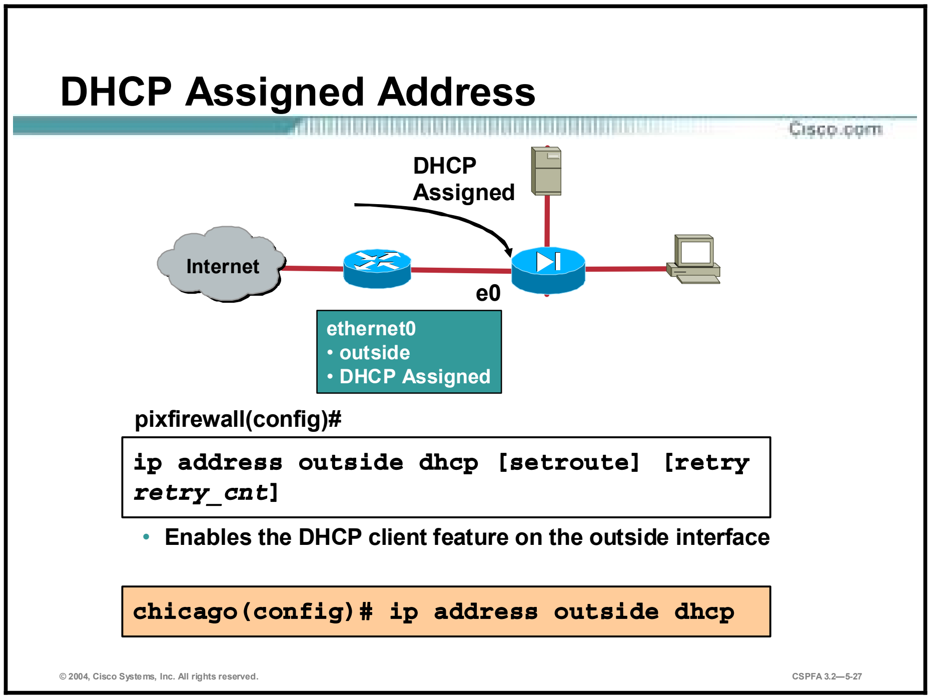
The syntax for the **ip address** commands is as follows:

ip address ***if_name ip_address*** **[*netmask*]**

show ip

clear ip

if_name	Describes the interface. This name is assigned by you, and must be used in all future configuration references to the interface.
ip_address	Specifies the IP address of the interface.
netmask	Specifies the network mask of IP address. If a network mask is not specified, the default network mask is assumed.



DHCP Assigned Address

pixfirewall(config)#

```
ip address outside dhcp [setroute] [retry
retry_cnt]
```

- Enables the DHCP client feature on the outside interface

```
chicago(config)# ip address outside dhcp
```

CSPFA 3.2—5-27

Instead of manually configuring an IP address on the PIX Firewall's outside interface, you can enable the PIX Firewall's Dynamic Host Configuration Protocol (DHCP) client feature to have the PIX Firewall dynamically retrieve an IP address from a DHCP server. With the PIX Firewall configured as a DHCP client, a DHCP server can configure the PIX Firewall's outside interface with an IP address, subnet mask, and optionally a default route. Use the **ip address dhcp** command to enable this feature. In the figure, the PIX Firewall is configured to receive an IP address on the outside interface via DHCP.

Use the **show ip address dhcp** command to view current information about your DHCP lease. Re-entering the **ip address dhcp** command with the **ip address outside dhcp** form enables you to release and renew a DHCP lease from the PIX Firewall. The **clear ip** command can also be used to release and renew the DHCP lease, but this clears the configuration of every PIX Firewall interface. To delete the DHCP leased IP address from the outside interface only, use the command **clear ip address outside dhcp**. The **debug dhcpc packet | detail | error** command provides debugging tools for the DHCP client feature.

The syntax for the **ip address** commands is as follows:

ip address *if_name ip_address* [*netmask*]

ip address outside dhcp [setroute] [retry *retry_cnt*]

show ip address dhcp

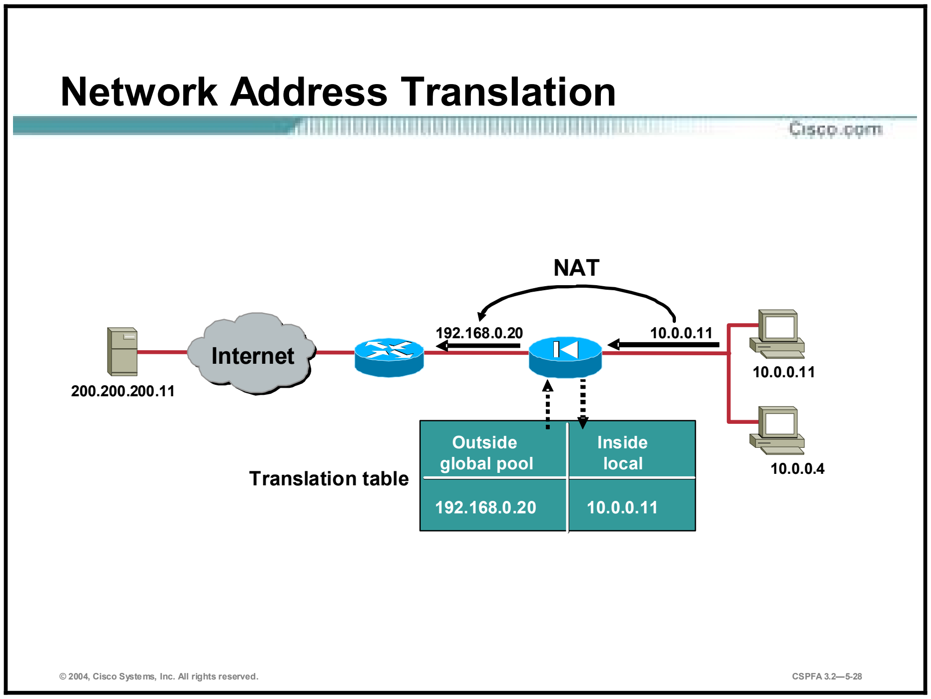
clear ip

clear ip address outside dhcp [setroute] [retry *retry_cnt*]

dhcp	Specifies that the PIX Firewall will use DHCP to obtain an IP address.

if_name	Describes the interface. This name is assigned by you, and must be used in all future configuration references to the interface.
ip_address	Specifies the IP address of the interface.
netmask	Specifies the network mask of ip_address. If a network mask is not specified, the default network mask is assumed.
outside	Specifies the interface from which the PIX Firewall will poll for information.
retry	Enables the PIX Firewall to retry a poll for DHCP information.
retry_cnt	Specifies the number of times the PIX Firewall will poll for DHCP information. The values available are 4 to 16. If no value is specified, the default is 4.
setroute	Tells the PIX Firewall to set the default route using the default gateway parameter the DHCP server returns.

Note The PIX Firewall DHCP client does not support failover configurations.



Network Address Translation

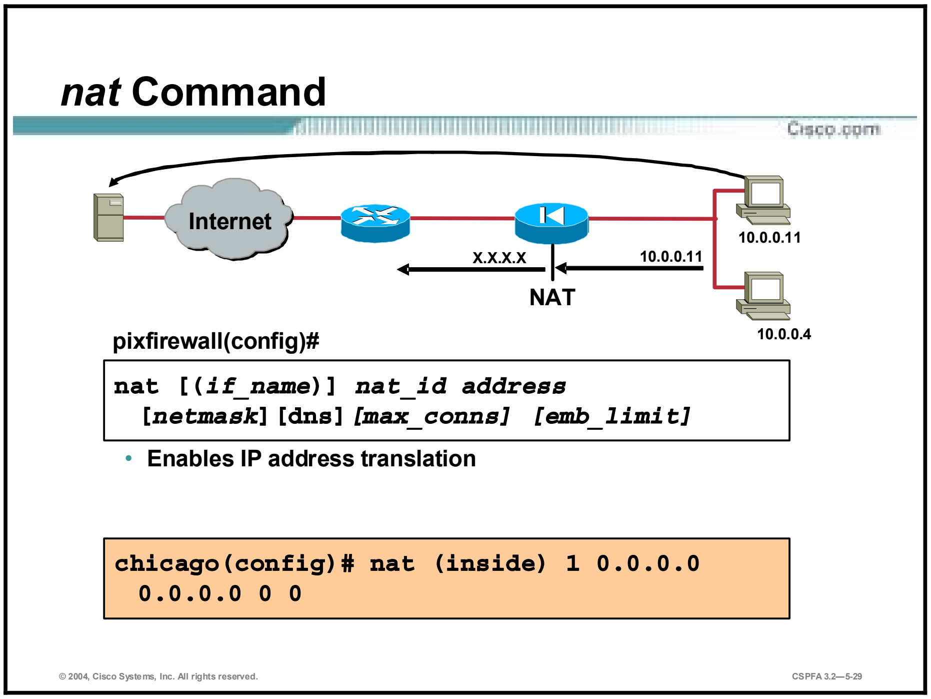
Outside global pool	Inside local
192.168.0.20	10.0.0.11

CSPFA 3.2—5-28

Network Address Translation (NAT) enables you to keep your internal IP addresses—those behind the PIX Firewall—unknown to external networks. NAT accomplishes this by translating the internal IP addresses, which are not globally unique, into globally accepted IP addresses before packets are forwarded to the external network. NAT is implemented in the PIX Firewall with the **nat** and **global** commands.

When an outbound IP packet that is sent from a device on the inside network reaches a PIX Firewall with NAT configured, the source address is extracted and compared to an internal table of existing translations. If the device's address is not already in the table, it is then translated. A new entry is created for that device, and it is assigned an IP address from a pool of global IP addresses. This global pool is configured with the **global** command. After this translation occurs, the table is updated and the translated IP packet is forwarded. After a user-configurable timeout period (or the default of three hours), during which there have been no translated packets for that particular IP address, the entry is removed from the table, and the global address is freed for use by another inside device.

In the figure, host 10.0.0.11 starts an outbound connection. The PIX Firewall translates the source address to 192.168.0.20. Packets from host 10.0.0.11 are seen on the outside as having a source address of 192.168.0.20.



nat Command

pixfirewall(config)#

```
nat [(if_name)] nat_id address
  [netmask] [dns] [max_conns]  [emb_limit]
```

- Enables IP address translation

```
chicago(config)# nat (inside) 1 0.0.0.0
  0.0.0.0 0 0
```

CSPFA 3.2—5-29

The first step in enabling NAT on a PIX Firewall is entering the **nat** command. The **nat** command can specify translation for a single host or a range of hosts. The nat command has two major components, *nat_id* and IP address or range of IP addresses. A *nat_id* is a number from 1 to 2147483647 which specifies the hosts for dynamic address translation. The dynamic addresses are chosen from a global address pool created with the **global** command. The **nat** command *nat_id* number must match the *nat_id* number in the **global** command if you want to use that specific global pool of IP addresses for the dynamic address translation.

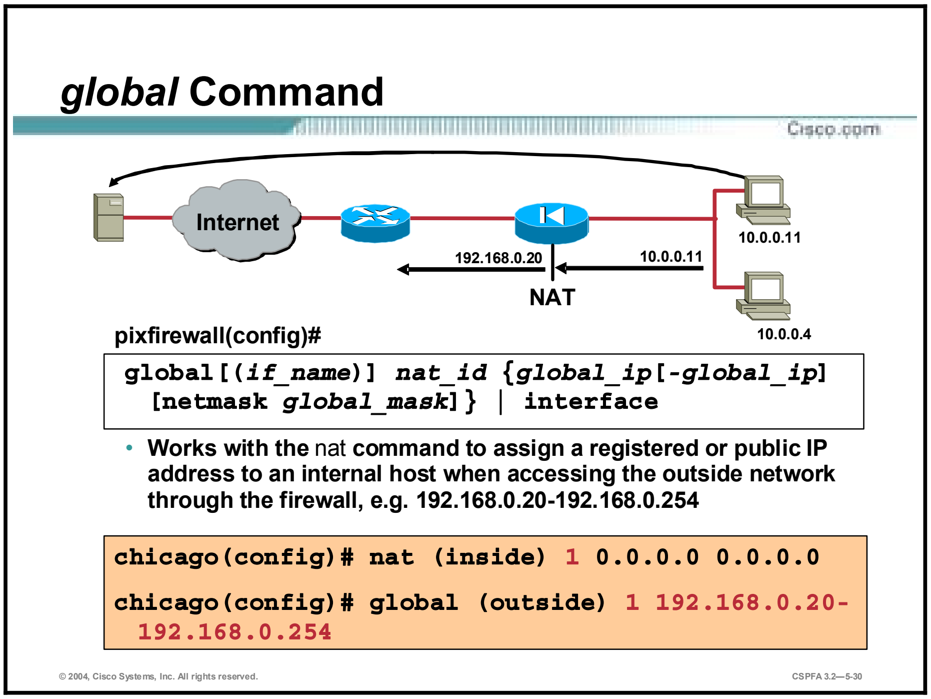
For example, the **nat (inside) 1 10.0.0.0 255.255.255.0** command means that all outbound connections from a host within the specified network, 10.0.0.0, can pass through the PIX Firewall (with address translation). The **nat (inside) 1 10.0.0.11 255.255.255.255** command means that only outbound connections originating from the inside host 10.0.0.11 are translated as the packet passes through the PIX Firewall. One can use 0.0.0.0 to allow all hosts to be translated. The 0.0.0.0 can be abbreviated as 0. As shown in the example, all inside hosts making outbound connections with the **nat (inside) 1 0.0.0.0 0.0.0.0** command are translated. The *nat_id* identifies the global address pool the PIX Firewall will use for the dynamic address translation. DNS, max-conns and emb-limit parameters are discussed in more depth in a later lesson.

The syntax for the **nat** command is as follows:

nat [(*if_name***)]** *nat_id address* **[***netmask***] [dns]/***max_conns] [emb_limit]*

if_name	The name of the interface attached to the network to be translated.
nat_id	A number greater than zero (0) that specifies the global address pool you want to use for dynamic address translation.
address	The IP address to be translated. You can use 0.0.0.0 to allow all hosts to start outbound connections. The 0.0.0.0 can be abbreviated as 0.

netmask	Network mask for the address. You can use 0.0.0.0 to allow all outbound connections to translate with IP addresses from the global pool.
dns	Specifies to use the created translation to rewrite the DNS address record.
max_conns	The maximum number of simultaneous connections the *local_ip* hosts are to allow. (Idle connections are closed after the idle timeout specified by the **timeout conn** command.)
emb_limit	The maximum number of embryonic connections per host. (An embryonic connection is a connection request that has not finished the necessary handshake between source and destination.) Set a small value for slower systems, and a higher value for faster systems. The default is 0, which allows unlimited embryonic connections.



global Command

pixfirewall(config)#

```
global[(if_name)] nat_id {global_ip[-global_ip]
  [netmask global_mask]} | interface
```

- Works with the nat command to assign a registered or public IP address to an internal host when accessing the outside network through the firewall, e.g. 192.168.0.20-192.168.0.254

```
chicago(config)# nat (inside) 1 0.0.0.0 0.0.0.0

chicago(config)# global (outside) 1 192.168.0.20-
  192.168.0.254
```

CSPFA 3.2—5-30

When an outbound IP packet that is sent from any device on the inside network reaches a PIX Firewall with NAT configured, the source address is extracted and compared to an internal table of existing translations. If the device's address is not already in the translation table, it is then translated. It is assigned an IP address from a pool of global IP addresses. In a PIX Firewall configuration, there may be more than one global pool configured. Each outbound network address translation is associated with a nat id. Each global pool has a corresponding nat id. The PIX uses the nat id of the outbound IP packet to identify which global pool of addresses to select a translation IP address from. The nat id of the outbound packet must match the nat id of the global pool. The PIX Firewall assigns addresses from the designated global pool starting from the low end to the high end of the range specified in the **global** command. The pool of global IP addresses is configured with the **global** command.

In the figure, host 10.0.0.11 starts an outbound connection. The nat id of the outbound packet is 1. In this instance, a global IP address pool of 192.168.0.20-254 is also identified with a nat id of 1. The PIX assigns an IP address of 192.168.0.20. It is the lowest available IP address of the range specified in the global command. Packets from host 10.0.0.11 are seen on the outside as having a source address of 192.168.0.20.

The syntax for the **global** command is as follows:

global [*(if_name)*] *nat_id {global_ip* [-*global_ip*] [netmask *global_mask*]} | interface

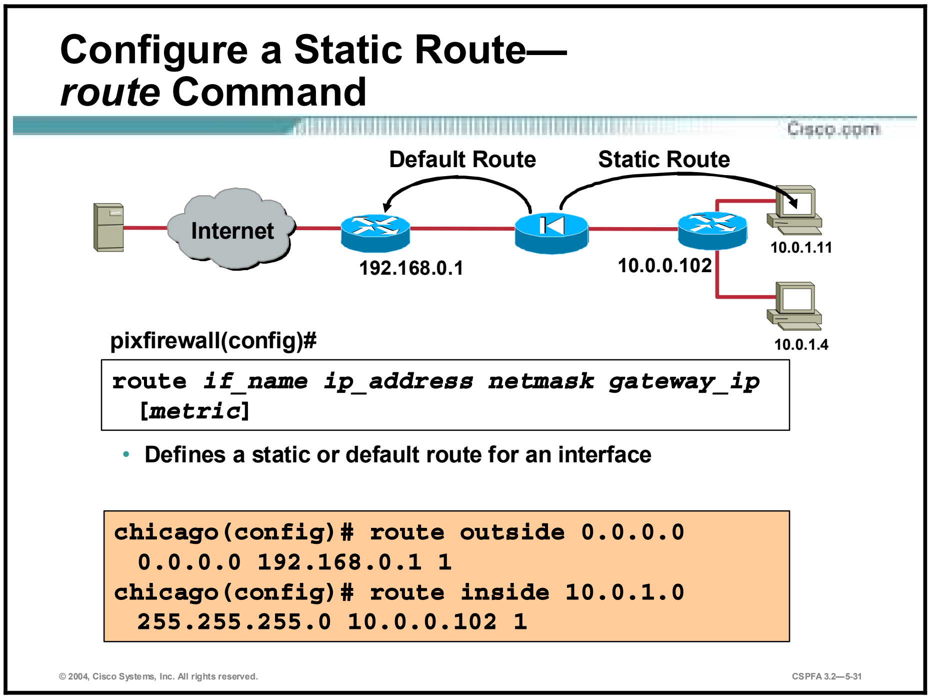
if_name	Describes the external network interface name where you will use the global addresses.
nat_id	Identifies the global pool and matches it with its respective **nat** command.
global_ip	Single IP addresses or the beginning IP address for a range of global IP addresses.
global_ip	A range of global IP addresses.

global_mask	The network mask for the global_ip address. If subnetting is in effect, use the subnet mask (for example, 255.255.255.128). If you specify an address range that overlaps subnets with the **netmask** command, this command will not use the broadcast or network address in the pool of global addresses. For example, if you use 255.255.255.128 and an address range of 192.150.50.20–192.150.50.140, the 192.150.50.127 broadcast address and the 192.150.50.128 network address will not be included in the pool of global addresses.
interface	Specifies PAT using the IP address at the interface.

If the **nat** command is used, the companion command, **global**, must be configured to define the pool of translated IP addresses.

Use the **no global** command to delete a global entry (for example, **no global (outside) 1 192.168.1.20–192.168.1.254 netmask 255.255.255.0**).

Note The PIX Firewall uses the global addresses to assign a virtual IP address to an internal NAT address. After adding, changing, or removing a global statement, use the **clear xlate** command to make the IP addresses available in the translation table.



Configure a Static Route— *route* Command

pixfirewall(config)#

```
route if_name ip_address netmask gateway_ip
  [metric]
```

- Defines a static or default route for an interface

```
chicago(config)# route outside 0.0.0.0
  0.0.0.0 192.168.0.1 1
chicago(config)# route inside 10.0.1.0
  255.255.255.0 10.0.0.102 1
```

CSPFA 3.2—5-31

Use the **route** command to enter a static route for an interface. To enter a default route, set *ip_address* and *netmask* to **0.0.0.0,** or the shortened form of 0. In the figure, a route command with the IP address of 0.0.0.0 identifies the command as the default route. The PIX transmits all destination packets not listed in its routing table out the outside interface to the router at IP address 192.168.0.1.

Create static routes to access specific networks beyond the locally connected networks. The effect of a static route is like stating "to send a packet to the specified network, give it to this router." For example, in the figure, PIX Firewall sends all packets destined to the 10.0.1.0 255.255.255.0 network out the inside interface to the router at IP address 10.0.0.102. This was accomplished by using the following static **route** command: **route inside 10.0.1.0 255.255.255.0 10.0.0.102 1.** The router knows how to route the packet to the destination network of 10.0.1.0.

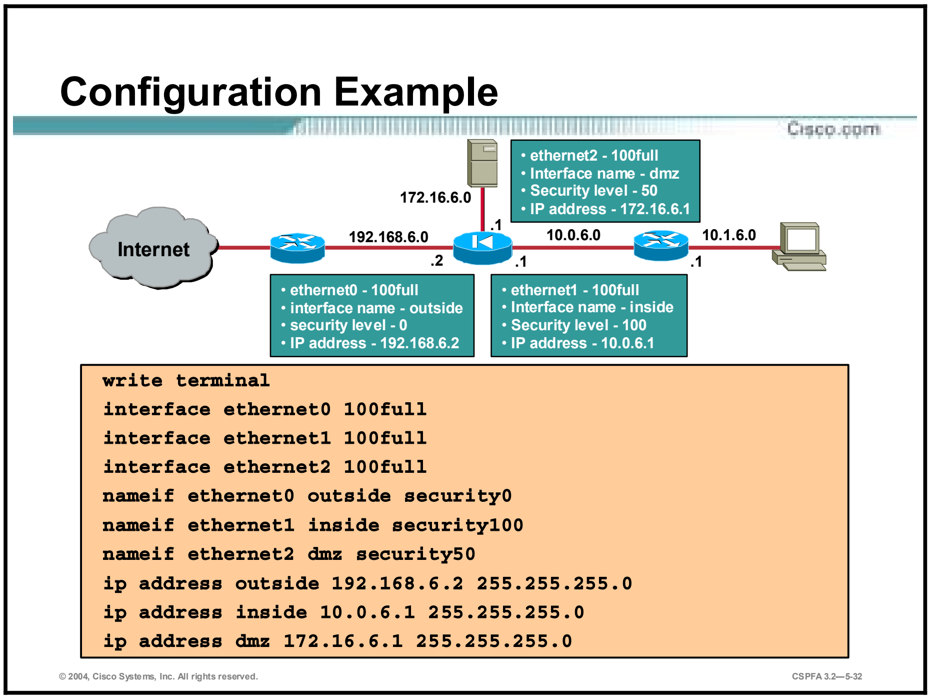
The syntax for the **route** command is as follows:

route ***if_name ip_address netmask gateway_ip*** [***metric***]

if_name	Describes the internal or external network interface name.
ip_address	Describes the internal or external network IP address. Use 0.0.0.0 to specify a default route. The 0.0.0.0 IP address can be abbreviated as 0.
netmask	Specifies a network mask to apply to ip_address. Use 0.0.0.0 to specify a default route. The 0.0.0.0 netmask can be abbreviated as 0.
gateway_ip	Specifies the IP address of the gateway router (the next hop address for this route).
metric	Specifies the number of hops to gateway_ip. If you are not sure, enter **1**. Your WAN administrator can supply this information or you can use a **traceroute** command to obtain the number of hops. The default is 1 if a metric is not specified.

All routes entered using the **route** command are stored in the configuration when it is saved.

You can use the IP address of one of the PIX Firewall's interfaces as the gateway address. If this is done, the PIX Firewall broadcasts an Address Resolution Protocol (ARP) request for the MAC address of the destination IP address in the packet instead of broadcasting a request for the MAC address of the gateway IP address.

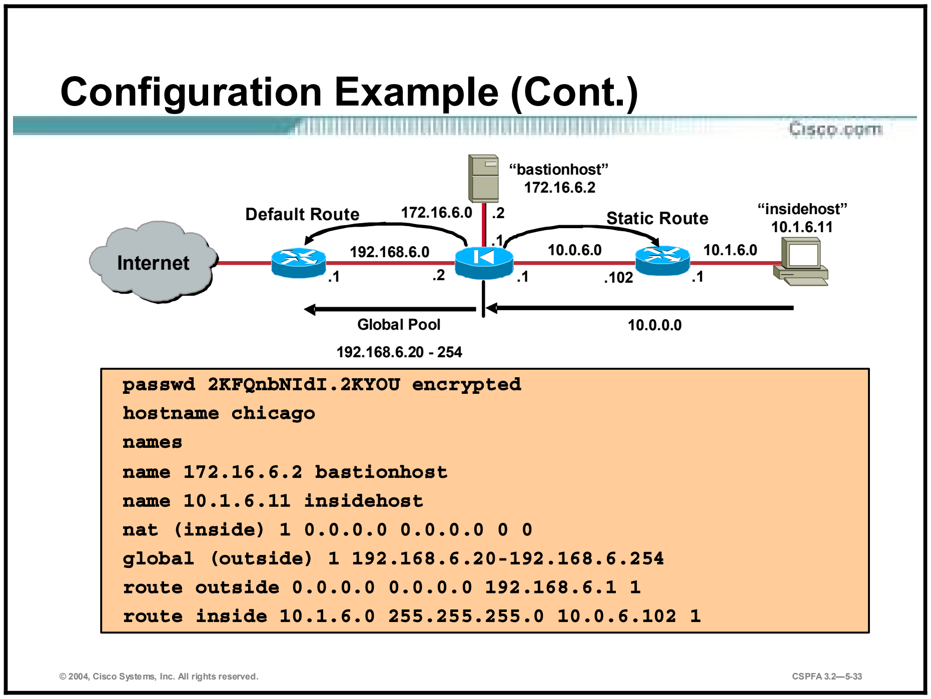


Configuration Example

```
write terminal
interface ethernet0 100full
interface ethernet1 100full
interface ethernet2 100full
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
ip address outside 192.168.6.2 255.255.255.0
ip address inside 10.0.6.1 255.255.255.0
ip address dmz 172.16.6.1 255.255.255.0
```

CSPFA 3.2—5-32

In the figure is a basic PIX Firewall configuration. There are three basic configuration commands in the example, **interface**, **nameif**, and **ip address**. Using the interface command, each of the interfaces is set for 100 Mbps full-duplex communications. Ethernet0 and Ethernet1 are set for their default name configuration, e.g. nameif ethernet0 outside security0. Using the **nameif** command, the additional interface, ethernet2, is configured as follows: nameif ethernet2 dmz security50. The last command is the **ip address** command. Each of the three interfaces is assigned an IP address and subnet mask, e.g. ip address outside 192.168.6.2 255.255.255.0. The configuration is continued on the next page.



Configuration Example (Cont.)

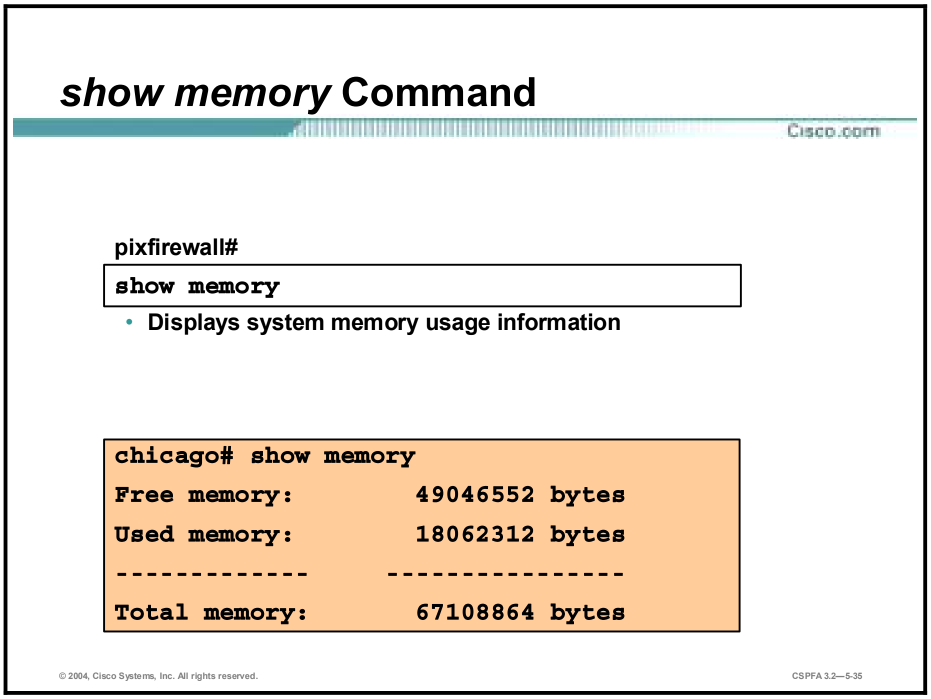
```
passwd 2KFQnbNIdI.2KYOU encrypted
hostname chicago
names
name 172.16.6.2 bastionhost
name 10.1.6.11 insidehost
nat (inside) 1 0.0.0.0 0.0.0.0 0 0
global (outside) 1 192.168.6.20-192.168.6.254
route outside 0.0.0.0 0.0.0.0 192.168.6.1 1
route inside 10.1.6.0 255.255.255.0 10.0.6.102 1
```

CSPFA 3.2—5-33

In the previous configuration example, the interfaces are configured using the nameif, interface, and IP address commands. In this figure, four features are configured, hostnames, names, NAT, and routes. Hostname allows the administrator to define the PIX CLI prompt, Chicago for example. The administrator can apply a name to any of the hosts, e.g. name 10.1.6.11 insidehost. Global and NAT commands enable the NAT feature in the PIX Firewall. In the example, outbound packets from any inside host, 0.0.0.0 0.0.0.0, are translated to one of the global pool IP addresses, 192.168.6.20 – 254. The last command is the route command. In the example a default route to the router at IP address 192.168.6.1 is added. The hosts on the 10.1.6.0 network by default cannot be reached by the PIX Firewall. To access these devices, a static route to the router at IP address 10.0.6.102 is defined. Any PIX packets bound for the 10.1.6.0 network are forwarded to the router at IP address 10.0.6.102.

Examining PIX Firewall Status

This topic contains the basic show commands needed to examine the status of the PIX Firewall.



show memory Command

Cisco.com

pixfirewall#

```
show memory
```

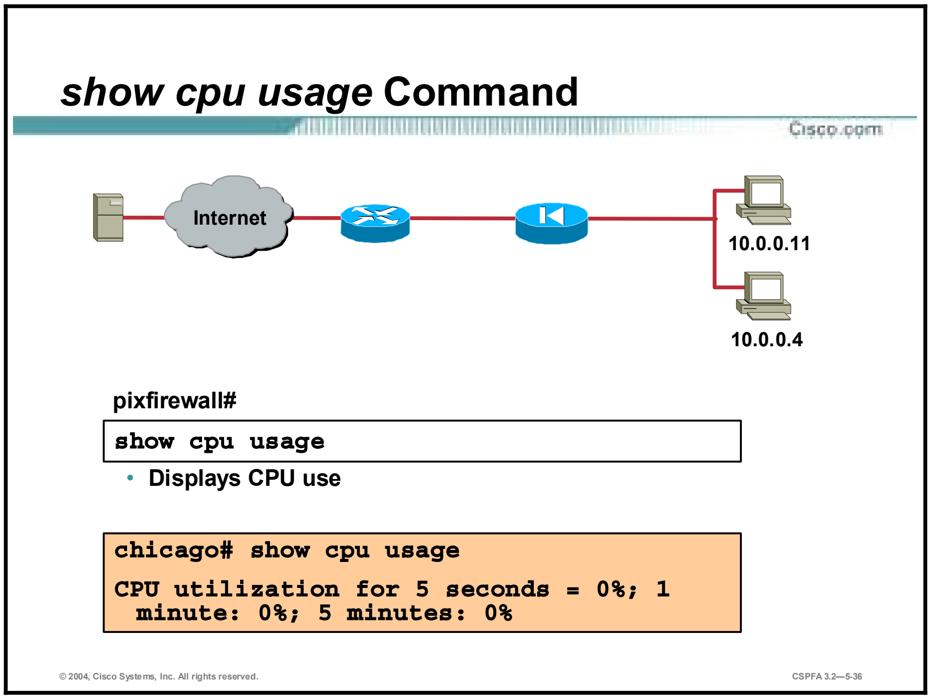
- Displays system memory usage information

```
chicago# show memory
Free memory:        49046552 bytes
Used memory:        18062312 bytes
-------------     ----------------
Total memory:       67108864 bytes
```

CSPFA 3.2—5-35

The **show** command enables you to view command information. There are several **show** commands that display system information. You can enter either **show** or **?** to view the names of the **show** commands and their descriptions.

The **show memory** command displays a summary of the maximum physical memory, current used memory, and current free memory available to the PIX Firewall operating system. The example in the figure shows sample output from the **show memory** command.



show cpu usage Command

Cisco.com
Internet
10.0.0.11
10.0.0.4

pixfirewall#

```
show cpu usage
```

- Displays CPU use

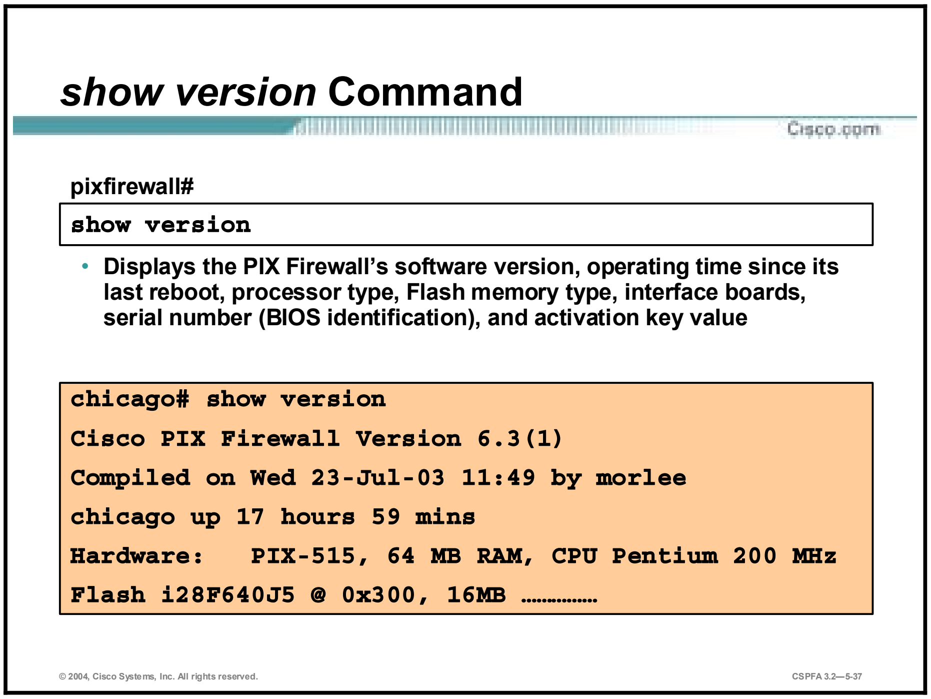
```
chicago# show cpu usage
CPU utilization for 5 seconds = 0%; 1
 minute: 0%; 5 minutes: 0%
```

CSPFA 3.2—5-36

The **show cpu usage** command displays CPU use. In the following example output for the **show cpu usage** command, 0% is the percentage of CPU used for 5 seconds, 0% is the average percentage of CPU use for one minute, and 0% is the average percentage utilization for five minutes:

```
CPU utilization for 5 seconds: 0%; 1 minute: 0%; 5 minutes: 0%
```

The percentage of usage is shown as NA (not available) if the usage is not available for any of the time intervals. This can happen if the user asks for CPU usage before the five-second, one-minute, or five-minute time interval has elapsed.



show version Command

Cisco.com

pixfirewall#

```
show version
```

- Displays the PIX Firewall's software version, operating time since its last reboot, processor type, Flash memory type, interface boards, serial number (BIOS identification), and activation key value

```
chicago# show version
Cisco PIX Firewall Version 6.3(1)
Compiled on Wed 23-Jul-03 11:49 by morlee
chicago up 17 hours 59 mins
Hardware:   PIX-515, 64 MB RAM, CPU Pentium 200 MHz
Flash i28F640J5 @ 0x300, 16MB …………
```

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—5-37

Use the **show version** command to display the PIX Firewall's software version, operating time since the last reboot, processor type, Flash memory type, interface boards, serial number (BIOS identification), and activation key value.

The serial number listed with the **show version** command in PIX Firewall software version 5.3 and higher is for the Flash memory BIOS. This number is different from the serial number on the chassis. To obtain a software upgrade, you need the serial number that appears in the **show version** command, not the chassis number.

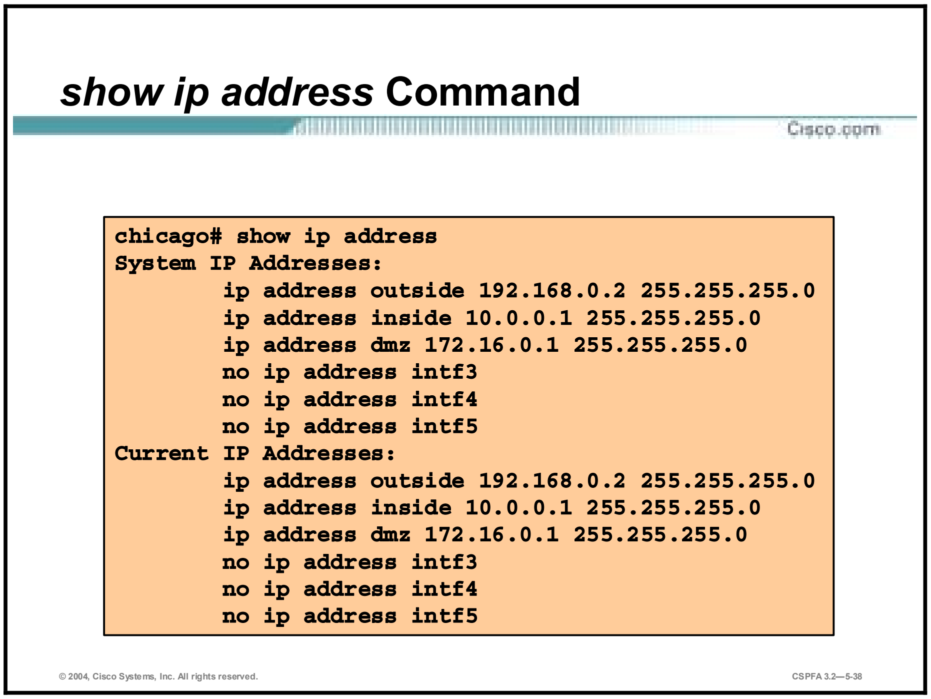
For PIX Firewall software version 6.2 and higher, the **show version** output appears as follows:

```
chicago# sh version


Cisco PIX Firewall Version 6.3(1)


Compiled on Wed 19-Mar-03 11:49 by morlee


chicago up 17 hours 59 mins


Hardware:   PIX-515, 64 MB RAM, CPU Pentium 200 MHz
Flash i28F640J5 @ 0x300, 16MB
BIOS Flash AT29C257 @ 0xfffd8000, 32KB


0: ethernet0: address is 0050.54ff.653a, irq 10
1: ethernet1: address is 0050.54ff.653b, irq 7
2: ethernet2: address is 00e0.b602.22c3, irq 11
3: ethernet3: address is 00e0.b602.22c2, irq 11
```

```
4: ethernet4: address is 00e0.b602.22c1, irq 11
5: ethernet5: address is 00e0.b602.22c0, irq 11
Licensed Features:
Failover:           Enabled
VPN-DES:            Enabled
VPN-3DES-AES:       Disabled
Maximum Interfaces: 6
Cut-through Proxy:  Enabled
Guards:             Enabled
URL-filtering:      Enabled
Inside Hosts:       Unlimited
Throughput:         Unlimited
IKE peers:          Unlimited

This PIX has an Unrestricted (UR) license.

Serial Number: 480350144 (0x1ca18fc0)
Running Activation Key: 0x51312668 0xb54b73ee 0x2e88e731 0xdf5e101a
Configuration last modified by enable_15 at 05:31:41.411 UTC Tue May
20 2003
```

In the above example, notice the following important bolded parameters:

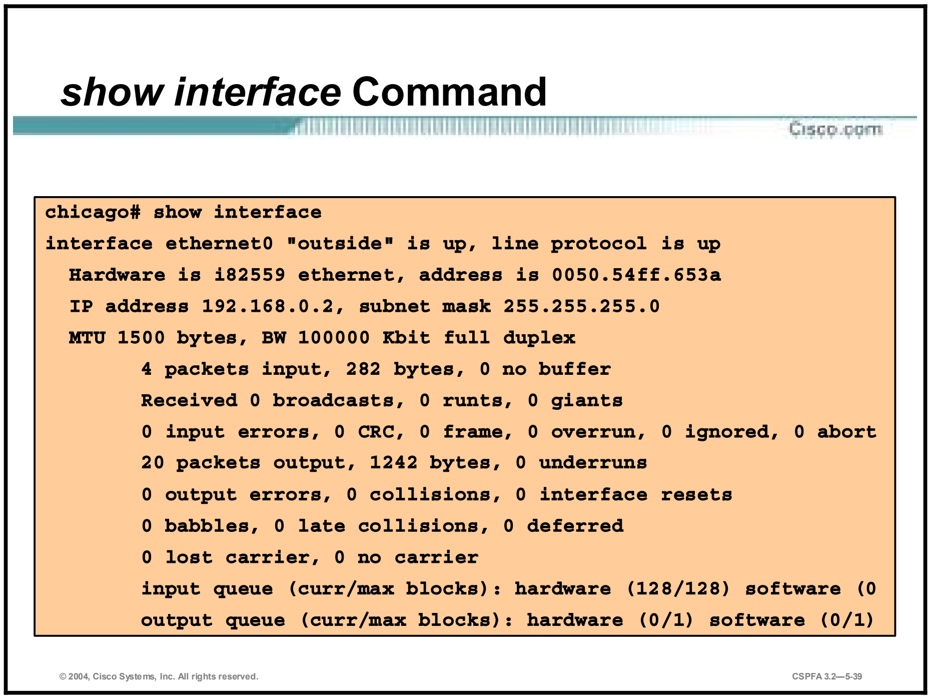
```
Hardware:   PIX-515, 64 MB RAM
Flash: 16MB
Licensed Features:
Failover:           Enabled
VPN-DES:            Enabled
VPN-3DES-AES:       Disabled
This PIX has an Unrestricted (UR) license.
Serial Number: 480350144
Running Activation Key: 0x51312668 0xb54b73ee 0x2e88e731 0xdf5e101a
```



show ip address Command

```
chicago# show ip address
System IP Addresses:
        ip address outside 192.168.0.2 255.255.255.0
        ip address inside 10.0.0.1 255.255.255.0
        ip address dmz 172.16.0.1 255.255.255.0
        no ip address intf3
        no ip address intf4
        no ip address intf5
Current IP Addresses:
        ip address outside 192.168.0.2 255.255.255.0
        ip address inside 10.0.0.1 255.255.255.0
        ip address dmz 172.16.0.1 255.255.255.0
        no ip address intf3
        no ip address intf4
        no ip address intf5
```

The **show ip address** command enables you to view which IP addresses are assigned to the network interfaces. The current IP addresses are the same as the system IP addresses on the failover active firewall. When the active firewall fails, the current IP addresses become that of the standby firewall. There is more on failover and failover IP addressing later in the course.



show interface Command

```
chicago# show interface
interface ethernet0 "outside" is up, line protocol is up
  Hardware is i82559 ethernet, address is 0050.54ff.653a
  IP address 192.168.0.2, subnet mask 255.255.255.0
  MTU 1500 bytes, BW 100000 Kbit full duplex
        4 packets input, 282 bytes, 0 no buffer
        Received 0 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        20 packets output, 1242 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128) software (0
        output queue (curr/max blocks): hardware (0/1) software (0/1)
```

CSPFA 3.2—5-39

The **show interface** command enables you to view network interface information. This is one of the first commands you should use when trying to establish connectivity.

The following are explanations of the information that is displayed after entering the **show interface** command:

- Ethernet—Indicates that you have used the **interface** command to configure the interface. The statement indicates whether the interface is inside or outside, and whether the interface is available (“up”) or not available (“down”).
- Line protocol up—A working cable is plugged into the network interface.
- Line protocol down—Either the cable plugged into the network interface is incorrect, or it is not plugged into the interface connector.
- Network interface type—Identifies the network interface.
- Interrupt vector—It is acceptable for interface cards to have the same interrupts because the PIX Firewall uses interrupts to get Token Ring information, but polls Ethernet cards.
- MAC address—Intel cards begin with “i” and 3Com cards begin with “3c”.
- MTU (maximum transmission unit)—The size in bytes that data can best be sent over the network.
- Packets input—Indicates that packets are being received in the PIX Firewall.
- Packets output—Indicates that packets are being sent from the PIX Firewall.
- Line duplex status—Indicates whether the PIX Firewall is running either full duplex (simultaneous packet transmission) or half duplex (alternating packet transmission).
- Line speed—10baseT is listed as 10000 Kbit. 100baseTX is listed as 100000 Kbit.

The following are explanations of **show interface** command output that can indicate interface problems:

- No buffer—Indicates the PIX Firewall is out of memory or slowed down due to heavy traffic and cannot keep up with the received data.
- Runts—Packets with less information than expected.
- Giants—Packets with more information than expected.
- CRC (cyclic redundancy check)—Packets that contain corrupted data (checksum error).
- Frame errors—Indicates framing errors.
- Ignored and aborted errors—This information is provided for future use, but is not currently checked; the PIX Firewall does not ignore or abort frames.
- Underruns—Occurs when the PIX Firewall is overwhelmed and cannot get data fast enough to the network interface card.
- Overruns—Occurs when the network interface card is overwhelmed and cannot buffer received information before more needs to be sent.
- Unicast rpf drops—Occurs when packets sent to a single network destination using reverse path forwarding are dropped.
- Output errors—(Maximum collisions.) The number of frames not transmitted because the configured maximum number of collisions was exceeded. This counter should only increment during heavy network traffic.
- Collisions—(Single and multiple collisions.) The number of messages retransmitted due to an Ethernet collision. This usually occurs on an overextended LAN when the Ethernet or transceiver cable is too long, there are more than two repeaters between stations, or there are too many cascaded multiport transceivers. A packet that collides is counted only once by the output packets.
- Interface resets—The number of times an interface has been reset. If an interface is unable to transmit for three seconds, the PIX Firewall resets the interface to restart transmission. During this interval, the connection state is maintained. An interface reset can also happen when an interface is looped back or shut down.
- Babbles—The transmitter has been on the interface longer than the time taken to transmit the largest frame. This counter is unused.
- Late collisions—The number of frames that were not transmitted because a collision occurred outside the normal collision window. A late collision is a collision that is detected late in the transmission of the packet. Normally, these should never happen. When two Ethernet hosts try to talk at once, they should collide early in the packet and both back off, or the second host should see that the first one is talking and wait.

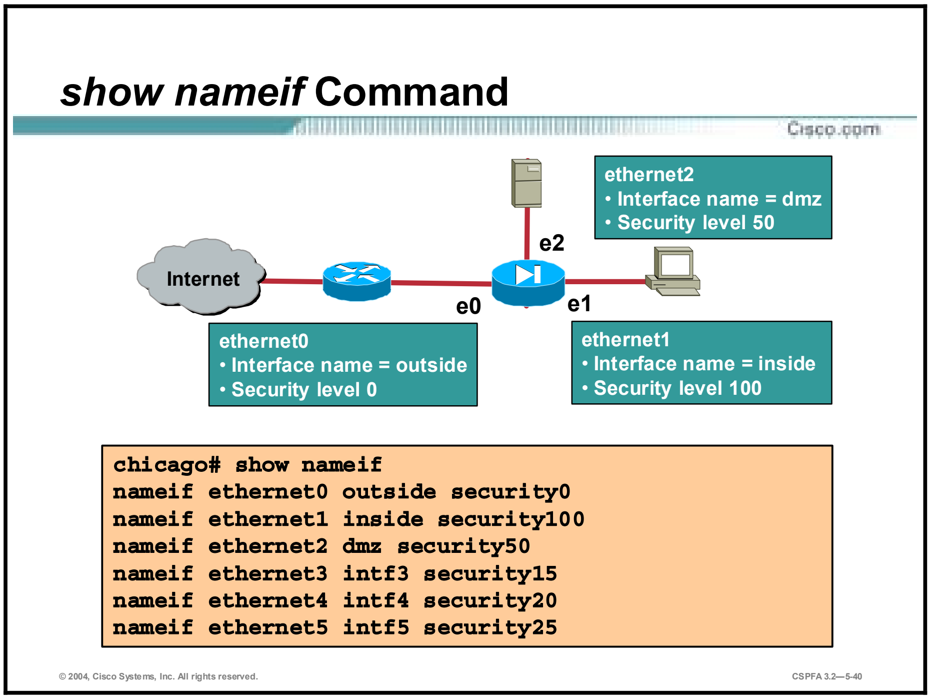
If you get a late collision, a device is jumping in and trying to send on the Ethernet while the PIX Firewall is partly finished sending the packet. The PIX Firewall does not resend the packet, because it may have freed the buffers that held the first part of the packet. This is not a real problem because networking protocols are designed to cope with collisions by resending packets. However, late collisions indicate a problem exists in your network. Common problems are large repeated networks and Ethernet networks running beyond the specification.

- Deferred—The number of frames that were deferred before transmission due to activity on the link.
- Lost carrier—The number of times the carrier signal was lost during transmission.

- No carrier—This counter is unused.
- Input queue—This is the input (receive) hardware and software queue.
  - Hardware—(Current and maximum blocks.) The number of blocks currently present on the input hardware queue, and the maximum number of blocks previously present on that queue.
  - Software—(Current and maximum blocks.) The number of blocks currently present on the input software queue, and the maximum number of blocks previously present on that queue.
- Output queue—This is the output (transmit) hardware and software queue.
  - Hardware—(Current and maximum blocks.) The number of blocks currently present on the output hardware queue, and the maximum number of blocks previously present on that queue.
  - Software—(Current and maximum blocks.) The number of blocks currently present on the output software queue, and the maximum number of blocks previously present on that queue.

Note The following counters are only valid for Ethernet interfaces: output errors, collisions, interface resets, babbles, late collisions, deferred, lost carrier, and no carrier.

Note Starting with PIX Firewall software version 6.0(1), FDDI, PL2, and Token Ring interfaces are not supported.



show nameif Command

ethernet2
- Interface name = dmz
- Security level 50

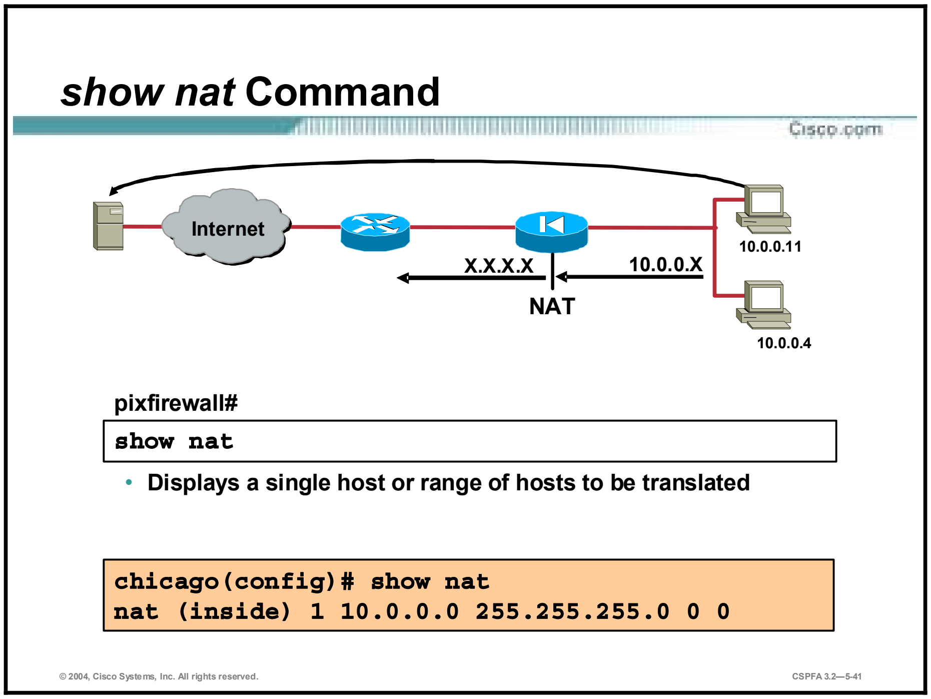
ethernet0
- Interface name = outside
- Security level 0

ethernet1
- Interface name = inside
- Security level 100

```
chicago# show nameif
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
nameif ethernet3 intf3 security15
nameif ethernet4 intf4 security20
nameif ethernet5 intf5 security25
```

CSPFA 3.2—5-40

Use the **show nameif** command to view the named interfaces. In the figure, the first two interfaces have the default names **inside** and **outside**. The **inside** interface has a default security level of 100; the **outside** interface has a default security level of 0. Ethernet2 is assigned a name of dmz with a security level of 50. Ethernet3 through Ethernet5 are unnamed and are assigned default security levels of 15, 20, and 25 respectively.



show nat Command

pixfirewall#

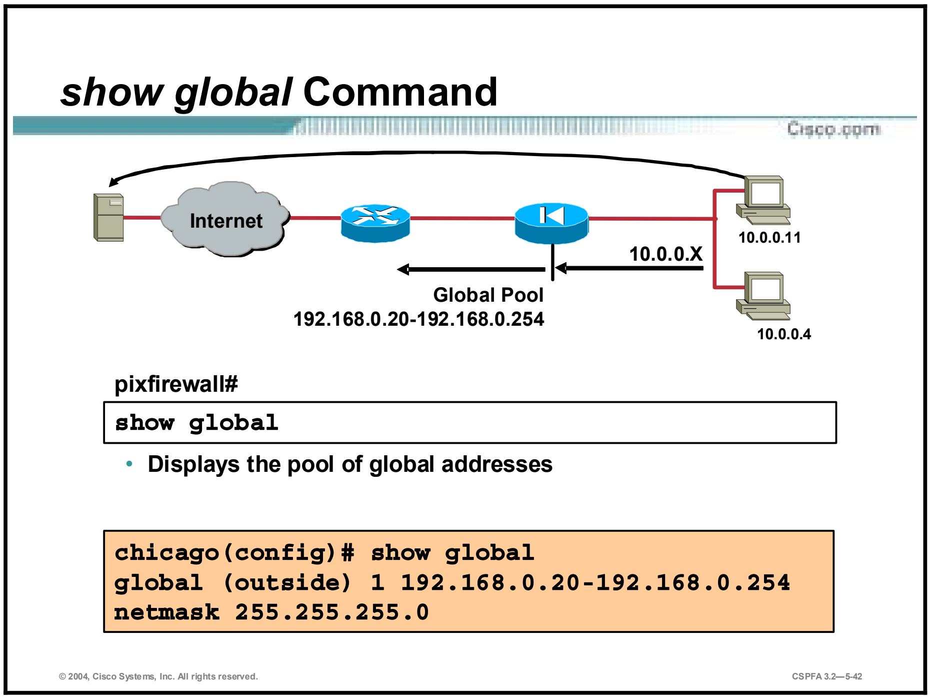
```
show nat
```

- Displays a single host or range of hosts to be translated

```
chicago(config)# show nat
nat (inside) 1 10.0.0.0 255.255.255.0 0 0
```

CSPFA 3.2—5-41

Use the show nat command to display a single host or range of hosts to be translated. In the figure, all hosts on the 10.0.0.0 network will be translated when traversing the PIX Firewall. The nat-id is 1.



show global Command

pixfirewall#

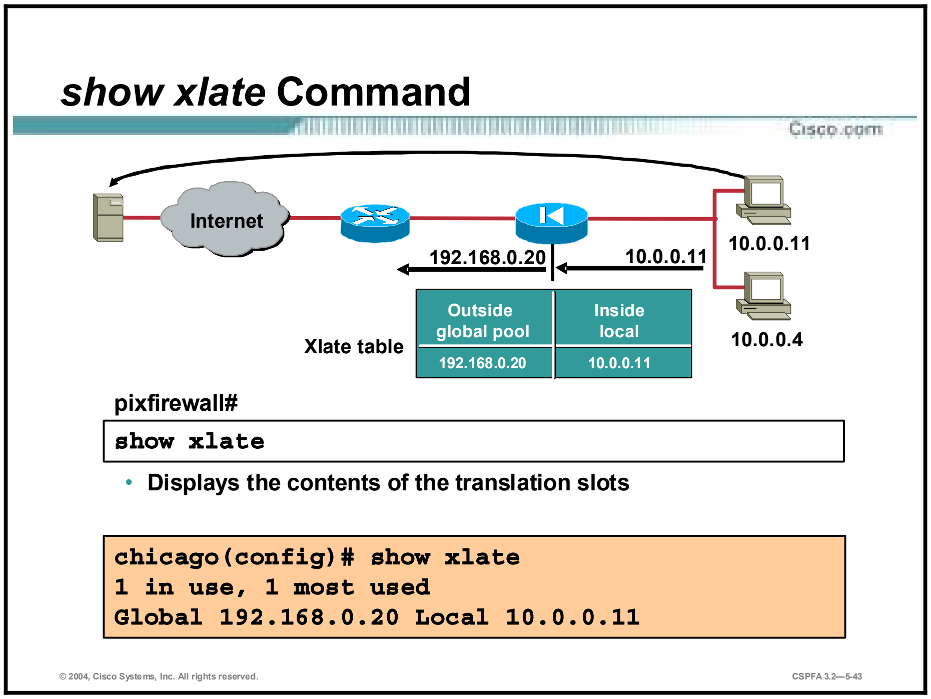
```
show global
```

- Displays the pool of global addresses

```
chicago(config)# show global
global (outside) 1 192.168.0.20-192.168.0.254
netmask 255.255.255.0
```

CSPFA 3.2—5-42

Show global displays the global pool(s) of addresses configured in the PIX Firewall. In the figure there is currently one pool configured. The pool is configured on the outside interface. The pool has an IP address range of 192.168.0.20 to 192.168.0.254. The nat_id is 1.



show xlate Command

Outside global pool	Inside local
192.168.0.20	10.0.0.11

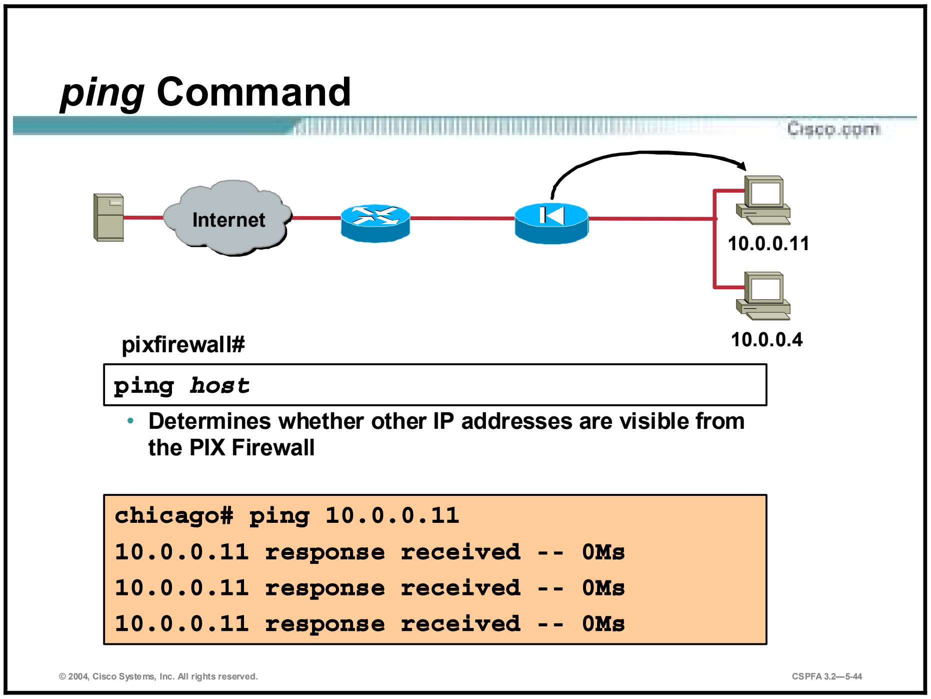
pixfirewall#

```
show xlate
```

- Displays the contents of the translation slots

```
chicago(config)# show xlate
1 in use, 1 most used
Global 192.168.0.20 Local 10.0.0.11
```

Show xlate displays the contents of the translation slot. In the figure, the number of currently used translations is 1 with a maximum count of 1. The current translation is a local IP address of 10.0.0.11 to a global IP address of 192.168.0.20.



ping Command

pixfirewall#

```
ping host
```

- Determines whether other IP addresses are visible from the PIX Firewall

```
chicago# ping 10.0.0.11
10.0.0.11 response received -- 0Ms
10.0.0.11 response received -- 0Ms
10.0.0.11 response received -- 0Ms
```

CSPFA 3.2—5-44

The **ping** command determines if the PIX Firewall has connectivity, or if a host is available (visible to the PIX Firewall) on the network. The command output shows if the ping was received. If the ping was received, then the host exists on the network. If the ping was not received, the command output displays “NO response received”. (At this time, one would use the **show interface** command to ensure that the PIX Firewall is connected to the network and is passing traffic.) By default, the **ping** command makes three attempts to reach an IP address.

If you want internal hosts to be able to ping external hosts, you must create an access-list for echo reply. This is discussed in another lesson. If you are pinging through the PIX Firewall between hosts or routers and the pings are not successful, use the **debug icmp trace** command to monitor the success of the ping.

After your PIX Firewall is configured and operational, you will not be able to ping the inside interface of the PIX Firewall from the outside network or from the outside interfaces of the PIX Firewall. If you can ping the inside networks from the inside interface and if you can ping the outside networks from the outside interface, the PIX Firewall is functioning normally and your routes are correct.

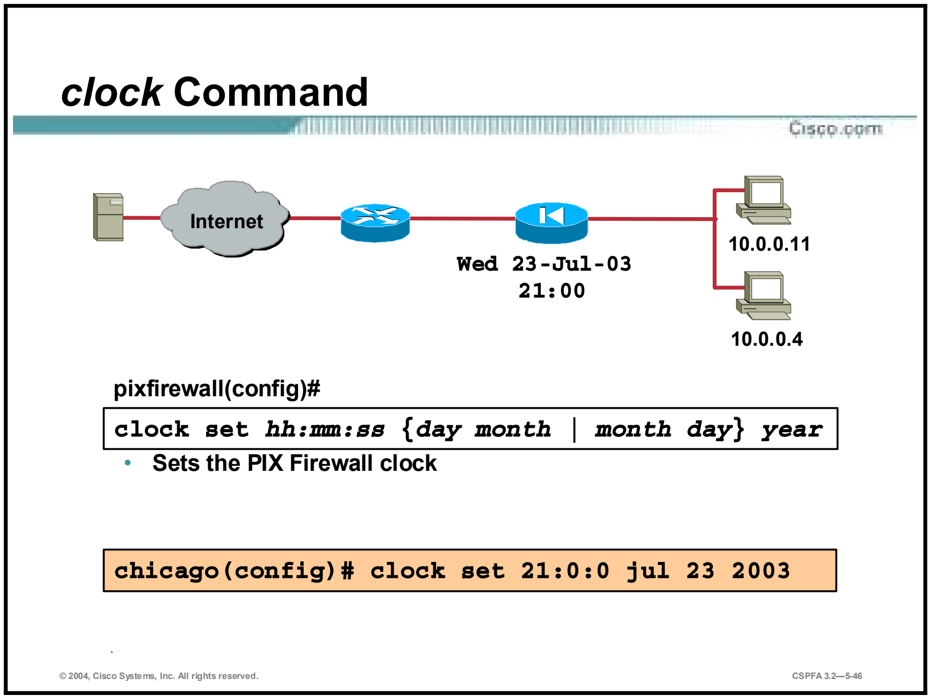
The syntax for the **ping** command is as follows:

ping [*if_name*] *host*

if_name	The network interface name. The address of the specified interface is used as the source address of the ping**.**
host	The name or IP address of the host being pinged.

Time Setting and NTP Support

This topic explains how to set the clock on the PIX Firewall and synchronize the times of devices operating over an IP data network.



clock Command

pixfirewall(config)#

```
clock set hh:mm:ss {day month | month day} year
```

- Sets the PIX Firewall clock

```
chicago(config)# clock set 21:0:0 jul 23 2003
```

CSPFA 3.2—5-46

The **clock** command sets the PIX Firewall clock. It enables you to specify the time, month, day, and year. The clock setting is retained in memory when the power is off by a battery on the PIX Firewall's motherboard. The PIX Firewall generates Syslog messages for system events and can log these messages to a Syslog server. If you want the messages to contain a time stamp value, you must enter the **logging timestamp** command. The **logging timestamp** command requires that the **clock set** command be used to ensure that the correct time appears on the Syslog messages. Syslog and its corresponding commands are explained in another lesson.

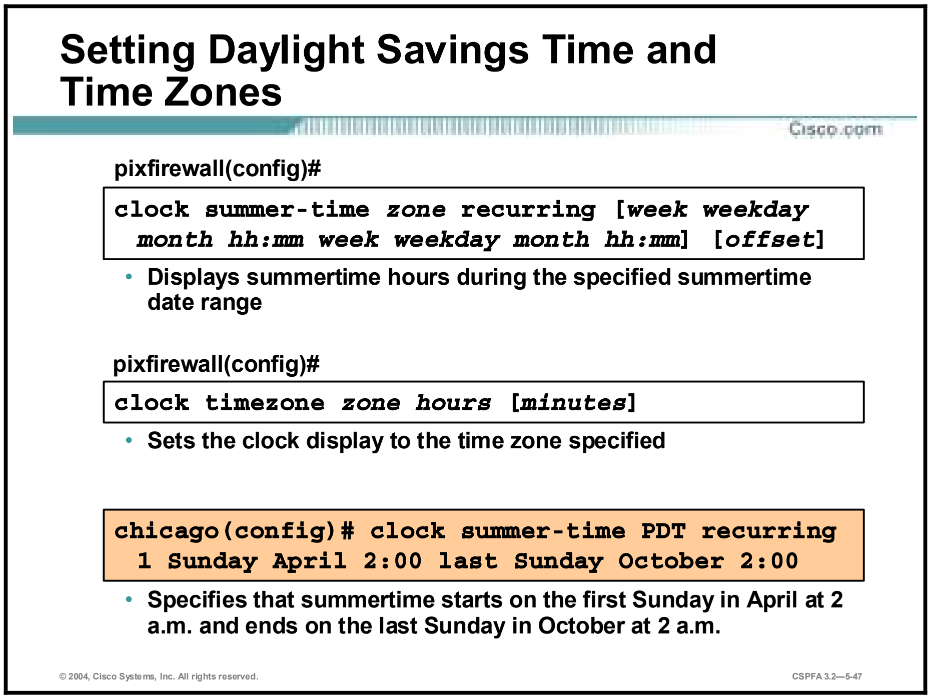
It is also important to ensure that the clock is correctly set if you use Public Key Infrastructure (PKI), which uses digital certificates for authentication of Virtual Private Network (VPN) peers. The Cisco PKI protocol uses the clock to make sure that a Certificate Revocation List (CRL) is not expired. Otherwise, the Certificate Authority (CA) may reject or allow certificates based on an incorrect timestamp. The lifetimes of certificates and CRLs are checked in coordinated universal time (UTC). If you are using certificates with IPSec for VPNs, set the PIX Firewall clock to UTC timezone to ensure that CRL checking works correctly. VPNs are discussed in another lesson.

You can view the time with the **show clock** command, which displays the time, time zone, day, and full date. You can remove the **clock set** command with the **clear clock** command.

The syntax for the **clock set** command is as follows:

clock set *hh:mm:ss month day year*

day	The day of the month to start, from 1 to 31.
hh:mm:ss	The hour: minutes: seconds expressed in 24-hour time (for example, 20:54:00 for 8:54 p.m.). Zeros can be entered as a single digit; for example, 21:0:0.
month	The month expressed as the first three characters of the month (for example, apr for April).
year	The year expressed as four digits (for example, 2000).



Setting Daylight Savings Time and Time Zones

pixfirewall(config)#

```
clock summer-time zone recurring [week weekday
  month hh:mm week weekday month hh:mm] [offset]
```

- Displays summertime hours during the specified summertime date range

pixfirewall(config)#

```
clock timezone zone hours [minutes]
```

- Sets the clock display to the time zone specified

```
chicago(config)# clock summer-time PDT recurring
  1 Sunday April 2:00 last Sunday October 2:00
```

- Specifies that summertime starts on the first Sunday in April at 2 a.m. and ends on the last Sunday in October at 2 a.m.

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—5-47

Although the PIX Firewall clock does not adjust itself for daylight savings time changes, you can configure it to display daylight savings time by using the **clock summer-time** command. The **summer-time** keyword causes the PIX Firewall to automatically switch to summertime (for display purposes only). The **recurring** keyword indicates that summer time should start and end on the days specified by the values that follow it. If no values are specified, the summer time rules default to United States rules.

You can also specify the exact date and times with the **date** version of the **clock summer-time** command. In the following example, daylight savings time (summer time) is configured to start on April 7, 2002 at 2 a.m. and end on October 27, 2002 at 2 a.m.

```
pixfirewall (config)# clock summer-time PDT date 7 April 2002 2:00
27 October 2002 2:00
```

Use the **clock timezone** command to set the time zone. The **clock timezone** command sets the time zone for display purposes only. Internally, the time is kept in UTC. The **no** form of the command is used to set the time zone to Coordinated Universal Time (UTC). The **clear clock** command removes summer time settings and sets the time zone to UTC.

The syntax for the **clock** commands is as follows:

clock summer-time *zone* recurring [*week weekday month hh:mm week weekday month hh:mm*] [*offset*]

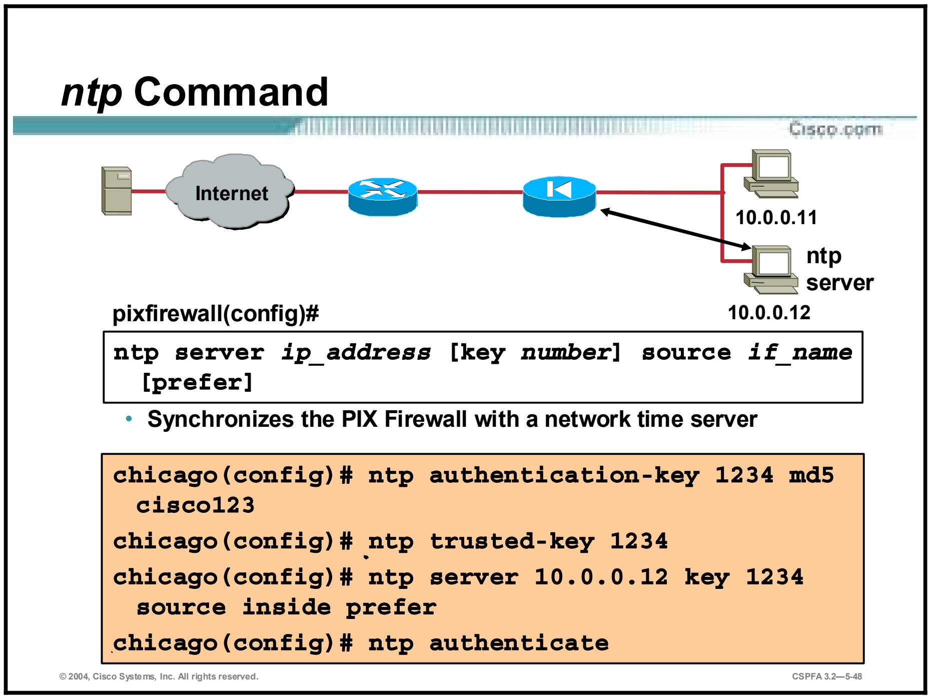
clock summer-time zone **date** {*day month* | *month day*} *year hh:mm* {*day month* | *month day*} *year hh:mm* [*offset*]

clock timezone *zone hours* [*minutes*]

show clock [detail]

summer-time	The **clock summer-time** command displays summertime hours during the specified summertime date range. This command affects the clock display time only.

zone	The name of the time zone.
recurring	Specifies the start and end dates for local summer "daylight savings" time. The first date entered is the start date and the second date entered is the end date. (The start date is relative to UTC and the end date is relative to the specified summer time zone.) If no dates are specified, United States Daylight Savings Time is used. If the start date month is after the end date month, the summer time zone is accepted and assumed to be in the Southern Hemisphere.
week	Specifies the week of the month. Enter 1, 2, 3, or 4 to specify the first, second, third, or fourth week of the month. Use first or last to specify a partial week at the beginning or end of a month. For example, week 5 of any month is specified by using last.
weekday	Specifies the day of the week. Enter **Monday**, **Tuesday**, **Wednesday**, **Thursday**, **Friday**, **Saturday**, or **Sunday**.
month	Specifies the month. Enter the first three characters of the month (for example, apr for April).
hh:mm	The hour and minutes expressed in 24-hour time (for example, 20:54 for 8:54 p.m.). Zeros can be entered as a single digit (for example, 21:0).
offset	The number of minutes to add during summertime. The default is 60 minutes.
date	Used as an alternative to the recurring form of the **clock summer-time** command. It specifies that summertime should start on the first date entered and end on the second date entered. If the start date month is after the end date month, the summer time zone is accepted and assumed to be in the Southern Hemisphere.
day	The day of the month to start. Enter a number from 1 to 31.
year	The year expressed as four digits (for example, 2000). The year range supported for the **clock** command is 1993 to 2035.
timezone	The **clock timezone** command sets the clock display to the time zone specified. It does not change internal PIX Firewall time, which remains UTC.
hours	The hours of offset from UTC.
minutes	The minutes of offset from UTC.
detail	Displays the clock source and current summertime settings.



ntp Command

Cisco.com
Internet
10.0.0.11
ntp
server
10.0.0.12

pixfirewall(config)#

```
ntp server ip_address [key number] source if_name
  [prefer]
```

- Synchronizes the PIX Firewall with a network time server

```
chicago(config)# ntp authentication-key 1234 md5
  cisco123
chicago(config)# ntp trusted-key 1234
chicago(config)# ntp server 10.0.0.12 key 1234
  source inside prefer
chicago(config)# ntp authenticate
```

CSPFA 3.2—5-48

The **ntp server** command synchronizes the PIX Firewall with the network timeserver you specify. You can configure the PIX Firewall to require authentication before synchronizing with the NTP server. To enable and support authentication, there are several forms of the ntp command that work with the ntp server command. The following are the ntp command forms and their uses:

- **ntp server**—Specifies an NTP server. The PIX Firewall listens for NTP packets (port 123) only on interfaces that have an NTP server configured. NTP packets that are not responses from a request by the PIX Firewall are dropped.
- **ntp authenticate**—Enables NTP authentication.
- **ntp authentication-key**—Defines the authentication keys for the **ntp** commands. If authentication is used, the PIX Firewall and NTP server must be configured with the same key.
- **ntp trusted-key**—Defines one or more key numbers that the NTP server needs to provide in its NTP packets for the PIX Firewall to accept synchronization with the NTP server. Use this command if NTP authentication is enabled.

You can use the **show ntp** command to display the current NTP configuration and the **show ntp status** command to display the NTP clock information. The **clear ntp** command removes the NTP configuration, including disabling authentication and removing all authentication keys and NTP server designations.

The syntax for the **ntp** commands is as follows:

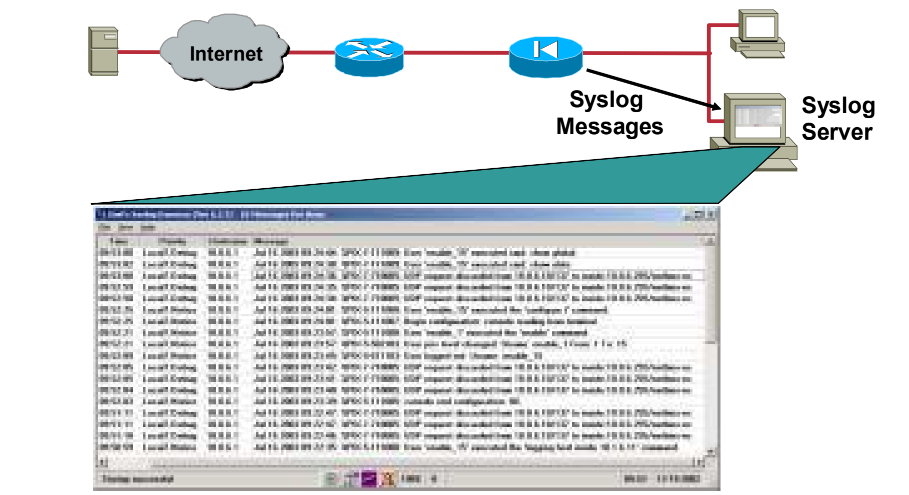
```
ntp authenticate

ntp authentication-key number md5 value

ntp server ip_address [key number] source if_name [prefer]

ntp trusted-key number
```

authenticate	Enables NTP authentication. If enabled, the PIX Firewall requires authentication before synchronizing with an NTP server.
authentication-key	Defines the authentication keys for use with other NTP commands.
if_name	Specifies the interface to use to send packets to the network timeserver.
ip_address	The IP address of the network timeserver with which to synchronize.
key	Specifies the authentication key.
md5	The authentication algorithm.
number	The authentication key number (1 to 4294967295).
prefer	Designates the network timeserver specified as the preferred server with which to synchronize time.
server	The network timeserver.
source	Specifies the network time source.
trusted-key	Specifies the trusted key against which to authenticate.
value	The key value, an arbitrary string of up to 32 characters. The key value is displayed as "***********" when the configuration is viewed by write terminal or show tech-support.

Syslog Configuration

This topic explains how to configure the PIX Firewall to send Syslog messages to its buffer and to a Syslog server.

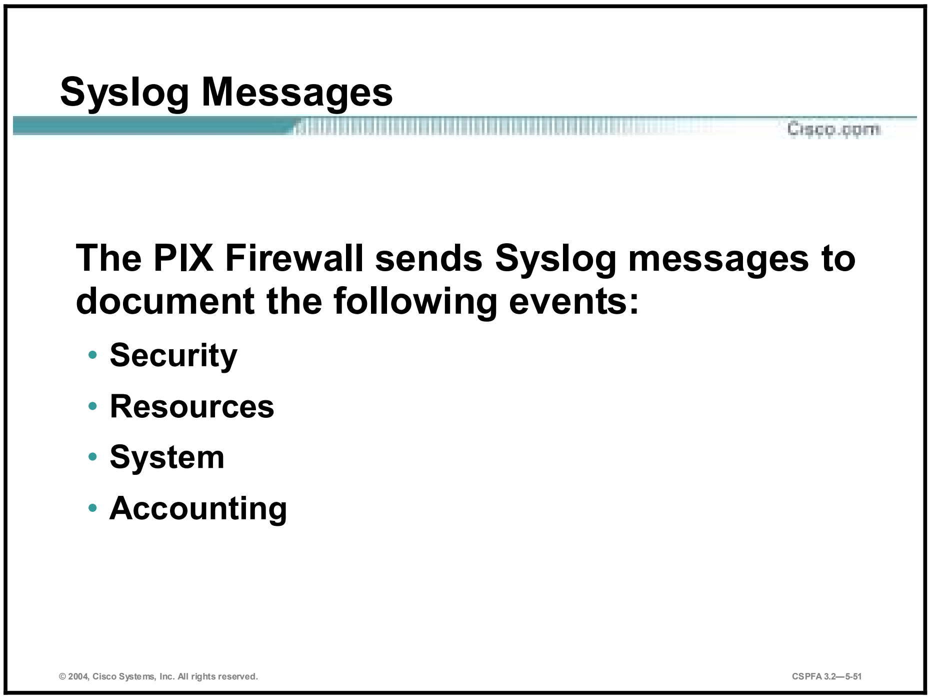
Configure Syslog Output to a Syslog Server



CSPFA 3.2—5-50

The PIX Firewall generates Syslog messages for system events, such as alerts and resource depletion. Syslog messages may be used to create log files, or displayed on the console of a designated Syslog host. If you do not already have a Syslog server at your place of business, you can download a copy of the software from the Cisco Connection Online (CCO) web site.

The PIX Firewall can send Syslog messages to any Syslog server. In the event that all Syslog servers or hosts are offline, the PIX Firewall Syslog Server stores up to 100 messages in its memory. Subsequent messages that arrive overwrite the buffer starting from the first line.



Syslog Messages

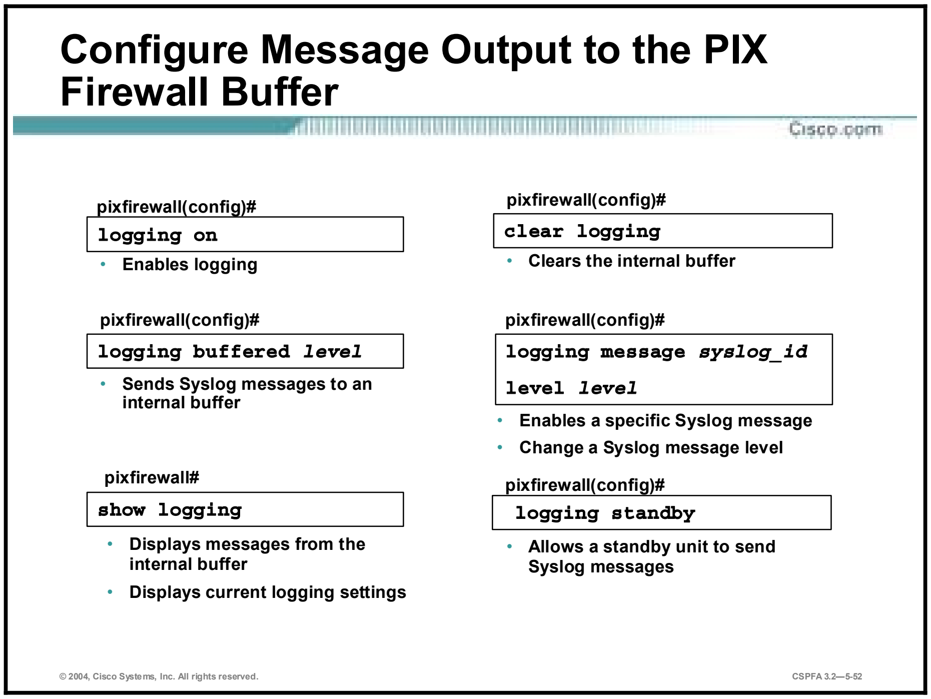
The PIX Firewall sends Syslog messages to document the following events:

- Security
- Resources
- System
- Accounting

CSPFA 3.2—5-51

The PIX Firewall sends Syslog messages to document the following events:

- Security—Dropped UDP packets and denied TCP connections.
- Resources—Notification of connection and translation slot depletion.
- System—Console and Telnet logins and logouts, and when the PIX Firewall reboot.
- Accounting—Bytes transferred per connection.



Configure Message Output to the PIX Firewall Buffer

Cisco.com

pixfirewall(config)#

```
logging on
```

- Enables logging

pixfirewall(config)#

```
logging buffered level
```

- Sends Syslog messages to an internal buffer

pixfirewall#

```
show logging
```

- Displays messages from the internal buffer
- Displays current logging settings

pixfirewall(config)#

```
clear logging
```

- Clears the internal buffer

pixfirewall(config)#

```
logging message syslog_id
level level
```

- Enables a specific Syslog message
- Change a Syslog message level

pixfirewall(config)#

```
logging standby
```

- Allows a standby unit to send Syslog messages

CSPFA 3.2—5-52

After enabling logging with the **logging on** command, use the **logging buffered** command to specify what Syslog messages appear on the PIX Firewall console as each message occurs. The level parameter of the **logging buffered** command enables you to limit the types of messages that appear on the console.

Note It is recommended that you do not use logging console command in production mode because its use degrades PIX Firewall performance.

Use the **show logging** command to list the current message buffer.

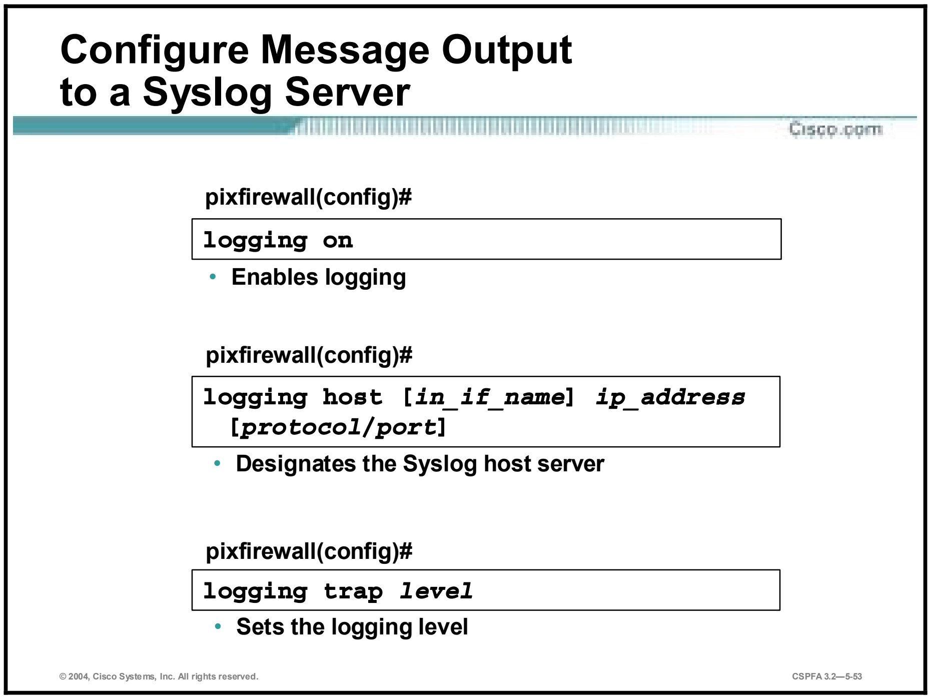
Use the **clear logging** command to clear the message buffer. New messages append to the end of the buffer.

Use the **logging message** command to specify a message to be allowed. Use with the **no** command to suppress a message. All Syslog messages are permitted unless explicitly disallowed. The “PIX Startup begin” message cannot be blocked and neither can more than one message per command statement. Specify a message number to disallow or allow. If a message is listed in the Syslog as %PIX-1-101001, use “101001” as the syslog_id. Refer to *the System Log Messages for the Cisco Secure PIX Firewall Version 6.2* guide for message numbers. Another logging message option is to change the level of a syslog message. To change the level of a syslog message use the logging message <syslog_id> level <level> form of the command.

Use the **logging standby** command to allow a failover standby PIX Firewall to send Syslog messages. This option is disabled by default. Enabling it ensures that the standby PIX Firewall’s Syslog messages stay synchronized if failover occurs; however, it doubles the amount of traffic on the Syslog server. You can disable this feature with the **no logging standby** command.

The following table shows the commands used in configuring a Syslog server:

buffered	Sends Syslog messages to an internal buffer that can be viewed with the **show logging** command.
show	Lists which logging options are enabled. If the **logging buffered** command is in use, the **show logging** command lists the current message buffer.
clear	Clears the buffer for use with the **logging buffered** command.
message	Specifies a message to be allowed. Use with the **no** command to suppress a message. Use the **clear logging disabled** command to reset the disallowed messages to the original set. Use the **show message disabled** command to list the suppressed messages you specified with the **no logging message** command. All Syslog messages are permitted unless explicitly disallowed. The “PIX Startup begin" message cannot be blocked and neither can more than one message per command statement.
syslog_id	Specifies a message number to disallow or allow. If a message is listed in Syslog as %PIX-1-101001, use "101001" as the *syslog_id*. Refer to the *System Log Messages for the Cisco Secure PIX Firewall Version 6.3* guide for message numbers. PIX Firewall documentation is available online: www.cisco.com/univercd/cc/td/doc/product/iaabu/pix.
standby	Allows the failover standby PIX Firewall to send Syslog messages. This option is disabled by default. You can enable it to ensure that the standby PIX Firewall’s Syslog messages stay synchronized if failover occurs. However, this option causes twice as much traffic on the Syslog server. This feature can be disabled with the **no logging standby** command.



Configure Message Output to a Syslog Server

pixfirewall(config)#

```
logging on
```

- Enables logging

pixfirewall(config)#

```
logging host [in_if_name] ip_address
  [protocol/port]
```

- Designates the Syslog host server

pixfirewall(config)#

```
logging trap level
```

- Sets the logging level

CSPFA 3.2—5-53

Complete the following to send messages to a Syslog server:

Step 1 Enable logging with the **logging on** command. Use the **no logging on** command to disable sending messages.

Step 2 Designate a host to receive the messages with the **logging host** command:

logging host [*in_if_name***]** *ip_address* **[***protocol* **/** *port***]**

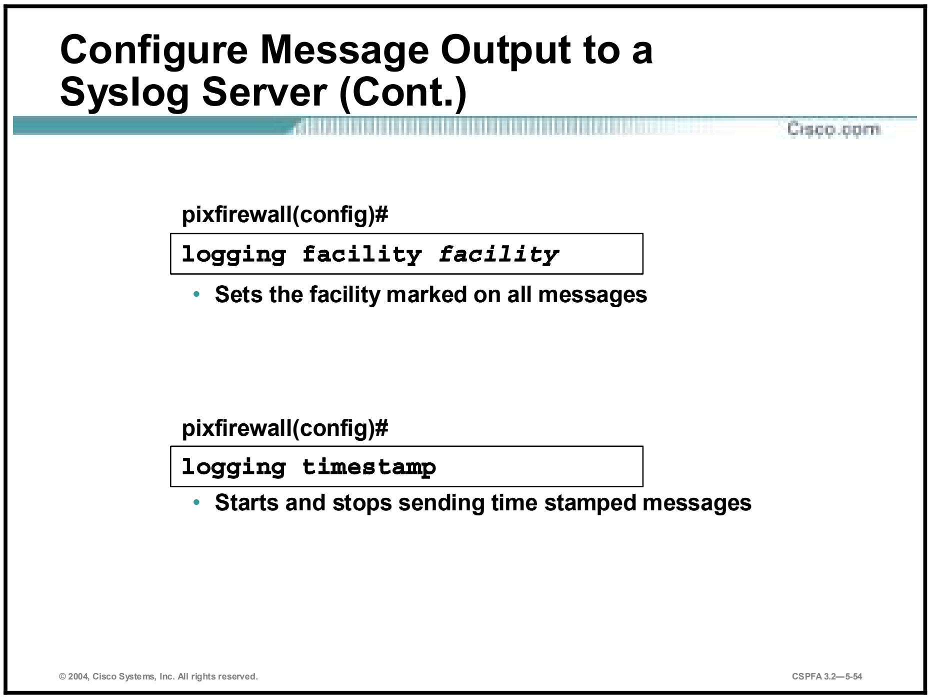
Replace in_if_name with the interface on which the server exists and ip_address with the IP address of the host. If the Syslog server is receiving messages on a non-standard port, you can replace a protocol with a UDP, and port with the new port value. The default protocol is UDP with a default port of 514, and the allowable range for changing the value is 1025 through 65535. You can also specify TCP with a default of 1470, and the allowable range is 1025 through 65535.

Note Multiple logging host commands are allowed for the PIX Firewall to send Syslog messages to multiple servers, but only one protocol, UDP or TCP, is permitted for a specific Syslog server. A subsequent command statement overrides the previous one.

Note The PIX Firewall sends only TCP Syslog messages to the PIX Firewall Syslog server.

Step 3 Specify the logging levels that will be forwarded to the Syslog server with the **logging trap** command:

logging trap ***level***



Configure Message Output to a Syslog Server (Cont.)

pixfirewall(config)#

```
logging facility facility
```

- Sets the facility marked on all messages

pixfirewall(config)#

```
logging timestamp
```

- Starts and stops sending time stamped messages

CSPFA 3.2—5-54

Step 4 Specify the logging facility to which the PIX Firewall will assign the Syslog messages with the **logging facility** command:

logging facility *facility*

Because network devices share the eight facilities, the logging facility command enables you to set the facility marked on all messages.

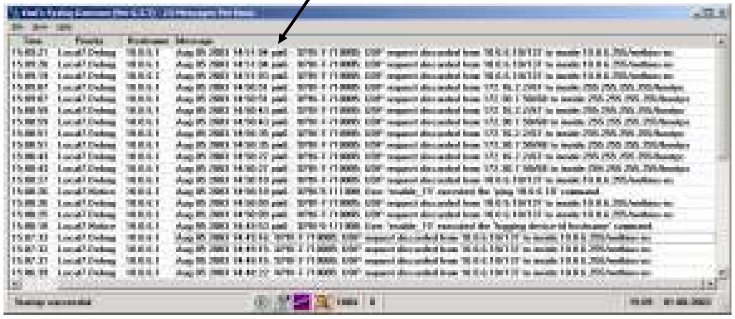
Step 5 If you want to send time-stamped messages to a Syslog server, use the **logging timestamp** command to enable time stamping. Use the **no logging timestamp** command to disable time-stamp logging.

The following table shows the commands used in configuring a Syslog server:

level	Specifies the Syslog message level as a number or string. The level you specify means that you want to use that level and those less than the level. Possible number and string level values follow: 0–emergencies—System unusable messages 1–alerts—Take immediate action 2–critical—Critical condition 3–errors—Error message 4–warnings—Warning message 5–notifications—Normal but significant condition 6–informational—Information message 7–debugging—Debug messages and log FTP commands and WWW URLs
syslog_id	Specifies a message number to disallow or allow. If a message is listed in Syslog as%PIX-1-101001, use "101001" as the syslog_id. Refer to Cisco PIX Firewall System Log Messages for message numbers.
in_if_name	Interface on which the Syslog server resides.

ip_address	Syslog server's IP address.
protocol	The protocol over which the Syslog message is sent; either TCP or UDP. PIX Firewall only sends TCP Syslog messages to the PIX Firewall Syslog server. You can only view the port and protocol values you previously entered by using the **write terminal** command and finding the command in the listing—the TCP protocol is listed as 6 and the UDP protocol is listed as 17.
port	The port from which the PIX Firewall sends either UDP or TCP Syslog messages. This must be same port at which the Syslog server listens. For the UDP port, the default is 514 and the allowable range for changing the value is 1025 through 65535. For the TCP port, the default is 1470, and the allowable range is 1025 through 65535. TCP ports only work with the PIX Firewall Syslog server.
facility	Specifies the Syslog facility. There are eight facilities: LOCAL0 (16) through LOCAL7 (23). The default is LOCAL4 (20). Hosts file the messages based on the facility number in the message.

Logging Device-ID



pixfirewall(config)#

```
logging device-id {hostname | ipaddress| string}
```

- Displays a unique device ID in syslog messages

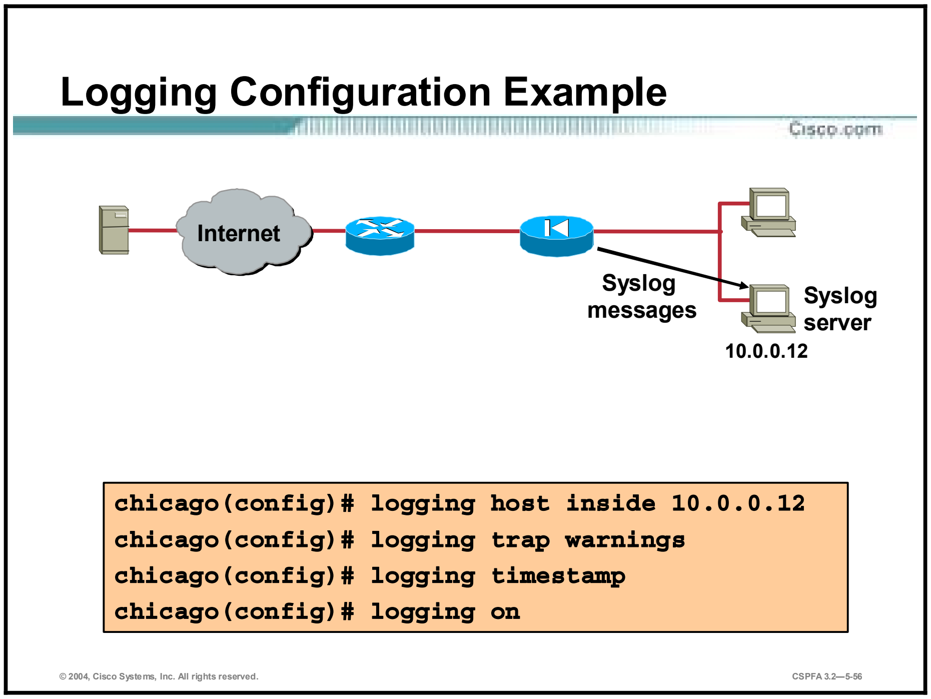
```
pix6(config)# logging device-id hostname
```

CSPFA 3.2—5-55

The **logging device-id** command displays a unique device ID in non-EMBLEM format syslog messages that are sent to the syslog server. This command is available in PIX Firewall software Version 6.2.2.115 and higher. If enabled, the PIX Firewall displays the device ID in all non-EMBLEM-formatted syslog messages. The device ID part of the syslog message is viewed through the syslog server only and not directly on the firewall.

If the **ipaddress** option is used, the device ID becomes the specified PIX Firewall interface IP address, regardless of the interface from which the message is sent. This provides a single consistent device ID for all messages sent from the device.

In the figure, logging device-id hostname command is configured. Notice the hostname, pix6, appears in the syslog output directly after the date and timestamp.



Logging Configuration Example

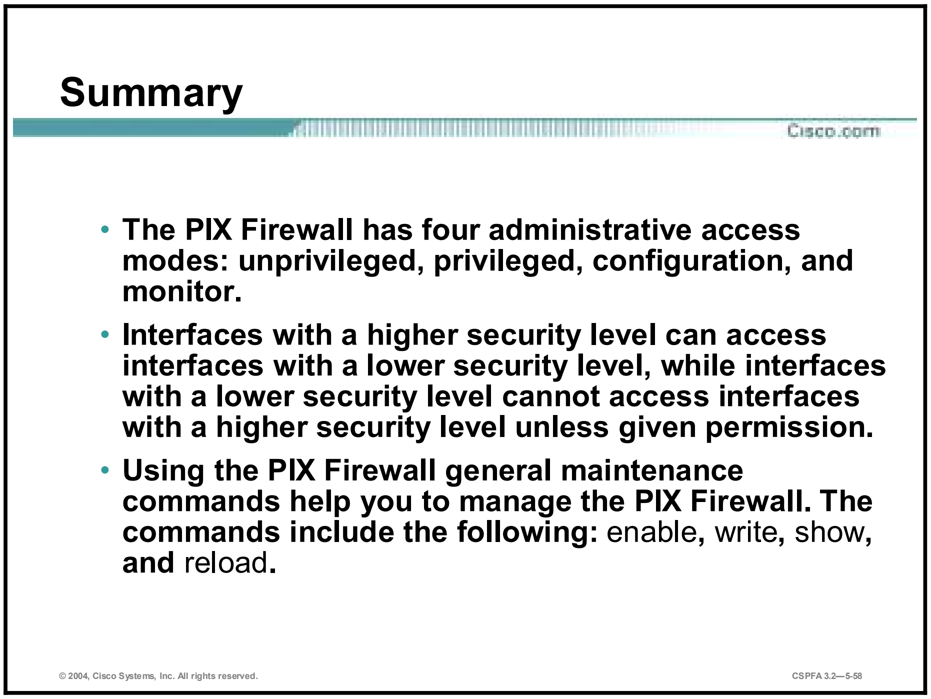
```
chicago(config)# logging host inside 10.0.0.12
chicago(config)# logging trap warnings
chicago(config)# logging timestamp
chicago(config)# logging on
```

CSPFA 3.2—5-56

In the figure, the PIX Firewall is configured to send the logging messages to Syslog server 10.0.0.12. The messages sent will consist of warning messages and higher. Each message is time stamped. Lastly, logging is turned on.

Summary

This topic summarizes what you learned in this lesson.

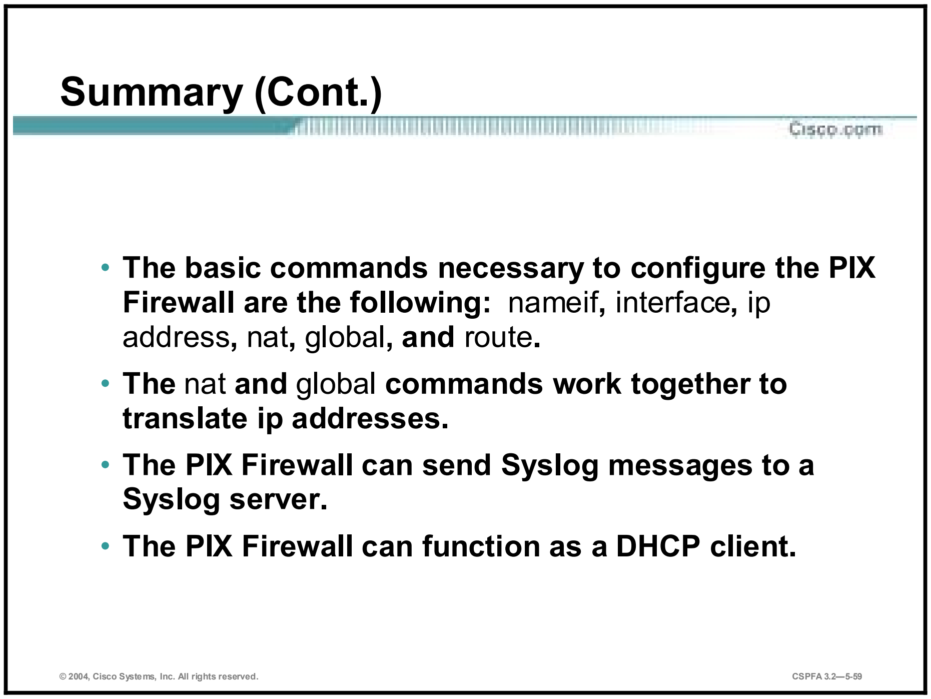


Summary

Cisco.com

- **The PIX Firewall has four administrative access modes: unprivileged, privileged, configuration, and monitor.**
- **Interfaces with a higher security level can access interfaces with a lower security level, while interfaces with a lower security level cannot access interfaces with a higher security level unless given permission.**
- **Using the PIX Firewall general maintenance commands help you to manage the PIX Firewall. The commands include the following:** enable**,** write**,** show**, and** reload**.**

CSPFA 3.2—5-58



Summary (Cont.)

- **The basic commands necessary to configure the PIX Firewall are the following:** nameif**,** interface**,** ip address**,** nat**,** global**, and** route**.**
- **The** nat **and** global **commands work together to translate ip addresses.**
- **The PIX Firewall can send Syslog messages to a Syslog server.**
- **The PIX Firewall can function as a DHCP client.**

CSPFA 3.2—5-59

Lab Exercise—Configure the PIX Firewall and Execute General Maintenance Commands

Complete the following lab exercises to practice what you have learned in this lesson.

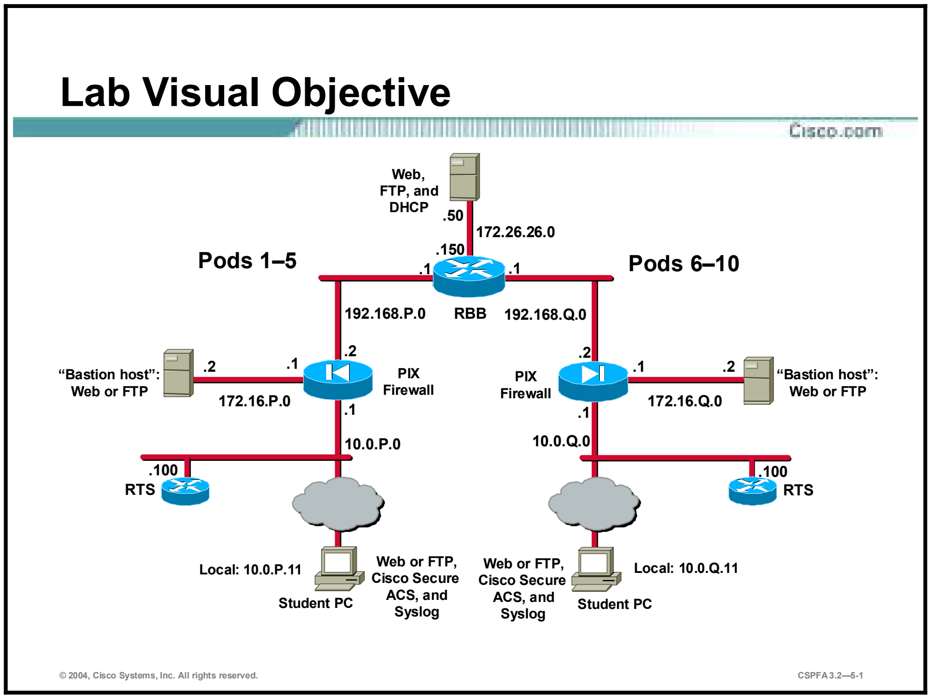
Objectives

In this lab exercise, you will complete the following tasks:

- Execute general commands.
- Configure PIX Firewall interfaces.
- Configure global addresses, NAT, and routing for inside and outside interfaces.
- Test the inside, outside, and DMZ interface connectivity.
- Configure Syslog output.
- Configure Syslog output to a Syslog server.

Visual Objective

The following illustration displays the lab topology for your classroom environment.



Lab Visual Objective

CSPFA 3.2—5-1

Task 1—Execute General Commands

Complete the steps in this task to familiarize yourself with the general maintenance commands. Observe the output of the commands carefully. The instructor will provide you with the procedures for access to the PIX Firewall console port, as this will vary according to your lab connectivity. After you access the PIX Firewall console port, the PIX Firewall prompt appears. If the prompt that appears is not the configuration mode prompt, enter configuration mode. Ask your instructor for assistance if necessary.

```
pixfirewall(config)#
```

Step 1 Erase the PIX Firewall's default configuration. When prompted to confirm, press **Enter**.

```
pixfirewall(config)# write erase
Erase PIX configuration in flash memory? [confirm] <Enter>
```

Step 2 Reboot the PIX Firewall. When prompted to confirm, press **Enter**.

```
pixfirewall(config)# reload
Proceed with reload? [confirm} <Enter>
```

Step 3 The PIX Firewall prompts you to bootstrap it through interactive prompts. Press **Control>Z** to escape. The unprivileged mode prompt appears.

```
Pre-configure PIX Firewall through interactive prompts [yes]?
<Control+Z>
pixfirewall>
```

Step 4 Display the list of help commands:

```
pixfirewall> ?
```

Step 5 Enter the privileged mode of the PIX Firewall. When prompted for a password, press **Enter**.

```
pixfirewall> enable
    password:
pixfirewall#
```

Step 6 Display the list of help commands:

```
pixfirewall# ?
```

Step 7 Use the **write terminal** command to display the PIX Firewall configuration on the terminal screen:

```
pixfirewall# write terminal
Building configuration . . .
:  Saved
:
PIX Version 6.3(1)
interface ethernet0 auto shutdown
interface ethernet1 auto shutdown
interface ethernet2 auto shutdown
interface ethernet3 auto shutdown
interface ethernet4 auto shutdown
interface ethernet5 auto shutdown
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 intf2 security4
nameif ethernet3 intf3 security6
nameif ethernet4 intf4 security8
nameif ethernet5 intf5 security10
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname pixfirewall
fixup protocol ftp 21
fixup protocol h323 h225 1720
fixup protocol h323 ras 1718-1719
fixup protocol http 80
fixup protocol ils 389
fixup protocol rsh 514
fixup protocol rtsp 554
fixup protocol sip 5060
fixup protocol sip udp 5060
fixup protocol skinny 2000
fixup protocol smtp 25
fixup protocol sqlnet 1521
names
pager lines 24
mtu outside 1500
mtu inside 1500
mtu intf2 1500
mtu intf3 1500
mtu intf4 1500
mtu intf5 1500
no ip address outside
no ip address inside
no ip address intf2
no ip address intf3
no ip address intf4
no ip address intf5
ip audit info action alarm
ip audit attack action alarm
no failover
failover timeout 0:00:00
failover poll 15
no failover ip address outside
no failover ip address inside
no failover ip address intf2
no failover ip address intf3
no failover ip address intf4
no failover ip address intf5
pdm history enable
arp timeout 14400
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 rpc 0:10:00 h225
1:00:00
timeout h323 0:05:00 mgcp 0:05:00 sip 0:30:00 sip_media 0:02:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server LOCAL protocol local
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
floodguard enable
```

```
telnet timeout 5
ssh timeout 5
console timeout 0
terminal width 80
Cryptochecksum:d41d8cd98f00b204e9800998ecf8427e
: end
[OK]
```

Step 8 Enter the **show memory** command:

```
pixfirewall# show memory
Free memory:        49046552 bytes
Used memory:        18062312 bytes
-------------     ----------------
Total memory:       67108864 bytes
```

Step 9 Enter the **show version** command:

```
pixfirewall# show version
Cisco PIX Firewall Version 6.3(1)
Compiled on Wed 19-Mar-03 11:49 by morlee
pixfirewall up 17 hours 59 mins

Hardware:   PIX-515, 64 MB RAM, CPU Pentium 200 MHz
Flash i28F640J5 @ 0x300, 16MB
BIOS Flash AT29C257 @ 0xfffd8000, 32KB

0: ethernet0: address is 0050.54ff.653a, irq 10
1: ethernet1: address is 0050.54ff.653b, irq 7
2: ethernet2: address is 00e0.b602.22c3, irq 11
3: ethernet3: address is 00e0.b602.22c2, irq 11
4: ethernet4: address is 00e0.b602.22c1, irq 11
5: ethernet5: address is 00e0.b602.22c0, irq 11
Licensed Features:
Failover:           Enabled
VPN-DES:            Enabled
VPN-3DES-AES:       Disabled
Maximum Interfaces: 6
Cut-through Proxy:  Enabled
Guards:             Enabled
URL-filtering:      Enabled
Inside Hosts:       Unlimited
Throughput:         Unlimited
IKE peers:          Unlimited
```

```
This PIX has an Unrestricted (UR) license.

Serial Number: 480350144 (0x1ca18fc0)
Running Activation Key: 0x51312668 0xb54b73ee 0x2e88e731 0xdf5e101a
Configuration last modified by enable_15 at 05:31:41.411 UTC Tue May
20 2003
```

Step 10 Enter the **show history** command:

```
pixfirewall# show history
enable
write terminal
sh memory
show version
show history
```

Step 11 Enter the configuration mode and change the hostname to pixP using the **hostname** command:

```
pixfirewall# configure terminal
pixfirewall(config)#
pixfirewall(config)# hostname pixP
```

(where P = pod number)

Step 12 Enable the use of names rather than IP addresses:

```
pixP(config)# names
```

Step 13 Assign the name bastionhost to the server on your DMZ:

```
pixP(config)# name 172.16.P.2 bastionhost
```

(where P = pod number)

Step 14 Assign the name insidehost to your student PC:

```
pixP(config)# name 10.0.P.11 insidehost
```

(where P = pod number)

Step 15 Save your configuration to Flash memory:

```
pixP(config)# write memory
Building configuration.
Cryptochecksum: e901c202 27a9db19 7e3c2878 0fc0966b
[OK]
```

Task 2—Configure PIX Firewall Interfaces

Complete the following steps to configure PIX Firewall Ethernet interfaces:

Step 1 Assign the PIX Firewall DMZ interface a name (dmz) and security level (50):

```
pixP(config)# nameif e2 dmz security50
pixP(config)# show nameif
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
nameif ethernet3 intf3 security6
nameif ethernet4 intf4 security8
nameif ethernet5 intf5 security10
```

Step 2 Enable the Ethernet 0, Ethernet 1, and Ethernet 2 interfaces for 100 Mbps Ethernet full duplex communication:

Note: By default the interfaces are disabled. You must enable all interfaces you intend to use.

```
pixP(config)# interface e0 100full
pixP(config)# interface e1 100full
pixP(config)# interface e2 100full
pixP(config)# show interface
interface ethernet0 "outside" is up, line protocol is up
  Hardware is i82558 ethernet, address is 0090.2724.fd0f
  MTU 1500 bytes, BW 100000 Kbit full duplex
        6 packets input, 360 bytes, 0 no buffer
        Received 6 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        0 packets output, 0 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks):hardware (128/128)software (0/0)
        output queue (curr/max blocks): hardware (0/0) software (0/0)
interface ethernet1 "inside" is up, line protocol is up
  Hardware is i82558 ethernet, address is 0090.2716.43dd
  MTU 1500 bytes, BW 100000 Kbit full duplex
        22811 packets input, 3365905 bytes, 0 no buffer
        Received 22811 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        0 packets output, 0 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks):hardware (128/128)software (0/1)
        output queue (curr/max blocks): hardware (0/0) software (0/0)
interface ethernet2 "dmz" is up, line protocol is up
  Hardware is i82558 ethernet, address is 0090.2725.060d
  MTU 1500 bytes, BW 100000 Kbit full duplex
        20283 packets input, 3034748 bytes, 0 no buffer
        Received 20283 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        0 packets output, 0 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks):hardware (128/128)software (0/1)
        output queue (curr/max blocks): hardware (0/0) software (0/0)
interface ethernet3 "intf3" is administratively down, line protocol is down
  Hardware is i82558 ethernet, address is 0090.2716.43dc
  MTU 1500 bytes, BW 100000 Kbit full duplex
        184 packets input, 15043 bytes, 0 no buffer
        Received 179 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        0 packets output, 0 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks):hardware (128/128)software (0/1)
        output queue (curr/max blocks): hardware (0/0) software (0/0)
interface ethernet4 "intf4" is administratively down, line protocol is down
  Hardware is i82558 ethernet, address is 0090.2716.43db
  MTU 1500 bytes, BW 100000 Kbit full duplex
        184 packets input, 15043 bytes, 0 no buffer
        Received 179 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        0 packets output, 0 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks):hardware (128/128)software (0/1)
        output queue (curr/max blocks): hardware (0/0) software (0/0)
interface ethernet5 "intf5" is administratively down, line protocol is down
  Hardware is i82558 ethernet, address is 0090.2716.43da
  MTU 1500 bytes, BW 100000 Kbit full duplex
        184 packets input, 15043 bytes, 0 no buffer
        Received 179 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        0 packets output, 0 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks):hardware (128/128)software (0/1)
        output queue (curr/max blocks): hardware (0/0) software (0/0)
```

Step 3 Assign IP addresses to the inside and DMZ network interface cards:

```
pixP(config)# ip address inside 10.0.P.1 255.255.255.0
pixP(config)# ip address dmz 172.16.P.1 255.255.255.0
```

(where P = pod number)

Step 4 Configure the PIX Firewall to retrieve its outside IP address from a DHCP server. Wait until you see the address appear before moving on to the next step:

```
pixP (config)# ip address outside dhcp
```

Step 5 Ensure that the IP addresses are correctly configured and are associated with the proper network interface:

```
pixP(config)# show ip address
System IP Addresses:
              ip address outside 192.168.P.2 255.255.255.0
              ip address inside 10.0.P.1 255.255.255.0
              ip address dmz 172.16.P.1 255.255.255.0
              no ip address intf3
              no ip address intf4
              no ip address intf5
Current IP Addresses:
              ip address outside 192.168.P.2 255.255.255.0
              ip address inside 10.0.P.1 255.255.255.0
              ip address dmz 172.16.P.1 255.255.255.0
              no ip address intf3
              no ip address intf4
              no ip address intf5
```

(where P = pod number)

Step 6 Write the configuration to the Flash memory:

```
pixP(config)# write memory
```

Task 3—Configure Global Addresses, NAT, and Routing for Inside and Outside Interfaces

Complete the following steps to configure a global address pool, NAT, and routing:

Step 1 Assign one pool of NIC-registered IP addresses for use by outbound connections:

```
pixP(config)# global (outside) 1 192.168.P.20-192.168.P.254 netmask
255.255.255.0
pixP(config)# show global
global (outside) 1 192.168.P.20-192.168.P.254 netmask 255.255.255.0
```

(where P = pod number)

Step 2 Configure the PIX Firewall to allow inside hosts to use NAT for outbound access:

```
pixP(config)# nat (inside) 1 10.0.P.0 255.255.255.0
```

(where P = pod number)

Step 3 Display the currently configured NAT:

```
pixP(config)# show nat
nat (inside) 1 10.0.P.0 255.255.255.0 0 0
```

(where P = pod number)

Step 4 Create a static route to the 10.1.P.0 network:

```
pixP(config)# route inside 10.1.P.0 255.255.255.0 10.0.P.102
```

(where P = pod number)

Step 5 Assign a default route:

```
pixP(config)# route outside 0 0 192.168.P.1
```

(where P = pod number)

Step 6 Display the currently configured routes:

```
pixP(config)# show route
outside 0.0.0.0 0.0.0.0 192.168.P.1 1 OTHER static
inside 10.0.P.0 255.255.255.0 10.0.P.1 1 CONNECT static
inside 10.1.P.0 255.255.255.0 10.0.P.102 1 OTHER static
dmz 172.16.P.0 255.255.255.0 172.16.P.1 1 CONNECT static
outside 192.168.P.0 255.255.255.0 192.168.P.2 1 CONNECT static
```

(where P = pod number)

Step 7 Write the current configuration to Flash memory:

```
pixP(config)# write memory
```

Step 8 Display a list of the most recently entered commands:

Your history inputs should be similar to the following:

```
pixP(config)#  show history
interface e0 100full
interface e1 100full
interface e2 100full
show interface
ip address inside 10.0.P.1 255.255.255.0
ip address dmz 172.16.P.1 255.255.255.0
ip address outside dhcp
show ip address
write memory
global (outside) 1 192.168.P.20-192.168.P.254 netmask 255.255.255.0
show global
nat (inside) 1 10.0.P.0 255.255.255.0
show nat
route inside 10.1.P.0 255.255.255.0 10.0.P.102
route outside 0 0 192.168.P.1
show route
write memory
show history
```

(where P = pod number)

Note: You can use the up and down cursor keys on your keyboard to recall commands.

Step 9 Write the current configuration to the terminal and verify that you have entered the previous commands correctly:

```
pixP(config)# write terminal

PIX Version 6.3(1)
interface ethernet0 100full
interface ethernet1 100full
interface ethernet2 100full
interface ethernet3 auto shutdown
interface ethernet4 auto shutdown
interface ethernet5 auto shutdown
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
nameif ethernet3 intf3 security15
nameif ethernet4 intf4 security20
nameif ethernet5 intf5 security25
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname pix6
fixup protocol ftp 21
fixup protocol h323 h225 1720
fixup protocol h323 ras 1718-1719
fixup protocol http 80
fixup protocol ils 389
fixup protocol rsh 514
fixup protocol rtsp 554
fixup protocol sip 5060
fixup protocol sip udp 5060
fixup protocol skinny 2000
fixup protocol smtp 25
fixup protocol sqlnet 1521
names
name 172.16.P.2 bastionhost
name 10.0.P.11 insidehost
logging on
mtu outside 1500
mtu inside 1500
mtu dmz 1500
mtu intf3 1500
mtu intf4 1500
mtu intf5 1500
ip address outside 192.168.P.2 255.255.255.0
ip address inside 10.0.P.1 255.255.255.0
ip address dmz 172.16.P.1 255.255.255.0
no ip address intf3
no ip address intf4
no ip address intf5
ip audit info action alarm
ip audit attack action alarm
no failover
failover timeout 0:00:00
failover poll 15
no failover ip address outside
no failover ip address inside
no failover ip address dmz
no failover ip address intf3
no failover ip address intf4
no failover ip address intf5
pdm history enable
arp timeout 14400
global (outside) 1 192.168.P.20-192.168.P.254 netmask 255.255.255.0
nat (inside) 1 10.0.P.0 255.255.255.0 0 0
route outside 0.0.0.0 0.0.0.0 192.168.P.1 1
route inside 10.1.P.0 255.255.255.0 10.0.P.102 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:0
timeout h323 0:05:00 mgcp 0:05:00 sip 0:30:00 sip 0:30:00 sip_media
0:02:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server LOCAL protocol local
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
floodguard enable
telnet timeout 5
ssh timeout 5
console timeout 0
terminal width 80
Cryptochecksum:00000000000000000000000000000000
: end
[OK]
```

(where P = pod number)

Step 10 Test the operation of the globals and NAT statements you configured by originating connections through the PIX Firewall by completing the following sub-steps:

1. Open a web browser on the Windows NT server.
2. Use the web browser to access the super server at IP address 172.26.26.50 by entering **http://172.26.26.50**.

Step 11 Observe the translation table:

```
pixP(config)# show xlate
```

Your display should appear similar to the following:

```
1 in use, 1 most used
Global 192.168.P.20 Local insidehost
```

(where P = pod number)

A global address chosen from the low end of the global range has been mapped to your NT laptop.

Task 4—Test the Inside, Outside, and DMZ Interface Connectivity

Complete the following steps to test and troubleshoot interface connectivity using the PIX Firewall **ping** command:

Step 1 Ping the inside interface:

```
pixP(config)# ping 10.0.P.1
10.0.P.1 response received –– 10ms
10.0.P.1 response received –– 10ms
10.0.P.1 response received –– 10ms
```

(where P = pod number)

Step 2 Ping your inside host:

```
pixP(config)# ping insidehost
insidehost response received –– 10ms
insidehost response received –– 10ms
insidehost response received –– 10ms
```

Step 3 Ping the outside interface:

```
pixP(config)# ping 192.168.P.2
192.168.P.2 response received –– 10ms
192.168.P.2 response received –– 10ms
192.168.P.2 response received –– 10ms
```

(where P = pod number)

Step 4 Ping the backbone router:

```
pixP(config)# ping 192.168.P.1
192.168.P.1 response received –– 10ms
192.168.P.1 response received –– 10ms
192.168.P.1 response received –– 10ms
```

(where P = pod number)

Step 5 Ping the DMZ interface:

```
pixP(config)# ping 172.16.P.1
172.16.P.1 response received –– 10ms
172.16.P.1 response received –– 10ms
172.16.P.1 response received –– 10ms
```

(where P = pod number)

Step 6 Ping bastionhost:

```
pixP(config)# ping bastionhost
bastionhost response received —— 10ms
bastionhost response received —— 10ms
bastionhost response received —— 10ms
```

Task 5—Configure Syslog Output

Complete the following steps and enter the commands as directed to configure Syslog output:

Step 1 Enable Syslog logging:

```
pixP(config)# logging on
```

Step 2 Begin storing messages to the PIX Firewall message buffer and set the logging level to debugging:

```
pixP(config)# logging buffered debugging
```

Step 3 From your Windows command line, telnet to the backbone router:

```
C:\> telnet 192.168.P.1
```

(where P = pod number)

Step 4 View the Syslog messages with the **show logging** command. New messages appear at the end of the display:

```
pixP(config)# show logging
Syslog logging: enabled
    Facility: 20
    Timestamp logging: disabled
    Standby logging: disabled
    Console logging: disabled
    Monitor logging: disabled
    Buffer logging: level debugging, 2 messages logged
    Trap logging: disabled
    History logging: disabled
111008: User 'enable_15' executed cmd : logging buffered debugging
302013: Built outbound TCP connection 2 for outside:192.168.P.1/23
(192.168.P.1/23) to inside:10.0.P.11/1036 (192.168.P.20/1036)
```

Step 5 Clear messages in the buffer and verify they are cleared:

```
pixP(config)# clear logging
pixP(config)# show logging
```

Step 6 Set the **logging buffered** command back to a minimal level:

```
pixP(config)# logging buffered alerts
```

Task 6—Configure Syslog Output to a Syslog Server

The instructor will provide you with the procedures for access to a Syslog server or host. This varies according to the type of Syslog server used for your classroom environment.

Note: Verify that the Syslog server or host is turned on and that the Syslog service is installed and started.

Step 1 Designate a host to receive the messages with the **logging host** command. For normal Syslog operations to any Syslog server, use the default message protocol:

```
pixP(config)# logging host inside insidehost
```

Step 2 Set the logging level to the Syslog server or host with **the logging trap** command. This command is used to start sending messages to the Syslog server or host. For the level, refer to the command reference table at the beginning of this exercise:

```
pixP(config)# logging trap debugging
```

Step 3 Open the Kiwi Syslog Daemon on your desktop.

Step 4 From your Windows command line, telnet the backbone router:

```
C:\> telnet 192.168.P.1
```

(where P = pod number)

Step 5 Observe the messages logged to the Kiwi Syslog Daemon display screen:

```
%PIX-6-302013:  Built outbound TCP connection 3 for
outside:192.168.P.1/23[192.168.P.1/23] to
inside:10.0.P.11/1037[192.168.P.20/1037]
```

(where P = pod number)

6

Translations and Connections

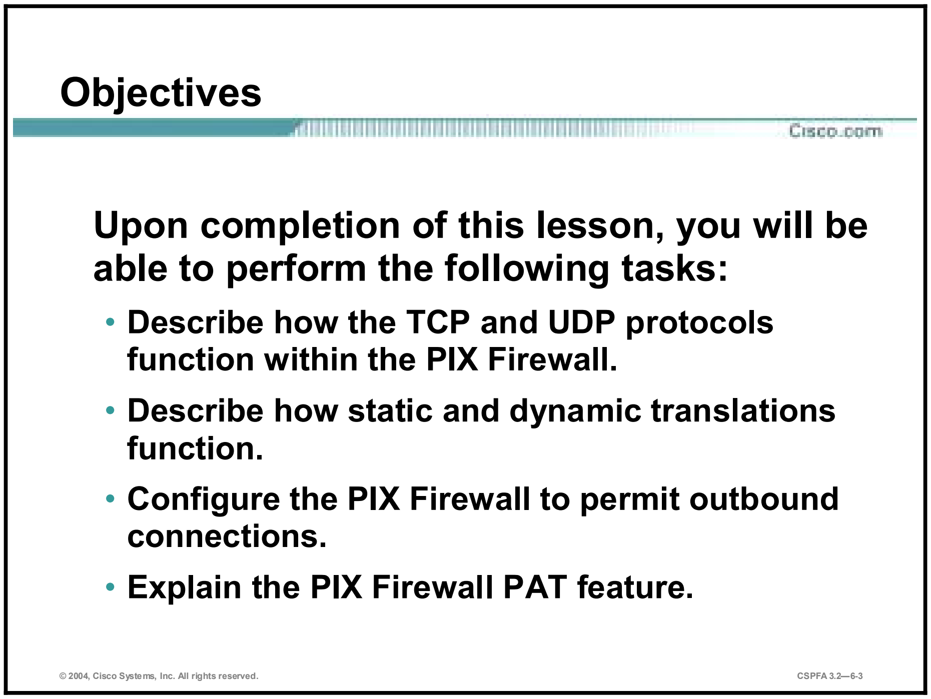
Overview

This lesson includes the following topics:

- Objectives
- Transport protocols
- Network Address Translation
- Port Address Translation
- **static** command
- Identity NAT command
- Connections and translations
- Configuring multiple interfaces
- Summary
- Lab exercise

Objectives

This topic lists the lesson’s objectives.



Objectives

Cisco.com

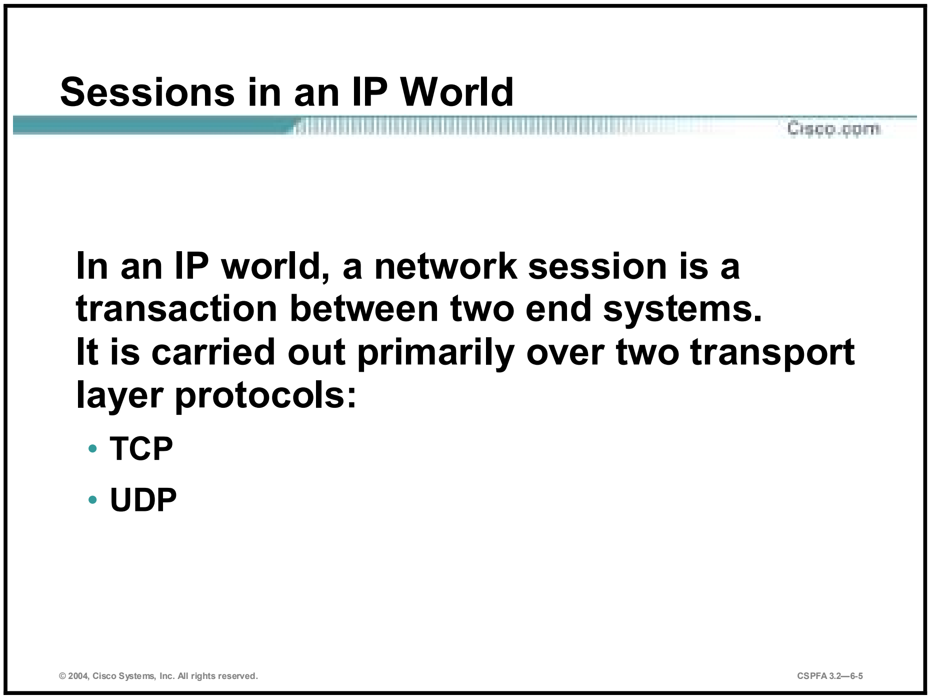
Upon completion of this lesson, you will be able to perform the following tasks:

- **Describe how the TCP and UDP protocols function within the PIX Firewall.**
- **Describe how static and dynamic translations function.**
- **Configure the PIX Firewall to permit outbound connections.**
- **Explain the PIX Firewall PAT feature.**

CSPFA 3.2—6-3

Transport Protocols

To gain a deeper understanding of how the Cisco PIX Firewall processes inbound and outbound transmissions, a brief review of the two primary transport protocols is warranted.



Sessions in an IP World

Cisco.com

In an IP world, a network session is a transaction between two end systems.
It is carried out primarily over two transport layer protocols:

- **TCP**
- **UDP**

© 2004, Cisco Systems, Inc. All rights reserved.

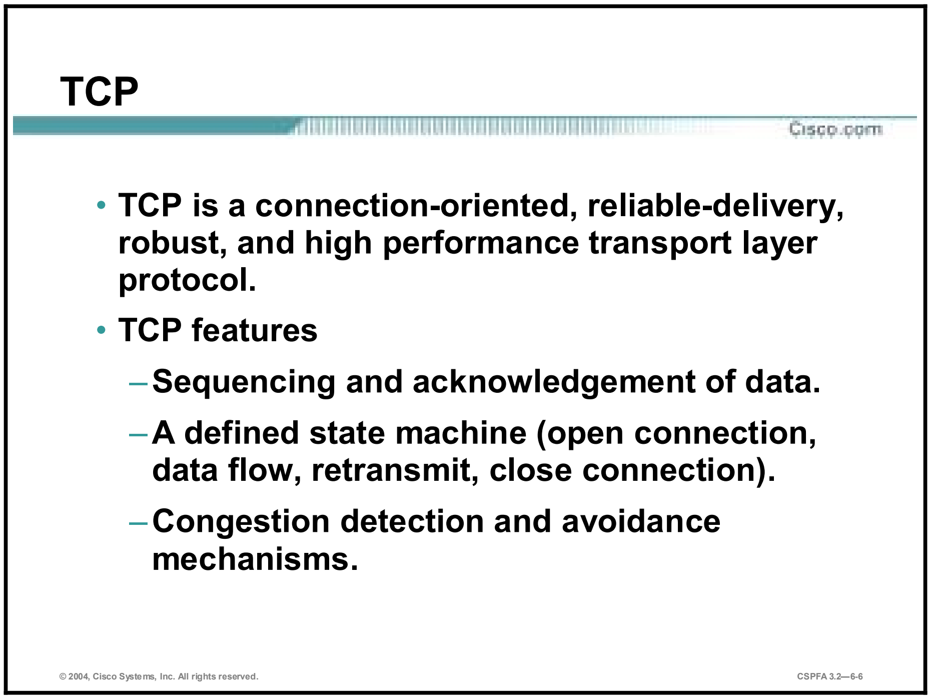
CSPFA 3.2—6-5

It is important to understand the transport protocols Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) to understand how the PIX Firewall operates. This topic aids in understanding the TCP and UDP protocols.

A network session is carried out over two transport layer protocols:

- TCP, which is easy to inspect
- UDP, which is difficult to inspect properly

Note: In the context of this training, the term *outbound* means connections from a more trusted side of the PIX Firewall to a less trusted side of the PIX Firewall. The term *inbound* means connections from a less trusted side of the PIX Firewall to a more trusted side of the PIX Firewall.



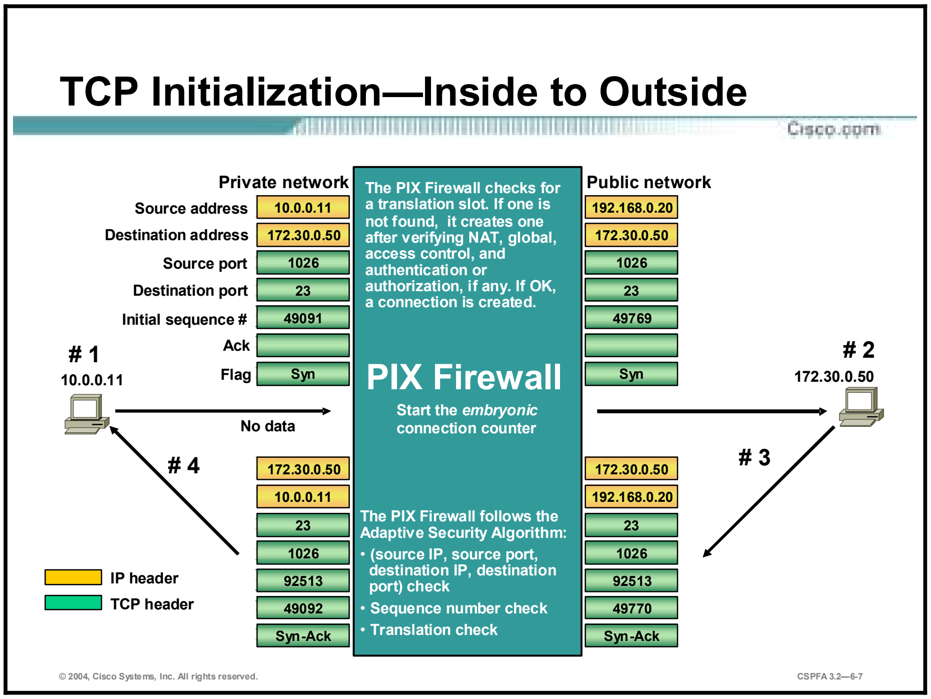
TCP

Cisco.com

- TCP is a connection-oriented, reliable-delivery, robust, and high performance transport layer protocol.
- TCP features
  - Sequencing and acknowledgement of data.
  - A defined state machine (open connection, data flow, retransmit, close connection).
  - Congestion detection and avoidance mechanisms.

CSPFA 3.2—6-6

TCP is a connection-oriented protocol. When a session from a more secure host inside the PIX Firewall is started, the PIX Firewall creates an entry in the session state filter. The PIX Firewall is able to extract network sessions from the network flow and actively verify their validity in real time. This stateful filter maintains the state of each network connection and checks subsequent protocol units against its expectations. When a TCP session is initiated through a PIX Firewall, the PIX Firewall records the network flow and looks for an acknowledgement from the device with which the host is trying to initiate communications. The PIX Firewall then allows traffic to flow between the hosts involved in the connection based on the three-way handshake.



TCP Initialization—Inside to Outside

Cisco.com

	Private network	Public network
Source address	10.0.0.11	192.168.0.20
Destination address	172.30.0.50	172.30.0.50
Source port	1026	1026
Destination port	23	23
Initial sequence #	49091	49769
Ack		
Flag	Syn	Syn

The PIX Firewall checks for a translation slot. If one is not found, it creates one after verifying NAT, global, access control, and authentication or authorization, if any. If OK, a connection is created.

PIX Firewall

Start the *embryonic* connection counter

1
10.0.0.11

No data

2
172.30.0.50

3

4

172.30.0.50	172.30.0.50
10.0.0.11	192.168.0.20
23	23
1026	1026
92513	92513
49092	49770
Syn-Ack	Syn-Ack

The PIX Firewall follows the Adaptive Security Algorithm:

- (source IP, source port, destination IP, destination port) check
- Sequence number check
- Translation check

🟨 IP header

🟩 TCP header

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—6-7

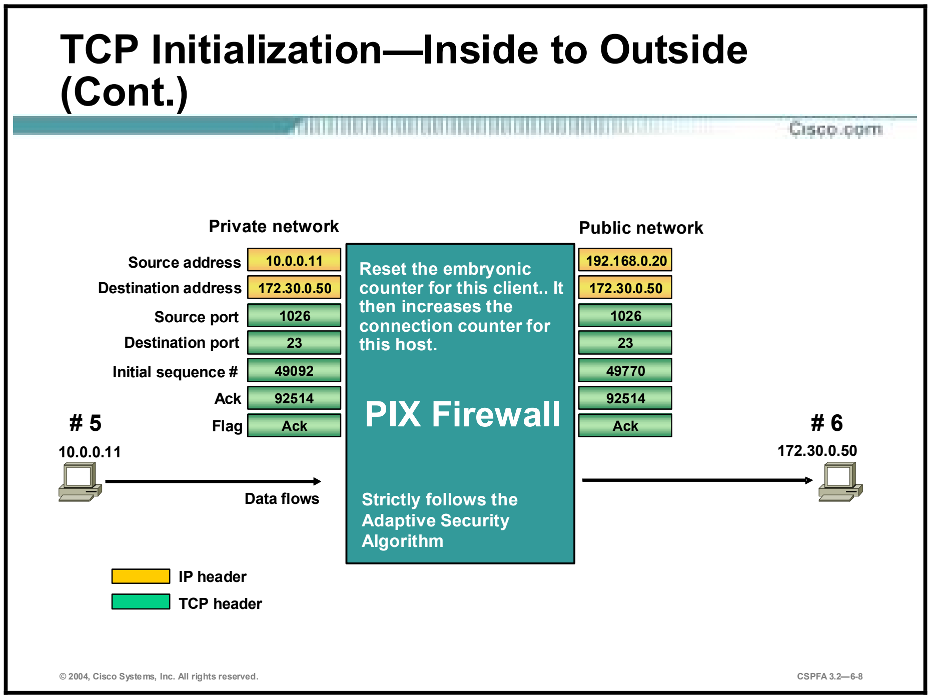
When a TCP session is established over the PIX Firewall, the following happens:

Step 1 The first Internet Protocol (IP) packet from an inside host causes the generation of a translation slot. The embedded TCP information is then used to create a connection slot in the PIX Firewall.

Step 2 The connection slot is marked as *embryonic* (not established yet).

Step 3 The PIX Firewall randomizes the initial sequence number of the connection, stores the delta value, and forwards the packet onto the outgoing interface.

The PIX Firewall now expects a synchronize/acknowledge (SYN/ACK) packet from the destination host. Then the PIX Firewall matches the received packet against the connection slot, computes the sequencing information, and forwards the return packet to the inside host.



TCP Initialization—Inside to Outside (Cont.)

Cisco.com

	Private network	Public network
Source address	10.0.0.11	192.168.0.20
Destination address	172.30.0.50	172.30.0.50
Source port	1026	1026
Destination port	23	23
Initial sequence #	49092	49770
Ack	92514	92514
Flag	Ack	Ack

Reset the embryonic counter for this client.. It then increases the connection counter for this host.

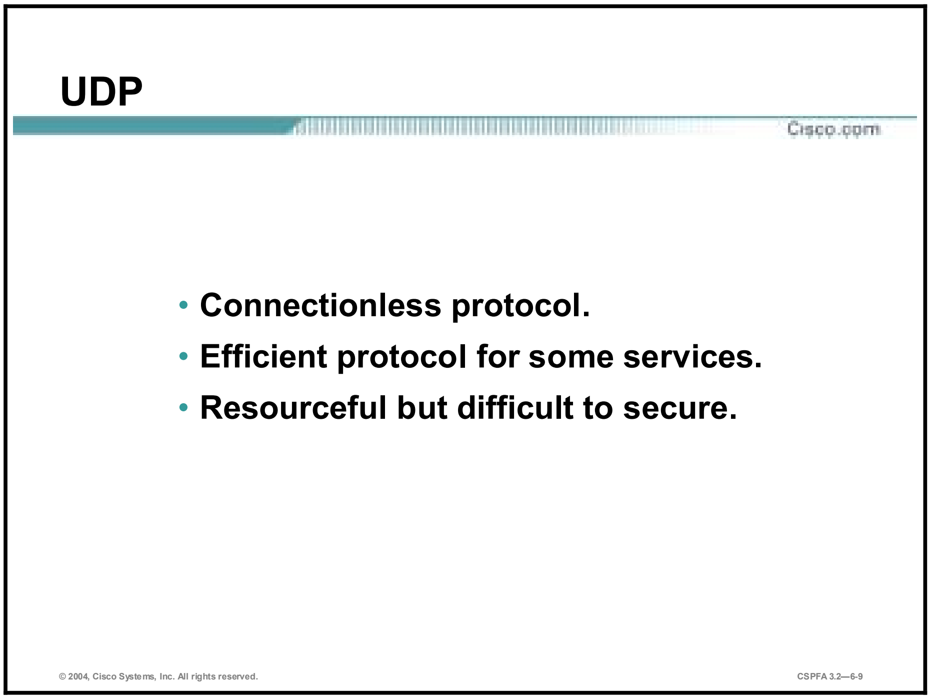
PIX Firewall

Strictly follows the Adaptive Security Algorithm

CSPFA 3.2—6-8

Step 4 The inside host completes the connection setup, the three-way handshake, with an ACK.

Step 5 The connection slot on the PIX Firewall is marked as connected, or active-established, and data is transmitted. The embryonic counter is then reset for this connection.



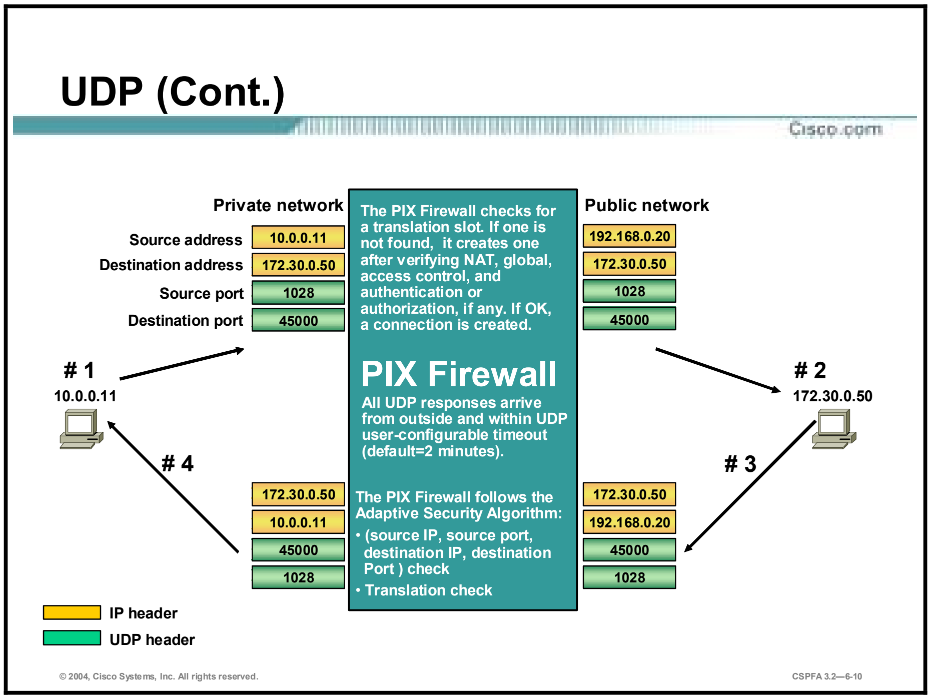
UDP

Cisco.com

- Connectionless protocol.
- Efficient protocol for some services.
- Resourceful but difficult to secure.

CSPFA 3.2—6-9

UDP is connectionless. The PIX Firewall must take other measures to ensure its security. Applications using UDP are difficult to secure properly because there is no handshaking or sequencing. It is difficult to determine the current state of a UDP transaction. It is also difficult to maintain the state of a session, as it has no clear beginning, flow state, or end. However, the PIX Firewall creates a UDP connection slot when a UDP packet is sent from a more secure to a less secure interface. All subsequent returned UDP packets matching the connection slot are forwarded to the inside network.



UDP (Cont.)

Cisco.com

Private network

Source address 10.0.0.11

Destination address 172.30.0.50

Source port 1028

Destination port 45000

The PIX Firewall checks for a translation slot. If one is not found, it creates one after verifying NAT, global, access control, and authentication or authorization, if any. If OK, a connection is created.

Public network

192.168.0.20

172.30.0.50

1028

45000

1

10.0.0.11

2

172.30.0.50

PIX Firewall

All UDP responses arrive from outside and within UDP user-configurable timeout (default=2 minutes).

4

3

172.30.0.50

10.0.0.11

45000

1028

The PIX Firewall follows the Adaptive Security Algorithm:

- (source IP, source port, destination IP, destination Port) check
- Translation check

172.30.0.50

192.168.0.20

45000

1028

🟨 IP header

🟩 UDP header

CSPFA 3.2—6-10

When the UDP connection slot is idle for more than the configured idle time, it is deleted from the connection table. The following are some UDP characteristics:

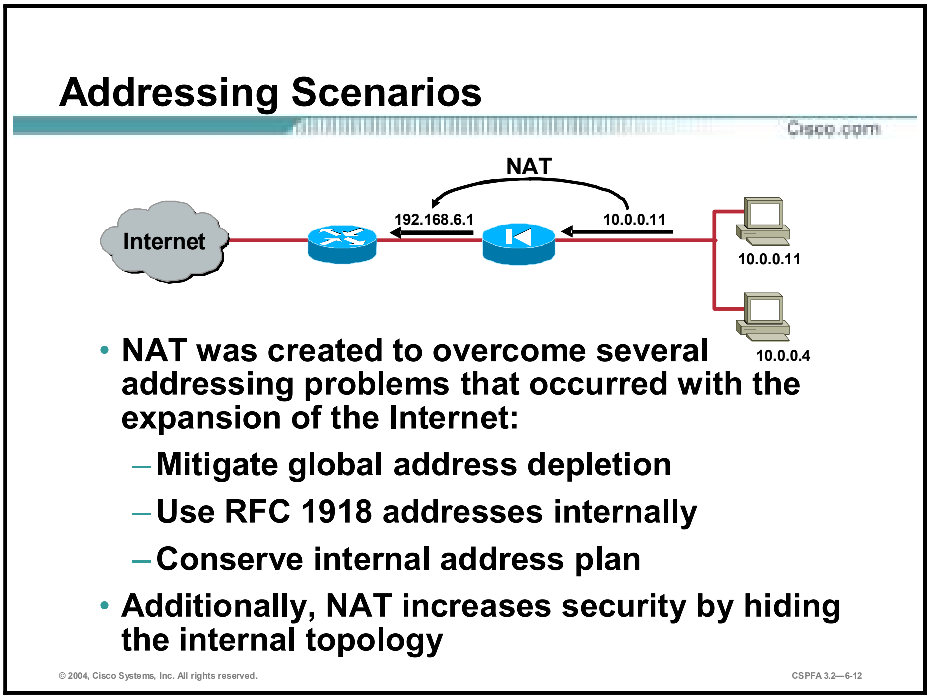
- UDP is an unreliable but efficient transport protocol.
- Spoofing UDP packets is very easy because there is no handshaking or sequencing. As there is no state machine, the initiator of the transaction or the current state usually cannot be determined.
- UDP has no delivery guarantees.
- There is no connection setup and termination.
- UDP has no congestion management or avoidance.

Services that use UDP can be generally divided into two categories:

- Request-reply, or ping-pong services, such as domain name system (DNS).
- Flow services, such as video, voice over IP (VoIP), and Network File System (NFS).

Network Address Translation

This topic describes the translation process in the PIX Firewall. There are two types of inside translations: dynamic and static.



Addressing Scenarios

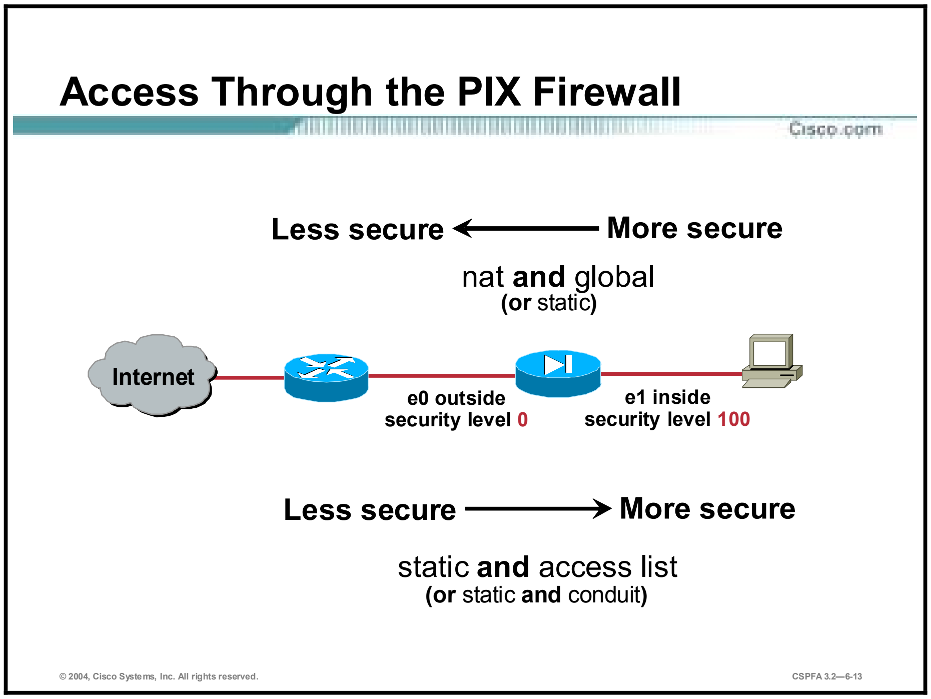
- NAT was created to overcome several addressing problems that occurred with the expansion of the Internet:
  - Mitigate global address depletion
  - Use RFC 1918 addresses internally
  - Conserve internal address plan
- Additionally, NAT increases security by hiding the internal topology

CSPFA 3.2—6-12

Invented in May 1994 by Paul Francis and Kjeld Borch Egevang, Network Address Translation (NAT) became a popular technique to save official network addresses and to hide network topology from the Internet. Paul Francis and Kjeld Borch Egevang have written several RFCs about NAT, most importantly RFC 1631: "The IP Network Address Translator (NAT)".

In the modern world, NAT is critical to mitigate global Internet address depletion. Very often, private networks are assigned numbers from network blocks defined in RFC 1918. Because these addresses are intended for local use only, NAT is required to connect to the Internet. NAT is sometimes used to preserve the inside addresses of an enterprise, for example, when changing the Internet service provider (ISP).

In the figure, the private network is using private IP addressing, 10.0.0.0/24. Before a packet can be sent to the Internet, it must be translated into a public, routable address. In this example, the PIX Firewall translates IP address 10.0.0.11 into routable IP address 192.168.6.1.



Access Through the PIX Firewall

Cisco.com

Less secure ← More secure

nat **and** global
(or static**)**

Less secure → More secure

static **and** access list
(or static **and** conduit**)**

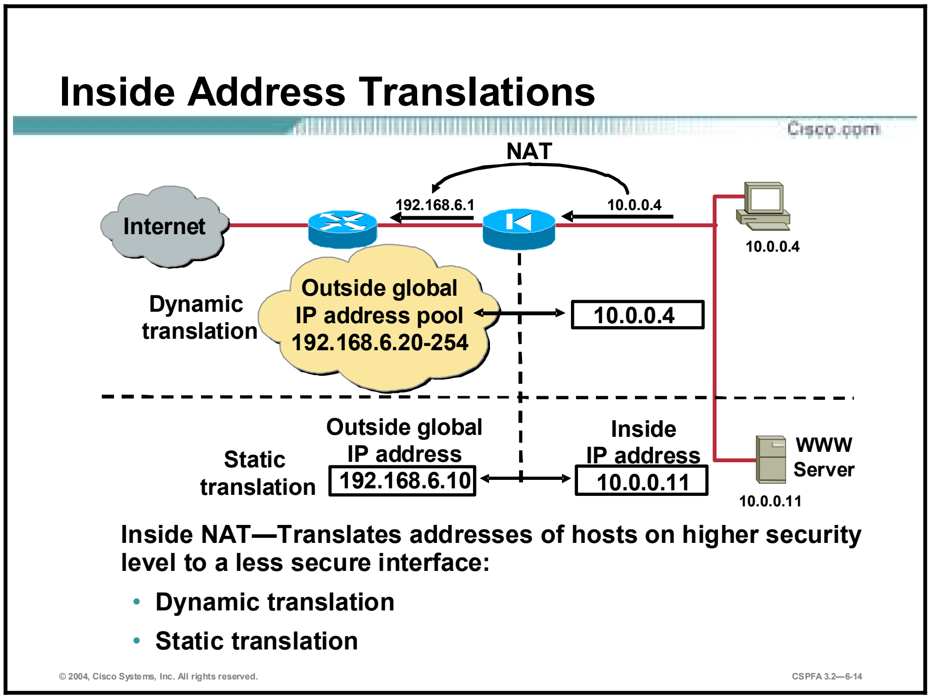
© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—6-13

When you are configuring multiple interfaces, remember that the security level designates whether an interface is inside (trusted) or outside (untrusted) relative to another interface. An interface is considered inside in relation to another interface if its security level is higher than the security level of the other interface, and is considered outside in relation to another interface if its security level is lower than the security level of the other interface.

The primary rule for security levels is that an interface with a higher security level can access an interface with a lower security level. The **nat** and **global** commands work together to enable your network to use any IP addressing scheme and to remain hidden from the external network.

An interface with a lower security level cannot access an interface with a higher security level unless you specifically allow it by implementing **static** and **access list** command pairs, or **static** and **conduit**. Outside **static** commands and access lists are covered later in the course.



Inside Address Translations

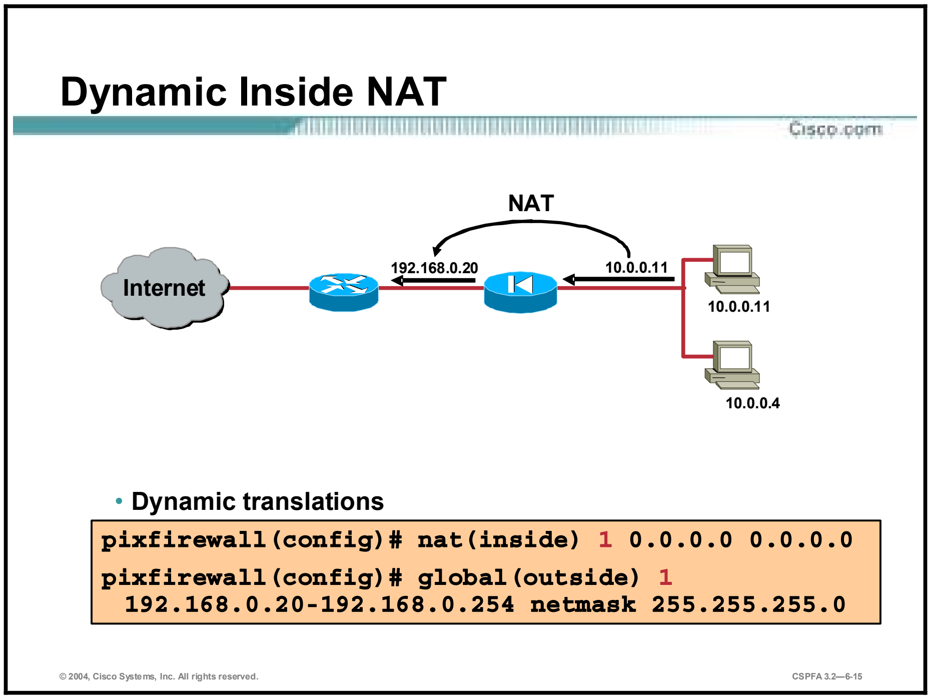
Inside NAT—Translates addresses of hosts on higher security level to a less secure interface:

- **Dynamic translation**
- **Static translation**

CSPFA 3.2—6-14

The PIX Firewall supports the following two main types of address translations:

- Dynamic translation—Translates host addresses on more secure interfaces to a range or pool of IP addresses on a less secure interface. This allows internal users to share registered IP addresses and hides internal addresses from view on the public Internet.
- Static translation—Provides a permanent, one-to-one mapping between an IP address on a more secure interface and an IP address on a less secure interface. This allows an inside host to access a less secure host, a server on the Internet, for instance, without exposing the actual IP address. Examples of static translation are static NAT and identity NAT.



Dynamic Inside NAT

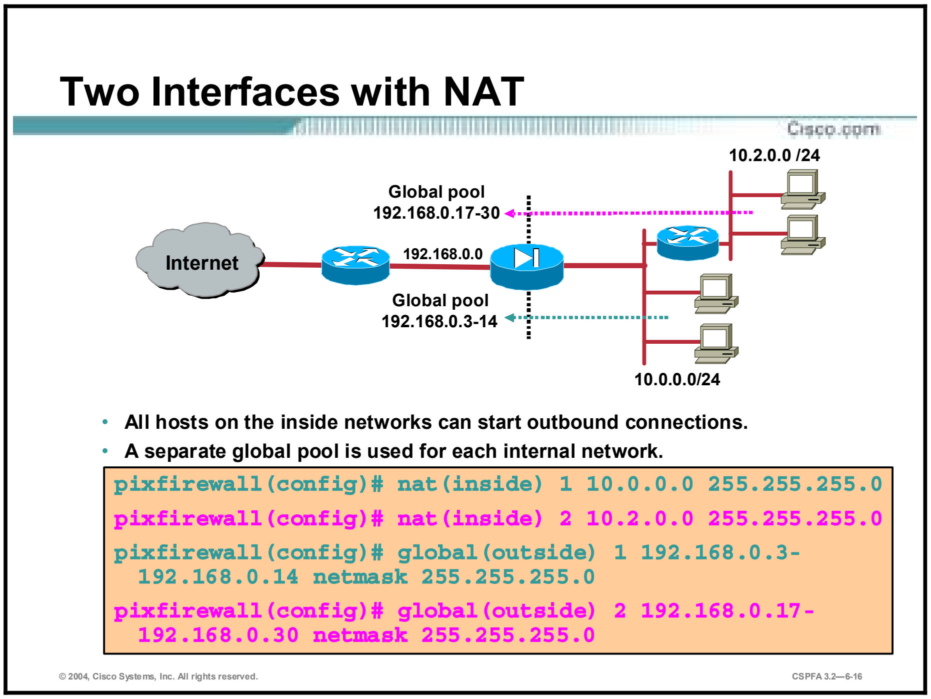
- Dynamic translations

```
pixfirewall(config)# nat(inside) 1 0.0.0.0 0.0.0.0
pixfirewall(config)# global(outside) 1
  192.168.0.20-192.168.0.254 netmask 255.255.255.0
```

CSPFA 3.2—6-15

Dynamic inside translations are used for local hosts and their outbound connections, and hide the host address from the Internet. With dynamic translations, you must first define which hosts are eligible for translation with the **nat** command, and then define the address pool with the **global** command. The pool for address allocation is chosen on the outgoing interface based on the nat_id selected with the **nat** command.

In the figure shown, all hosts on the inside network are eligible for translation. The global pool of addresses assigned by the **global** command is 192.168.0.20 through 192.168.0.254, enabling up to 235 individual IP addresses.



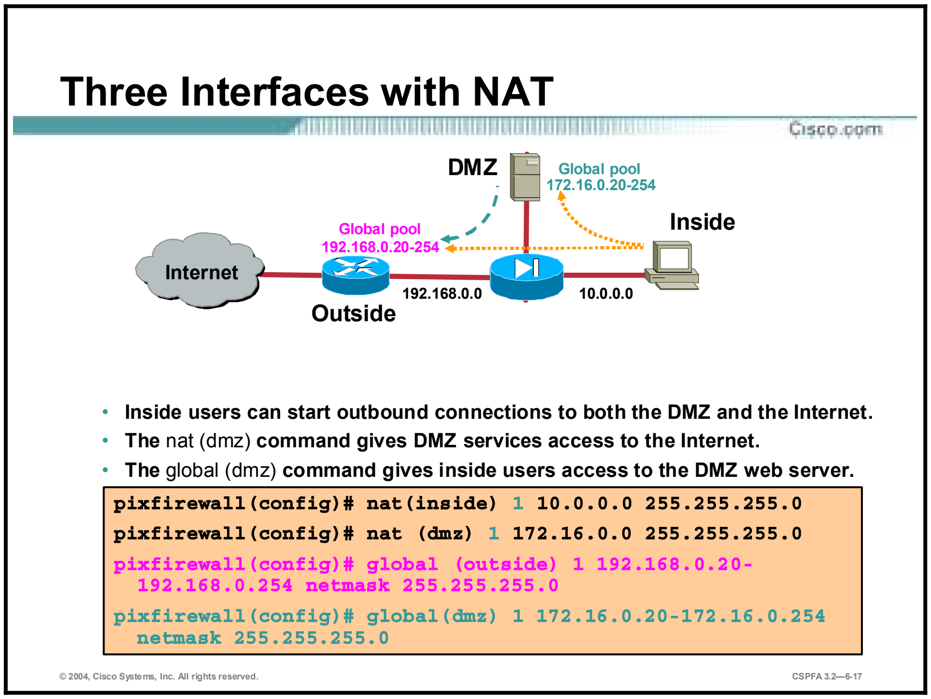
Two Interfaces with NAT

- All hosts on the inside networks can start outbound connections.
- A separate global pool is used for each internal network.

```
pixfirewall(config)# nat(inside) 1 10.0.0.0 255.255.255.0
pixfirewall(config)# nat(inside) 2 10.2.0.0 255.255.255.0
pixfirewall(config)# global(outside) 1 192.168.0.3-
  192.168.0.14 netmask 255.255.255.0
pixfirewall(config)# global(outside) 2 192.168.0.17-
  192.168.0.30 netmask 255.255.255.0
```

CSPFA 3.2—6-16

In the figure, the first **nat** command statement permits all hosts on the 10.0.0.0 network to start outbound connections using the IP addresses from a global pool. The second **nat** command statement permits all hosts on the 10.2.0.0 network to do the same. The nat_id in the first **nat** command statement tells the PIX Firewall to translate the 10.0.0.0 addresses to those in the global pool containing the same nat_id. Likewise, the nat_id in the second **nat** command statement tells the PIX Firewall to translate addresses for hosts on network 10.2.0.0 to the addresses in the global pool containing nat_id 2.



Three Interfaces with NAT

- **Inside users can start outbound connections to both the DMZ and the Internet.**
- **The** nat (dmz) **command gives DMZ services access to the Internet.**
- **The** global (dmz) **command gives inside users access to the DMZ web server.**

```
pixfirewall(config)# nat(inside) 1 10.0.0.0 255.255.255.0
pixfirewall(config)# nat (dmz) 1 172.16.0.0 255.255.255.0
pixfirewall(config)# global (outside) 1 192.168.0.20-
  192.168.0.254 netmask 255.255.255.0
pixfirewall(config)# global(dmz) 1 172.16.0.20-172.16.0.254
  netmask 255.255.255.0
```

CSPFA 3.2—6-17

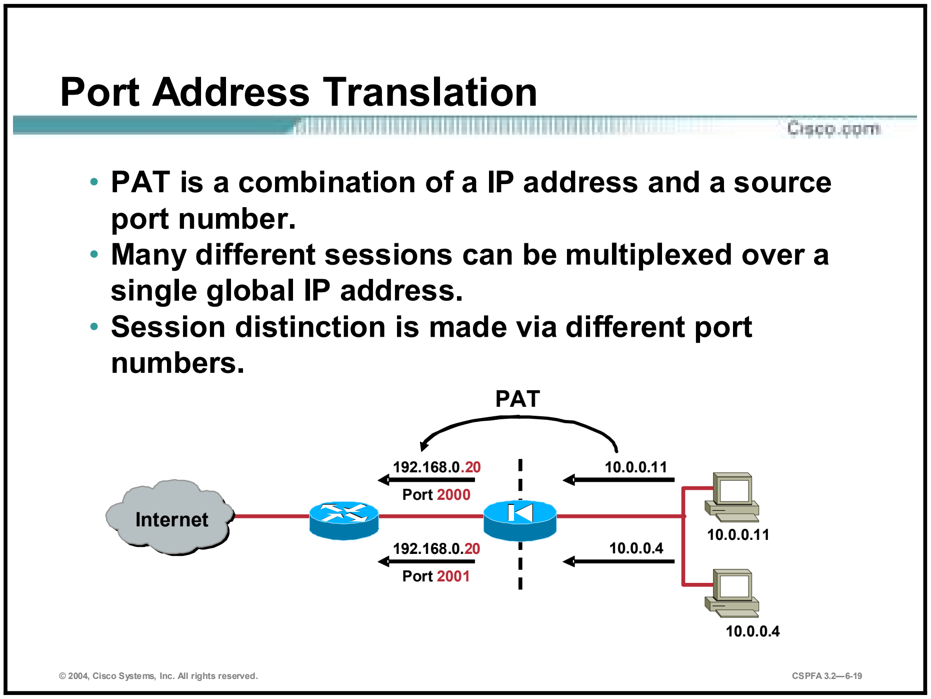
In the figure, the first **nat** command statement enables hosts on the inside interface, which has a security level of 100, to start connections to hosts on interfaces with lower security levels. In this case, that includes hosts on the outside interface and hosts on the demilitarized zone (DMZ). The second **nat** command statement enables hosts on the DMZ, which has a security level of 50, to start connections to hosts on interfaces with lower security levels. In this case, that includes only the outside interface.

Because both global pools and the **nat (inside)** command statement use a nat_id of 1, addresses for hosts on the 10.0.0.0 network can be translated to those in either global pool. Therefore, when users on the inside interface access hosts on the DMZ, their source addresses will be translated to addresses in the 172.16.0.20–172.16.0.254 range from the **global (dmz)** command statement. When they access hosts on the outside, their source addresses will be translated to addresses in the 192.168.0.20–192.168.0.254 range from the **global (outside)** command statement.

When users on the DMZ access hosts on the outside, their source addresses will always be translated to addresses in the 192.168.0.20–192.168.0.254 range from the **global (outside)** command statement.

Port Address Translation

This topic describes how to configure a PIX Firewall to take advantage of Port Address Translation (PAT).



Port Address Translation

Cisco.com

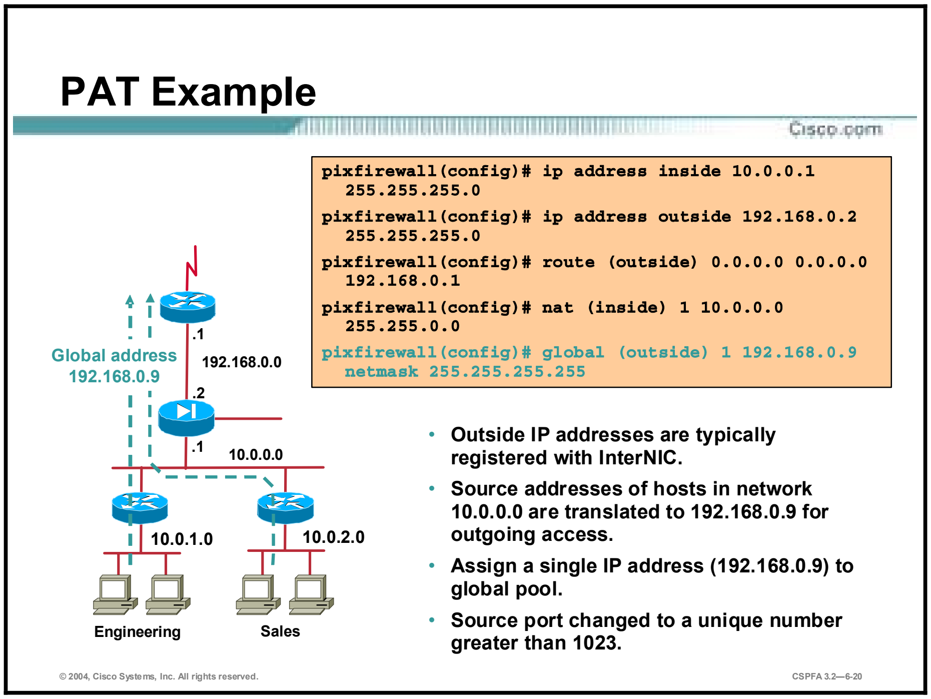
- PAT is a combination of a IP address and a source port number.
- Many different sessions can be multiplexed over a single global IP address.
- Session distinction is made via different port numbers.

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—6-19

Typically, an enterprise network receives only a small number of addresses from its ISP, while the number of hosts is much larger. To resolve this situation, configure PAT, which is an enhancement of NAT.

Using PAT, multiple connections originating from different hosts on the inside networks can be multiplexed by a single global IP address. The multiplexing identifier is the source port number. In the figure, the IP addresses of the two hosts on the inside network are translated to a PAT IP address of 192.168.0.20 source ports 2000 and 2001.



PAT Example

Cisco.com

```
pixfirewall(config)# ip address inside 10.0.0.1
  255.255.255.0
pixfirewall(config)# ip address outside 192.168.0.2
  255.255.255.0
pixfirewall(config)# route (outside) 0.0.0.0 0.0.0.0
  192.168.0.1
pixfirewall(config)# nat (inside) 1 10.0.0.0
  255.255.0.0
pixfirewall(config)# global (outside) 1 192.168.0.9
  netmask 255.255.255.255
```

- Outside IP addresses are typically registered with InterNIC.
- Source addresses of hosts in network 10.0.0.0 are translated to 192.168.0.9 for outgoing access.
- Assign a single IP address (192.168.0.9) to global pool.
- Source port changed to a unique number greater than 1023.

CSPFA 3.2—6-20

The PIX Firewall PAT feature expands a company's address pool:

- One outside IP address is used for approximately 4,000 inside hosts (practical limit, theoretical limit is greater than 64,000).
- PAT maps TCP and UDP port numbers to a single IP address.
- PAT provides security by hiding the inside source address by using a single IP address from the PIX.
- PAT can be used with NAT.
- A PAT address can be a virtual address, different from the outside address. Do not use PAT when running multimedia applications through the PIX Firewall. Multimedia applications need access to specific ports and can conflict with port mappings provided by PAT.

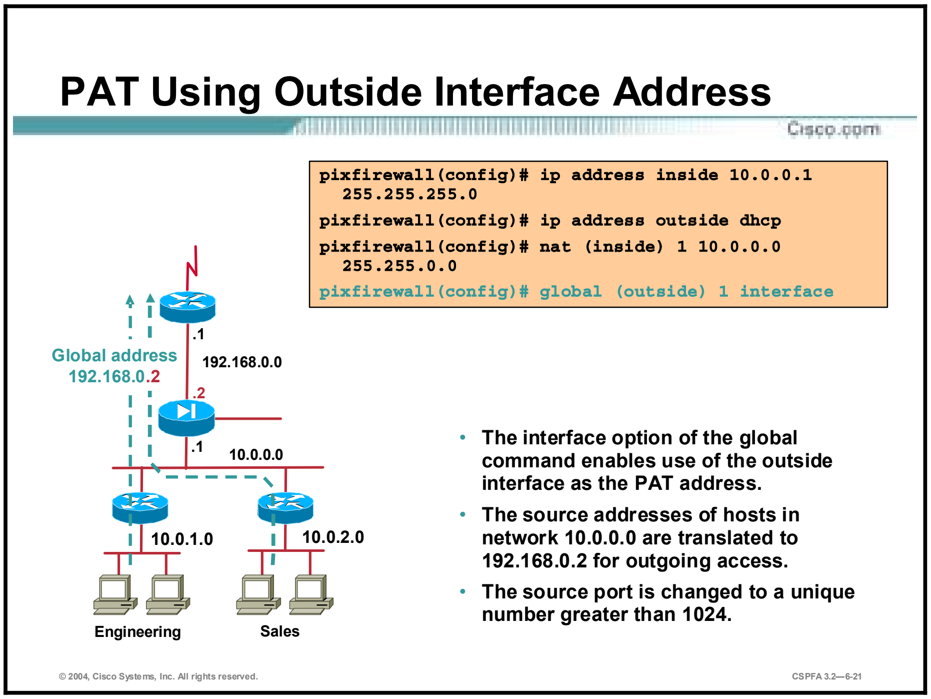
In this example of PAT, the XYZ Company has only three registered IP addresses. One address is taken by the perimeter router, one by the PIX Firewall, and one by the global address.

The example configuration is as follows:

```
ip address inside 10.0.0.1 255.255.255.0
ip address outside 192.168.0.2 255.255.255.0
route outside 0.0.0.0 0.0.0.0 192.168.0.1
```

IP addresses are assigned to the internal and external interfaces. A single registered IP address is put into the global pool and is shared by all outgoing access for network 10.0.0.0:

```
nat (inside) 1 10.0.0.0 255.255.0.0
global (outside) 1 192.168.0.9 netmask 255.255.255.255
```



PAT Using Outside Interface Address

Cisco.com

```
pixfirewall(config)# ip address inside 10.0.0.1
  255.255.255.0
pixfirewall(config)# ip address outside dhcp
pixfirewall(config)# nat (inside) 1 10.0.0.0
  255.255.0.0
pixfirewall(config)# global (outside) 1 interface
```

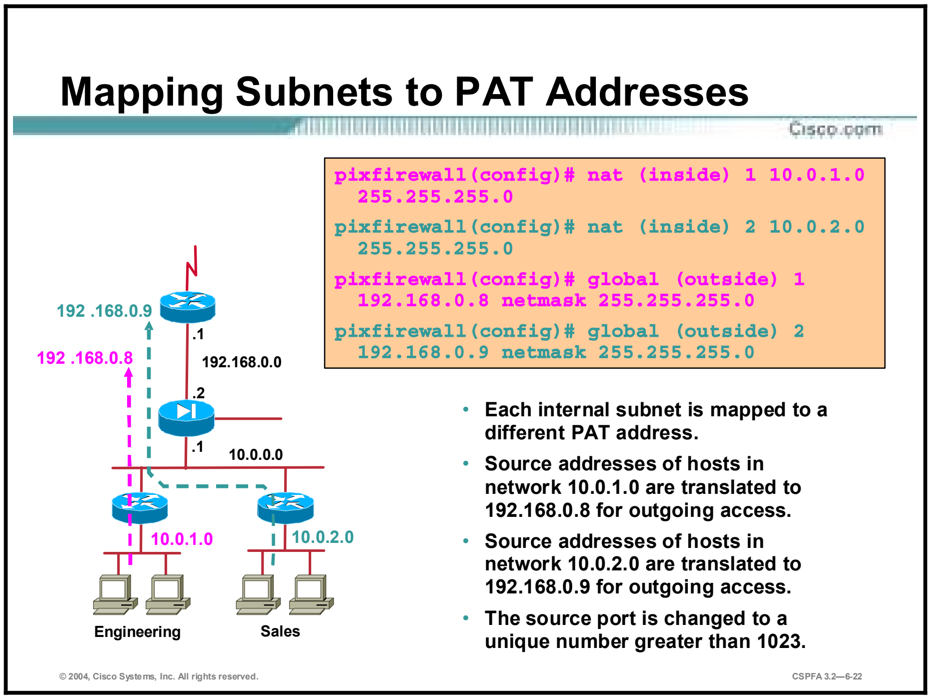
- The interface option of the global command enables use of the outside interface as the PAT address.
- The source addresses of hosts in network 10.0.0.0 are translated to 192.168.0.2 for outgoing access.
- The source port is changed to a unique number greater than 1024.

CSPFA 3.2—6-21

You can use the IP address of the outside interface as the PAT address by using the interface option of the **global** command. This is important when using the PIX Firewall Dynamic Host Configuration Protocol (DHCP) client feature. It allows the DHCP retrieved address to be used for PAT. DHCP support is discussed later in this course.

In the figure, source addresses for hosts on network 10.0.0.0 are translated to 192.168.0.2 for outgoing access, and the source port is changed to a unique number greater than 1023.

Note: When PAT is enabled on an interface, there should be no loss of TCP, UDP, and Internet Control Message Protocol (ICMP) services. These services allow termination at outside interface of the PIX Firewall unit.



Mapping Subnets to PAT Addresses

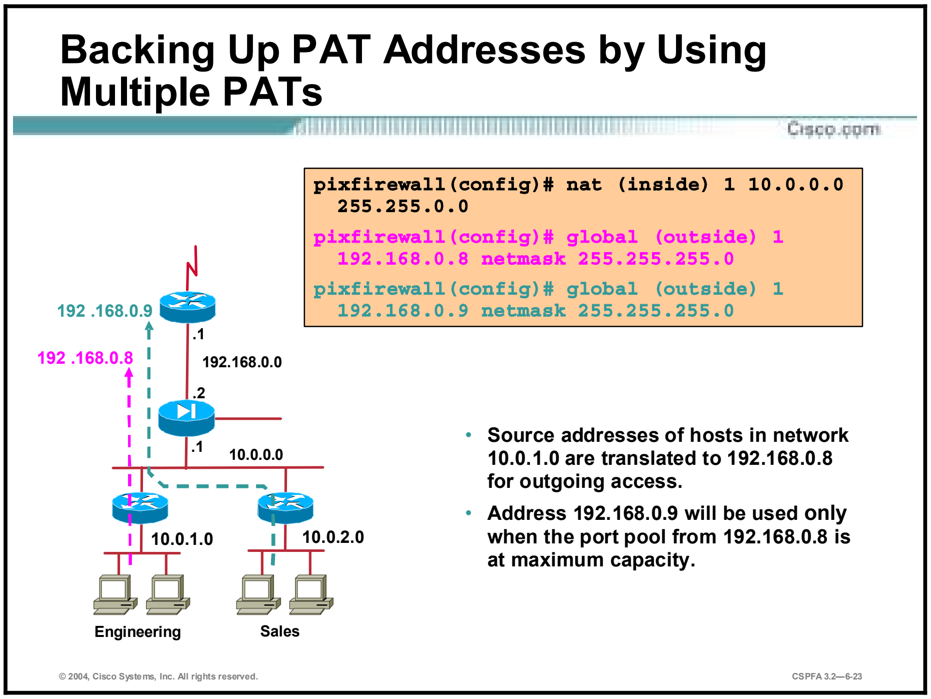
Cisco.com

```
pixfirewall(config)# nat (inside) 1 10.0.1.0
  255.255.255.0
pixfirewall(config)# nat (inside) 2 10.0.2.0
  255.255.255.0
pixfirewall(config)# global (outside) 1
  192.168.0.8 netmask 255.255.255.0
pixfirewall(config)# global (outside) 2
  192.168.0.9 netmask 255.255.255.0
```

- Each internal subnet is mapped to a different PAT address.
- Source addresses of hosts in network 10.0.1.0 are translated to 192.168.0.8 for outgoing access.
- Source addresses of hosts in network 10.0.2.0 are translated to 192.168.0.9 for outgoing access.
- The source port is changed to a unique number greater than 1023.

CSPFA 3.2—6-22

With Cisco PIX Firewall Software Version 5.2 and higher, you can specify multiple PATs to track use among different subnets. In the figure, network 10.0.1.0 and network 10.0.2.0 are mapped to different PAT addresses. This is done by using a separate **nat** and **global** command pair for each network. Outbound sessions from hosts on internal network 10.0.1.0 will appear to originate from address 192.168.0.8, and outbound sessions from hosts on internal network 10.0.2.0 will appear to originate from address 192.168.0.9.



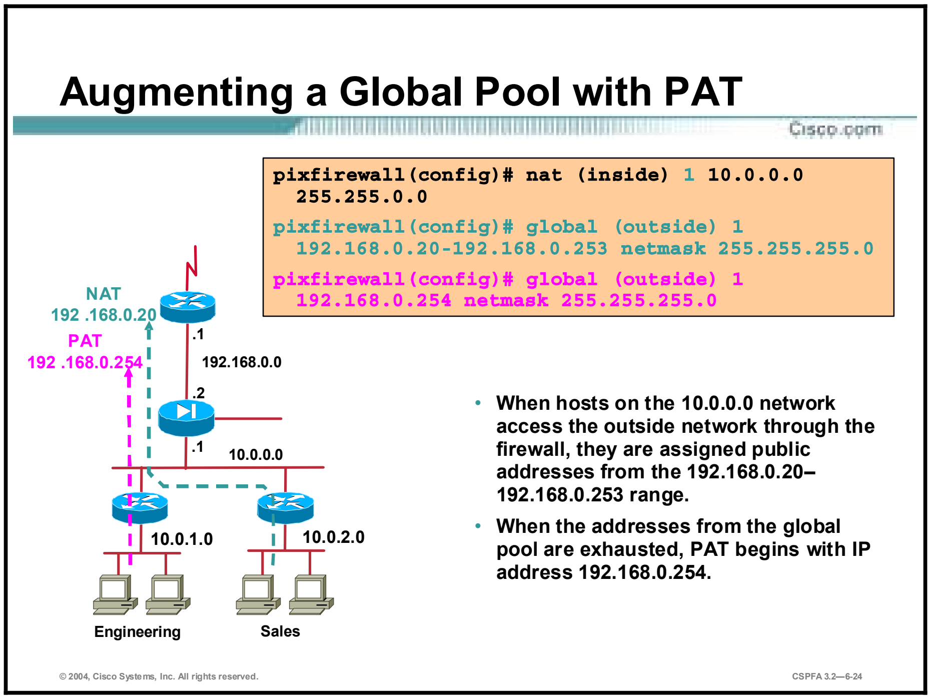
Backing Up PAT Addresses by Using Multiple PATs

```
pixfirewall(config)# nat (inside) 1 10.0.0.0
  255.255.0.0
pixfirewall(config)# global (outside) 1
  192.168.0.8 netmask 255.255.255.0
pixfirewall(config)# global (outside) 1
  192.168.0.9 netmask 255.255.255.0
```

- Source addresses of hosts in network 10.0.1.0 are translated to 192.168.0.8 for outgoing access.
- Address 192.168.0.9 will be used only when the port pool from 192.168.0.8 is at maximum capacity.

With PIX Firewall Software Version 5.2 and higher, you can also back up your PAT address by configuring multiple globals with the same nat_id.

In the figure, address 192.168.0.9 will be used for all outbound connections from network 10.0.0.0/16 when the port pool from 192.168.0.8 is at maximum capacity.



Augmenting a Global Pool with PAT

```
pixfirewall(config)# nat (inside) 1 10.0.0.0
  255.255.0.0
pixfirewall(config)# global (outside) 1
  192.168.0.20-192.168.0.253 netmask 255.255.255.0
pixfirewall(config)# global (outside) 1
  192.168.0.254 netmask 255.255.255.0
```

- When hosts on the 10.0.0.0 network access the outside network through the firewall, they are assigned public addresses from the 192.168.0.20–192.168.0.253 range.
- When the addresses from the global pool are exhausted, PAT begins with IP address 192.168.0.254.

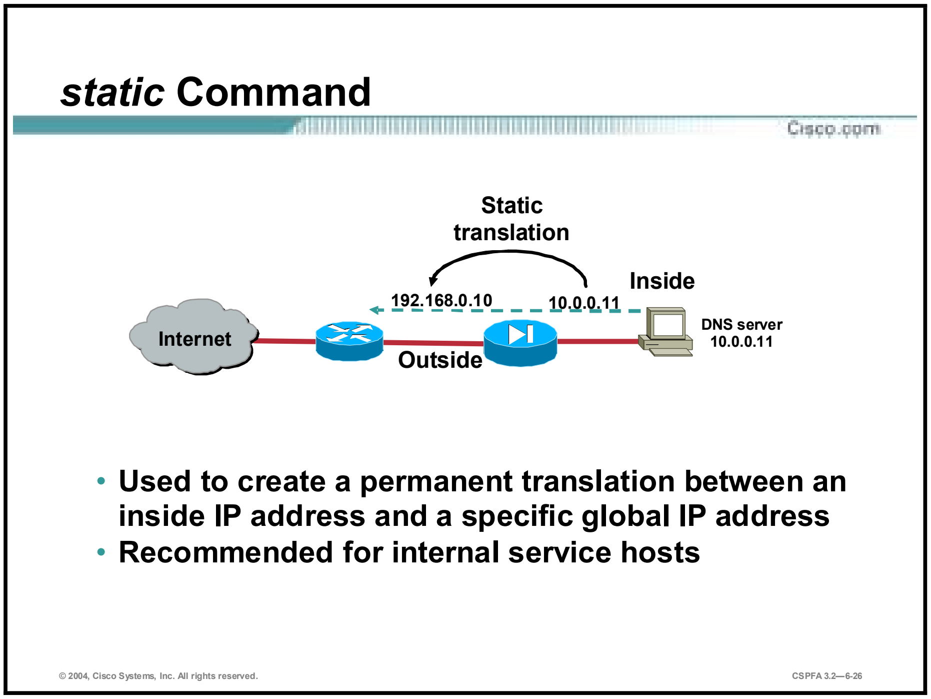
CSPFA 3.2—6-24

You can augment a pool of global addresses with PAT. When all IP addresses from the global pool are in use, the PIX Firewall begins PAT using the single IP address shown in the second **global** command.

In the figure, hosts on the 10.0.0.0/16 internal network are assigned addresses from the global pool 192.168.0.20 through 192.168.0.253 as they initiate outbound connections. When the addresses from the global pool are exhausted, packets from all hosts on network 10.0.0.0/16 appear to originate from 192.168.0.254.

static Command

This topic describes how to configure a permanent mapping between two IP addresses through a PIX Firewall.



static Command

- Used to create a permanent translation between an inside IP address and a specific global IP address
- Recommended for internal service hosts

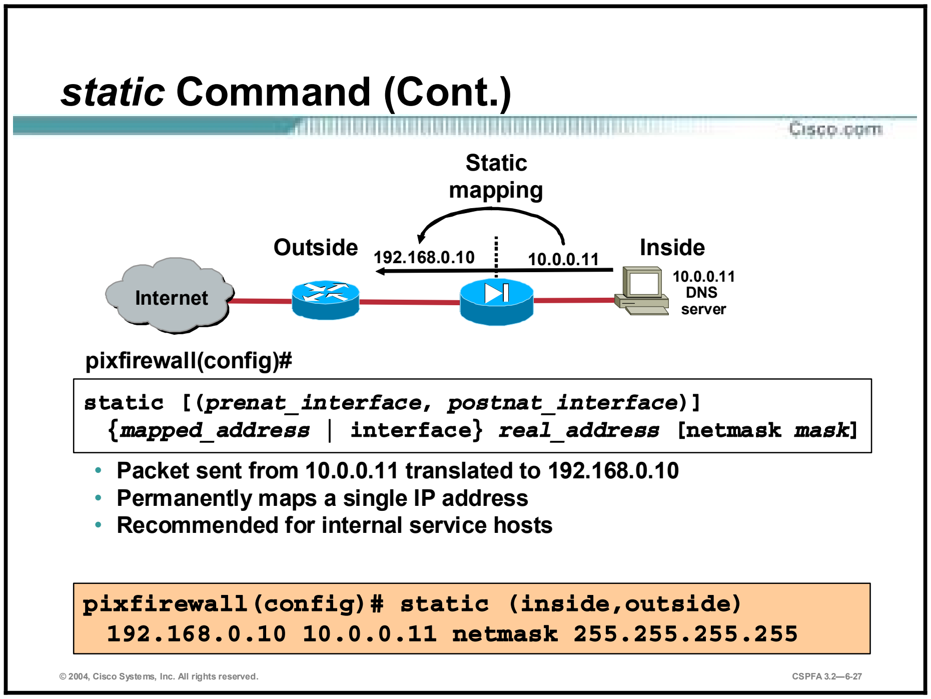
CSPFA 3.2—6-26

Use static translations when you want an inside host to always appear with a fixed address on the PIX Firewall global network. Static translations are used to map an inside host address to an outside, global address:

- Use the **static** command for outbound connections to ensure that packets leaving an inside host are always mapped to a specific global IP address (for example, an inside DNS or Simple Mail Transfer Protocol [SMTP] host).
- Use the **static** command for outbound connections that must be mapped to the same global IP address.

The following information can help you determine when to use static translations in the PIX Firewall:

- Do not create static translations with overlapping IP addresses. Each IP address should be unique.
- Static commands take precedence over **nat** and **global** command pairs.
- If a global IP address will be used for PAT, do not use the same global IP address for a static translation.



static Command (Cont.)

pixfirewall(config)#

```
static [(prenat_interface, postnat_interface)]
  {mapped_address | interface} real_address [netmask mask]
```

- Packet sent from 10.0.0.11 translated to 192.168.0.10
- Permanently maps a single IP address
- Recommended for internal service hosts

```
pixfirewall(config)# static (inside,outside)
  192.168.0.10 10.0.0.11 netmask 255.255.255.255
```

CSPFA 3.2—6-27

The **static** command creates a permanent mapping (called a static translation slot or xlate) between a local IP address and a global IP address. For outbound connections, use the **static** command to specify a global address to which the actual IP address of a local host will be translated. In the figure, when a packet from the client station 10.0.0.11 goes out through the PIX Firewall, it will have the source IP address of 192.168.0.10.

The syntax for the **static** command is as follows:

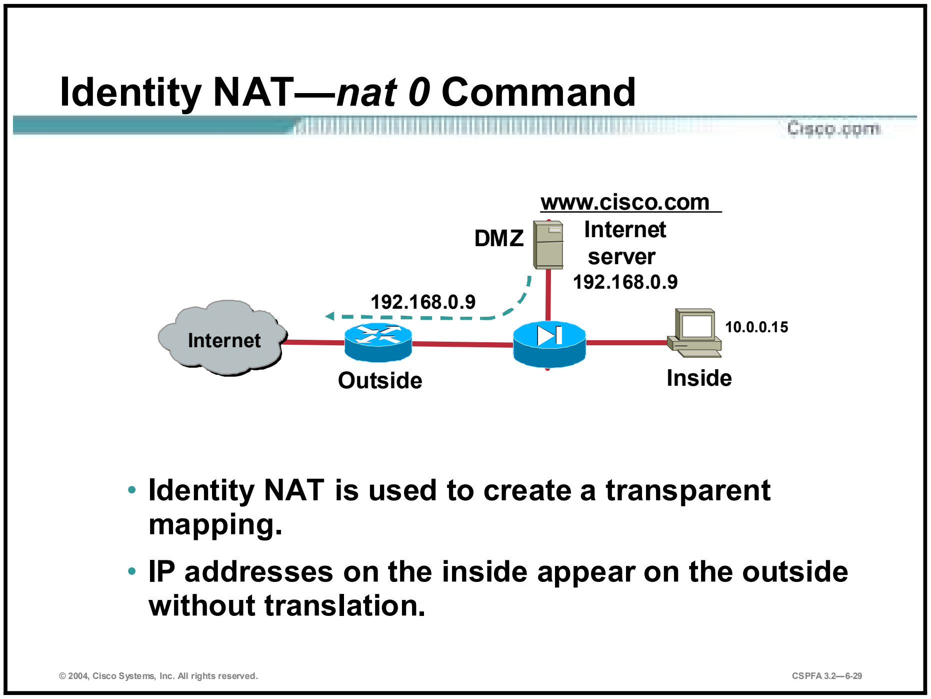
static [(prenat_interface, postnat_interface)] mapped_address | **interface** real_address [**netmask** mask]

prenat_interface	The network interface name. Usually the higher security level interface, in which case the translation is applied to the higher security level address.
postnat_interface	The lower security level interface when prenat_interface is the higher security level interface.
mapped_address	The address into which real_address is translated.
interface	Specifies to overload the global address from the interface.
real_address	The address to be mapped.
mask	The network mask that applies to both mapped_address and real_address.

Statics take precedence over **nat** and **global** command pairs. Use the **show static** command to view static statements in the configuration.

Identity NAT Command

This topic describes how to configure a transparent mapping of a host address through a PIX Firewall.

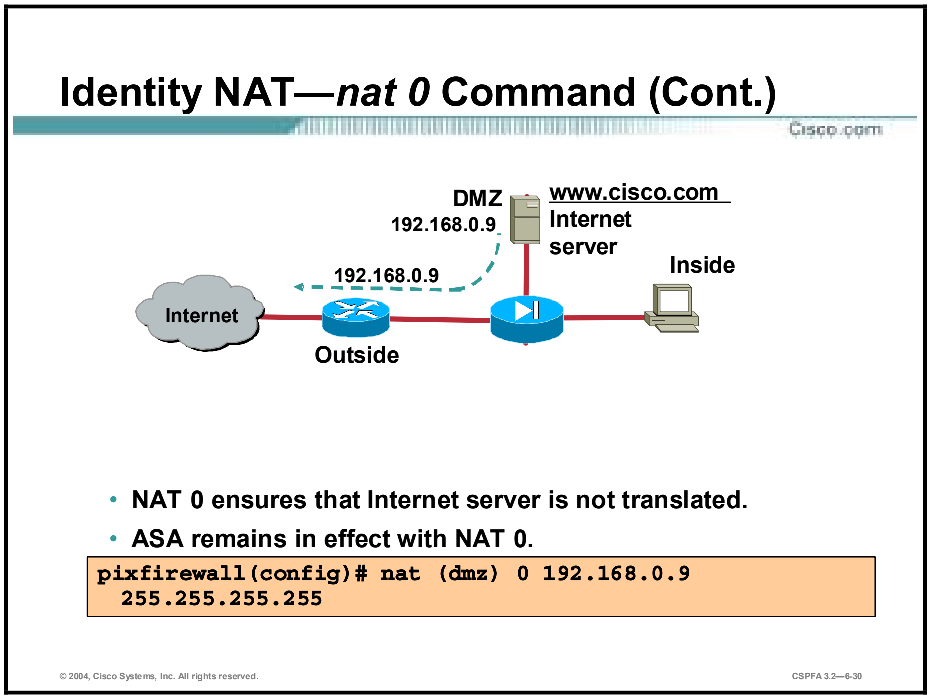


Identity NAT—*nat 0* Command

- Identity NAT is used to create a transparent mapping.
- IP addresses on the inside appear on the outside without translation.

CSPFA 3.2—6-29

Another feature to control outbound connections is the ability to control which internal IP addresses are visible on the outside. The **nat 0** command lets you disable address translation so that inside IP addresses are visible on the outside without address translation. Use this feature when you have InterNIC-registered IP addresses on your inside network that you want to be accessible on the outside network. Use of the **nat 0** command depends on your security policy. If your policy allows internal clients to have their IP addresses exposed to the Internet, then use the **nat 0** command to provide that service.



Identity NAT—*nat 0* Command (Cont.)

- NAT 0 ensures that Internet server is not translated.
- ASA remains in effect with NAT 0.

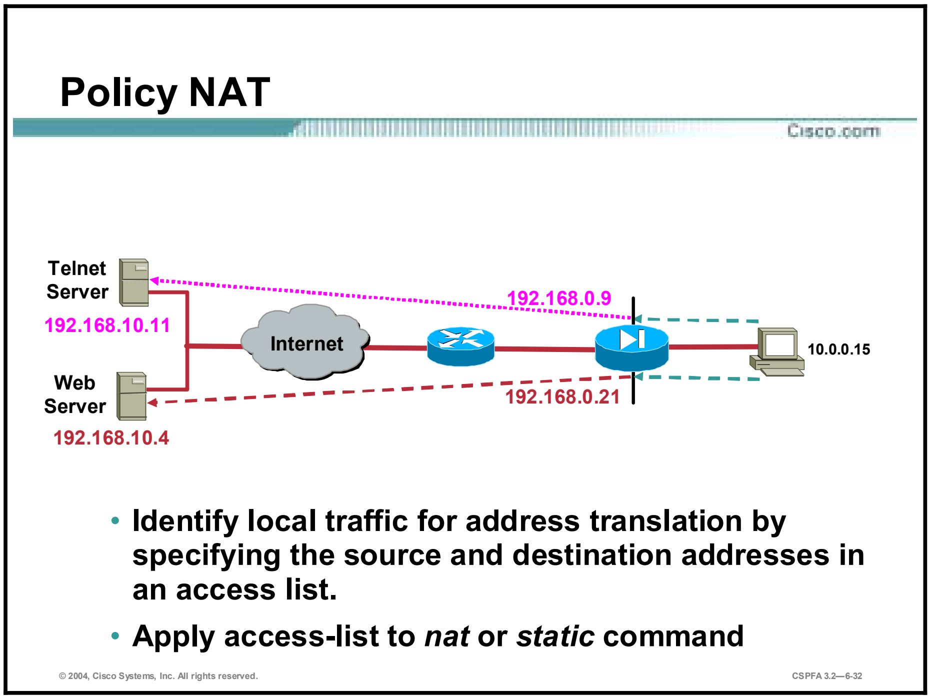
```
pixfirewall(config)# nat (dmz) 0 192.168.0.9
  255.255.255.255
```

CSPFA 3.2—6-30

In the figure shown, the address 192.168.0.9 is not translated. When you enter **nat (DMZ) 0 192.168.0.9 255.255.255.255**, the PIX Firewall displays the following message: nat 0 192.168.0.9 will be non-translated. It is important to note that NAT 0 enables the Internet server address to be visible on the outside interface. The administrator also needs to add a static in combination with a access-list to allow users on the outside to connect with the Internet server.

Policy NAT

This topic describes how to configure a Policy NAT.

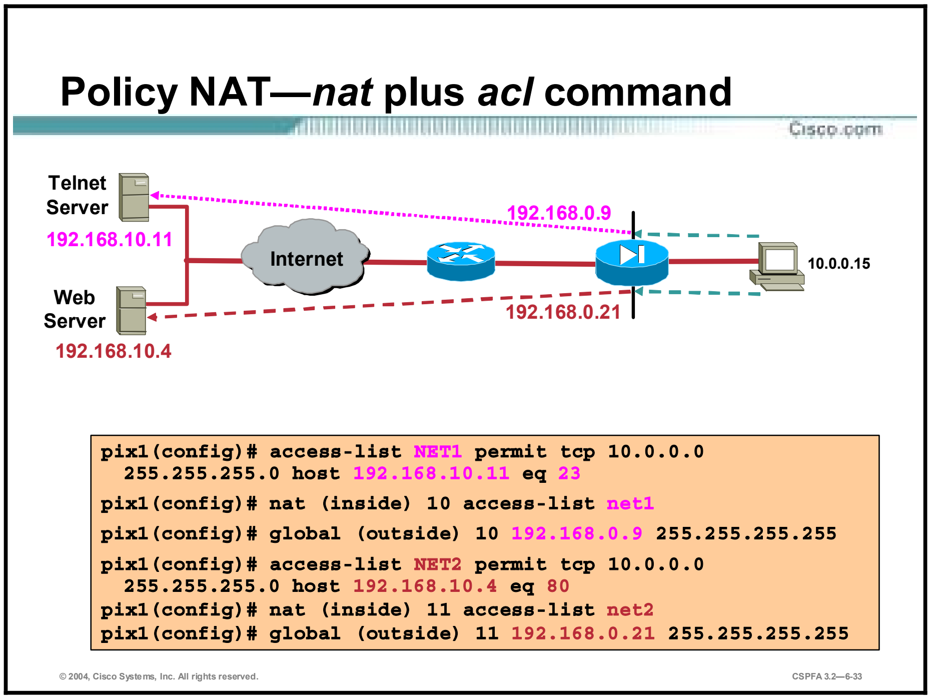


Policy NAT

- Identify local traffic for address translation by specifying the source and destination addresses in an access list.
- Apply access-list to *nat* or *static* command

CSPFA 3.2—6-32

Policy NAT enables the administrator to identify local traffic for address translation by specifying the source and destination addresses (and ports) in an access list. Regular NAT uses source addresses (and optionally ports) only. With policy NAT, the administrator can specify both the source and destination addresses (and optionally ports). In the example, if a host on the 10.0.0.0/24 network attempts to perform a telnet session with server 192.168.10.11, the PIX Firewall will translate the host address to a global address of 192.168.0.9. If the same host attempts a connection to web server 192.168.10.4, the pix will translate the source host address to 192.168.0.21. With policy NAT, the PIX Firewall can assign a source translation address based on the destination address.

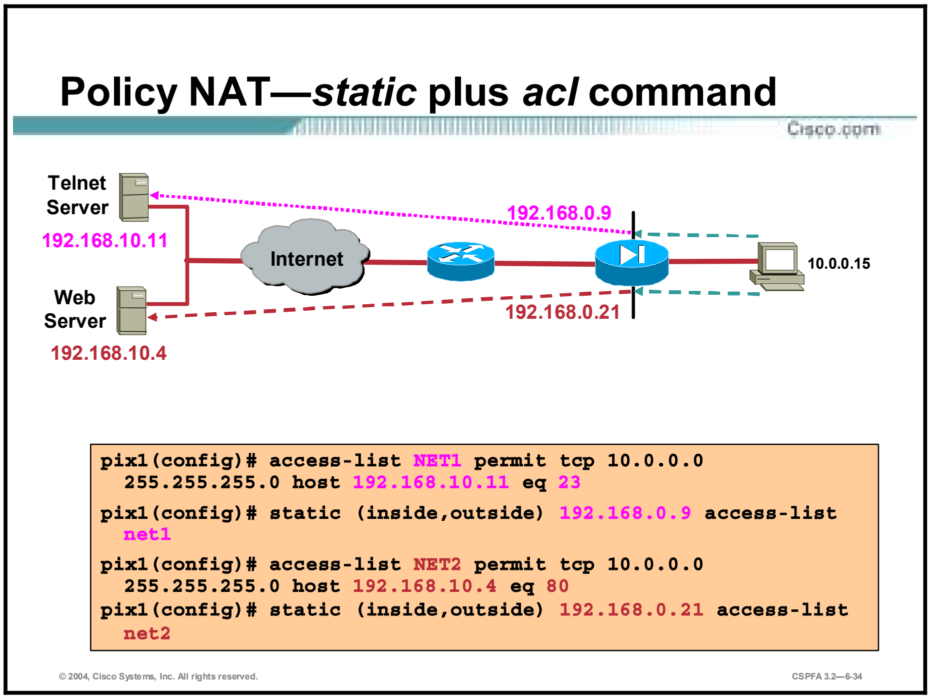


Policy NAT—*nat* plus *acl* command

```
pix1(config)# access-list NET1 permit tcp 10.0.0.0
  255.255.255.0 host 192.168.10.11 eq 23
pix1(config)# nat (inside) 10 access-list net1
pix1(config)# global (outside) 10 192.168.0.9 255.255.255.255
pix1(config)# access-list NET2 permit tcp 10.0.0.0
  255.255.255.0 host 192.168.10.4 eq 80
pix1(config)# nat (inside) 11 access-list net2
pix1(config)# global (outside) 11 192.168.0.21 255.255.255.255
```

CSPFA 3.2—6-33

With policy NAT, you can create multiple NAT statements that identify the same local address as long as the source/port and destination/port combination is unique for each statement. You can then match different global addresses to each source/port and destination/port pair. In the figure, a host on the 10.0.0.0/24 network is accessing two different servers. When the host accesses the server at 192.168.10.11, the local address is translated to 192.168.0.9. When the host accesses the server at 192.168.10.4, the local address is translated to 192.168.0.21. To accomplish this, an access-list is configured. The access-list defines the source and destination addresses and optional ports, access-list net1 for example. The access-list is then applied to a NAT statement, nat (inside) 10 access-list net1. When a packet traversing the PIX firewall matches the NAT ACL statement, the PIX translates the source address according to the corresponding global statement, global (outside) 10 192.168.0.9 255.255.255.255.



Policy NAT—*static* plus *acl* command

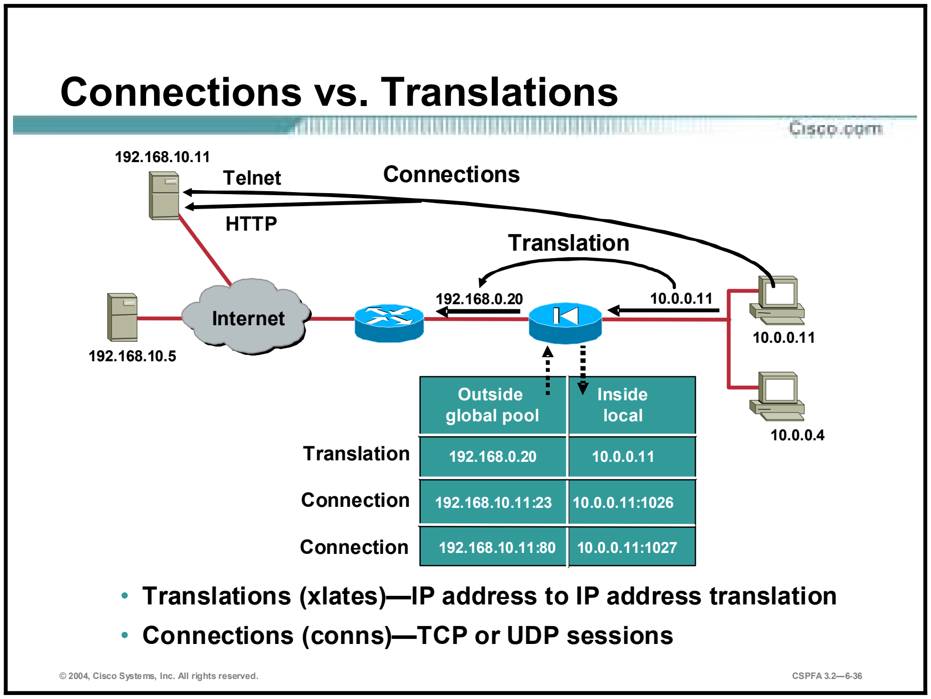
```
pix1(config)# access-list NET1 permit tcp 10.0.0.0
  255.255.255.0 host 192.168.10.11 eq 23
pix1(config)# static (inside,outside) 192.168.0.9 access-list
  net1
pix1(config)# access-list NET2 permit tcp 10.0.0.0
  255.255.255.0 host 192.168.10.4 eq 80
pix1(config)# static (inside,outside) 192.168.0.21 access-list
  net2
```

CSPFA 3.2—6-34

With policy NAT, an access list can also be applied to a static statement. The PIX Firewall translates the source address of every packet which matches the ACL to the source address defined in the static statement. In the figure, a host on the 10.0.0.0/24 network is accessing two different servers. When the host accesses the server at 192.168.10.11, the local address is translated to 192.168.0.9. When the host accesses the server at 192.168.10.4, the local address is translated to 192.168.0.21. To accomplish this, an access-list is configured. The access-list defines the source and destination addresses and optional ports, access-list net1 for example. The access-list is then applied to a static statement, static (inside,outside) 192.168.0.9 access-list net1. When a packet traversing the PIX firewall matches the access list statement, the PIX translates the source address according to the corresponding static statement, static (inside,outside) 192.168.0.9 access-list net1.

Connections and Translations

This topic describes the difference between connections and translations and how to verify them in a PIX Firewall.



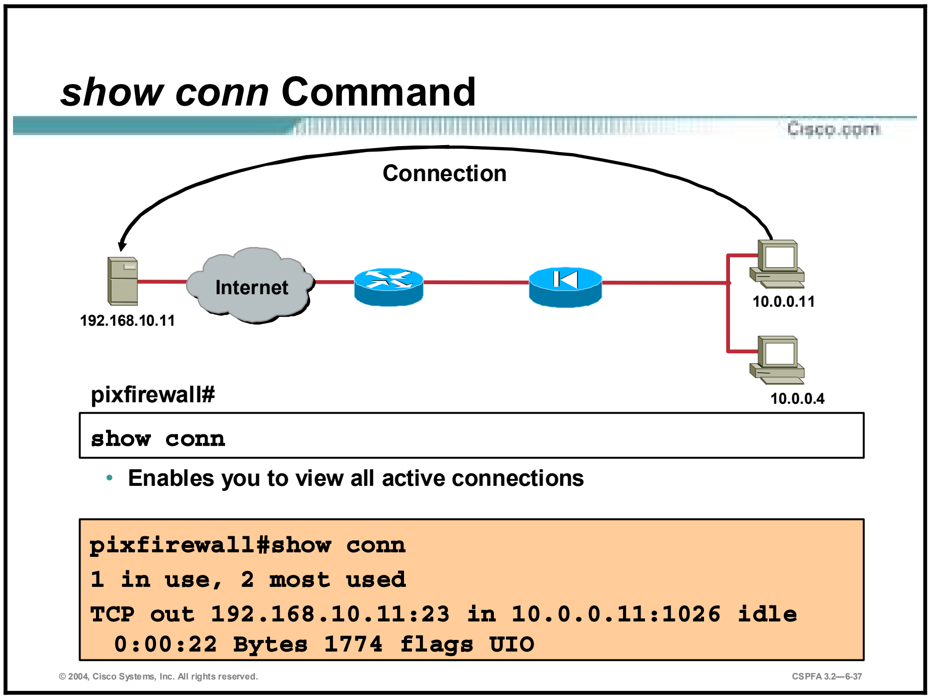
Connections vs. Translations

	Outside global pool	Inside local
Translation	192.168.0.20	10.0.0.11
Connection	192.168.10.11:23	10.0.0.11:1026
Connection	192.168.10.11:80	10.0.0.11:1027

- **Translations (xlates)—IP address to IP address translation**
- **Connections (conns)—TCP or UDP sessions**

CSPFA 3.2—6-36

Translations are at the IP layer, and connections are at the transport layer—TCP specifically. Connections are subsets of translations. You can have many connections open under one translation.



show conn Command

pixfirewall#

```
show conn
```

- Enables you to view all active connections

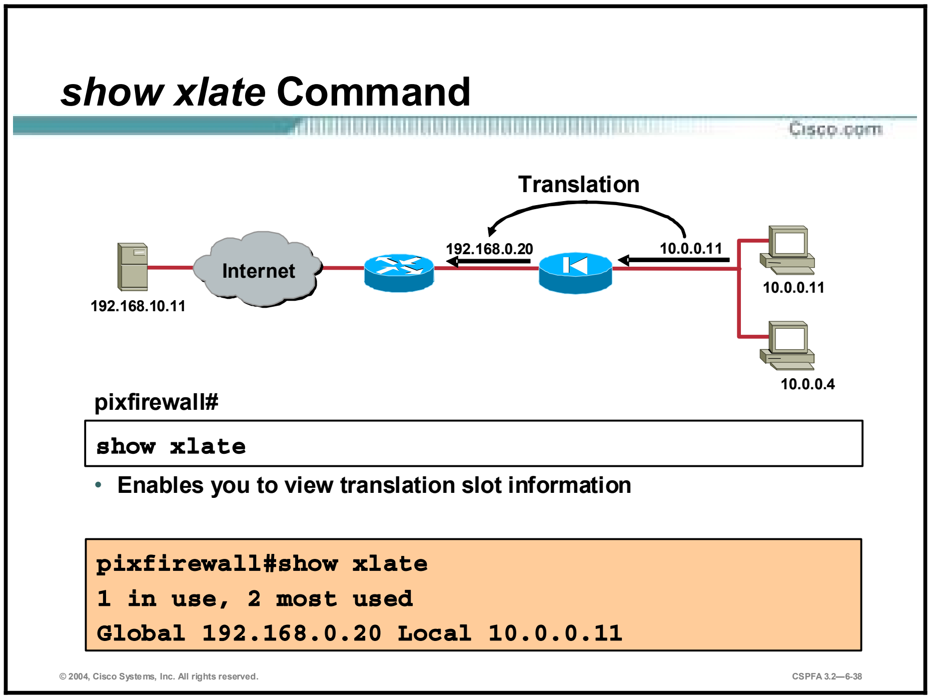
```
pixfirewall#show conn
1 in use, 2 most used
TCP out 192.168.10.11:23 in 10.0.0.11:1026 idle
  0:00:22 Bytes 1774 flags UIO
```

The **show conn** command displays the number of, and information about, active TCP connections.

The syntax for the **show conn** command is as follows:

show conn [count] | [detail]|[protocol tcp | udp | ***protocol*****][{foreign | local}** ***ip [-ip*****2]] [netmask** ***mas*****k]] [{lport | fport}** ***port1 [-port*****2]]**

count	Display only the number of used connections. The precision of the displayed count may vary depending on traffic volume and the type of traffic passing through the PIX Firewall unit.
detail	If specified, displays translation type and interface information.
{foreign | local} ***ip*** **[-*ip*2]** **netmask** ***mask***	Display active connections by the foreign IP address or by local IP address. Qualify foreign or local active connections by network mask.
{lport | fport} ***port1*****[-*port2*]**	Display foreign or local active connections by port. See “Ports” in the lesson “Using PIX Firewall Commands” for a list of valid port literal names.
protocol tcp | udp |*protocol*	Display active connections by protocol type. ***protocol*** is a protocol specified by number. See “Protocols” in the lesson “Using PIX Firewall Commands” for a list of valid protocol literal names.



show xlate Command

pixfirewall#

```
show xlate
```

- Enables you to view translation slot information

```
pixfirewall#show xlate
1 in use, 2 most used
Global 192.168.0.20 Local 10.0.0.11
```

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—6-38

The **xlate** command enables you to show or clear the contents of the translation (xlate) slots. Translation slots can remain indefinitely after key changes have been made. Always use **clear xlate** or reload after adding, changing, or removing **alias**, **access-list**, **global**, **nat**, **route**, or **static** commands in your configuration.

The syntax for the **xlate** command is as follows:

clear xlate [*global_ip* [*local_ip*]]

show xlate [*global_ip* [*local_ip*]]

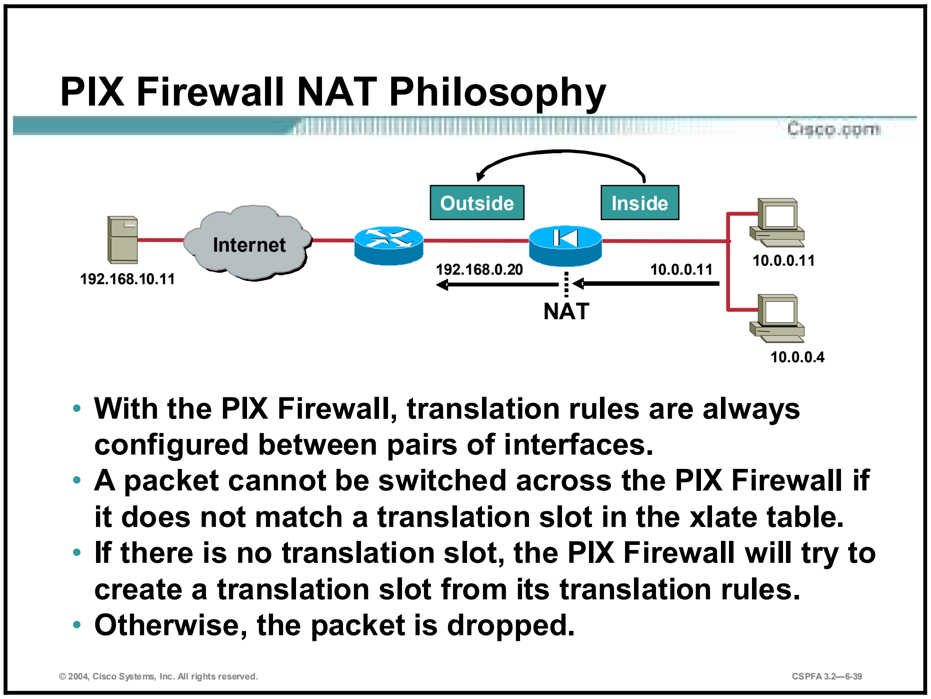
global_ip	The registered IP address to be used from the global pool.
local_ip	The local IP address from the inside network.

The show **timeout** command displays the idle time for connection and translation slots. If the slot has not been used for the idle time specified, the resource is returned to the free pool. TCP connection slots are freed approximately 60 seconds after a normal connection close sequence.

The following is sample output from the **show timeout** command:

show timeout

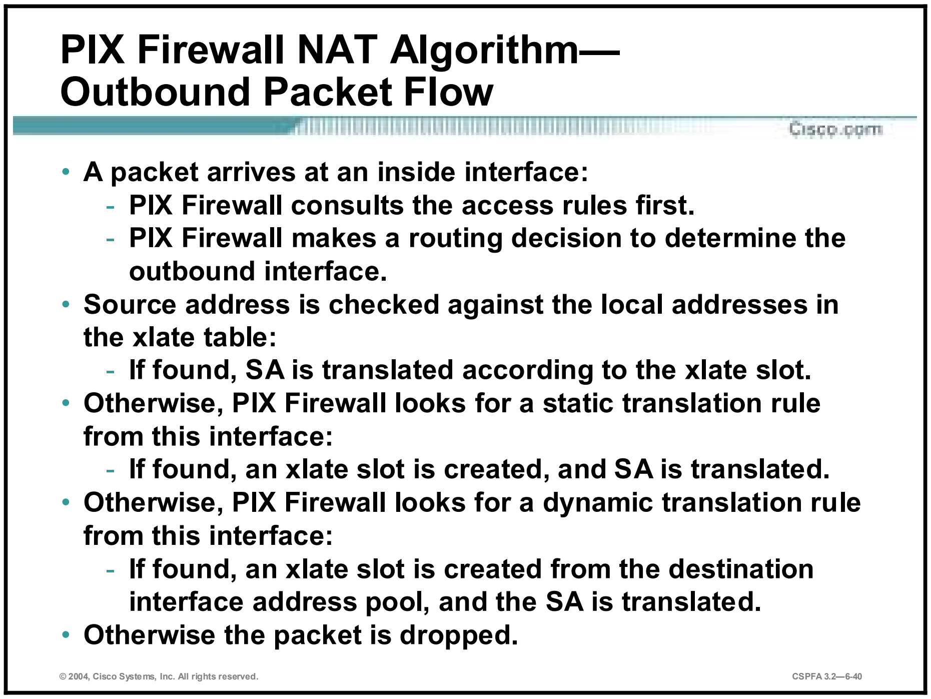
```
timeout xlate 3:00:00

timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00
```



PIX Firewall NAT Philosophy

- With the PIX Firewall, translation rules are always configured between pairs of interfaces.
- A packet cannot be switched across the PIX Firewall if it does not match a translation slot in the xlate table.
- If there is no translation slot, the PIX Firewall will try to create a translation slot from its translation rules.
- Otherwise, the packet is dropped.

Translations will be built for every source/destination interface pair. This permits the PIX Firewall to translate the same internal host to different addresses depending on the destination. A packet will not be forwarded by the PIX Firewall if the packet does not match any of the existing translation entries or if a translation entry cannot be established according to the translation rules. The exception to this is nat 0 which does not translate an address and does not use a translation slot.



PIX Firewall NAT Algorithm—Outbound Packet Flow

Cisco.com

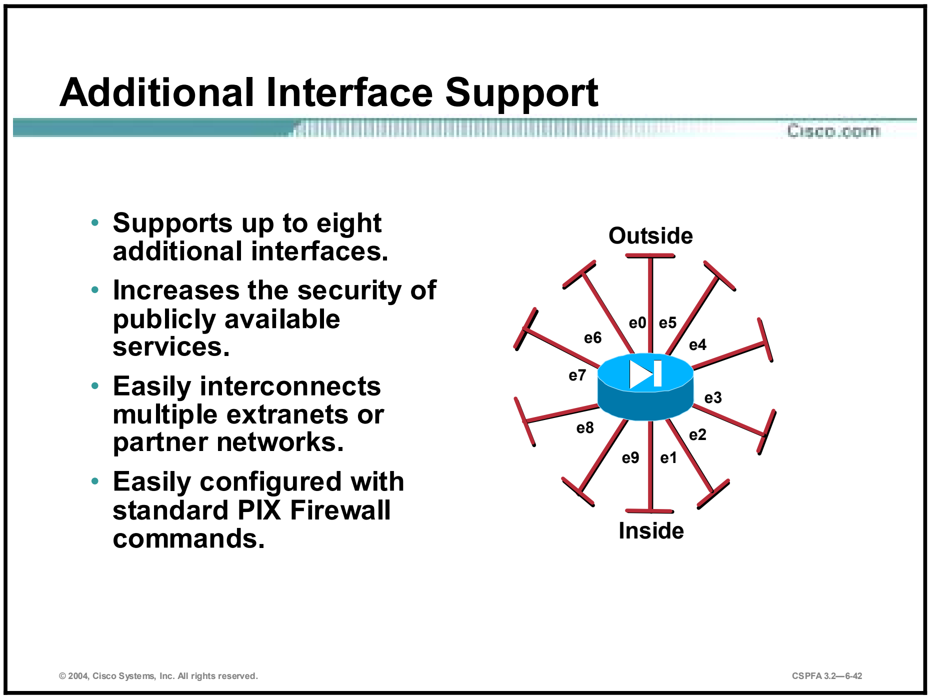
- A packet arrives at an inside interface:
  - PIX Firewall consults the access rules first.
  - PIX Firewall makes a routing decision to determine the outbound interface.
- Source address is checked against the local addresses in the xlate table:
  - If found, SA is translated according to the xlate slot.
- Otherwise, PIX Firewall looks for a static translation rule from this interface:
  - If found, an xlate slot is created, and SA is translated.
- Otherwise, PIX Firewall looks for a dynamic translation rule from this interface:
  - If found, an xlate slot is created from the destination interface address pool, and the SA is translated.
- Otherwise the packet is dropped.

CSPFA 3.2—6-40

For outbound connections, the destination interface is evaluated according to the routing table; it must be at a lower security level than the originating interface. The PIX Firewall then compares the source address with the inside-local entries in the xlate table. If a match is found, the translation is used. If no translation for this source/destination interface pair exists, a new translation is created according to the translation rules. If no match for this new connection is found, the packet is dropped.

Configuring Multiple Interfaces

This topic describes how to configure multiple interfaces on the PIX Firewall.



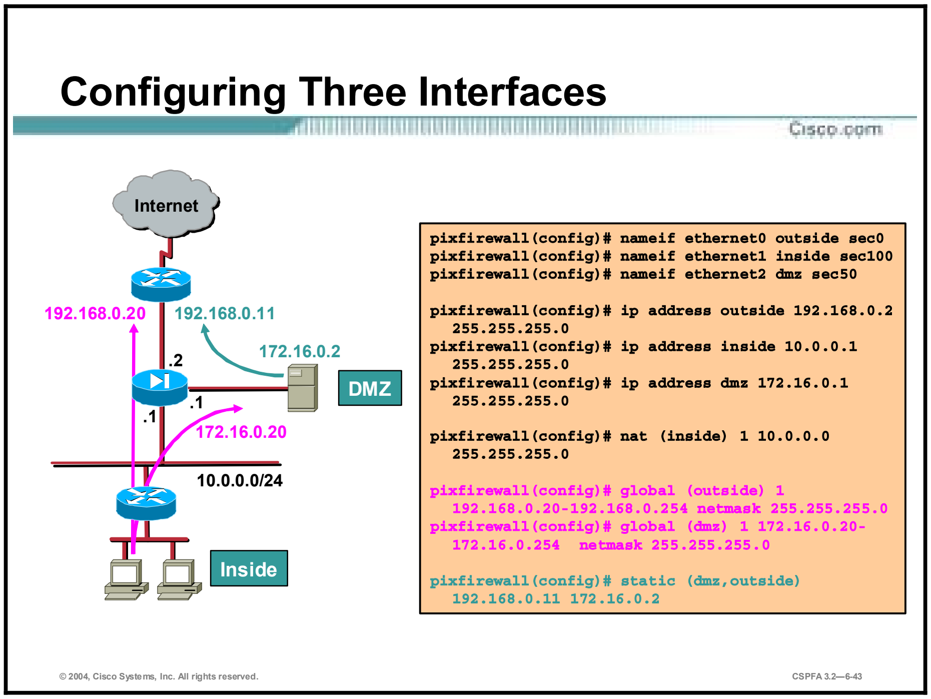
Additional Interface Support

Cisco.com

- Supports up to eight additional interfaces.
- Increases the security of publicly available services.
- Easily interconnects multiple extranets or partner networks.
- Easily configured with standard PIX Firewall commands.

CSPFA 3.2—6-42

The PIX Firewall supports up to eight additional physical interfaces for platform extensibility and security policy enforcement on publicly accessible services. The multiple physical interfaces enable the PIX Firewall to protect publicly accessible web, mail, and DNS servers on the DMZ. Web-based and traditional Electronic Data Interchange (EDI) applications that link vendors and customers are also more secure and scalable when implemented using a physically separate network. As the trend toward building these extranet and partnernet applications accelerates, the PIX Firewall is already prepared to accommodate them.



Configuring Three Interfaces

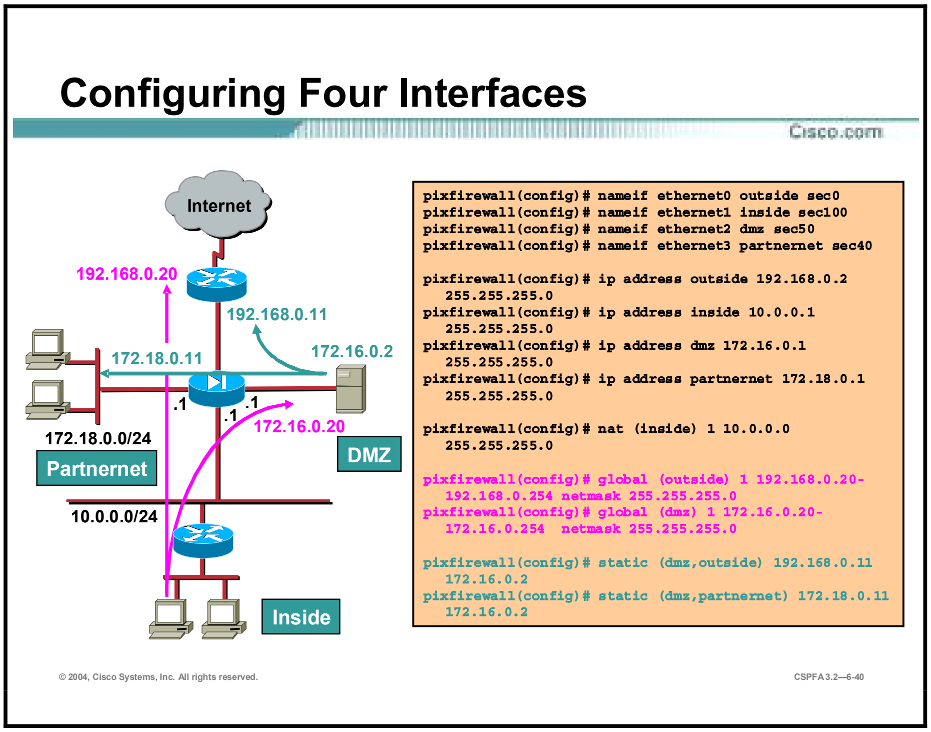
```
pixfirewall(config)# nameif ethernet0 outside sec0
pixfirewall(config)# nameif ethernet1 inside sec100
pixfirewall(config)# nameif ethernet2 dmz sec50

pixfirewall(config)# ip address outside 192.168.0.2
   255.255.255.0
pixfirewall(config)# ip address inside 10.0.0.1
   255.255.255.0
pixfirewall(config)# ip address dmz 172.16.0.1
   255.255.255.0

pixfirewall(config)# nat (inside) 1 10.0.0.0
   255.255.255.0

pixfirewall(config)# global (outside) 1
   192.168.0.20-192.168.0.254 netmask 255.255.255.0
pixfirewall(config)# global (dmz) 1 172.16.0.20-
   172.16.0.254  netmask 255.255.255.0

pixfirewall(config)# static (dmz,outside)
   192.168.0.11 172.16.0.2
```

CSPFA 3.2—6-43

A third interface is configured as shown in the figure. When your PIX Firewall is equipped with three or more interfaces, use the following guidelines to configure it while employing NAT:

- The outside interface cannot be renamed or given a different security level.
- An interface is always outside with respect to another interface that has a higher security level. Packets cannot flow between interfaces that have the same security level.
- Use a single default route statement to the outside interface only. Set the default route with the **route** command.
- Use the **nat** command to let users on the respective interfaces start outbound connections. Associate the nat_id with the nat_id in the **global** command statement. The valid identification numbers can be any positive number up to two billion.
- After you have completed a configuration in which you add, change, or remove a **global** statement, save the configuration and enter the **clear xlate** command so that the IP addresses will be updated in the translation table.
- To permit access to servers on protected networks from a less secure interface, use the **static** and **access-list** commands. The **static** and **access-list** commands are covered later in the course.

In the figure, hosts on the inside network can access the outside network. The original 10.0.0.0/24 address is assigned an address from the global pool of 192.168.0.20-254. When an inside host accesses the DMZ, the original address is assigned an address from the global pool of 172.16.0.20-254. Last, the DMZ server is always translated to an outside address of 192.168.0.11.



Configuring Four Interfaces

```
pixfirewall(config)# nameif ethernet0 outside sec0
pixfirewall(config)# nameif ethernet1 inside sec100
pixfirewall(config)# nameif ethernet2 dmz sec50
pixfirewall(config)# nameif ethernet3 partnernet sec40

pixfirewall(config)# ip address outside 192.168.0.2
   255.255.255.0
pixfirewall(config)# ip address inside 10.0.0.1
   255.255.255.0
pixfirewall(config)# ip address dmz 172.16.0.1
   255.255.255.0
pixfirewall(config)# ip address partnernet 172.18.0.1
   255.255.255.0

pixfirewall(config)# nat (inside) 1 10.0.0.0
   255.255.255.0

pixfirewall(config)# global (outside) 1 192.168.0.20-
   192.168.0.254 netmask 255.255.255.0
pixfirewall(config)# global (dmz) 1 172.16.0.20-
   172.16.0.254  netmask 255.255.255.0

pixfirewall(config)# static (dmz,outside) 192.168.0.11
   172.16.0.2
pixfirewall(config)# static (dmz,partnernet) 172.18.0.11
   172.16.0.2
```

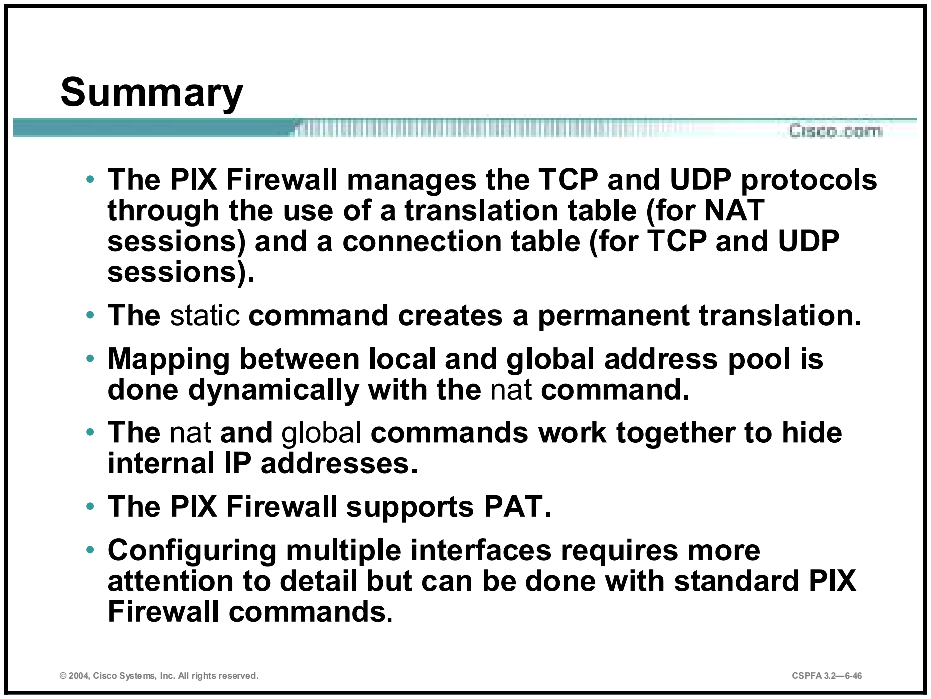
CSPFA 3.2—6-40

In the figure shown, the PIX Firewall has four interfaces. Users on the inside have access to the DMZ and the outside. The server 172.16.0.2 is visible on the outside as 192.168.0.11 and on the partnernet as 172.18 0.11. Configuring four interfaces requires more attention to detail, but they are configured with standard PIX Firewall commands. To enable users on a higher security level interface to access hosts on a lower security interface, use the **nat** and **global** commands (for example, when users on the inside interface have access to the web server on the DMZ interface).

To let users on a lower security level interface, such as users on the partnernet interface, access hosts on a higher security interface (DMZ), use the **static** and **access-list** commands. As seen in the figure, the partnernet has a security level of 40 and the DMZ has a security level of 50. The DMZ will use **nat** and **global** commands to speak with the partnernet and will use **static** commands and **access-list** commands to receive traffic originating from the partnernet. **Static** and **access-list** commands are covered later in this course.

Summary

This topic summarizes what you learned in this lesson.



Summary

Cisco.com

- **The PIX Firewall manages the TCP and UDP protocols through the use of a translation table (for NAT sessions) and a connection table (for TCP and UDP sessions).**
- **The** static **command creates a permanent translation.**
- **Mapping between local and global address pool is done dynamically with the** nat **command.**
- **The** nat **and** global **commands work together to hide internal IP addresses.**
- **The PIX Firewall supports PAT.**
- **Configuring multiple interfaces requires more attention to detail but can be done with standard PIX Firewall commands.**

CSPFA 3.2—6-46

Lab Exercise—Configuring Access Through the PIX Firewall

Complete the following lab exercise to practice what you learned in this chapter.

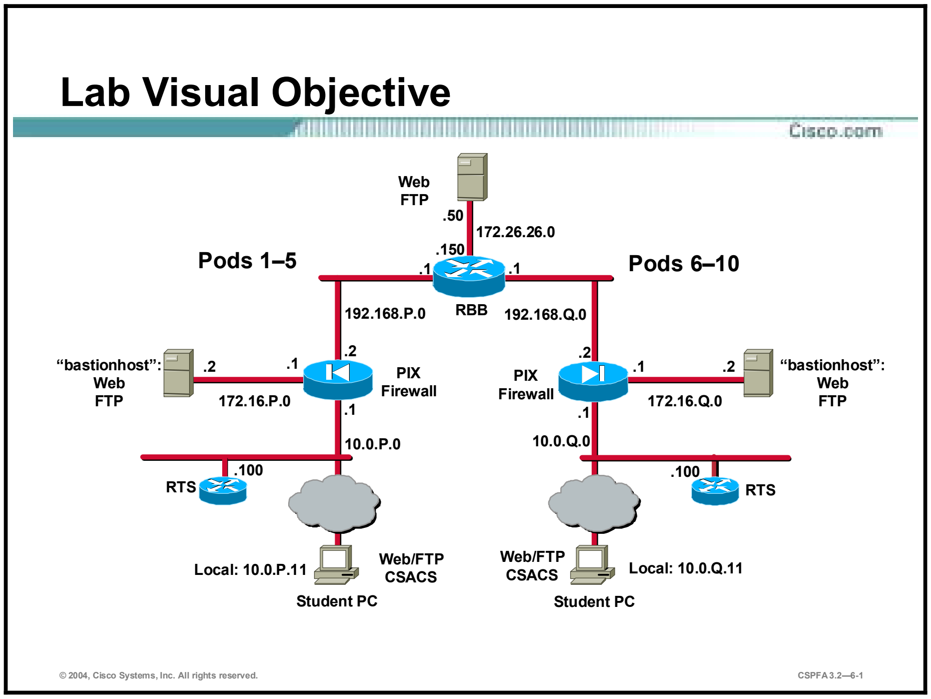
Objectives

In this lab exercise, you will complete the following tasks:

- Configure the PIX Firewall to allow users on the inside interface to access the bastion host.
- Establish a Telnet connection to the backbone router.

Visual Objective

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—6-1

Task 1—Configure the PIX Firewall to Allow Users on the Inside Interface to Access the Bastion Host

Configure the PIX Firewall to allow access to the DMZ from the inside network.

Step 1 Assign one pool of IP addresses for hosts on the public DMZ:

```
pixP(config)# global (dmz) 1 172.16.P.20-172.16.P.254 netmask
255.255.255.0
```

(where P = pod number)

Step 2 Clear the translation table so that the global IP address will be updated in the table:

```
pixP(config)# clear xlate
```

Step 3 Write the current configuration to Flash memory:

```
pixP(config)# write memory
```

Step 4 Test web access to your bastion host from your PC by completing the following sub-steps:

1. Open a web browser on your PC.
2. Use the web browser to access your bastion host by entering **http://172.16.P.2.** (where P = pod number)
3. The home page of the bastion host should appear on your web browser.

Step 5 Use the **show arp, show conn**, and **show xlate** commands to observe the transaction:

```
pixP(config)# show arp
outside 192.168.P.1 00e0.1e41.8762
inside insidehost 00e0.b05a.d509
dmz bastionhost 00e0.1eb1.78df
pixP(config)# show xlate
1 in use, 1 most used
Global 172.16.P.20 Local insidehost
```

Q1) How many translations are in use in the translation table?

A) ____________________

List the translations table host entries below:

Global ____________________ Local ____________________

```
pixP(config)# show conn
2 in use, 2 most used
TCP out bastionhost:80 in insidehost:1076 idle 0:00:07 Bytes 461 flags
UIO
TCP out bastionhost:80 in insidehost:1075 idle 0:00:07 Bytes 1441
flags UIO
```

(where P = pod number)

Q2) How many connections are in use in the connection table?

A) ____________________

List the connections table host entries below:

TCP out (host)_____________: (port)___ in (host)______________: (port)_____

TCP out (host)_____________: (port)___ in (host)______________: (port)_____

Step 6 Test FTP access to the bastion host from your PC by completing the following sub-steps:

1. Establish an FTP session to the bastion host by choosing **Start>Run>ftp 172.16.P.2**. You have reached the bastion host if you receive the message “Connected to 172.16.P.2.” (where P = pod number)

2. Log in to the FTP session:

```
User (172.16.P.2(none)): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password: Cisco
```

(where P = pod number)

Step 7 Use the **show conn** and **show xlate** commands to observe the FTP connection.

Q3) How many translations are in use in the translation table?

A) ____________________

List the translations table host entries below:

Global ____________________ Local ____________________

Q4) How many connections are in use in the connection table?

A) ____________________

List the connections table host entries below:

TCP out (host)_____________: (port)___ in (host)______________: (port)_____

TCP out (host)_____________: (port)___ in (host)______________: (port)_____

TCP out (host)_____________: (port)___ in (host)______________: (port)_____

Task 2—Establish a Telnet Connection to the Backbone Router

Establish a Telnet connection from the insidehost to the backbone router. View the translation and the connection tables.

Step 1 From your Windows command line, establish a Telnet session to the backbone router:

```
C:\> telnet 192.168.P.1
```

(where P = pod number)

Step 2 Use the **show conn** and **show xlate** commands to observe the transaction.

Q1) How many translations are in the translation table?

A) ____________________

Q2) List the translations table host entries below:

Global ____________________ Local ____________________

Global ____________________ Local ____________________

Q3) How many connections are in the connection table?

A) ____________________

List the connection host entries below:

TCP out (host)_____________: (port)___ in (host)______________: (port)_____

TCP out (host)_____________: (port)___ in (host)______________: (port)_____

TCP out (host)_____________: (port)___ in (host)______________: (port)_____

TCP out (host)_____________: (port)___ in (host)______________: (port)_____

Step 3 Clear the translation table.

```
pixP(config)# clear xlate
```

Step 4 Show the translation table.

```
pixP(config)# show xlate
```

Q4) How many translations are in use?

A) ____________________

Step 5 Show the connection table.

```
pixP(config)# show conn
```

Q5) How many connections are in use?

A) ____________________

Step 6 Quit the FTP session if you were able to connect and log in:

```
ftp> quit
```

Step 7 Close the Telnet command prompt window.

Step 8 Close the web browser.

Step 9 Save your configuration.

```
pixP(config)# write memory
```

Step 10 Write the current configuration to the terminal and verify that you have entered the previous commands correctly. Your configuration should appear similar to the following:

```
pixP(config)# write terminal
Building configuration...
: Saved
:
PIX Version 6.3(1)
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
nameif ethernet3 intf3 security15
nameif ethernet4 intf4 security20
nameif ethernet5 intf5 security25
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname pixP
fixup protocol ftp 21
fixup protocol http 80
fixup protocol h323 h225 1720
fixup protocol h323 ras 1718-1719
fixup protocol ils 389
fixup protocol rsh 514
fixup protocol rtsp 554
fixup protocol smtp 25
fixup protocol sqlnet 1521
fixup protocol sip 5060
fixup protocol skinny 2000
names
name 172.16.P.2 bastionhost
name 10.1.P.11 insidehost
pager lines 24
logging on
logging buffered debugging
logging trap debugging
logging host inside insidehost
interface ethernet0 100full
interface ethernet1 100full
interface ethernet2 100full
interface ethernet3 auto shutdown
interface ethernet4 auto shutdown
interface ethernet5 auto shutdown
mtu outside 1500
mtu inside 1500
mtu dmz 1500
mtu intf3 1500
mtu intf4 1500
mtu intf5 1500
ip address outside dhcp
ip address inside 10.0.P.1 255.255.255.0
ip address dmz 172.16.P.1 255.255.255.0
no ip address intf3
no ip address intf4
no ip address intf5
ip audit info action alarm
ip audit attack action alarm
no failover
failover timeout 0:00:00
failover poll 15
no failover ip address outside
no failover ip address inside
no failover ip address dmz
no failover ip address intf3
no failover ip address intf4
no failover ip address intf5
```

```
pdm history enable
arp timeout 14400
global (outside) 1 192.168.P.20-192.168.P.254 netmask 255.255.255.0
global (dmz) 1 172.16.P.20-172.16.P.254 netmask 255.255.255.0
nat (inside) 1 10.1.P.0 255.255.255.0 0 0
route outside 0.0.0.0 0.0.0.0 192.168.P.1 1
route inside 10.1.P.0 255.255.255.0 10.0.P.102 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 rpc 0:10:00 h323
0:05:00 si
p 0:30:00 sip_media 0:02:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server LOCAL protocol local
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
floodguard enable
no sysopt route dnat
telnet timeout 5
ssh timeout 5
terminal width 80
Cryptochecksum:65677978f6b81613892109e0f68af9d6
: end
```

7

Access Control Lists and Content Filtering

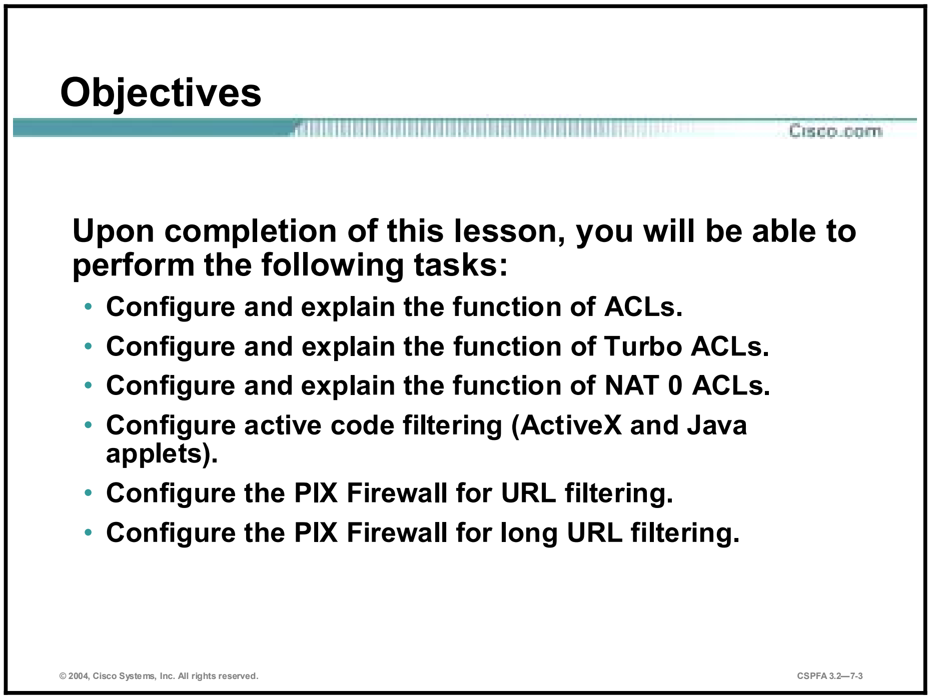
Overview

This lesson includes the following topics:

- Objectives
- ACLs
- Using ACLs
- Malicious active code filtering
- URL filtering
- Summary
- Lab exercise

Objectives

This topic lists the lesson's objectives.



Objectives

Cisco.com

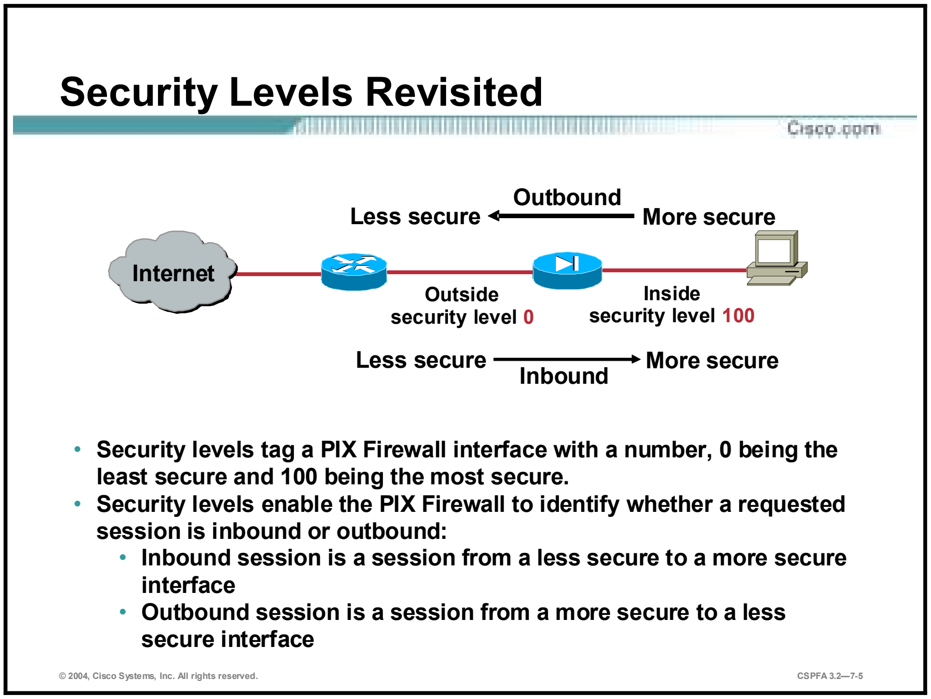
Upon completion of this lesson, you will be able to perform the following tasks:

- Configure and explain the function of ACLs.
- Configure and explain the function of Turbo ACLs.
- Configure and explain the function of NAT 0 ACLs.
- Configure active code filtering (ActiveX and Java applets).
- Configure the PIX Firewall for URL filtering.
- Configure the PIX Firewall for long URL filtering.

CSPFA 3.2—7-3

ACLs

This topic discusses access control through the Cisco Secure PIX Firewall using an access control list (ACL).



Security Levels Revisited

Cisco.com

Outbound

Less secure ← More secure

Internet

Outside
security level 0

Inside
security level 100

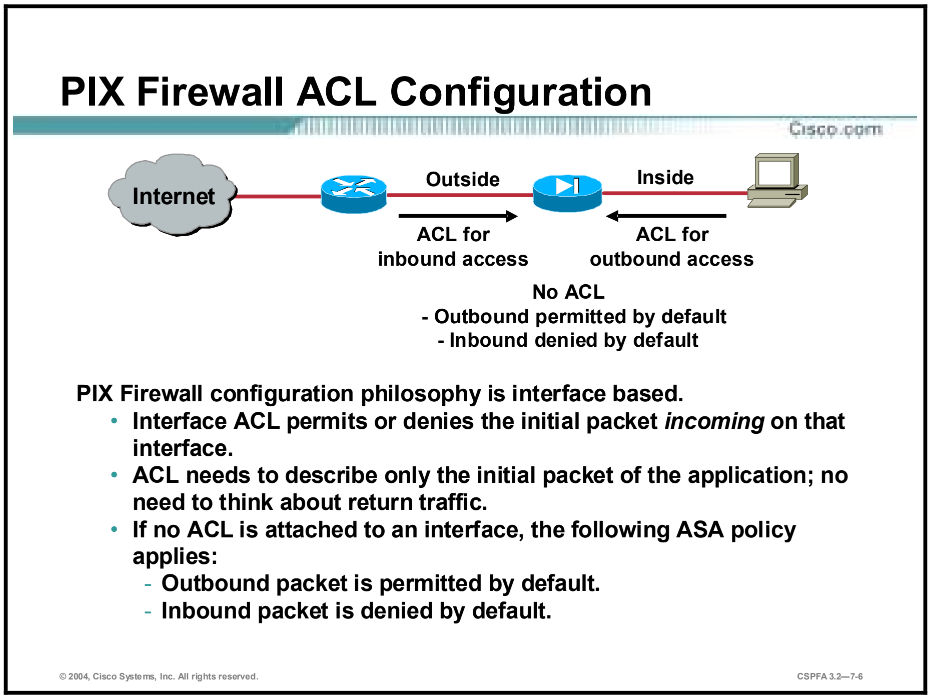
Less secure → More secure

Inbound

- Security levels tag a PIX Firewall interface with a number, 0 being the least secure and 100 being the most secure.
- Security levels enable the PIX Firewall to identify whether a requested session is inbound or outbound:
  - Inbound session is a session from a less secure to a more secure interface
  - Outbound session is a session from a more secure to a less secure interface

CSPFA 3.2—7-5

The configuration of every PIX defaults to an inside interface with a level of 100 and an outside interface with a level of 0. There's nothing more secure than the internal network, and nothing less secure than the external network. By default, once address translation is configured, all communications are permitted in an outbound direction, from a more secure to a less secure level. By default, all communications are prohibited in an inbound direction, from a less secure to a more secure level.



PIX Firewall ACL Configuration

Cisco.com

Internet

Outside

Inside

ACL for
inbound access

ACL for
outbound access

No ACL
- Outbound permitted by default
- Inbound denied by default

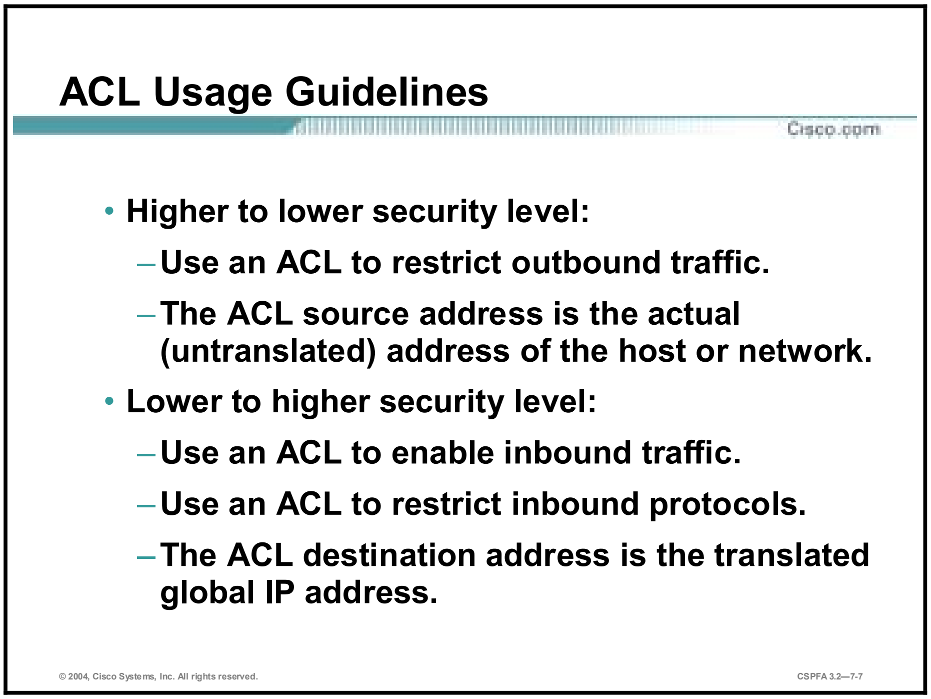
PIX Firewall configuration philosophy is interface based.

- Interface ACL permits or denies the initial packet *incoming* on that interface.
- ACL needs to describe only the initial packet of the application; no need to think about return traffic.
- If no ACL is attached to an interface, the following ASA policy applies:
  - Outbound packet is permitted by default.
  - Inbound packet is denied by default.

CSPFA 3.2—7-6

ASA check applies to every packet of a "conversation". Access-lists (ACLs) are only evaluated once per connection. ACLs can work in both directions. Once it is configured, it is activated with an **access-group** command. If no ACL is attached to an interface, the following default ASA policy applies:

- Outbound permitted by default unless explicitly denied
- Inbound denied by default unless explicitly permitted


ACL Usage Guidelines

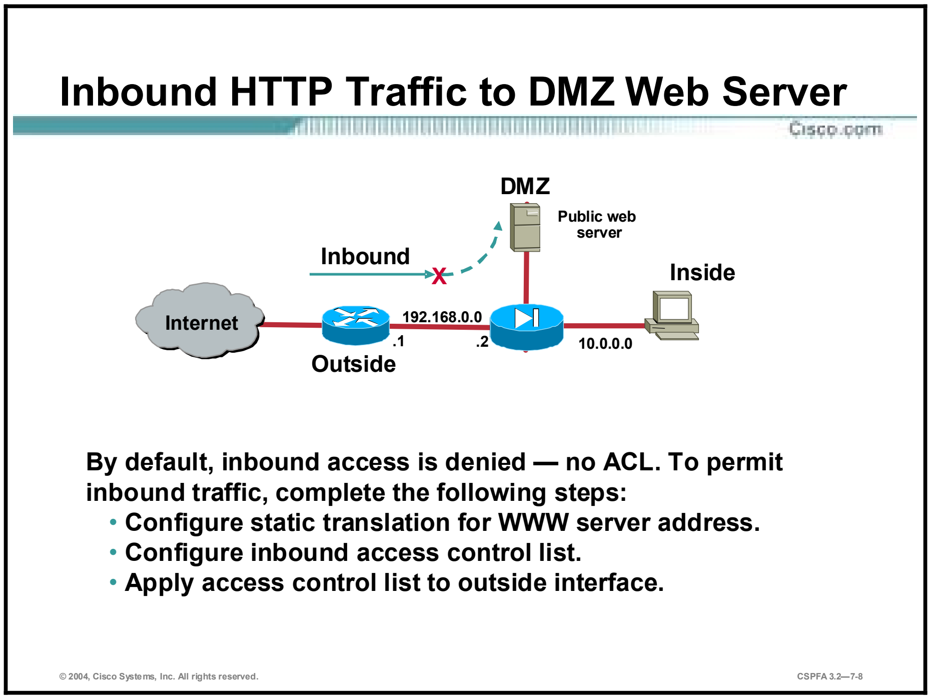
- Higher to lower security level:
  - Use an ACL to restrict outbound traffic.
  - The ACL source address is the actual (untranslated) address of the host or network.
- Lower to higher security level:
  - Use an ACL to enable inbound traffic.
  - Use an ACL to restrict inbound protocols.
  - The ACL destination address is the translated global IP address.

CSPFA 3.2—7-7

The **access-list** command is used to permit or deny traffic. The following are guidelines to use when designing and implementing ACLs:

- Higher to lower security:
  - The ACL is used to restrict outbound traffic.
  - The source address argument of the ACL command is the actual address of the host or network.
- Lower to higher:
  - The ACL is used to restrict inbound traffic.
  - The destination address argument of the ACL command is the translated global IP address.

Note ACLs are ALWAYS checked BEFORE translation is performed on the PIX Firewall.



Inbound HTTP Traffic to DMZ Web Server

By default, inbound access is denied — no ACL. To permit inbound traffic, complete the following steps:

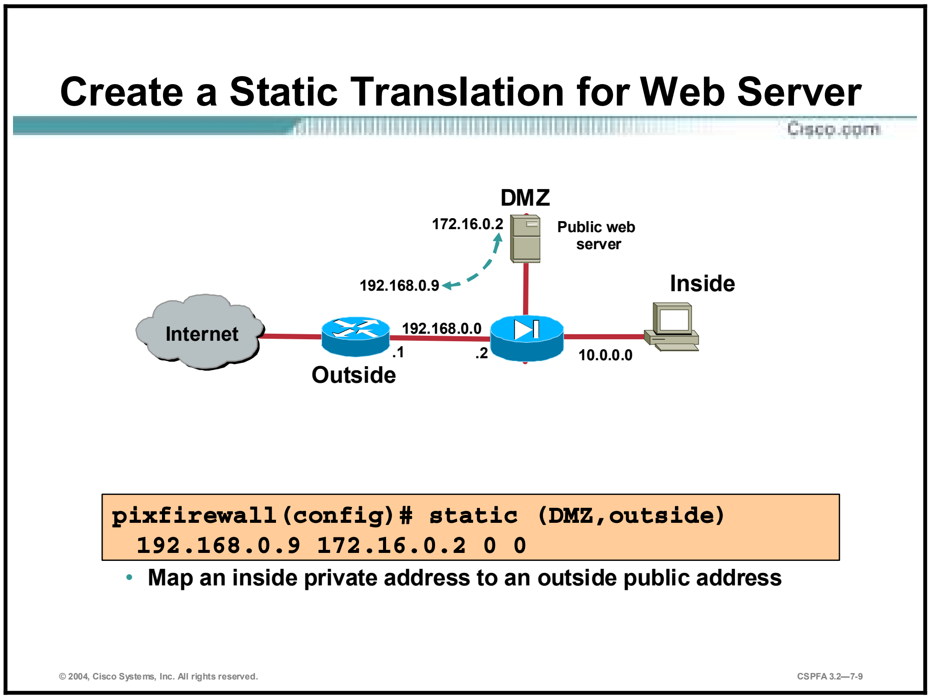
- **Configure static translation for WWW server address.**
- **Configure inbound access control list.**
- **Apply access control list to outside interface.**

CSPFA 3.2—7-8

In the figure, a network administrator wants to grant access to their company’s public web server. The web server is isolated on the PIX DMZ. By default, any inbound access to the web server from the Internet is denied. To grant access to Internet users, the administrator must complete the following steps:

- Configure a static translation for the web server address. In this way, the web server address is hidden from the Internet users.
- Configure an inbound ACL that grants access to specific hosts and protocols.
- Apply the ACL to an interface.

There is more information on each of these steps later in the lesson.



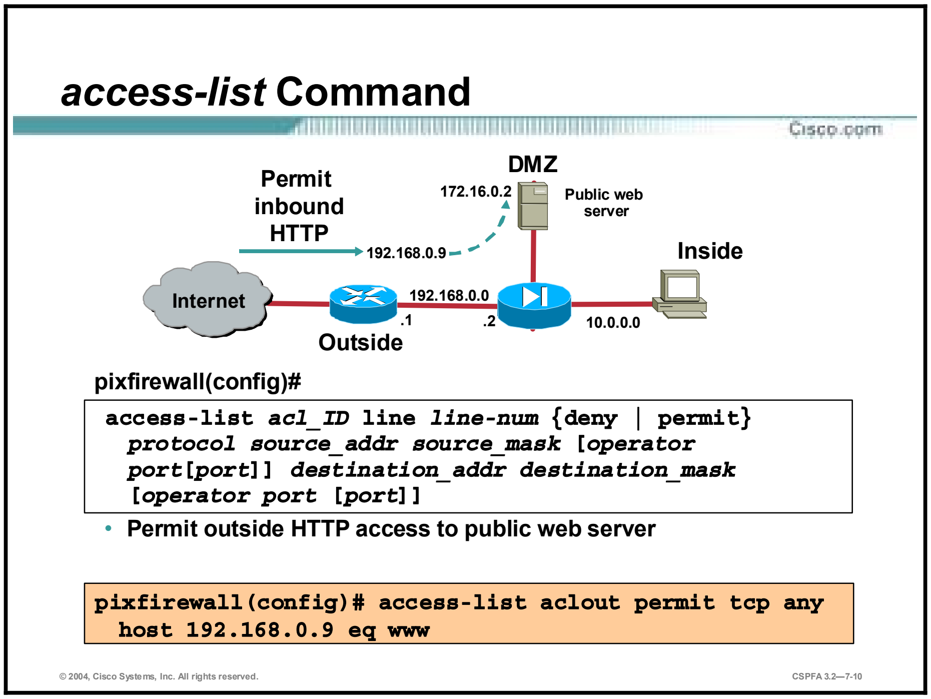
Create a Static Translation for Web Server

```
pixfirewall(config)# static (DMZ,outside)
  192.168.0.9 172.16.0.2 0 0
```

- Map an inside private address to an outside public address

CSPFA 3.2—7-9

The first step is to map the IP address of the web server to a “fixed” outside address. This will hide the true address of the web server. Internet hosts access the DMZ web server via the outside IP address. The PIX will perform the necessary translations to send the packet from the outside interface to the DMZ interface. To accomplish this, a static command is used. IP address 192.168.0.9 on the outside interface is mapped to 172.16.0.2, on the DMZ.



access-list Command

pixfirewall(config)#

```
access-list acl_ID line line-num {deny | permit}
  protocol source_addr source_mask [operator
  port[port]] destination_addr destination_mask
  [operator port [port]]
```

- Permit outside HTTP access to public web server

```
pixfirewall(config)# access-list aclout permit tcp any
  host 192.168.0.9 eq www
```

© 2004, Cisco Systems, Inc. All rights reserved.   CSPFA 3.2—7-10

The **access-list** command enables you to specify if an IP address is permitted or denied access to a port or protocol. By default, all access in an access list is denied. You must explicitly permit it.

When specifying the IP address of a host as a source or destination, use the host keyword instead of the network mask 255.255.255.255. For example, use the following ACL entry to permit HTTP traffic from any host to host 192.168.0.9:

```
access-list aclout permit tcp any host 192.168.0.9 eq www
```

The **show access-list** command lists the **access-list** command statements in the configuration. The **show access-list** command also lists a hit count that indicates the number of times an element has been matched during an **access-list** command search.

The **clear access-list** command removes all references to the deleted access list from the PIX Firewall configuration. If the acl_id argument is specified, it clears only the corresponding ACL. If the **counters** option is specified as well, it clears the hit count for the specified ACL.

The **no access-list** command removes an **access-list** command from the configuration. If you remove all the **access-list** command statements in an ACL group, the **no access-list** command also removes the corresponding **access-group** command from the configuration.

Note The **access-list** command uses the same syntax as the Cisco IOS software **access-list** command except that the subnet mask in the PIX Firewall **access-list** command is reversed from the Cisco IOS software version of this command. For example, a wildcard mask specified as 0.0.0.255 in the Cisco IOS **access-list** command would be specified as a subnet mask of 255.255.255.0 in the PIX Firewall **access-list** command.

The syntax for the **access-list** commands is as follows:

```
access-list acl_ID deny | permit protocol source_addr source_mask [operator port [port]] destination_addr destination_mask operator port [port]
```

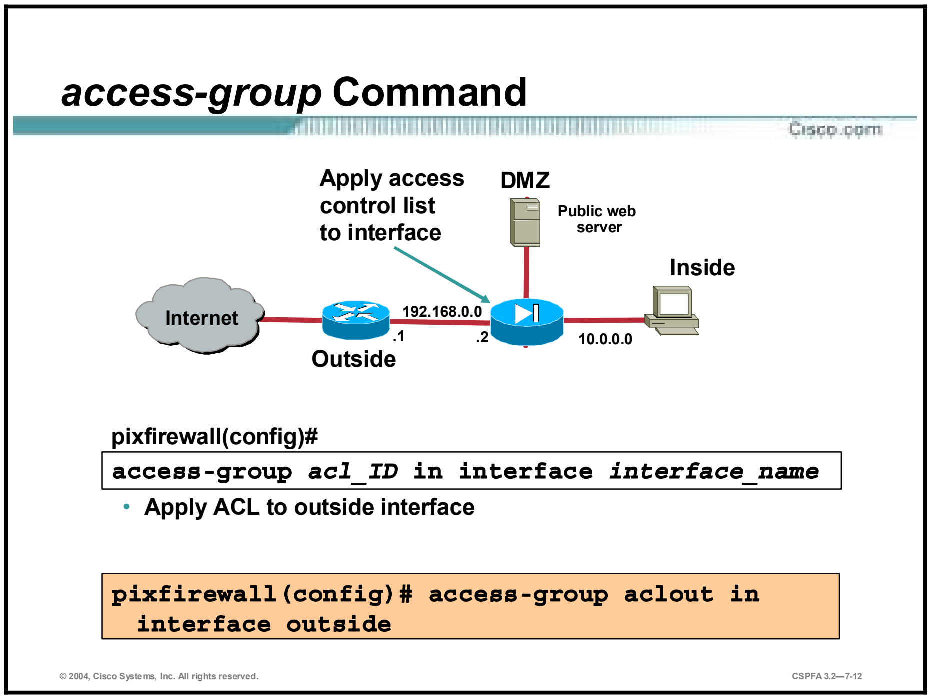
```
access-list acl_ID deny | permit icmp source_addr source_mask destination_addr destination_mask [icmp_type]
```

```
show access-list
```

```
clear access-list [acl_ID] [counters]
```

acl_ID	Name of an ACL. You can use either a name or number.
deny	Does not allow a packet to travel through the PIX Firewall. By default, the PIX Firewall denies all inbound packets unless you specifically permit access.
permit	Selects a packet to travel through the PIX Firewall.
protocol	Name or number of an IP protocol. It can be one of the keywords icmp, ip, tcp, or udp, or an integer in the range 1 to 254 representing an IP protocol number. To match any Internet protocol, including ICMP, TCP, and UDP, use the keyword ip.
source_addr	Address of the network or host from which the packet is being sent.
source_mask	Netmask bits (mask) to be applied to source_addr, if the source address is for a network mask.
operator	A comparison operand that lets you specify a port or a port range. Use without an operator and port to indicate all ports. Valid operand keywords are lt, gt, eq, neq, and range. Use range and a port range to permit or deny access to only those ports named in the range. For example, use range 10 1024 to permit or deny access only to ports 10 through 1024.
port	Services you permit or to which you deny access. Specify services by the port that handles it, such as smtp for port 25, www for port 80, and so on. You can specify ports by either a literal name or a number in the range of 0 to 65535.
destination_addr	IP address of the network or host to which the packet is being sent.
destination_mask	Netmask bits (mask) to be applied to destination_addr, if the destination address is a network mask.
icmp_type	Permit or deny access to ICMP message types. Omit this option to mean all ICMP types. ICMP message types are not supported for use with IPSec. When the **access-list** command is used in conjunction with the **crypto map** command, the icmp_type is ignored.

Note For inbound connections, destination_addr is the global address. For outbound connections, source_addr is the address before NAT has been performed.



access-group Command

pixfirewall(config)#

```
access-group acl_ID in interface interface_name
```

- Apply ACL to outside interface

```
pixfirewall(config)# access-group aclout in
  interface outside
```

CSPFA 3.2—7-12

The **access-group** command binds an ACL to an interface. The ACL is applied to traffic inbound to an interface. Only one ACL can be bound to an interface using the **access-group** command. In the figure, the outside ACL is bound to the outside interface.

The **no access-group** command unbinds the acl_ID from the interface interface_name.

The **show access-group** command displays the current ACL bound to the interfaces.

The **clear access-group** command removes all entries from an ACL indexed by acl_ID. If acl_ID is not specified, all **access-group** command statements are removed from the configuration.

The syntax for the **access-group** commands is as follows:

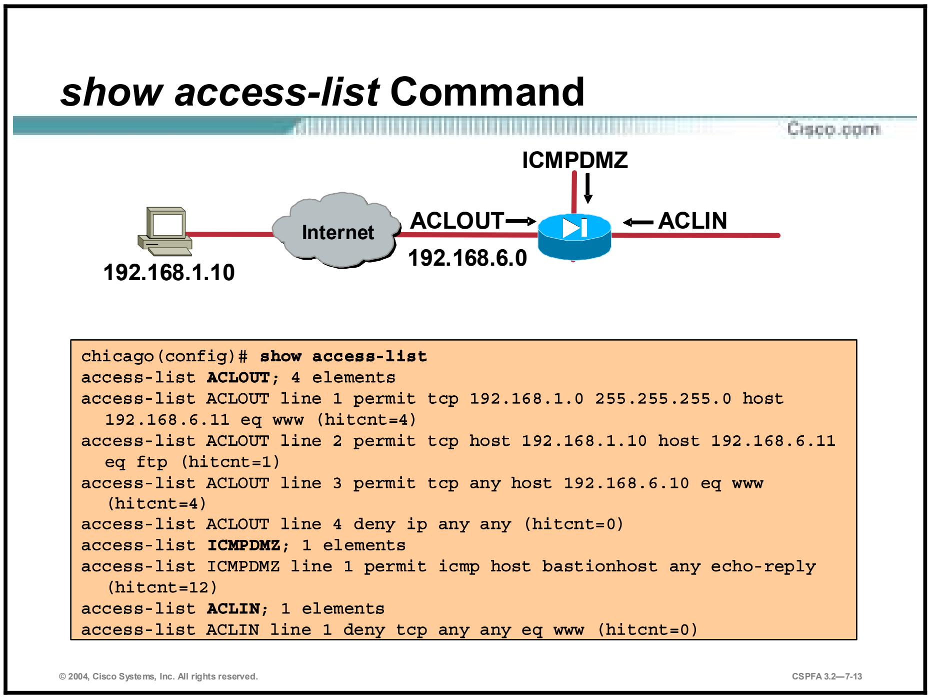
access-group ***acl_ID*** **in interface** ***interface_name***

no access-group ***acl_ID*** **in interface** ***interface_name***

show access-group ***acl_ID*** **in interface** ***interface_name***

clear access-group

acl_ID	The name associated with a given ACL.
in interface	Filters inbound packets at the given interface.
interface_name	The name of the network interface.

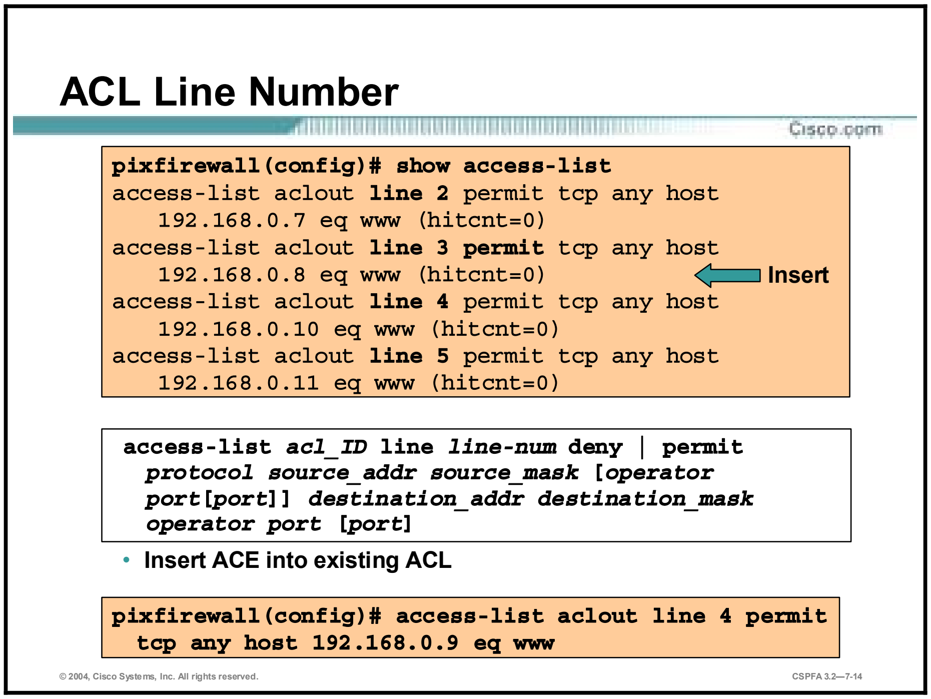


show access-list Command

```
chicago(config)# show access-list
access-list ACLOUT; 4 elements
access-list ACLOUT line 1 permit tcp 192.168.1.0 255.255.255.0 host
  192.168.6.11 eq www (hitcnt=4)
access-list ACLOUT line 2 permit tcp host 192.168.1.10 host 192.168.6.11
  eq ftp (hitcnt=1)
access-list ACLOUT line 3 permit tcp any host 192.168.6.10 eq www
  (hitcnt=4)
access-list ACLOUT line 4 deny ip any any (hitcnt=0)
access-list ICMPDMZ; 1 elements
access-list ICMPDMZ line 1 permit icmp host bastionhost any echo-reply
  (hitcnt=12)
access-list ACLIN; 1 elements
access-list ACLIN line 1 deny tcp any any eq www (hitcnt=0)
```

CSPFA 3.2—7-13

The **show access-list** command lists all the configured ACLs and the access control entries (ACEs). In the figure, there are three ACLs, ACLOUT, ICMPDMZ, and ACLIN. Within each ACL, there are one or more ACEs. Each ACE is denoted by a line number. In the figure, ACLout has six ACEs. The ACEs are numbered from line 1 to line 6.



ACL Line Number

```
pixfirewall(config)# show access-list
access-list aclout line 2 permit tcp any host
   192.168.0.7 eq www (hitcnt=0)
access-list aclout line 3 permit tcp any host
   192.168.0.8 eq www (hitcnt=0)
access-list aclout line 4 permit tcp any host
   192.168.0.10 eq www (hitcnt=0)
access-list aclout line 5 permit tcp any host
   192.168.0.11 eq www (hitcnt=0)
```

Insert

```
access-list acl_ID line line-num deny | permit
  protocol source_addr source_mask [operator
  port[port]] destination_addr destination_mask
  operator port [port]
```

- Insert ACE into existing ACL

```
pixfirewall(config)# access-list aclout line 4 permit
  tcp any host 192.168.0.9 eq www
```

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—7-14

To view the configured ACLs, use the **show access-list** command. Notice that the **access-list** commands are listed by ACL line number. The line number was not part of the original command line, but was added by the operating system. Individual ACEs are given a single line number. All ACEs pertaining to an object group are given the same line number. Object groups are covered later in this course.

Line numbers give the administrator the ability to insert, or delete, ACEs within a list of existing ACEs. Use the **access-list** id line line-num command to insert an **access-list** command statement, and the **no access-list** id line line-num command to delete an **access-list** command statement. Each ACE and remark has an associated line number. Line numbers can be used to insert or delete elements at any position in an access list. These numbers are maintained internally in increasing order, starting from 1 (for example, in a sequence such as 1, 2, 3 …). A user can insert a new entry between two consecutive ACEs by choosing the line number of the ACE with the higher line number. The line numbers are always maintained in increasing order, with an individual line number for each ACE. However, all ACEs resulting from a single object group **access-list** command statement have a single line number. Consequently, you cannot insert an ACE in the middle of object group ACEs. Line numbers are displayed by the **show access-list** command. However, they are not shown in your configuration.

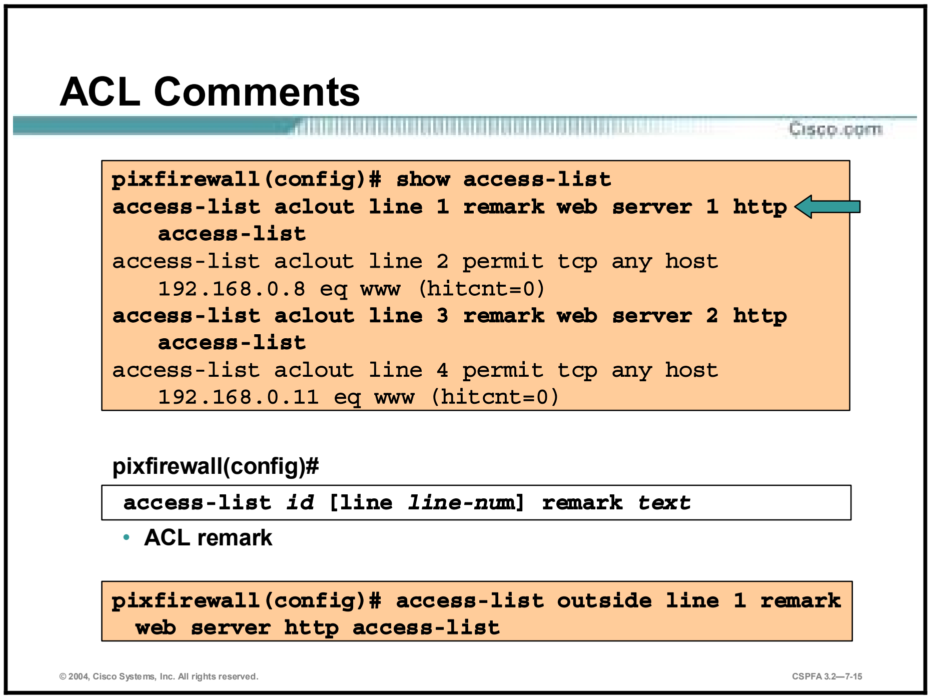
In the figure, the administrator adds an ACE to the existing ACL. Entering “line 4” in the **access-list** command line inserts this command into the fourth position in the ACL. This forces the existing line 4 ACE down one position in the ACL. The line 4 **access-list** command line becomes the new number 4 ACE. The current number 4 ACE becomes the new number 5 ACE.

The syntax for the **access-list line number** commands is as follows:

access-list line number

access-list *id* [line *line-num*]

id	Name of an ACL. You can use either a name or number.
line *line-num*	The line number at which to insert a remark or an ACE.



ACL Comments

```
pixfirewall(config)# show access-list
access-list aclout line 1 remark web server 1 http
    access-list
access-list aclout line 2 permit tcp any host
    192.168.0.8 eq www (hitcnt=0)
access-list aclout line 3 remark web server 2 http
    access-list
access-list aclout line 4 permit tcp any host
    192.168.0.11 eq www (hitcnt=0)
```

pixfirewall(config)#

```
access-list id [line line-num] remark text
```

- ACL remark

```
pixfirewall(config)# access-list outside line 1 remark
  web server http access-list
```

CSPFA 3.2—7-15

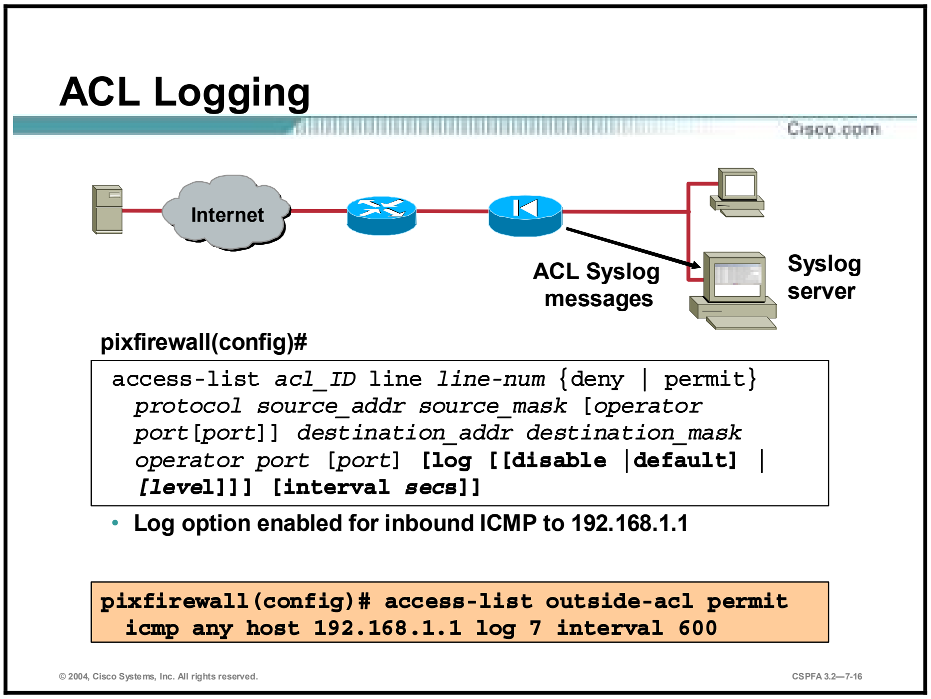
The **access-list remark** command enables users to include comments (remarks) about entries in any ACL. You can use remarks to make the ACL easier to scan and interpret. Each remark line is limited to 100 characters. The ACL remark can be placed before or after an **access-list** command statement, but it should be placed in a consistent position so that it is clear which remark describes which **access-list** command. For example, it would be confusing to have some remarks before the associated **access-list** commands and some remarks after the associated **access-list** commands. In the figure, the administrator added remarks pertaining to line 3 and 5 ACE. The new remarks are line 2 and 4 ACE. To add a remark above an existing access-list entry, type in the access-list remark using the existing ACE line number. The remark forces the existing ACE down. To view existing access-list entries, use the **show access-list** command.

The syntax for the **access-list remark** commands is as follows:

access-list remark

access-list *id* [**line** *line-num*] **remark** *text*

id	Name of an ACL. You can use either a name or number.
line *line-num*	The line number at which to insert a remark or an ACE.
remark *text*	The text of the remark to add before or after an **access-list** command statement, up to 100 characters in length.



ACL Logging

Cisco.com

pixfirewall(config)#

```
access-list acl_ID line line-num {deny | permit}
  protocol source_addr source_mask [operator
  port[port]] destination_addr destination_mask
  operator port [port] [log [[disable |default] |
  [level]]] [interval secs]]
```

- Log option enabled for inbound ICMP to 192.168.1.1

```
pixfirewall(config)# access-list outside-acl permit
  icmp any host 192.168.1.1 log 7 interval 600
```

CSPFA 3.2—7-16

When you specify the **log** option, it generates Syslog message 106100 for the ACE to which it is applied. (Syslog message 106100 is generated for every matching permit or deny ACE flow passing through the firewall.) The first-match flow is cached. Subsequent matches increment the hit count displayed in the **show access-list** command (hitcnt) for the ACE, and new 106100 messages will be generated at the end of the interval defined by the **interval *secs*** argument if the hit count for the flow is not zero.

The following example illustrates the use of ACL-based logging in an Internet Control Message Protocol (ICMP) context:

1. An inbound ICMP echo request to 192.168.1.1 arrives on the outside interface.

2. An ACL called **outside-acl** is applied for the access check.

3. The packet is permitted by the first ACE of **outside-acl,** which has the **log** option enabled.

4. The log flow (ICMP, 192.168.10.12, 0, 192.168.1.1, 8) has not been cached, so the following Syslog message is generated and the log flow is cached:

```
106100: access-list outside-acl permitted icmp
outside/192.168.10.12(0) -> inside/192.168.1.1(8) hit-cnt 1 (first
hit)
```

5. Twenty such packets arrive on the outside interface within the next 10 minutes (600 seconds). Because the log flow has been cached, the log flow is located and the hit count of the log flow is incremented for each packet.

6. At the end of 10th minute, the following Syslog message is generated and the hit count of the log flow is reset to 0:

```
106100: access-list outside-acl permitted icmp
outside/192.168.10.12(0) -> inside/192.168.1.1(8) hit-cnt 20 (600-
second interval)
```

7. No such packets arrive on the outside interface within the next 10 minutes. The hit count of the log flow remains 0.

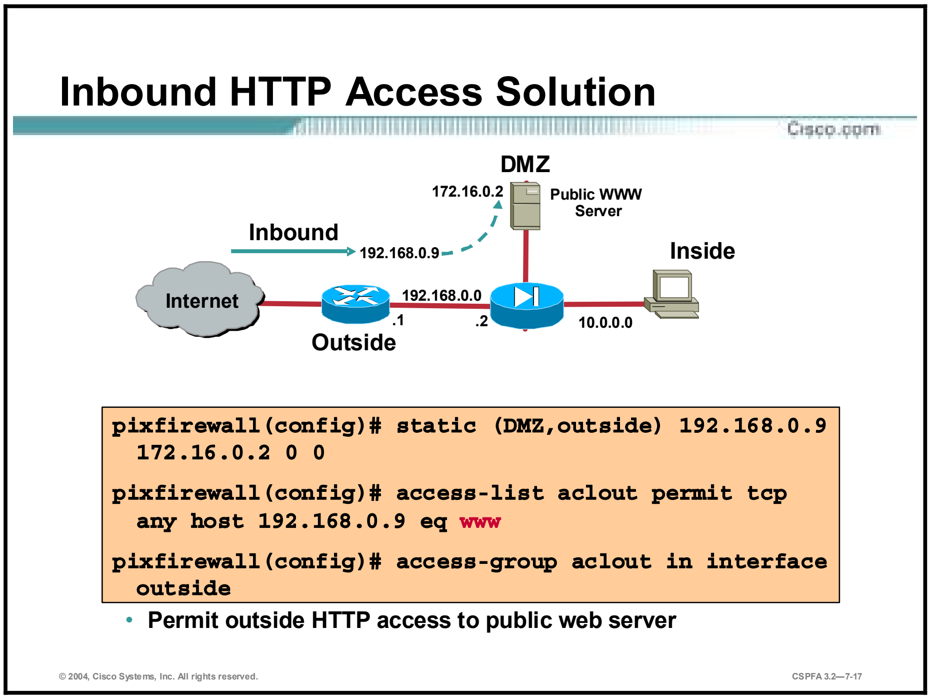
8. At the end of 20th minute, the cached flow (ICMP, 192.168.10.12, 0, 192.168.1.1, 8) is deleted because of the 0 hit count.

The syntax for the **access-list log** commands is as follows:

access-list log

access-list ***id*** **[log [[disable |default] |** ***[level*****]]] [interval** ***secs*****]]**

id	Name of an ACL. You can use either a name or number.
log	When the **log** option is specified, it generates Syslog message 106100 for the access list element (ACE) to which it is applied. (Syslog message 106100 is generated for every matching permit or deny ACE flow passing through the firewall.) The first-match flow is cached. Subsequent matches increment the hit count displayed in the **show access-list** command, **hitcnt**, for the ACE, and new 106100 messages will be generated at the end of the interval defined by **interval** *secs* if the hit count for the flow is not zero.
log disable	If the **log disable** option is specified, ACL logging is completely disabled. No Syslog message, including message 106023, will be generated.
log default	The **log default** option restores the default ACL logging behavior.
interval ***secs***	The time interval in seconds, from 1 to 600, at which to generate a 106100 Syslog message. The ***secs*** value is also used as the timeout value for deleting an inactive flow.



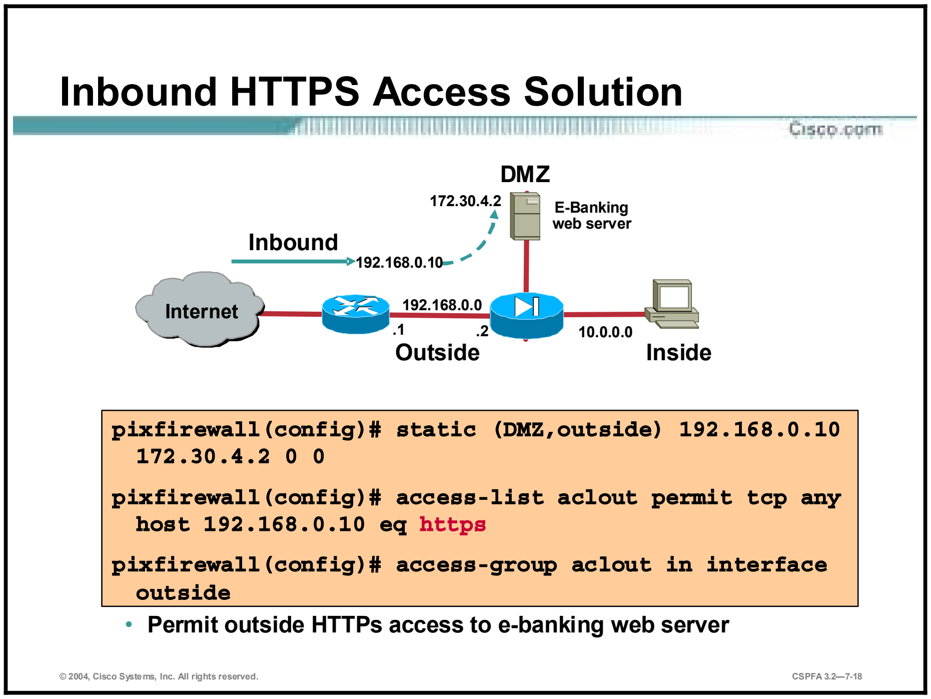
Inbound HTTP Access Solution

```
pixfirewall(config)# static (DMZ,outside) 192.168.0.9
  172.16.0.2 0 0

pixfirewall(config)# access-list aclout permit tcp
  any host 192.168.0.9 eq www

pixfirewall(config)# access-group aclout in interface
  outside
```

- **Permit outside HTTP access to public web server**

CSPFA 3.2—7-17

In the figure is a solution for HTTP access from the Internet to the public web server on the DMZ for HTTP traffic. The static was configured to translate the web server IP address to an outside IP address. An ACL was configured to enable Internet hosts to connect to the web server with HTTP traffic.



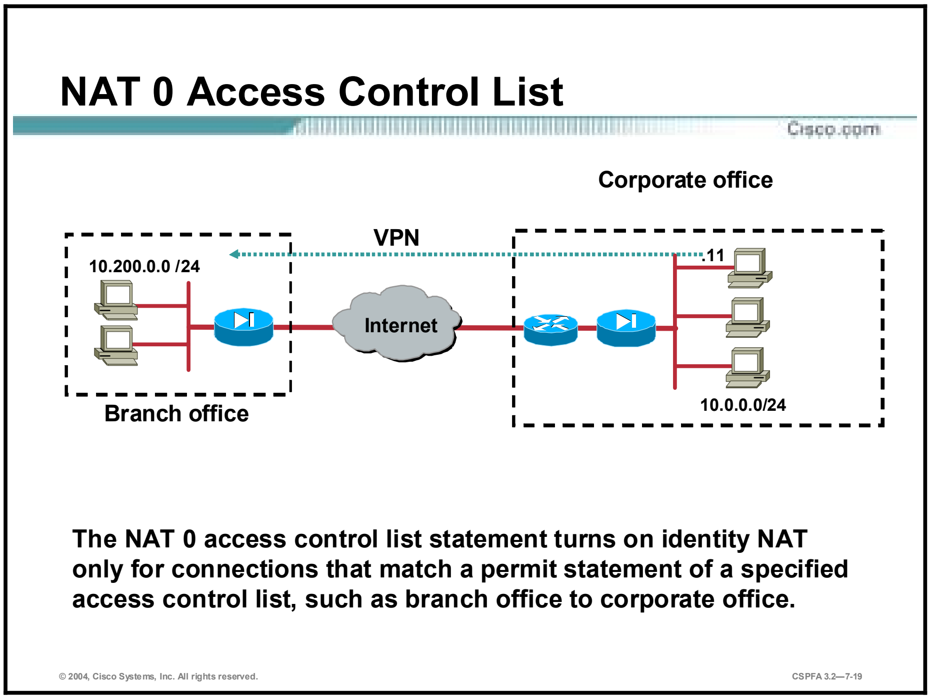
Inbound HTTPS Access Solution

```
pixfirewall(config)# static (DMZ,outside) 192.168.0.10
  172.30.4.2 0 0
pixfirewall(config)# access-list aclout permit tcp any
  host 192.168.0.10 eq https
pixfirewall(config)# access-group aclout in interface
  outside
```

- Permit outside HTTPs access to e-banking web server

CSPFA 3.2—7-18

In the figure, the company as an E-Banking solution located on the DMZ. The administrator wants to establish secure communications between the Internet users and the E-Banking web server. To establish and inbound HTTPS connection, the administrator configured a static, an ACL, and applied the ACL to an interface.



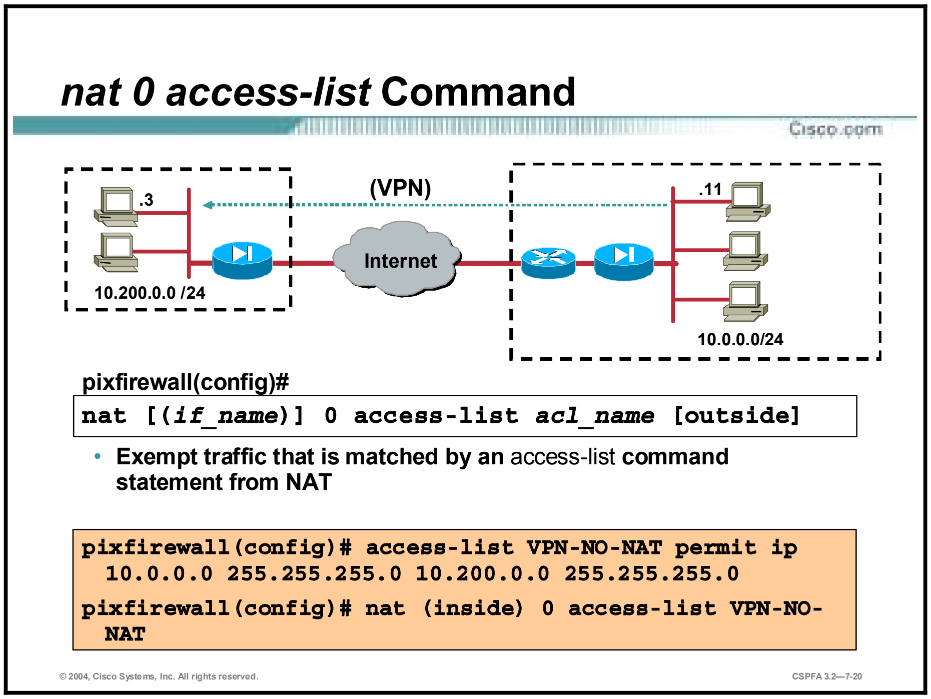
NAT 0 Access Control List

The NAT 0 access control list statement turns on identity NAT only for connections that match a permit statement of a specified access control list, such as branch office to corporate office.

CSPFA 3.2—7-19

As explained in a previous lesson, the **nat 0** command enables you to exempt a host or network from NAT. The **nat 0 access-list** command takes this a step further by enabling you to exempt from NAT any traffic that is matched by an **access-list** entry. Destination-sensitive **nat 0 access-list** is usually used in VPN scenarios.

In the figure, users in the corporate office wish to communicate with the branch site over a VPN tunnel. To accomplish this, the administrator employs **nat 0 access-list**. The IP source network, 10.0.0.0/24, and IP destination network, 10.200.0.0/24, are defined in the access-list. The access-list is applied to the **nat 0** command. Any VPN traffic originating at 10.0.0.0/24 and destined for 10.200.0.0/24 is not translated by the PIX. **Nat 0 access-list** supports both inbound and outbound connections with no restrictions.



nat 0 access-list Command

pixfirewall(config)#

```
nat [(if_name)] 0 access-list acl_name [outside]
```

- **Exempt traffic that is matched by an** access-list **command statement from NAT**

```
pixfirewall(config)# access-list VPN-NO-NAT permit ip
  10.0.0.0 255.255.255.0 10.200.0.0 255.255.255.0
pixfirewall(config)# nat (inside) 0 access-list VPN-NO-
  NAT
```

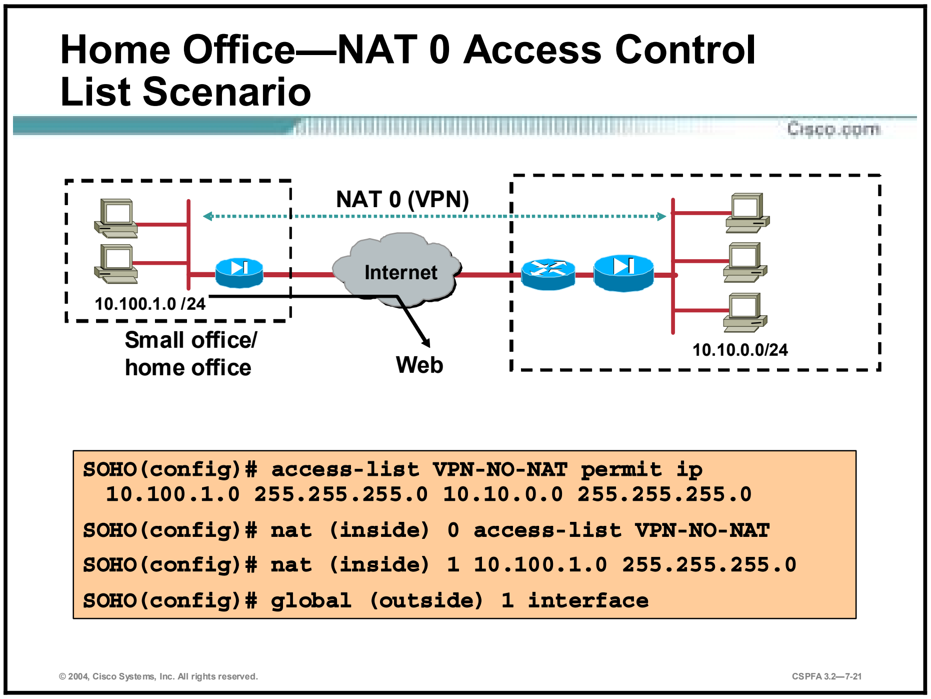
CSPFA 3.2—7-20

The **nat 0 access-list** command enables you to exempt from NAT any traffic that is matched by an **access-list** entry. The example in the figure shows use of the **nat 0 access-list** command to permit internal host 10.0.0.11 to bypass NAT when connecting to outside host 10.200.0.3.

The syntax of the **nat 0 access-list** command is as follows:

nat [(*if_name*)] 0 access-list *acl_name* [outside]

if_ name	The internal network interface name. If the interface is associated with an ACL, the if_name is the higher security level interface name.
access-list	Associates an **access-list** command with the **nat 0** command.
acl_name	The name you use to identify the **access-list** command statement.
outside	Specifies that the **nat** command apply to the outside interface address.



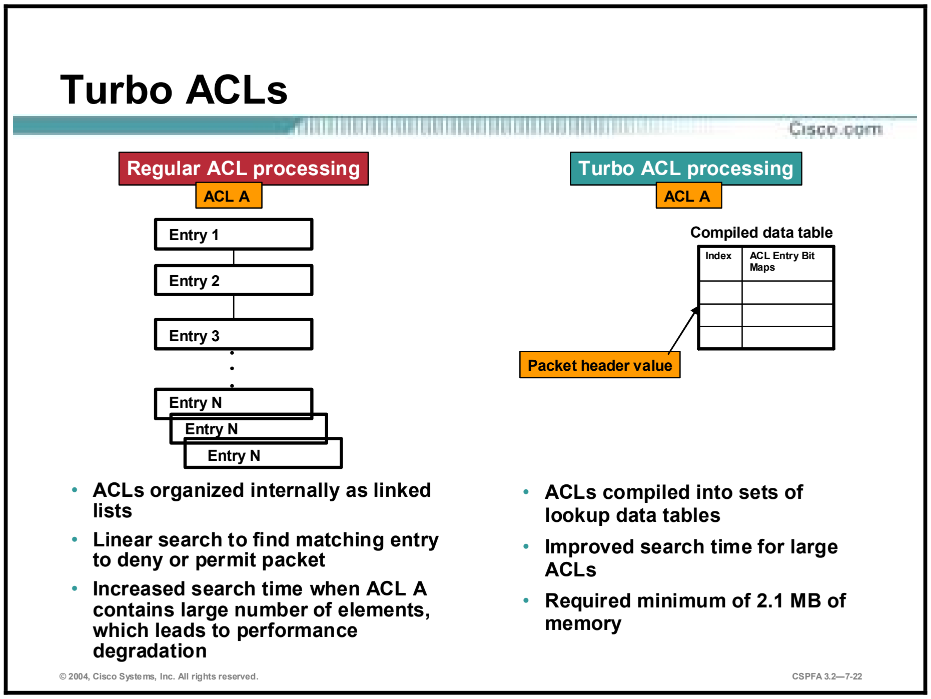
Home Office—NAT 0 Access Control List Scenario

```
SOHO(config)# access-list VPN-NO-NAT permit ip
  10.100.1.0 255.255.255.0 10.10.0.0 255.255.255.0
SOHO(config)# nat (inside) 0 access-list VPN-NO-NAT
SOHO(config)# nat (inside) 1 10.100.1.0 255.255.255.0
SOHO(config)# global (outside) 1 interface
```

CSPFA 3.2—7-21

In the figure, the home office/small office worker wants to access the corporate network via VPN without local translation, and the Internet with a translated address. To access the corporate network, **nat 0 access-list** is configured. The access-list, VPN-NO-NAT, defines both the source network of the traffic, 10.100.1.0, and the destination network, 10.10.0.0. Any traffic that matches the access-list statement is not translated. Corporate traffic is not translated by the PIX.

The second scenario is to translate any traffic bound for the Internet. The **nat** (inside) 1 statement defines the source network, 10.100.1.0. The global address is based on the IP address of the outside interface. The **nat 0** command takes precedence over the **nat** (inside) 1 command. Any packets that match the ACL are transmitted without translation. Any 10.100.1.0 network packets that do not match the VPN-NO-NAT access list are translated by the PIX.



Turbo ACLs

Regular ACL processing

ACL A

Entry 1

Entry 2

Entry 3

Entry N

Entry N

Entry N

- ACLs organized internally as linked lists
- Linear search to find matching entry to deny or permit packet
- Increased search time when ACL A contains large number of elements, which leads to performance degradation

Turbo ACL processing

ACL A

Compiled data table

Index	ACL Entry Bit Maps

Packet header value

- ACLs compiled into sets of lookup data tables
- Improved search time for large ACLs
- Required minimum of 2.1 MB of memory

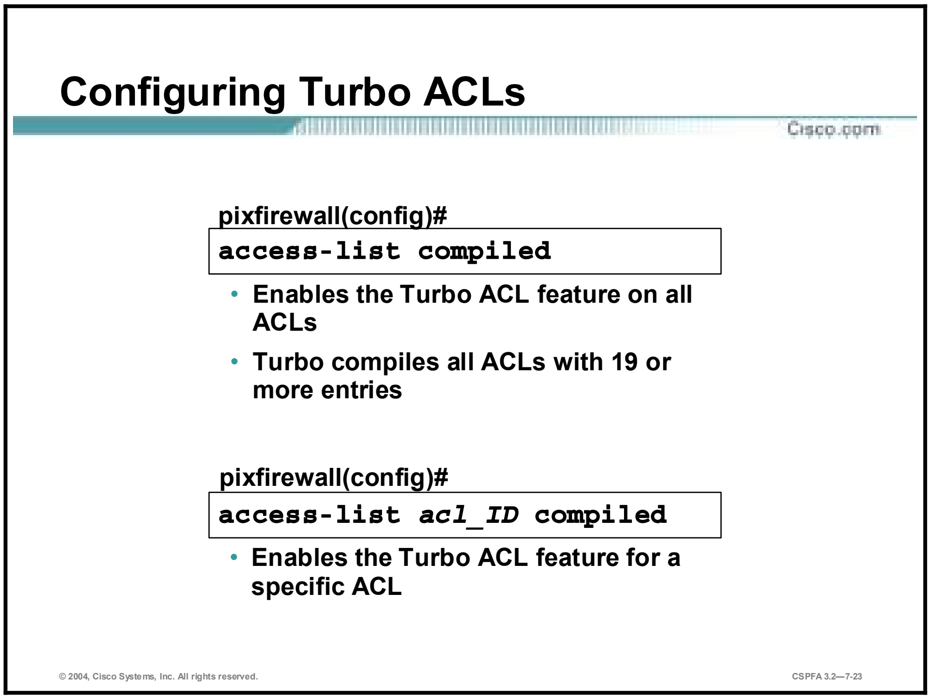
CSPFA 3.2—7-22

An ACL typically consists of multiple ACL entries, organized internally by the PIX Firewall as a linked list. When a packet is compared to the access list control, the PIX Firewall searches this linked list linearly to find a matching element. The matching element is then examined to determine if the packet is to be transmitted or dropped. With a linear search, the average search time increases proportionally to the size of the ACL.

Turbo ACLs improve the average search time for ACLs containing a large number of entries by causing the PIX Firewall to compile tables for ACLs. You can enable this feature globally and then disable it for specific ACLs, or you can enable it only for specific ACLs. For short ACLs, the Turbo ACL feature does not improve performance. A Turbo ACL search of an ACL of any length requires about the same amount of time as a regular search of an ACL consisting of approximately twelve to eighteen entries. For this reason, even when enabled, the Turbo ACL feature is only applied to ACLs with nineteen or more entries.

The Turbo ACL feature requires significant amounts of memory and is most appropriate for high-end PIX Firewall models, such as the PIX Firewall 525 or 535. The minimum memory required for Turbo ACL support is 2.1 MB, and approximately 1 MB of memory is required for every 2,000 ACL elements. The actual amount of memory required depends not only on the number of ACL elements but also on the complexity of the entries. Furthermore, when you add or delete an element from a turbo-enabled ACL, the internal data tables associated with the ACL are regenerated. This produces an appreciable load on the PIX Firewall CPU.

Note Turbo ACLs are not supported in the PIX Firewall 501 model.



Configuring Turbo ACLs

pixfirewall(config)#

```
access-list compiled
```

- Enables the Turbo ACL feature on all ACLs
- Turbo compiles all ACLs with 19 or more entries

pixfirewall(config)#

```
access-list acl_ID compiled
```

- Enables the Turbo ACL feature for a specific ACL

CSPFA 3.2—7-23

Turbo ACLs can be configured globally or on a per-ACL basis. Use the **access-list compiled** command to configure Turbo ACLs on all ACLs having 19 or more entries. This command causes the Turbo ACL process to scan through all existing ACLs. During the scanning, it marks every ACL to be turbo-configured, and compiles any ACL that has nineteen or more access control entries and has not yet been compiled.

You can enable the Turbo ACL feature for individual ACLs with the **access-list *acl_ID* compiled** command. You can also use the no form of this command after you globally configure Turbo ACLs to disable the Turbo ACL feature for specific ACLs.

The command **no access-list compiled**, which is the default, causes the PIX Firewall's Turbo ACL process to scan through all compiled ACLs and mark each one as non-turbo. It also deletes all existing Turbo ACL structures.

To view your Turbo ACL configuration, use the **show access-list** command. When Turbo ACLs are configured, this command displays the memory usage of each individually turbo-compiled ACL and the shared memory usage for all the turbo-compiled ACLs. If no ACLs are turbo-compiled, no turbo-statistics are displayed.

The syntax for the **access-list compiled** commands is as follows:

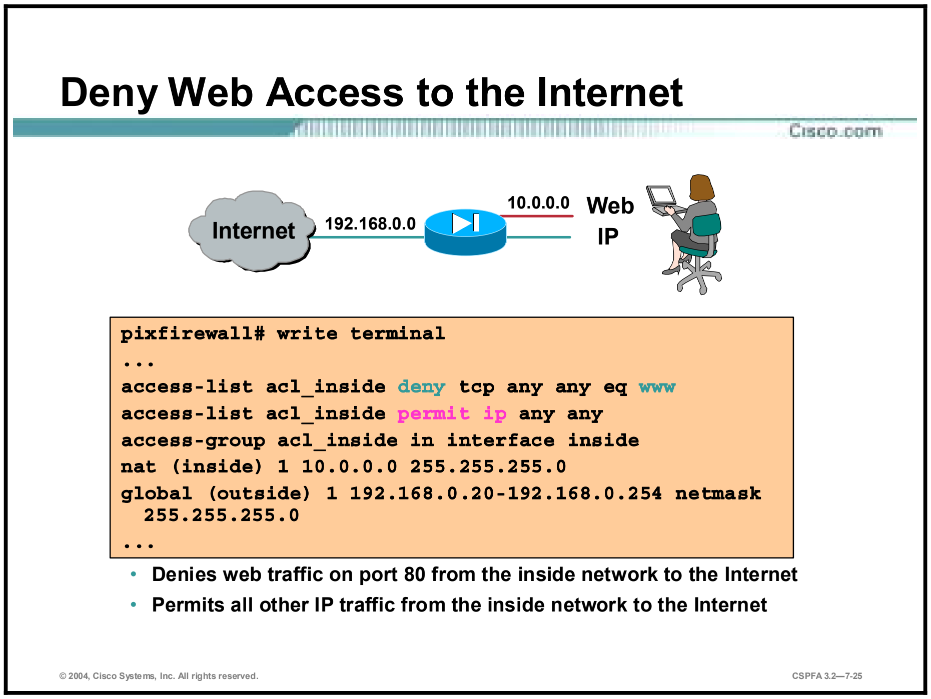
access-list compiled

access-list *acl_ID* compiled

acl_ID	The acl_ID of an existing ACL. The acl_ID is the name of an ACL, which can be a name or a number.

Using ACLs

This topic explains how to use access control lists (ACLs) in a variety of scenarios.



Deny Web Access to the Internet

Cisco.com

Internet
192.168.0.0
10.0.0.0
Web
IP

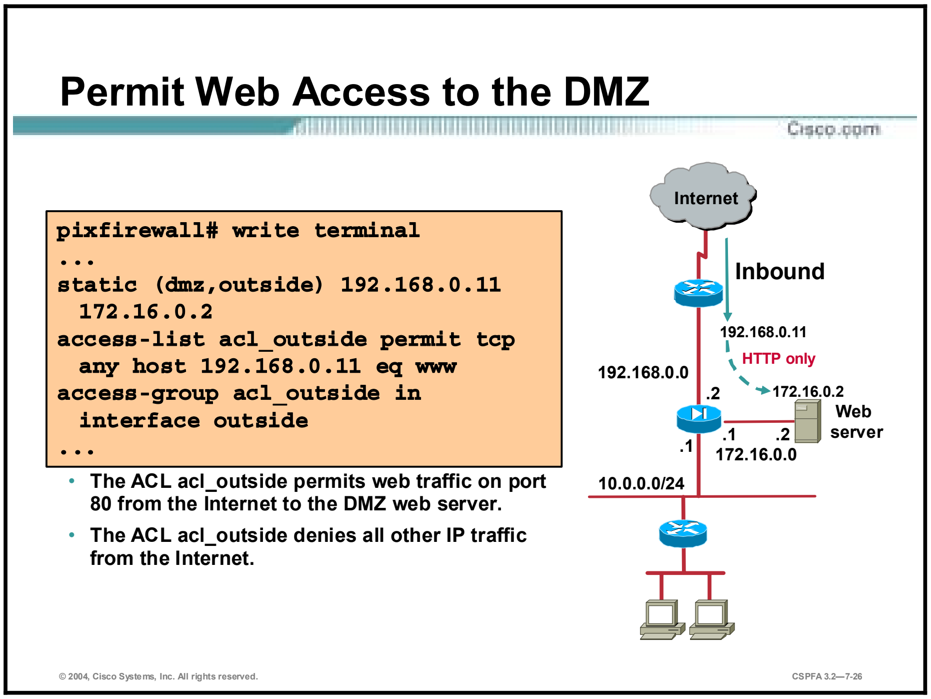
```
pixfirewall# write terminal
...
access-list acl_inside deny tcp any any eq www
access-list acl_inside permit ip any any
access-group acl_inside in interface inside
nat (inside) 1 10.0.0.0 255.255.255.0
global (outside) 1 192.168.0.20-192.168.0.254 netmask
  255.255.255.0
...
```

- Denies web traffic on port 80 from the inside network to the Internet
- Permits all other IP traffic from the inside network to the Internet

CSPFA 3.2—7-25

In the figure, the ACL acl_inside is applied to the inside interface. The ACL acl_inside denies HTTP connections from an internal network, but lets all other IP traffic through. Applying an ACL to the inside interface restricts internal users establishing outside web connections.

Note The internal network addresses (10.0.0.0) are dynamically translated (192.168.0.20–254) to allow outbound connections.



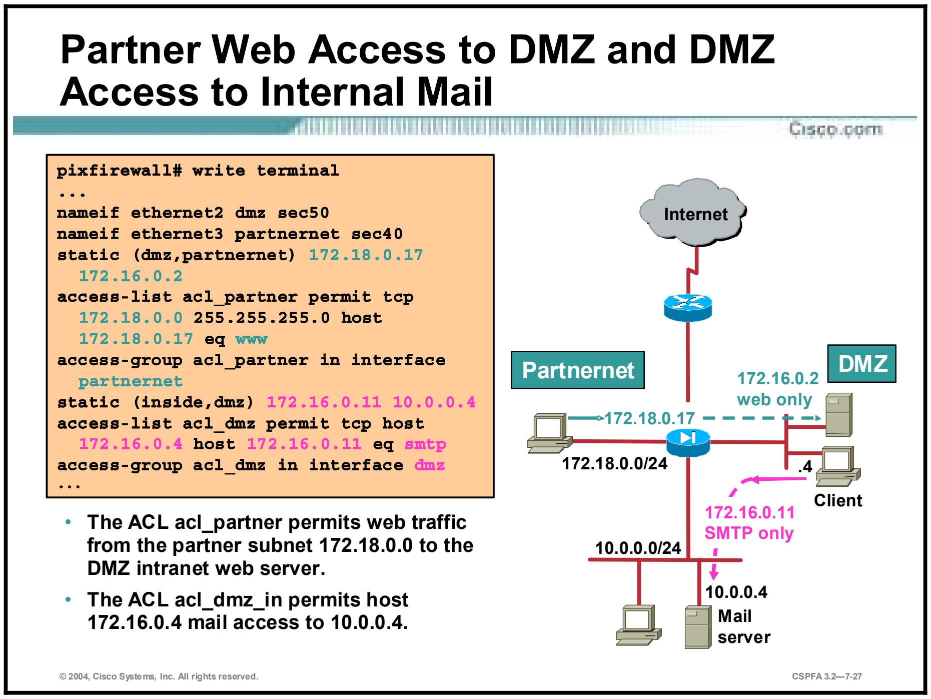
Permit Web Access to the DMZ

```
pixfirewall# write terminal
...
static (dmz,outside) 192.168.0.11
  172.16.0.2
access-list acl_outside permit tcp
  any host 192.168.0.11 eq www
access-group acl_outside in
  interface outside
...
```

- The ACL acl_outside permits web traffic on port 80 from the Internet to the DMZ web server.
- The ACL acl_outside denies all other IP traffic from the Internet.

CSPFA 3.2—7-26

In the figure, the IP address of the web server is translated to an outside IP address of 192.168.0.11. The ACL acl_outside is applied to traffic inbound to the outside interface. The ACL acl_outside permits Web connections from the Internet to a public Internet web server, 192.168.0.11. All other IP traffic is denied access to the DMZ or inside networks.



Partner Web Access to DMZ and DMZ Access to Internal Mail

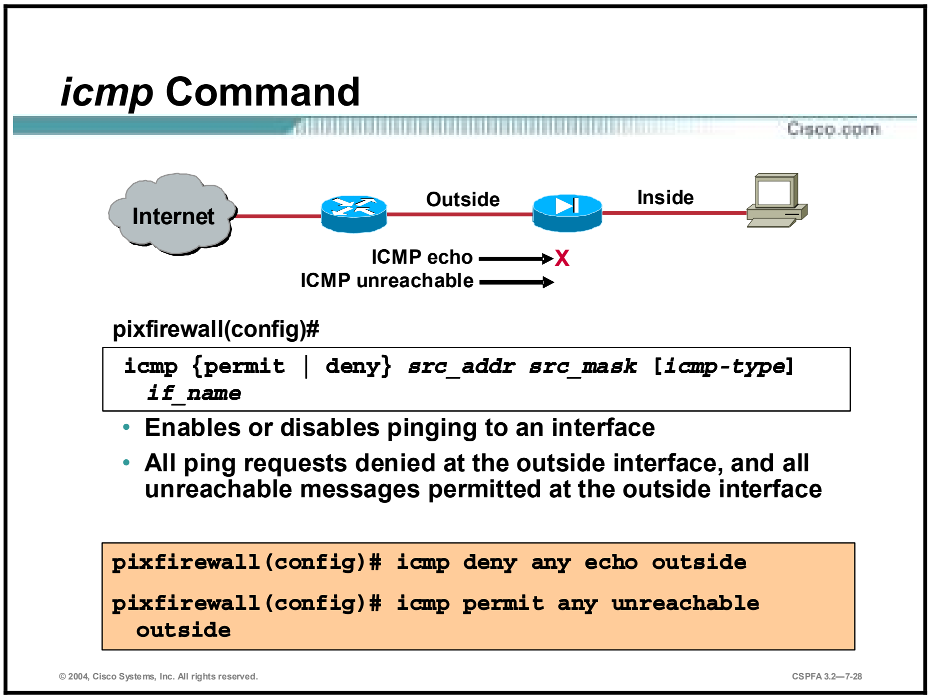
```
pixfirewall# write terminal
...
nameif ethernet2 dmz sec50
nameif ethernet3 partnernet sec40
static (dmz,partnernet) 172.18.0.17
  172.16.0.2
access-list acl_partner permit tcp
  172.18.0.0 255.255.255.0 host
  172.18.0.17 eq www
access-group acl_partner in interface
  partnernet
static (inside,dmz) 172.16.0.11 10.0.0.4
access-list acl_dmz permit tcp host
  172.16.0.4 host 172.16.0.11 eq smtp
access-group acl_dmz in interface dmz
...
```

- The ACL acl_partner permits web traffic from the partner subnet 172.18.0.0 to the DMZ intranet web server.
- The ACL acl_dmz_in permits host 172.16.0.4 mail access to 10.0.0.4.

CSPFA 3.2—7-27

In the figure, the web server is statically translated from 172.16.0.2 to 172.18.0.17. The ACL acl_partner is applied to traffic inbound to the partnernet interface. The ACL acl_partner permits Web connections from the hosts on network 172.18.0.0/24 to the DMZ web server via its statically mapped address, 172.18.0.17. All other traffic from the Partner network is denied.

In the second scenario, the client on the DMZ is trying to connect to the mail server on the inside network. The mail server IP address is statically translated to 172.16.0.11 by the PIX. The ACL acl_dmz is applied to traffic inbound to the DMZ interface. The ACL acl_dmz permits the host 172.16.0.4 mail access to the internal mail server on the inside interface via the mail server’s statically mapped address, 172.16.0.11. All other traffic originating from the DMZ network is denied.



icmp Command

pixfirewall(config)#

```
icmp {permit | deny} src_addr src_mask [icmp-type]
  if_name
```

- Enables or disables pinging to an interface
- All ping requests denied at the outside interface, and all unreachable messages permitted at the outside interface

```
pixfirewall(config)# icmp deny any echo outside
pixfirewall(config)# icmp permit any unreachable
  outside
```

CSPFA 3.2—7-28

You can enable or disable pinging to a PIX Firewall interface. With pinging disabled, the PIX Firewall cannot be detected on the network. The **icmp** command implements this feature, which is also referred to as configurable proxy pinging.

Note By default, pinging through the PIX Firewall to a PIX Firewall interface is not allowed. Pinging an interface from a host on that interface is allowed.

To use the **icmp** command, configure an **icmp** command statement that permits or denies ICMP traffic that terminates at the PIX Firewall. If the first matched entry is a permit entry, the ICMP packet continues to be processed. If the first matched entry is a deny entry or an entry is not matched, the PIX Firewall discards the ICMP packet and generates the %PIX-3-313001 Syslog message. An exception is when an **icmp** command statement is not configured, in which case, permit is assumed.

Note Cisco recommends that you grant permission for the ICMP unreachable message type (type 3). Denying ICMP unreachable messages disables ICMP Path MTU discovery, which can halt IPSec and PPTP traffic. See RFC 1195 and RFC 1435 for details about Path MTU Discovery.

The **clear icmp** command removes **icmp** command statements from the configuration.

The syntax for the **icmp** commands is as follows:

icmp {permit | deny} ***src_addr src_mask*** **[*icmp-type*]** ***if_name***

clear icmp

show icmp

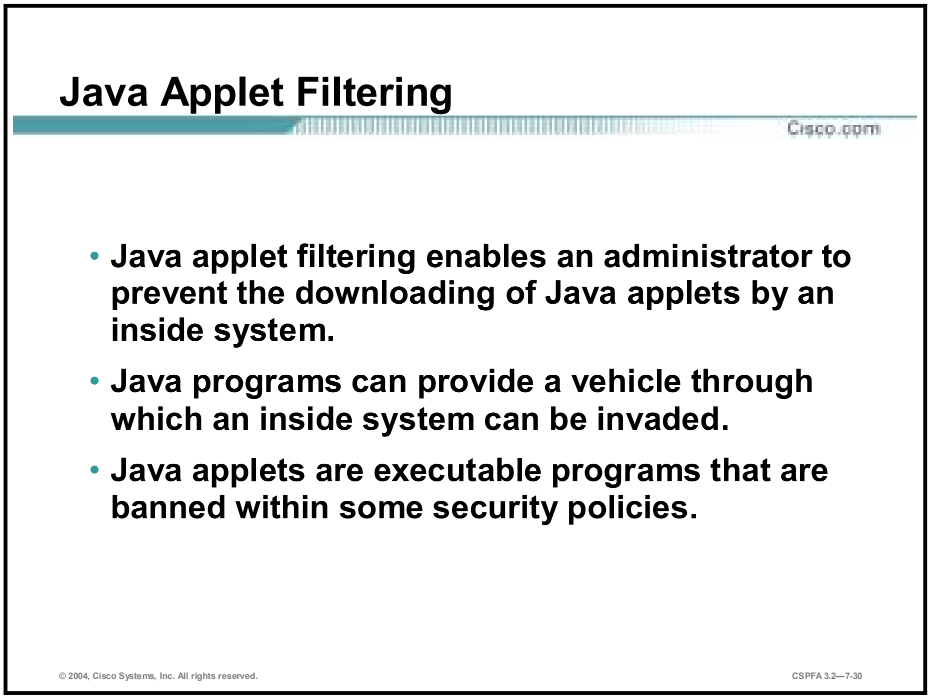
permit | deny	Permit or deny the ability to ping a PIX Firewall interface.
src_addr	Address that is either permitted or denied ability to ping an interface. Use **host** src_addr to specify a single host.
src_mask	(Optional.) Network mask. Specify if a network address is specified.
icmp-type	(Optional.) ICMP message type as described in Table ICMP Type Literals.
if_name	Interface name that can be pinged.

The following table lists the ICMP Type Literals that can be used in the type argument of the **icmp** command to designate which message types are permitted or denied:

ICMP	Type Literal
0	echo-reply
3	unreachable
4	source-quench
5	redirect
6	alternate-address
8	echo
9	router-advertisement
10	router-solicitation
11	time-exceeded
12	parameter-problem
13	timestamp-reply
14	timestamp-request
15	information-request
16	information-reply
17	mask-request
18	mask-reply
31	conversion-error
32	mobile-redirect

Malicious Active Code Filtering

The PIX Firewall can filter malicious active codes. Malicious active codes can be used in such applications as Java and ActiveX.



Java Applet Filtering

Cisco.com

- Java applet filtering enables an administrator to prevent the downloading of Java applets by an inside system.
- Java programs can provide a vehicle through which an inside system can be invaded.
- Java applets are executable programs that are banned within some security policies.

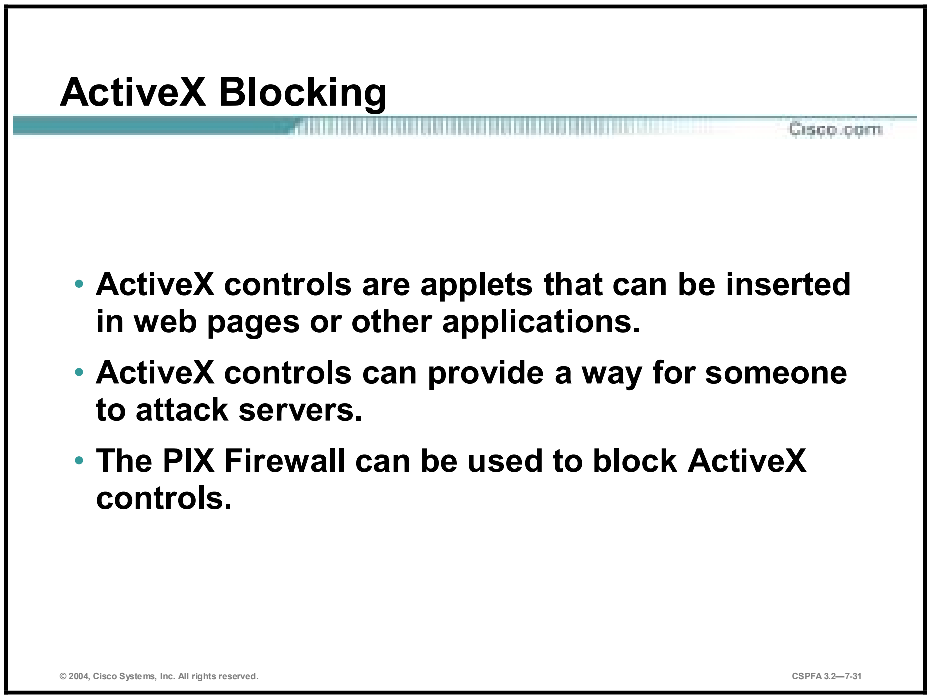
CSPFA 3.2—7-30

Java filtering enables an administrator to prevent Java applets from being downloaded by an inside system. Java applets are executable programs that are banned by many site security policies.

Java programs can provide a vehicle through which an inside system can be invaded or compromised. Java applets may be downloaded when you permit access to port 80 (HTTP), and some Java applets can contain hidden code that can destroy data on the internal network.

The PIX Firewall's Java applet filter can stop Java applications on a per-client or per-IP address basis. When Java filtering is enabled, the PIX Firewall searches for the programmed "cafe babe" string and, if found, drops the Java applet. A sample Java class code snippet looks like the following:

```
00000000: café babe 003 002d 0099 0900 8345 0098
```



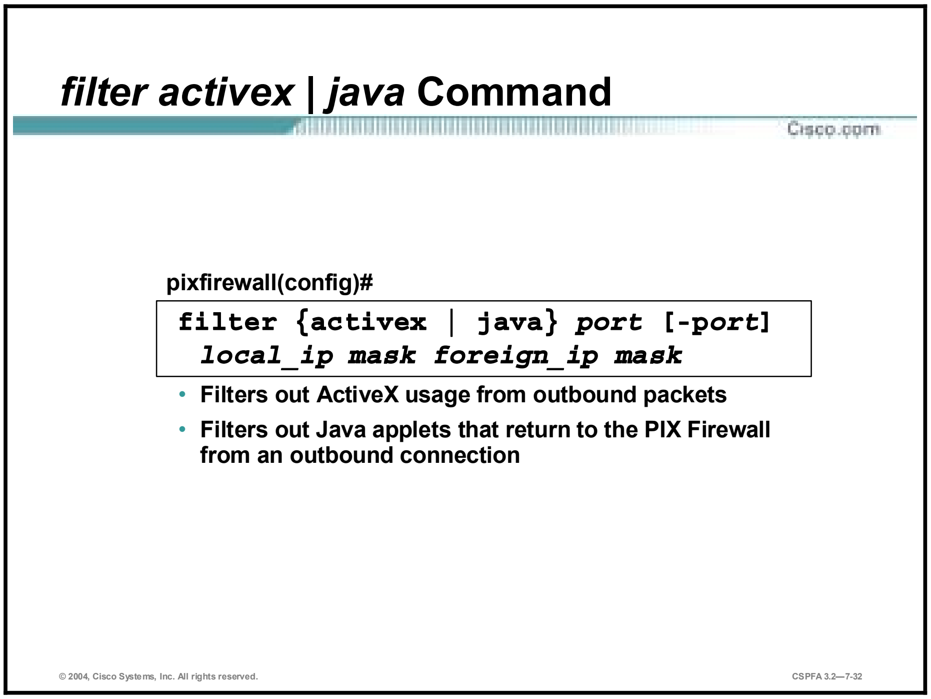
ActiveX Blocking

Cisco.com

- ActiveX controls are applets that can be inserted in web pages or other applications.
- ActiveX controls can provide a way for someone to attack servers.
- The PIX Firewall can be used to block ActiveX controls.

CSPFA 3.2—7-31

ActiveX controls, formerly known as Object Linking and Embedding (OLE) or Object Linking and Embedding control (OCX), are applets that can be inserted in web pages—often used in animations—or in other applications. ActiveX controls create a potential security problem because they can provide a way for someone to attack servers. Because of this potential security problem, you can use the PIX Firewall to block all ActiveX controls.



filter activex | *java* Command

pixfirewall(config)#

```
filter {activex | java} port [-port]
  local_ip mask foreign_ip mask
```

- Filters out ActiveX usage from outbound packets
- Filters out Java applets that return to the PIX Firewall from an outbound connection

CSPFA 3.2—7-32

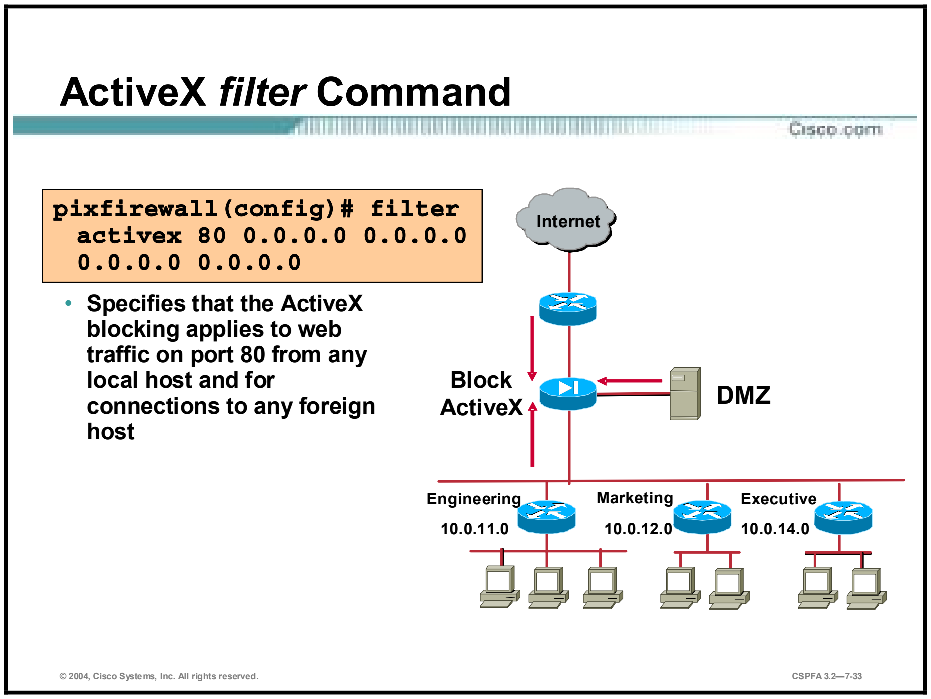
The **filter {activex | java}** command filters out ActiveX or Java usage from outbound packets.

The syntax for the **filter {activex | java}** command is as follows:

filter {activex | java} ***port*** **[-*port*]** ***local_ip mask foreign_ip mask***

activex	Block outbound ActiveX and other HTML <object> tags from outbound packets.
java	Block Java applets returning from an outbound connection.
port	The port(s) at which Internet traffic is received on the PIX Firewall.
local_ip	The IP address of the host from which access is sought.
mask	Subnet mask.
foreign_ip	The IP address of the host to which access is sought.

Note ActiveX blocking does not occur when users access an IP address referenced by the **alias** command.



ActiveX *filter* Command

```
pixfirewall(config)# filter
  activex 80 0.0.0.0 0.0.0.0
  0.0.0.0 0.0.0.0
```

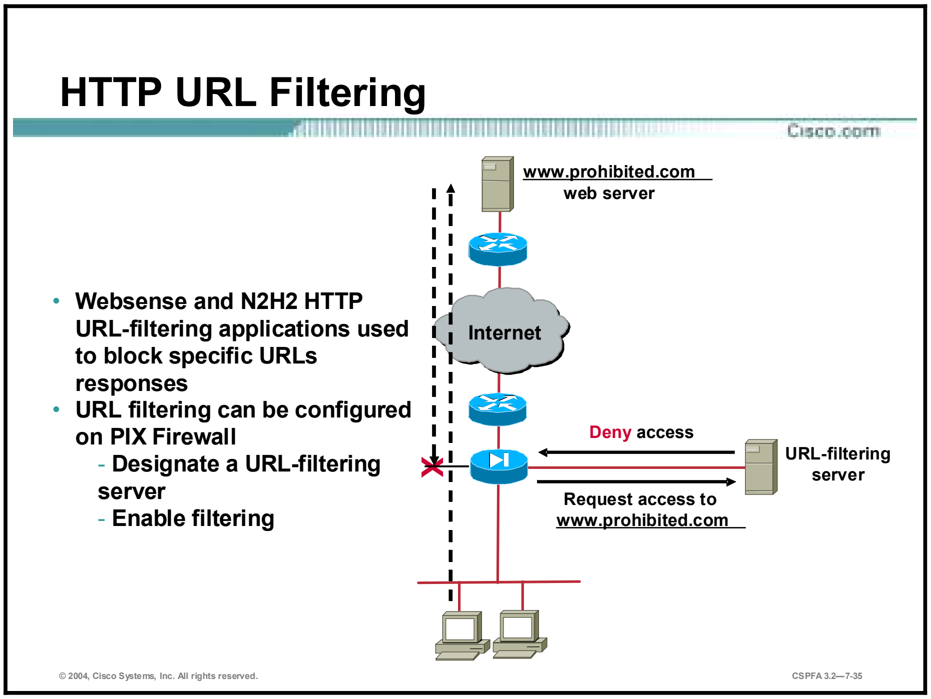
- Specifies that the ActiveX blocking applies to web traffic on port 80 from any local host and for connections to any foreign host

CSPFA 3.2—7-33

The **filter** command enables or disables outbound URL or HTML filtering. In the figure above, the command specifies that ActiveX is being filtered on port 80 from any internal host and for connection to any external host.

URL Filtering

This topic discusses how to configure the PIX Firewall for URL filtering.



HTTP URL Filtering

- Websense and N2H2 HTTP URL-filtering applications used to block specific URLs responses
- URL filtering can be configured on PIX Firewall
  - Designate a URL-filtering server
  - Enable filtering

© 2004, Cisco Systems, Inc. All rights reserved.

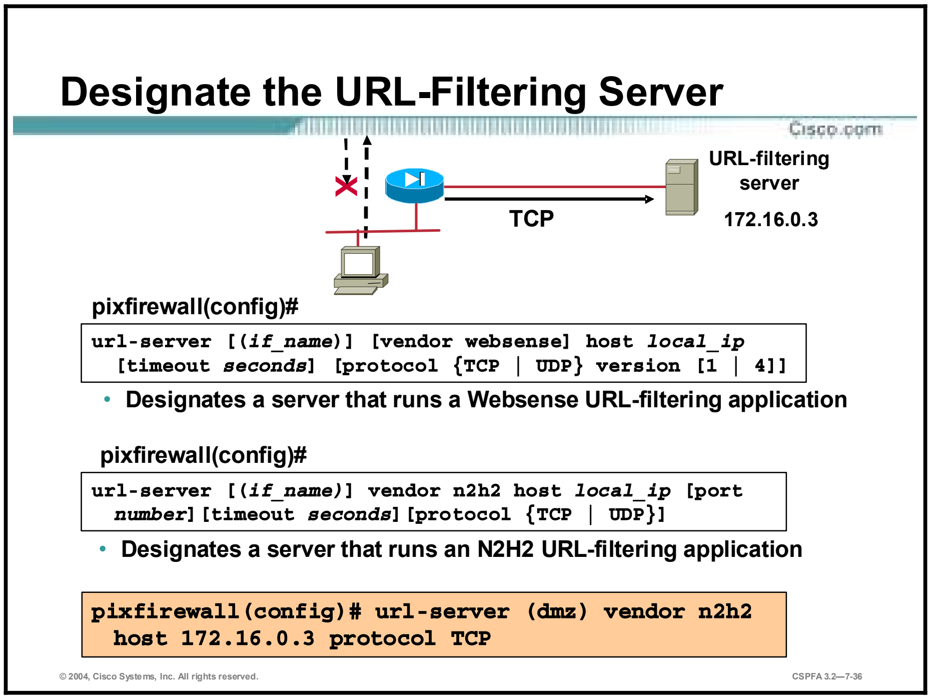
CSPFA 3.2—7-35

URL-filtering applications provide URL filtering for the PIX Firewall, enabling network administrators to effectively monitor and control network traffic. URL-filtering applications are used to block specific URLs because the PIX Firewall cannot. The PIX Firewall can be enabled to work with a Websense or N2H2 URL-filtering application. This is useful since between the hours of 9 a.m. and 5 p.m.:

- 30 to 40 percent of Internet surfing is not business related.
- 70 percent of all Internet porn traffic occurs.
- More than 60 percent of online purchases are made.

When the PIX Firewall receives a request to access a URL from users, it queries the URL-filtering server to determine whether to return, or block, the requested web page. The URL-filtering server checks its configurations to determine whether the URL should be blocked. If the URL should be blocked, URL-filtering applications can display blocking messages or direct the user requesting the URL to a specified web site.

Information about Websense, N2H2, and other Cisco Partners can be located on Cisco.com.



Designate the URL-Filtering Server

pixfirewall(config)#

```
url-server [(if_name)] [vendor websense] host local_ip
  [timeout seconds] [protocol {TCP | UDP} version [1 | 4]]
```

- Designates a server that runs a Websense URL-filtering application

pixfirewall(config)#

```
url-server [(if_name)] vendor n2h2 host local_ip [port
  number] [timeout seconds] [protocol {TCP | UDP}]
```

- Designates a server that runs an N2H2 URL-filtering application

```
pixfirewall(config)# url-server (dmz) vendor n2h2
  host 172.16.0.3 protocol TCP
```

CSPFA 3.2—7-36

Before you can begin URL filtering, you must designate at least one server on which a Websense or N2H2 URL-filtering application will run. The limit is 16 URL servers. You can use only one application at a time, either N2H2 or Websense. Additionally, changing your configuration on the PIX Firewall does not update the configuration on the application server; this must be done separately, according to the individual vendor's instructions.

Use the **url-server** command to designate the server on which it runs, and then enable the URL filtering service with the **filter url** command.

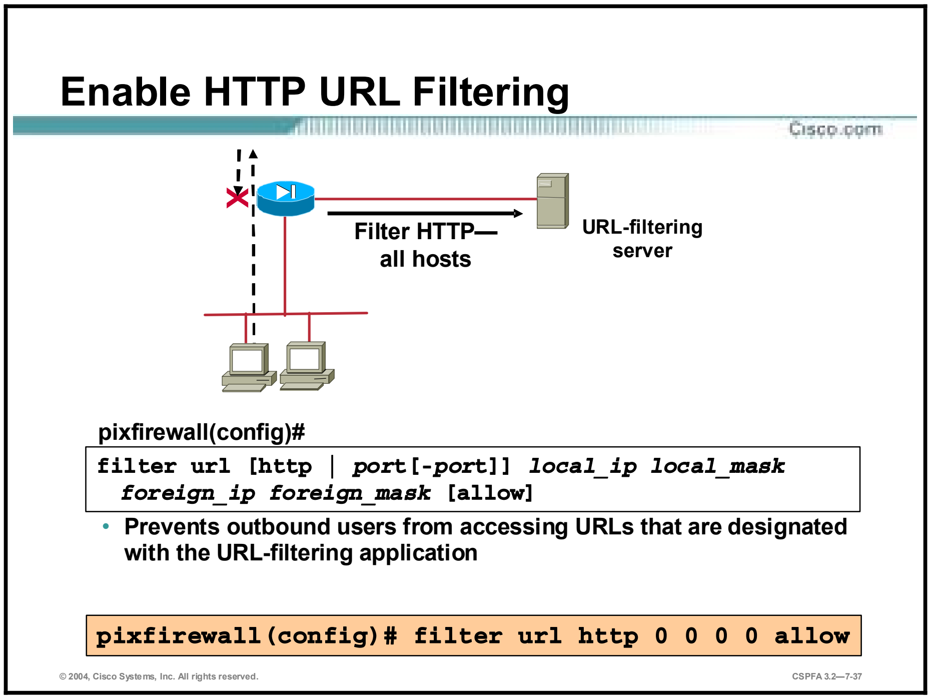
The syntax for the **url-server** commands is as follows:

url-server [(*if_name*)] [vendor websense] host *local_ip* [timeout *seconds*] [protocol {TCP | UDP} version [1 | 4]]

if_name	The network interface where the authentication server resides. If not specified, the default is inside.
vendor websense	Indicates that the URL filtering service vendor is Websense.
host *local_ip*	The server that runs the URL filtering application.
timeout *seconds*	The maximum idle time permitted before the PIX Firewall switches to the next server you specified. The default is 5 seconds.
protocol	The protocol can be configured using TCP or UDP keywords. The default is TCP protocol, version 1.
version	The version of the protocol can be configured using the keywords 1 or 4. The default is TCP protocol, version 1. TCP can be configured using version 1 or version 4. UDP can be configured using version 4 only.

url-server [(*if_name*)] vendor n2h2 host *local_ip* [port *number*] [timeout *seconds*] [protocol TCP | UDP]

if_name	The network interface where the authentication server resides. If not specified, the default is inside.
vendor n2h2	Indicates URL filtering service vendor is N2H2.
host *local_ip*	The server that runs the URL filtering application.
port *number*	The N2H2 server port. The PIX Firewall also listens for UDP replies on this port. The default port number is 4005.
timeout *seconds*	The maximum idle time permitted before the PIX Firewall switches to the next server you specified. The default is 5 seconds.
protocol	The protocol can be configured using TCP or UDP keywords. The default is TCP.



Enable HTTP URL Filtering

pixfirewall(config)#

```
filter url [http | port[-port]] local_ip local_mask
  foreign_ip foreign_mask [allow]
```

- Prevents outbound users from accessing URLs that are designated with the URL-filtering application

```
pixfirewall(config)# filter url http 0 0 0 0 allow
```

CSPFA 3.2—7-37

After designating which server runs the URL-filtering application, use the **filter url** command to tell the PIX Firewall to send URL requests to that server for filtering.

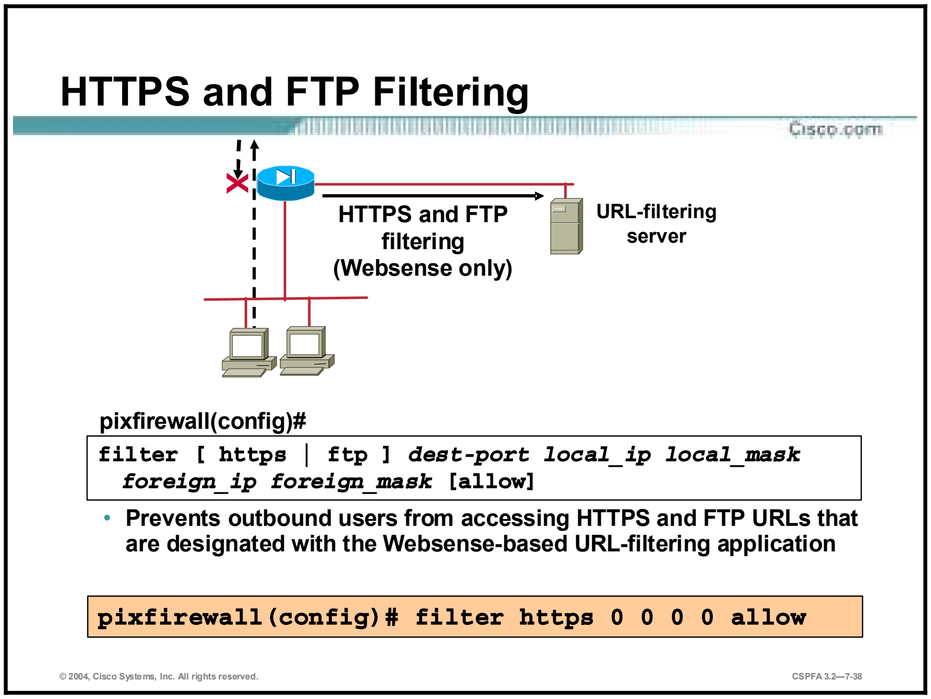
The example command in the figure above instructs the PIX Firewall to send all URL requests to the URL-filtering server to be filtered. The allow option in the **filter** command is crucial to the use of the PIX Firewall URL filtering feature. If you use the allow option and the URL-filtering server goes offline, the PIX Firewall lets all URL requests continue without filtering. If the allow option is not specified, all port 80 URL requests are stopped until the server is back online.

The syntax for the **filter url** command is as follows:

filter url ***port*** **[-*port*] | except *local_ip local_mask foreign_ip foreign_mask* [allow] [proxy-block] [longurl-truncate | longurl-deny] [cgi-truncate]**

url	Filters URLs from data moving through the PIX Firewall.
port	The port that receives Internet traffic on the PIX Firewall. Typically, this is port 80, but other values are accepted. The http or url literal can be used for port 80.
except	Creates an exception to a previous **filter** condition.
local_ip	The IP address of the host from which access is sought.
local_mask	Network mask of *local_ip*.
foreign_ip	The IP address of the host to which access is sought.
foreign_mask	Network mask of *foreign_ip*.
allow	Enables outbound connections to pass through the PIX Firewall without filtering when the URL-filtering server is unavailable.
proxy-block	Prevents users from connecting to an HTTP proxy server.

longurl-truncate	Enables outbound URL traffic whether or not the URL buffer is available.
longurl-deny	Denies the URL request if the URL is over the URL buffer size limit or the URL buffer is not available.
cgi-truncate	Sends a CGI script as an URL.



HTTPS and FTP Filtering

Cisco.com

HTTPS and FTP filtering (Websense only)

URL-filtering server

pixfirewall(config)#

```
filter [ https | ftp ] dest-port local_ip local_mask
  foreign_ip foreign_mask [allow]
```

- Prevents outbound users from accessing HTTPS and FTP URLs that are designated with the Websense-based URL-filtering application

```
pixfirewall(config)# filter https 0 0 0 0 allow
```

CSPFA 3.2—7-38

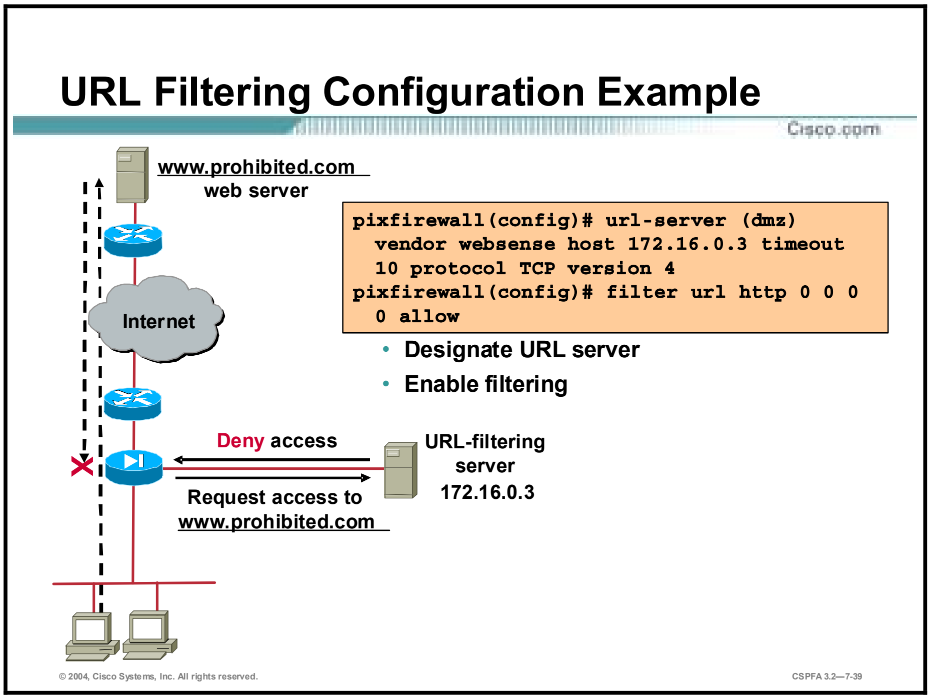
This feature extends Web-based URL filtering to HTTPS and FTP. The **filter ftp** and **filter https** commands were added to the **filter** command in Cisco PIX Firewall Software Version 6.3. The **filter ftp** command enables FTP filtering. The **filter https** command enables HTTPS filtering. The **filter ftp** and **filter https** commands are available with Websense URL filtering only.

The example command in the figure instructs the PIX Firewall to send all URL requests to the URL-filtering server to be filtered. The **allow** option in the **filter** command is crucial to the use of the PIX Firewall URL-filtering feature. If you use the **allow** option and the URL-filtering server goes offline, the PIX Firewall lets all FTP and HTTPS URL requests continue without filtering. If the allow option is not specified, all FTP and HTTPS URL requests are stopped until the server is back online.

The syntax for the **filter ftp and filter https** commands is as follows:

filter [https | ftp] ***dest-port local_ip local_mask foreign_ip foreign_mask*** **[allow]**

https | ftp	Filters HTTPS or FTP URLs from data moving through the PIX Firewall.
dest-port	The destination port number.
local_ip	The IP address of the host from which access is sought.
local_mask	Network mask of ***local_ip***.
foreign_ip	The IP address of the host to which access is sought.
foreign_mask	Network mask of ***foreign_ip***.
allow	Enables outbound connections to pass through the PIX Firewall without filtering when the URL-filtering server is unavailable.

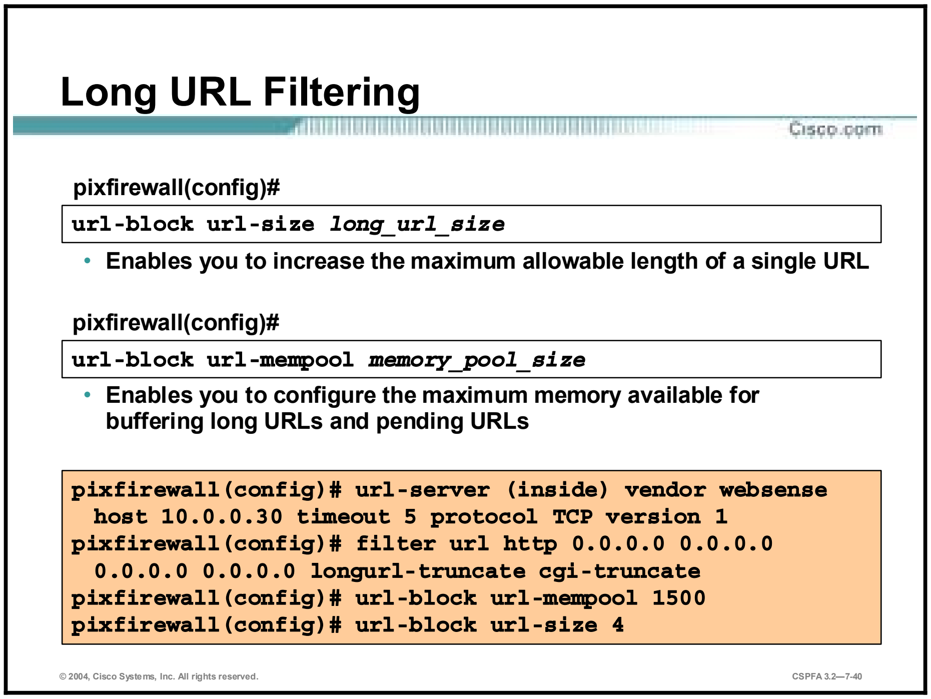


URL Filtering Configuration Example

```
pixfirewall(config)# url-server (dmz)
  vendor websense host 172.16.0.3 timeout
  10 protocol TCP version 4
pixfirewall(config)# filter url http 0 0 0
  0 allow
```

- Designate URL server
- Enable filtering

The commands in the figure instruct the PIX to send all URL requests to the Websense URL-filtering server at 172.16.0.3. Additionally, if the URL-filtering server goes offline, the PIX allows all URL requests to continue without filtering.



Long URL Filtering

pixfirewall(config)#

```
url-block url-size long_url_size
```

- Enables you to increase the maximum allowable length of a single URL

pixfirewall(config)#

```
url-block url-mempool memory_pool_size
```

- Enables you to configure the maximum memory available for buffering long URLs and pending URLs

```
pixfirewall(config)# url-server (inside) vendor websense
  host 10.0.0.30 timeout 5 protocol TCP version 1
pixfirewall(config)# filter url http 0.0.0.0 0.0.0.0
  0.0.0.0 0.0.0.0 longurl-truncate cgi-truncate
pixfirewall(config)# url-block url-mempool 1500
pixfirewall(config)# url-block url-size 4
```

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—7-40

PIX Firewall versions 6.1 and earlier do not support the filtering of URLs longer than 1159 bytes. PIX Firewall version 6.2 supports the filtering of URLs up to 6 KB for the Websense filtering server. You can increase the maximum allowable length of a single URL (for Websense only), by entering the **url-block url-size** command.

When a user issues a request for a long URL, the PIX Firewall breaks the long URL down into multiple IP packets and copies it to buffer memory. The URL is then passed to TCP for sending to the Websense server. You can use the **url-block url-mempool** command to configure the maximum memory available for buffering long URLs. Both the **url-block url-size** and the **url-block url-mempool** commands can be removed by using their no forms.

Note The PIX Firewall does not support long URLs for the Websense UDP Server.

If a URL still exceeds the maximum size or URL buffers are not available, new command options in the **filter url** command enable you to control the behavior. These command options are shown on the previous page. The longurl-truncate option causes the PIX Firewall to send only the hostname or IP address portion of the URL for evaluation to the filtering server when the URL is longer than the maximum length permitted. You can use the longurl-deny option to deny outbound URL traffic if the URL is longer than the maximum permitted.

If the long URL request is a CGI request, the cgi-truncate option, another new **filter url** command option, speeds up its processing. With the cgi-truncate option enabled, the PIX Firewall passes only the CGI script name and location as the URL to the Websense server. The PIX Firewall omits the parameter list, which can be quite long.

With PIX Firewall software version 6.2, you can also use the **url-block block** command to enable the PIX Firewall to buffer the response from the web server until the filtering server responds. This decreases the time the client must wait for an HTTP response. With PIX Firewall software versions 6.1 and earlier, responses from web servers are dropped until the PIX Firewall receives a response from the filtering server, so the client must wait for the web server to resend the response. The response buffering feature works with both the Websense and N2H2 filtering applications. Use the no form of the **url-block block** command to disable response buffering.

Note PIX Firewall version 6.2 supports a maximum URL length of 1159 bytes for the N2H2 filtering server.

The syntax for the **url-block** commands is as follows:

url-block url-mempool ***memory_pool_size***

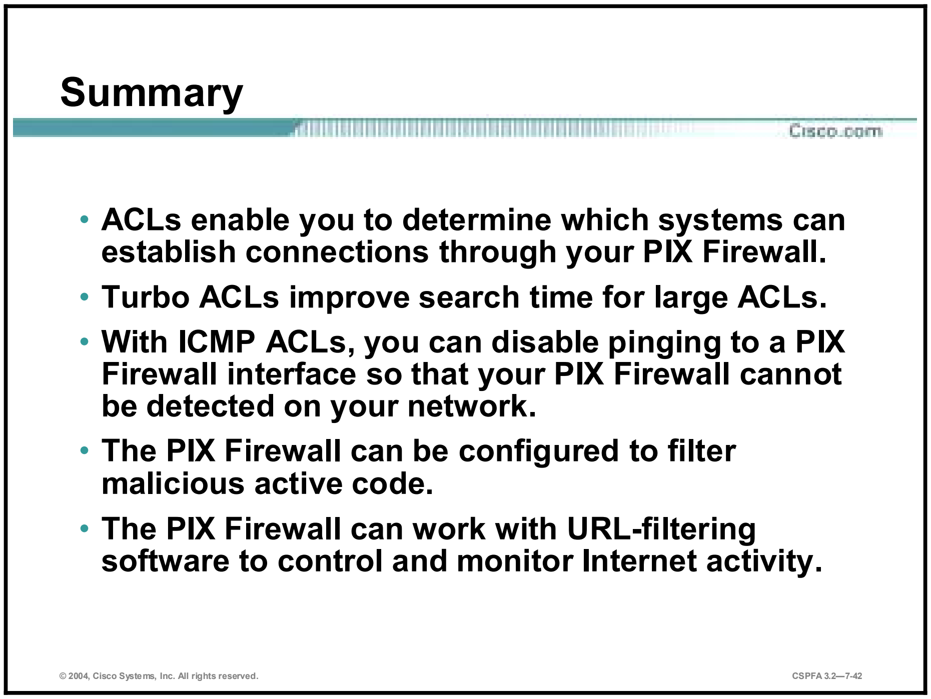
url-block url-size ***long_url_size***

url-block block ***block_buffer_limit***

memory_pool_size	A value from 2 to 10000 for a maximum memory allocation of 2 KB to 10 MB.
long_url_size	A value from 2 to 6 KB for a maximum allowable URL length.
block_buffer_limit	The maximum number of blocks allowed in the HTTP response buffer.

Summary

This topic summarizes the information you learned in this lesson.



Summary

Cisco.com

- ACLs enable you to determine which systems can establish connections through your PIX Firewall.
- Turbo ACLs improve search time for large ACLs.
- With ICMP ACLs, you can disable pinging to a PIX Firewall interface so that your PIX Firewall cannot be detected on your network.
- The PIX Firewall can be configured to filter malicious active code.
- The PIX Firewall can work with URL-filtering software to control and monitor Internet activity.

CSPFA 3.2—7-42

Lab Exercise—Configure ACLs in the PIX Firewall

Complete the following lab exercise to practice what you learned in this lesson.

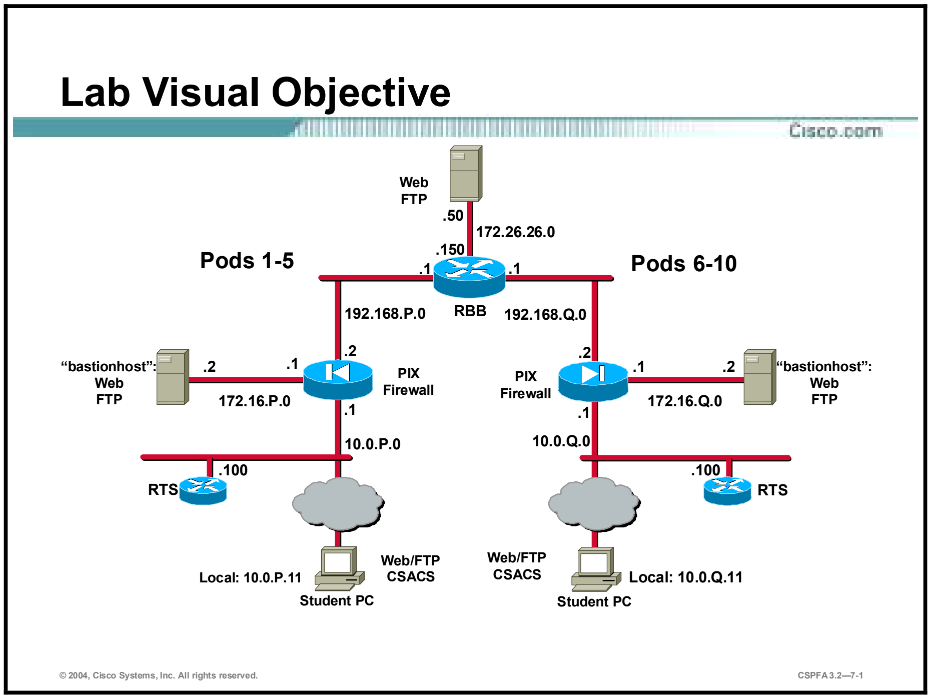
Objectives

In this lab exercise, you will complete the following tasks:

- Disable pinging to an interface.
- Configure an inbound ACL.
- Test and verify the inbound ACL.
- Configure an outbound ACL.
- Test and verify the outbound ACL.
- Filter malicious active code.
- Configure the PIX Firewall to work with a URL-filtering server.

Visual Objective

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

CSPFA 3.2—7-1

Task 1—Disable Pinging to an Interface

Complete the following steps to configure an ICMP ACL to prevent pinging to your PIX Firewall interfaces:

Step 1 From your inside host, ping the inside interface of your PIX Firewall:

```
C:\>ping 10.0.P.1
Pinging 10.0.P.1 with 32 bytes of data:
Reply from 10.0.P.1: bytes=32 time<10ms TTL=253
Reply from 10.0.P.1: bytes=32 time<10ms TTL=253
Reply from 10.0.P.1: bytes=32 time<10ms TTL=253
Reply from 10.0.P.1: bytes=32 time<10ms TTL=253
```

(where P = pod number)

Step 2 From the inside host, ping the outside interface. By default, pinging through the PIX Firewall to a PIX Firewall interface is not allowed:

```
C:\>ping 192.168.P.2
Pinging 192.168.P.2 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
```

(where P = pod number)

Step 3 Use the **icmp** command to prevent pinging the inside interface:

```
pixP(config)# icmp deny any echo inside
```

Step 4 View your ICMP ACL:

```
pixP(config)# show icmp
icmp deny any echo inside
```

Step 5 From your inside host, ping your inside PIX Firewall interface. The ICMP ACL causes the ping to fail:

```
C:\>ping 10.0.P.1
Pinging 10.0.P.1 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
```

(where P = pod number)

Step 6 Enable pinging to your PIX Firewall's inside interface:

```
pixP(config)# clear icmp
```

Step 7 Verify that you can again ping the inside interface of your PIX Firewall:

```
C:\>ping 10.0.P.1
Pinging 10.0.P.1 with 32 bytes of data:
Reply from 10.0.P.1: bytes=32 time<10ms TTL=253
Reply from 10.0.P.1: bytes=32 time<10ms TTL=253
Reply from 10.0.P.1: bytes=32 time<10ms TTL=253
Reply from 10.0.P.1: bytes=32 time<10ms TTL=253
```

(where P = pod number)

Step 8 Save your configuration:

```
pixP(config)# write memory
```

Task 2—Configure an Inbound ACL

Complete the following lab exercise steps to configure ACLs:

- Allow inbound web traffic from a peer pod's network to your bastion host.
- Allow inbound ftp traffic from a peer pod's internal host to your bastion host.
- Allow inbound web traffic to your inside host.
- Allow inbound pings to your inside host and bastion host.
- Allow icmp echo-replies from hosts on the outside and dmz interfaces.
- Deny all other inbound traffic.

Step 1 Configure the following statics for your bastion host and for your inside host:

```
pixP(config)# static (dmz,outside) 192.168.P.11 bastionhost netmask
255.255.255.255
```

```
pixP(config)# static (inside,outside) 192.168.P.10 insidehost netmask
255.255.255.255
```

(where P = pod number)

Step 2 Test web access to the bastion hosts of peer pod groups by completing the following sub-steps. You should be unable to access your peer's bastion host via its static mapping:

1. Open a web browser on your student PC.
2. Use the web browser to access the bastion host of your peer pod group by entering: **http://192.168.Q.11**.
   (where Q = peer pod number)
3. Have a peer pod group attempt to access your bastion host in the same way.

Step 3 Test web access to the inside host of your peer's pod by completing the following sub-steps. You should be unable to access your peer's inside host via its static mapping:

1. Open a web browser on your student PC.
2. Use the web browser to access the inside host of your peer pod group by entering **http://192.168.Q.10**.
   (where Q = peer pod number)
3. Have a peer pod group attempt to access your inside host in the same way.

Step 4 Test FTP access to the bastion hosts of peer pod groups by completing the following sub-steps. You should be unable to access your peer's bastion host via ftp:

1. On your FTP client, attempt to access the bastion hosts of another pod group: **Start>Run>ftp 192.168.Q.11**.
   (where Q = peer pod number)
2. Have a peer pod group use FTP to attempt to access your bastion host.

Step 5 Create an ACL to permit inbound web and ftp access to the bastion host:

```
pixP(config)# access-list ACLIN permit tcp 192.168.Q.0 255.255.255.0
host 192.168.P.11 eq www
```

```
pixP(config)# access-list ACLIN permit tcp host 192.168.Q.10 host
192.168.P.11 eq ftp
```

(where P = pod number, Q = peer pod number)

Step 6 Add commands to permit inbound web traffic to the inside host, permit inbound pings, permit icmp echo replies to the inside host, and deny all other traffic from the Internet:

```
pixP(config)# access-list ACLIN permit tcp any host 192.168.P.10 eq
www
```

```
pixP(config)# access-list ACLIN permit icmp any any echo
```

```
pixP(config)# access-list ACLIN permit icmp any host 192.168.P.10
echo-reply
```

```
pixP(config)# access-list ACLIN deny ip any any
```

(where P = pod number)

Step 7 Bind the ACL to the outside interface:

```
pixP(config)# access-group ACLIN in interface outside
```

Step 8 Create an access-list to allow icmp echo-replies from the bastion host:

```
pixP(config)# access-list ICMPDMZ permit icmp host bastionhost any
echo-reply
```

Step 9 Bind the new ACL to the dmz interface:

```
pixP(config)# access-group ICMPDMZ in interface dmz
```

Step 10 Display the access list and observe the hit counts:

```
pixP(config)# show access-list

access-list aclin; 6 elements

access-list aclin line 1 permit tcp 192.168.1.0 255.255.255.0 host
192.168.6.11

eq www (hitcnt=0)

access-list aclin line 2 permit tcp host 192.168.1.10 host
192.168.6.11 eq ftp

(hitcnt=0)

access-list aclin line 3 permit tcp any host 192.168.6.10 eq www
(hitcnt=0)

access-list aclin line 4 permit icmp any any echo (hitcnt=0)

access-list aclin line 5 permit icmp any host 192.168.6.10 echo-reply
(hitcnt=0)

access-list aclin line 6 deny ip any any (hitcnt=0)

access-list icmpdmz; 1 elements

access-list icmpdmz line 1 permit icmp host bastionhost any echo-reply
(hitcnt=0)
```

(where P = pod number, Q = peer pod number)

Task 3—Test and Verify the Inbound ACL

Complete the following steps to test your inbound ACL:

Step 1 Have a peer inside host ping your inside host:

```
C:\>ping 192.168.Q.10

Pinging 192.168.Q.10 with 32 bytes of data:

Reply from 192.168.Q.10: bytes=32 time<10ms TTL=122

Reply from 192.168.Q.10: bytes=32 time<10ms TTL=122

Reply from 192.168.Q.10: bytes=32 time<10ms TTL=122

Reply from 192.168.Q.10: bytes=32 time<10ms TTL=122
```

(where Q = peer pod number)

Step 2 Have a peer inside host ping your bastion host:

```
C:\>ping 192.168.Q.11
```

```
Pinging 192.168.Q.11 with 32 bytes of data:
Reply from 192.168.Q.11: bytes=32 time<10ms TTL=125
Reply from 192.168.Q.11: bytes=32 time<10ms TTL=125
Reply from 192.168.Q.11: bytes=32 time<10ms TTL=125
Reply from 192.168.Q.11: bytes=32 time<10ms TTL=125
```

(where Q = peer pod number)

Step 3 From your student PC, ping your bastion host:

```
C:\>ping 172.16.P.2
Pinging 172.16.P.2 with 32 bytes of data:
Reply from 172.16.P.2: bytes=32 time<10ms TTL=126
Reply from 172.16.P.2: bytes=32 time<10ms TTL=126
Reply from 172.16.P.2: bytes=32 time<10ms TTL=126
Reply from 172.16.P.2: bytes=32 time<10ms TTL=126
```

(where P = pod number)

Step 4 From your student PC, ping the super server:

```
C:\>ping 172.26.26.50
Pinging 172.26.26.50 with 32 bytes of data:
Reply from 172.26.26.50: bytes=32 time<10ms TTL=125
Reply from 172.26.26.50: bytes=32 time<10ms TTL=125
Reply from 172.26.26.50: bytes=32 time<10ms TTL=125
Reply from 172.26.26.50: bytes=32 time<10ms TTL=125
```

Step 5 Test web access to the bastion hosts of peer pod groups by completing the following sub-steps. You should now be able to access your peer's bastion host via its static mapping:

1. Open a web browser on your student PC.
2. Use the web browser to access the bastion host of your peer pod group by entering: **http://192.168.Q.11**.
   (where Q = peer pod number)
3. Have a peer pod group attempt to access your bastion host in the same way.

Step 6 Test web access to the inside hosts of peer pod groups by completing the following sub-steps. You should now be able to access the IP address of the static mapped to the inside host of the opposite pod group:

1. Open a web browser on the client PC.
2. Use the web browser to access the inside host of your peer pod group by entering: **http://192.168.Q.10**.
   (where Q = peer pod number)
3. Have a peer pod group attempt to access your inside host in the same way.

Step 7 Test FTP access to the bastion hosts of peer pod groups by completing the following sub-steps. You should now be able to access your peer's bastion host via ftp:

1. On your FTP client, attempt to access the bastion host of a peer pod group: **Start>Run>ftp 192.168.Q.11**.
   (where Q = peer pod number)

2. Have a peer pod group use FTP to attempt to access your bastion host.

Step 8 Display the access lists again and observe the hit counts:

```
pixP(config)# show access-list

access-list aclin; 6 elements

access-list aclin line 1 permit tcp 192.168.1.0 255.255.255.0 host
192.168.6.11

eq www (hitcnt=2)

access-list aclin line 2 permit tcp host 192.168.1.10 host
192.168.6.11 eq ftp

(hitcnt=0)

access-list aclin line 3 permit tcp any host 192.168.6.10 eq www
(hitcnt=2)

access-list aclin line 4 permit icmp any any echo (hitcnt=20)

access-list aclin line 5 permit icmp any host 192.168.6.10 echo-reply
(hitcnt=12)

access-list aclin line 6 deny ip any any (hitcnt=0)

access-list icmpdmz; 1 elements

access-list icmpdmz line 1 permit icmp host bastionhost any echo-reply
(hitcnt=12)
```

(where P = pod number, Q = peer pod number)

Task 4—Configure an Outbound ACL

Complete the following lab exercise steps to configure ACLs:

- Deny outbound web traffic.
- Allow outbound ftp traffic from your internal network to 172.26.26.50.

Step 1 Test web access to the Internet by completing the following sub-steps. You should be able to access 172.26.26.50:

1. Open a web browser on your student PC.

2. Use the web browser to access Internet host 172.26.26.50 by entering: **http://172.26.26.50**.

Step 2 Test FTP access to Internet host 172.26.26.50. You should be able to access host 172.26.26.50 via ftp. On your FTP client, attempt to access host 172.26.26.50: **Start>Run>ftp 172.26.26.50**.

Step 3 Create an ACL that prevents users on the internal network from making outbound HTTP connections:

```
pixP(config)# access-list ACLOUT deny tcp any any eq www
```

Note: This access list prevents all outbound connections.

Step 4 Enter the **access-group** command to create an access group that will bind the ACL to an interface:

```
pixP(config)# access-group ACLOUT in interface inside
```

Step 5 Display the access list you configured, and observe the hit count:

```
pixP(config)# show access-list

access-list ACLIN; 6 elements

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq www

(hitcnt=4)

access-list ACLIN line 2 permit tcp host 192.168.Q.10 host
192.168.P.11 eq ftp (hitcnt=

1)

access-list ACLIN line 3 permit tcp any host 192.168.P.10 eq www
(hitcnt=4)

access-list ACLIN line 4 permit icmp any any echo (hitcnt=20)

access-list ACLIN line 5 permit icmp any host 192.168.P.10 echo-reply
(hitcnt=12)

access-list ACLIN line 6 deny ip any any (hitcnt=0)

access-list ICMPDMZ; 1 elements

access-list ICMPDMZ line 1 permit icmp host bastionhost any echo-reply
(hitcnt=12)

access-list ACLOUT; 1 elements

access-list ACLOUT line 1 deny tcp any any eq www (hitcnt=0)
```

(where P = pod number, Q = peer pod number)

Step 6 Test web access to the Internet by completing the following sub-steps. You should be **unable** to access the Internet host:

1. Open a web browser on your student PC.
2. Use the web browser to access the Internet by entering: **http://172.26.26.50.**

Step 7 Test FTP access to an Internet host. The FTP connection should **fail** as well as the HTTP due to the implicit deny:

On your FTP client, attempt to access host 172.26.26.50: **Start>Run>ftp 172.26.26.50**.

Step 8 Display your access list again and note that the hit count has incremented:

```
pixP(config)# show access-list

access-list ACLIN; 6 elements

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq www

(hitcnt=4)

access-list ACLIN line 2 permit tcp host 192.168.Q.10 host
192.168.P.11 eq ftp (hitcnt=

1)

access-list ACLIN line 3 permit tcp any host 192.168.P.10 eq www
(hitcnt=4)
```

```
access-list ACLIN line 4 permit icmp any any echo (hitcnt=20)
access-list ACLIN line 5 permit icmp any host 192.168.P.10 echo-reply
(hitcnt=12)
access-list ACLIN line 6 deny ip any any (hitcnt=0)
access-list ICMPDMZ; 1 elements
access-list ICMPDMZ line 1 permit icmp host bastionhost any echo-reply
(hitcnt=12)
access-list ACLOUT; 1 elements
access-list ACLOUT line 1 deny tcp any any eq www (hitcnt=6)
```

(where P = pod number, Q = peer pod number)

Step 9 Add an additional command to the ACL to permit outbound ftp access to host 172.26.26.50:

```
pixP(config)# access-list ACLOUT permit tcp 10.0.P.0 255.255.255.0
host 172.26.26.50 eq ftp
```

(where P = pod number)

Step 10 Add another access-list command statement to deny other outbound IP traffic:

```
pixP(config)# access-list ACLOUT deny ip any any
```

Note: This access list statement is needed only to enable you to view the hit counts.

Step 11 View your access list again:

```
pixP(config)# show access-list ACLOUT
access-list ACLOUT line 1 deny tcp any any eq www (hitcnt=2)
access-list ACLOUT line 2 permit tcp 10.0.P.0 255.255.255.0 host
172.26.26.50 eq ftp (hitcnt=0)
access-list ACLOUT line 3 deny ip any any (hitcnt=0)
```

(where P = pod number)

Task 5—Test and Verify the Outbound ACL

Complete the following steps to test your outbound ACL:

Step 1 Test web access to the Internet by completing the following sub-steps. You should be **unable** to access the Internet host due to the deny ACL:

1. Open a web browser on your student PC.
2. Use the web browser to access the Internet by entering: **http://172.26.26.50**.

Step 2 Test FTP access to an Internet host by doing the following on your FTP client. At this point, you should be able to connect using FTP: **Start>Run>ftp 172.26.26.50**.

Step 3 Test FTP access to a peer pod's bastion host by attempting to access the peer pod's bastion host on your FTP client. You should be unable to connect using FTP: **Start>Run>ftp 192.168.Q.11**.

(where Q = peer pod number)

Step 4 View your outbound access list again and observe the hit counts:

```
pixP(config)# show access-list ACLOUT

access-list ACLOUT deny tcp any any eq www (hitcnt=2)

access-list ACLOUT permit tcp 10.0.P.0 255.255.255.0 host 172.26.26.50
eq ftp (hitcnt=1)

access-list ACLOUT deny ip any any (hitcnt=3)
```

(where P = pod number)

Caution: Be sure to enter the following command exactly as shown. If you omit the ACL name, all access list statements are removed.

Step 5 Remove the outbound ACL:

```
pixP(config)# clear access-list ACLOUT
```

Step 6 Verify that the outbound ACL has been removed:

```
pixP(config)# show access-list

access-list ACLIN; 6 elements

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq www (hitcnt=4)

access-list ACLIN line 2 permit tcp host 192.168.Q.10 host
192.168.P.11 eq ftp (hitcnt=1)

access-list ACLIN line 3 permit tcp any host 192.168.P.10 eq www
(hitcnt=4)

access-list ACLIN line 4 permit icmp any any echo (hitcnt=20)

access-list ACLIN line 5 permit icmp any host 192.168.P.10 echo-reply
(hitcnt=12)

access-list ACLIN line 6 deny ip any any (hitcnt=0)

access-list ICMPDMZ; 1 elements

access-list ICMPDMZ line 1 permit icmp host bastionhost any echo-reply
(hitcnt=12)
```

(where P = pod number, Q = peer pod number)

Step 7 Save your configuration:

```
pixP(config)# write memory
```

Task 6—Filter Malicious Active Code

Complete the following steps to configure ActiveX and filter Java. You will not be able to test this task.

Step 1 Enter the **filter activex** command to block ActiveX from any local host and for connections to any foreign host on port 80:

```
pixP(config)# filter activex 80 0 0 0 0
```

Step 2 Enter the **filter java** command to block Java applets:

```
pixP(config)# filter java 80 0 0 0 0
```

Step 3 Use the following command to show you the filters:

```
pixP(config)# show filter
filter activex 80 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
filter java 80 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
```

Task 7—Configure the PIX Firewall to Work with a URL-Filtering Server

Complete the following steps to configure the PIX Firewall to work with a URL-filtering server:

Step 1 Enter the **url-server** command to designate the URL-filtering server:

```
pixP(config)# url-server (inside) host insidehost timeout 5 protocol
TCP version 4
```

Step 2 Show the designated url-server by entering the following command:

```
pixP(config)# show url-server
url-server (inside) vendor websense host insidehost timeout 5 protocol
TCP version 4
```

Step 3 Enter the **filter url http** command to prevent outbound users from accessing WWW URLs that are designated with the filtering application:

```
pixP(config)# filter url http 0 0 0 0 allow
```

Step 4 Display the **filter url http** command by using the following command:

```
pixP(config)# show filter url
filter activex 80 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
filter java 80 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
filter url http 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 allow
```

Step 5 Remove the **url-server** command:

```
pixP(config)# no url-server (inside) host insidehost
```

(where P = your pod number)

Step 6 Remove the **filter url** command:

```
pixP(config)# no filter url http 0 0 0 0 allow
```

Step 7 Save your configuration:

```
pixP(config)# write memory
```

8

Object Grouping

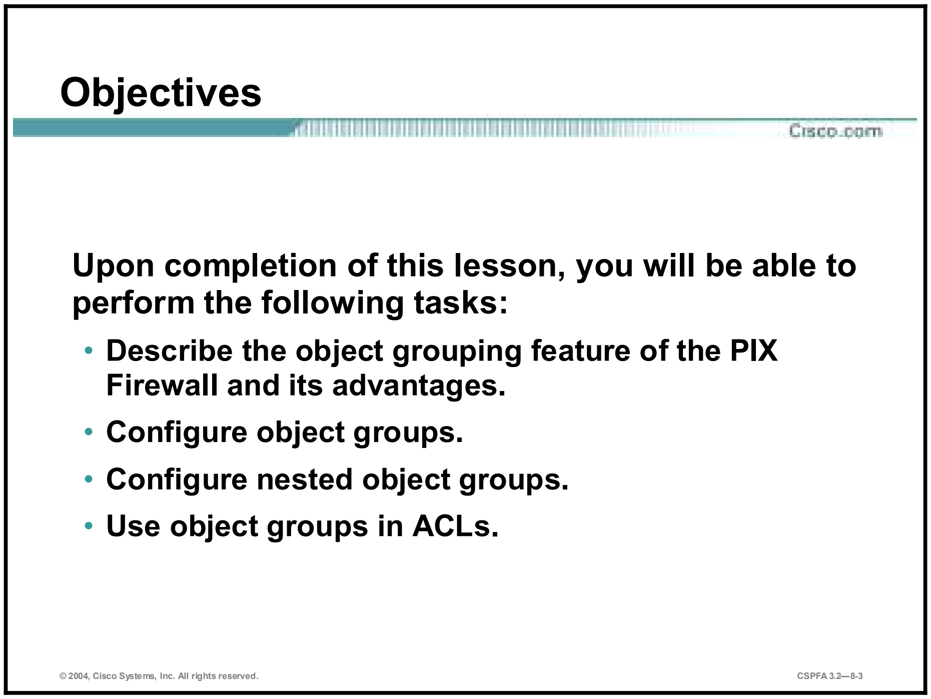
Overview

This lesson includes the following topics:

- Objectives
- Overview of object grouping
- Getting started with object groups
- Configuring object groups
- Nested object groups
- Summary
- Lab exercise

Objectives

This topic lists the lesson's objectives.



Objectives

Cisco.com

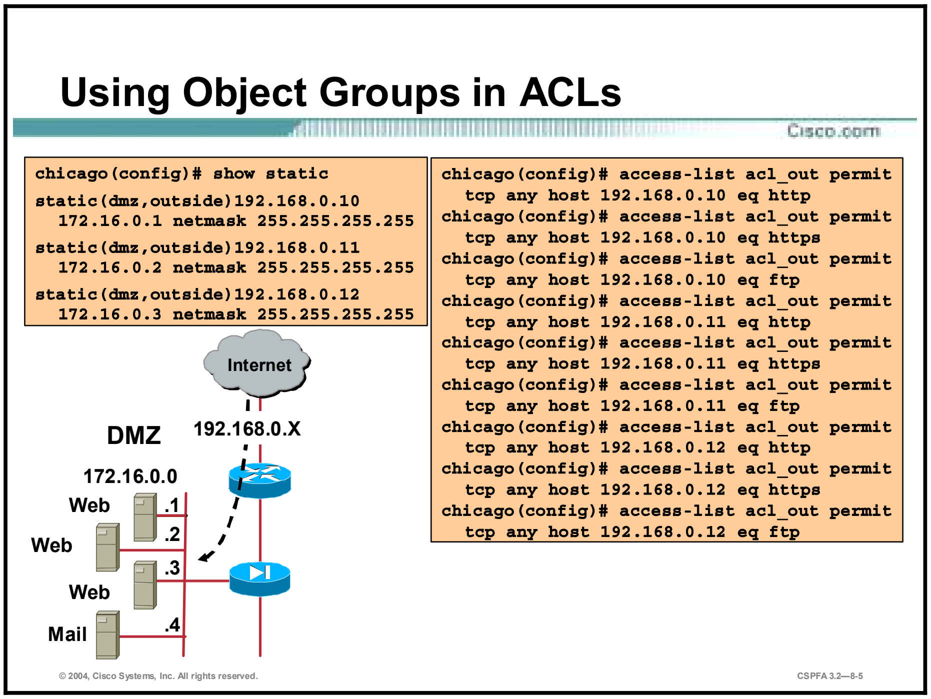
Upon completion of this lesson, you will be able to perform the following tasks:

- **Describe the object grouping feature of the PIX Firewall and its advantages.**
- **Configure object groups.**
- **Configure nested object groups.**
- **Use object groups in ACLs.**

CSPFA 3.2—8-3

Overview of Object Grouping

This topic introduces the object grouping concepts.



Using Object Groups in ACLs

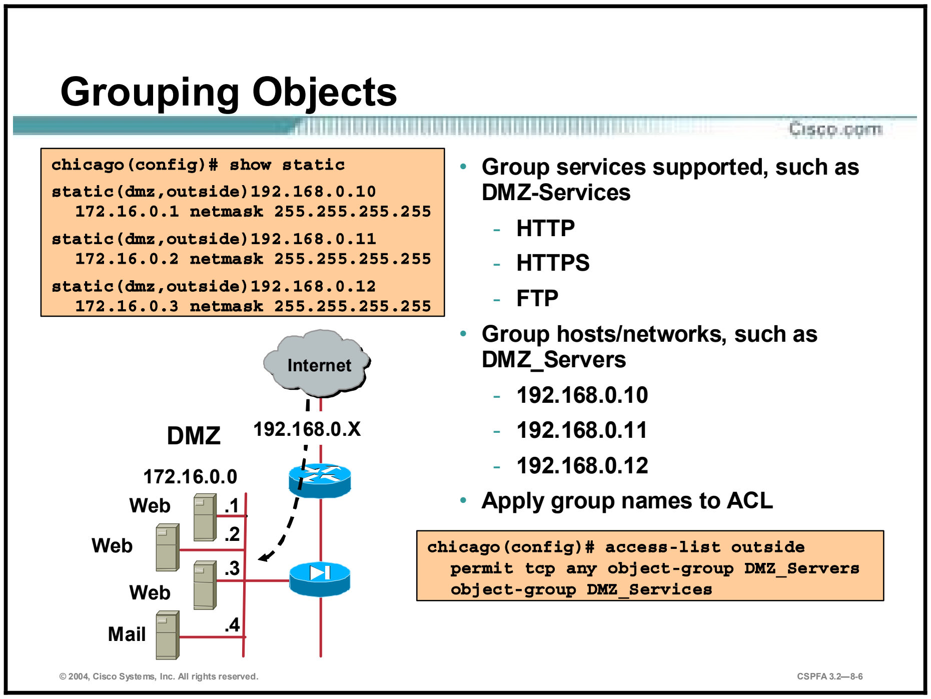
```
chicago(config)# show static
static(dmz,outside)192.168.0.10
  172.16.0.1 netmask 255.255.255.255
static(dmz,outside)192.168.0.11
  172.16.0.2 netmask 255.255.255.255
static(dmz,outside)192.168.0.12
  172.16.0.3 netmask 255.255.255.255
```

```
chicago(config)# access-list acl_out permit
  tcp any host 192.168.0.10 eq http
chicago(config)# access-list acl_out permit
  tcp any host 192.168.0.10 eq https
chicago(config)# access-list acl_out permit
  tcp any host 192.168.0.10 eq ftp
chicago(config)# access-list acl_out permit
  tcp any host 192.168.0.11 eq http
chicago(config)# access-list acl_out permit
  tcp any host 192.168.0.11 eq https
chicago(config)# access-list acl_out permit
  tcp any host 192.168.0.11 eq ftp
chicago(config)# access-list acl_out permit
  tcp any host 192.168.0.12 eq http
chicago(config)# access-list acl_out permit
  tcp any host 192.168.0.12 eq https
chicago(config)# access-list acl_out permit
  tcp any host 192.168.0.12 eq ftp
```

CSPFA 3.2—8-5

An access control list (ACL) can cause the PIX Firewall to allow a designated client to access a particular server for a specific service. When there is only one client, one host, and one service, only a minimum number of lines in an ACL are needed. However, as the number of clients, servers, and services increases, the number of lines in an ACL required increase exponentially.

In the figure, the Internet hosts are granted HTTP, HTTPS, and FTP access to specific hosts on the DMZ. An ACL is configured for each individual host and protocol combination. There are three hosts, three protocols, and nine lines in the ACL.



Grouping Objects

```
chicago(config)# show static
static(dmz,outside)192.168.0.10
  172.16.0.1 netmask 255.255.255.255
static(dmz,outside)192.168.0.11
  172.16.0.2 netmask 255.255.255.255
static(dmz,outside)192.168.0.12
  172.16.0.3 netmask 255.255.255.255
```

- Group services supported, such as DMZ-Services
  - HTTP
  - HTTPS
  - FTP
- Group hosts/networks, such as DMZ_Servers
  - 192.168.0.10
  - 192.168.0.11
  - 192.168.0.12
- Apply group names to ACL

```
chicago(config)# access-list outside
  permit tcp any object-group DMZ_Servers
  object-group DMZ_Services
```

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—8-6

You can group network objects such as hosts and services to simplify the task of creating and applying ACLs. This reduces the number of access control entries (ACEs) required to implement complex security policies. For example, a security policy that normally contains 3,300 ACEs within an ACL might require only several hundred ACEs within that ACL after hosts and services are properly grouped.

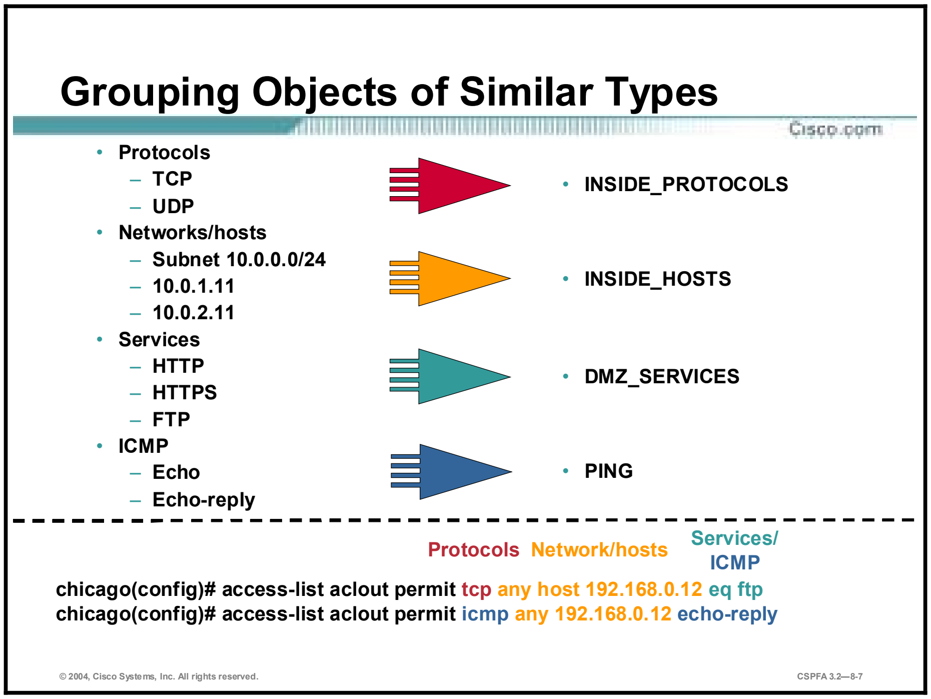
Applying a PIX Firewall object group to a PIX Firewall command is the equivalent of applying every element of the object group to the command. For example, the group DMZ_Servers contains servers 192.168.0.10, 192.168.0.11, and 192.168.0.12. The group DMZ_Services supports HTTP, HTTPS, and FTP protocols. Applying the group DMZ_Servers and DMZ_Services to an ACE is the same as applying all the individual hosts and protocols to the ACE. Therefore, the command:

```
access-list outside permit tcp any object-group DMZ_Servers object-
group DMZ_Services
```

is equivalent to the following:

```
chicago(config)# access-list outside permit tcp any host 192.168.0.10
eq http

chicago(config)# access-list outside permit tcp any host 192.168.0.10
eq https

chicago(config)# access-list outside permit tcp any host 192.168.0.10
eq ftp

chicago(config)# access-list outside permit tcp any host 192.168.0.11
eq http

chicago(config)# access-list outside permit tcp any host 192.168.0.11
eq https

chicago(config)# access-list outside permit tcp any host 192.168.0.11
eq ftp
```

```
chicago(config)# access-list outside permit tcp any host 192.168.0.12
eq http
chicago(config)# access-list outside permit tcp any host 192.168.0.12
eq https
chicago(config)# access-list outside permit tcp any host 192.168.0.12
eq ftp
```



Grouping Objects of Similar Types

Cisco.com

- Protocols
  - TCP
  - UDP
- Networks/hosts
  - Subnet 10.0.0.0/24
  - 10.0.1.11
  - 10.0.2.11
- Services
  - HTTP
  - HTTPS
  - FTP
- ICMP
  - Echo
  - Echo-reply

- INSIDE_PROTOCOLS
- INSIDE_HOSTS
- DMZ_SERVICES
- PING

```
chicago(config)# access-list aclout permit tcp any host 192.168.0.12 eq ftp
chicago(config)# access-list aclout permit icmp any 192.168.0.12 echo-reply
```

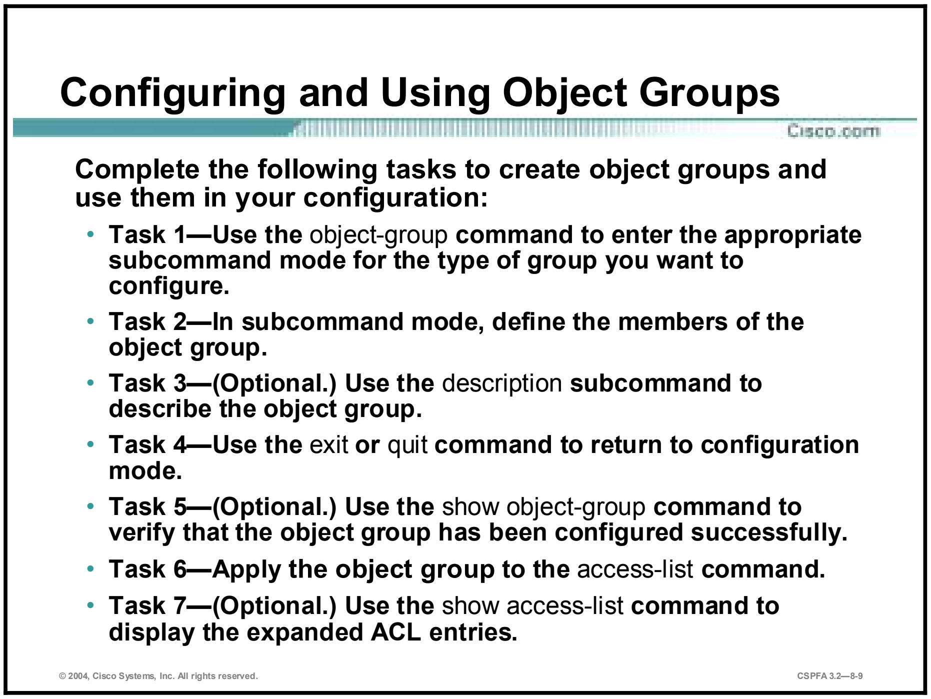
CSPFA 3.2—8-7

Object grouping provides a way to group objects of a similar type so that a single ACL can apply to all the objects in the group. You can create the following types of object groups:

- Network—Used to group client hosts, server hosts, or subnets.
- Protocol—Used to group protocols. It can contain one of the keywords icmp, ip, tcp, or udp, or an integer in the range 1 to 254 representing an IP protocol number. Use the keyword ip to match any Internet protocol, including Internet Control Message Protocol (ICMP), TCP, and UDP.
- Service—Used to group TCP or UDP port numbers assigned to a different service.
- ICMP-type—Used to group ICMP message types to which you permit or deny access.

Getting Started with Object Groups

This topic explains the **object-group** command and its subcommand mode.



Configuring and Using Object Groups

Cisco.com

Complete the following tasks to create object groups and use them in your configuration:

- **Task 1—Use the** object-group **command to enter the appropriate subcommand mode for the type of group you want to configure.**
- **Task 2—In subcommand mode, define the members of the object group.**
- **Task 3—(Optional.) Use the** description **subcommand to describe the object group.**
- **Task 4—Use the** exit **or** quit **command to return to configuration mode.**
- **Task 5—(Optional.) Use the** show object-group **command to verify that the object group has been configured successfully.**
- **Task 6—Apply the object group to the** access-list **command.**
- **Task 7—(Optional.) Use the** show access-list **command to display the expanded ACL entries.**

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—8-9

Complete the following steps to configure an object group and to use it for configuring ACLs:

Step 1 Use the **object-group** command to enter the appropriate subcommand mode for the type of group you want to configure. When you enter the **object-group** command, the system enters the appropriate subcommand mode for the type of object you specify in the object-group command. All subcommands entered from the subcommand prompt apply to the object group identified by the **object-group** command.

Step 2 In subcommand mode, define the members of the object group. In subcommand mode, you can enter object grouping subcommands as well as all other PIX Firewall commands, including **show** commands and **clear** commands. Enter a question mark (?) in the subcommand mode to view the permitted subcommands.

Step 3 (Optional.) Use the **description** subcommand to describe the object group.

Step 4 Return to configuration mode by entering the **exit** command or the **quit** command. When you enter any valid configuration command other than one designed for object grouping, the subcommand mode is terminated.

Step 5 (Optional.) Use the **show object-group** command to verify that the object group has been configured successfully. This command displays a list of the currently configured object groups of the specified type. Without a parameter, the command displays all object groups.

Step 6 Apply the object group to the **access-list** command. Replace the parameters of the **access-list** commands with the corresponding object group, as summarized by the following:

- Replace the protocol parameter with one Protocol object group preceded by the keyword object-group.

- Replace the source IP address and subnet mask with one Network object group preceded by the keyword object-group.
- Replace the destination IP address and subnet mask with one Network object group preceded by the keyword object-group.
- Replace the port parameter with one Service object group preceded by the keyword object-group.
- Replace the icmp-type parameter with one ICMP-Type object group preceded by the keyword object-group.

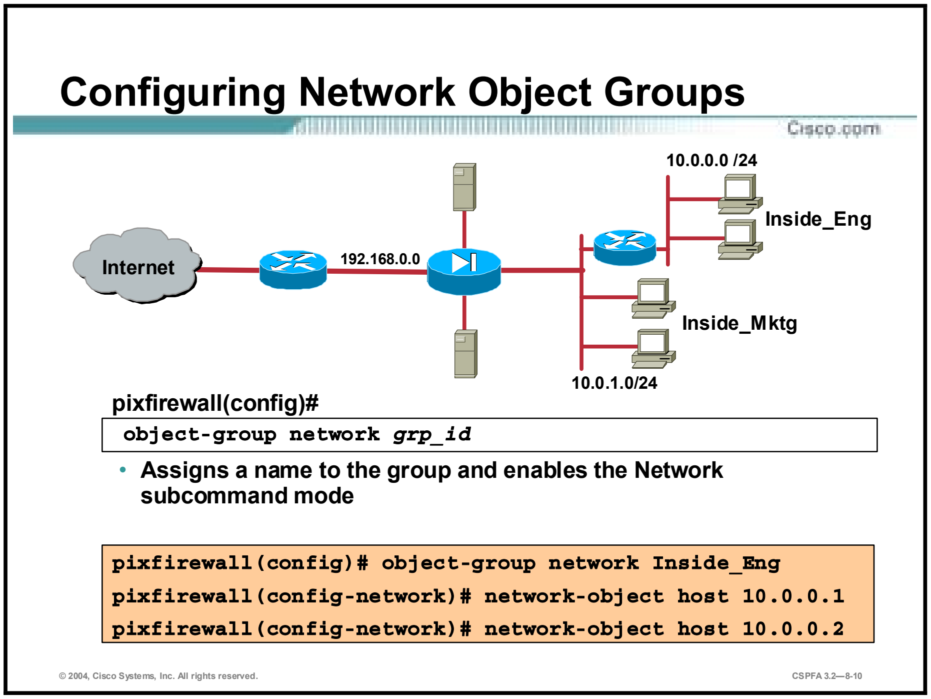
For example, the following command permits access to the members of the Network object group TrustedHosts:

```
access-list EXAMPLE permit tcp any object-group DMZHosts
```

Step 7 (Optional.) Use the **show access-list** command to display the expanded ACEs.

Configuring Object Groups

This topic explains how to configure Network, Service, Protocol, and ICMP-Type object groups.



Configuring Network Object Groups

pixfirewall(config)#

```
object-group network grp_id
```

- Assigns a name to the group and enables the Network subcommand mode

```
pixfirewall(config)# object-group network Inside_Eng
pixfirewall(config-network)# network-object host 10.0.0.1
pixfirewall(config-network)# network-object host 10.0.0.2
```

CSPFA 3.2—8-10

To configure a Network object group, first enter the **object-group network** command to name the Network object and enable the Network object subcommand mode. Once inside the subcommand mode, you can use the **network-object** command to add a single host or network to the Network object group.

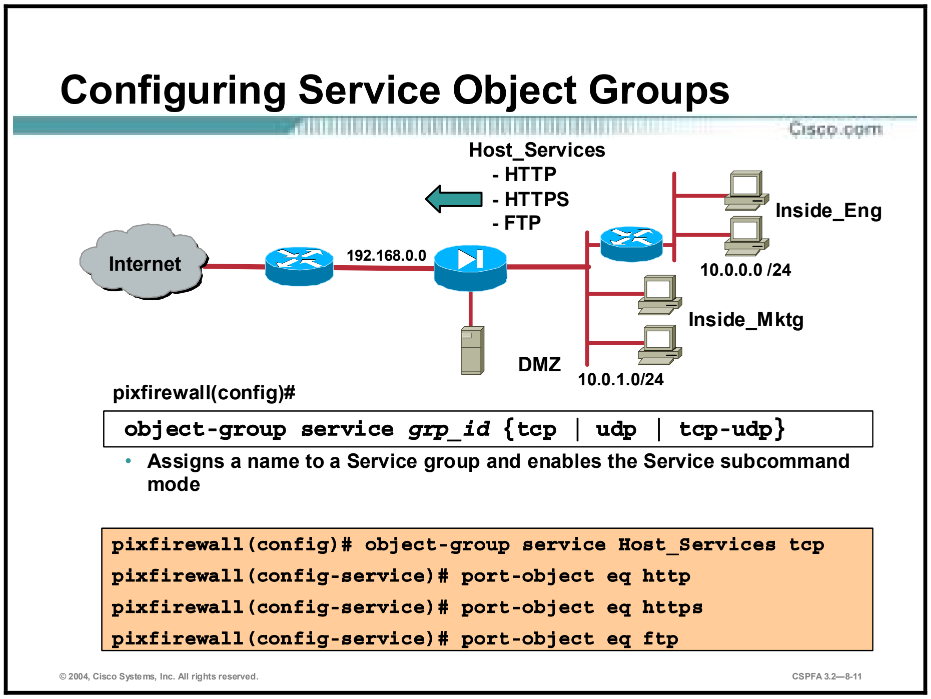
The syntax for the **network-object** command is as follows:

network-object host ***host_addr*** **|** ***host_name***

network-object ***net_addr netmask***

host	Keyword used with the host_addr parameter to define a host object.
host-addr	The host IP address.
host_name	The hostname (if the hostname is already defined using the **name** command).
net_addr	The network address. Used with netmask to define a subnet object.
netmask	The netmask. Used with net_addr to define a subnet object.

In the figure, the administrator wants to add specific 10.0.0.0/24 hosts to the Inside_Eng object group. The object group Inside_Eng is defined first. In the subcommand mode, the individual hosts are added to the group, hosts 10.0.0.1 and 10.0.0.2.



Configuring Service Object Groups

pixfirewall(config)#

```
object-group service grp_id {tcp | udp | tcp-udp}
```

- Assigns a name to a Service group and enables the Service subcommand mode

```
pixfirewall(config)# object-group service Host_Services tcp
pixfirewall(config-service)# port-object eq http
pixfirewall(config-service)# port-object eq https
pixfirewall(config-service)# port-object eq ftp
```

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—8-11

To configure a Service object group, first enter the **object-group service** command to name the Service object and enable the Service object subcommand mode. Using the tcp option specifies that the Service object group contains ports that are used for TCP only. Using the udp option specifies that the Service object group contains ports that are used for UDP only. Using the tcp-udp option specifies that the Service object group contains ports that are used for both TCP and UDP.

Once inside the subcommand mode, you can use the **port-object** command to add a TCP or UDP port number to the Service object group. You can also add a range of TCP or UDP port numbers to the Service object group.

The syntax for the **port-object** commands is as follows:

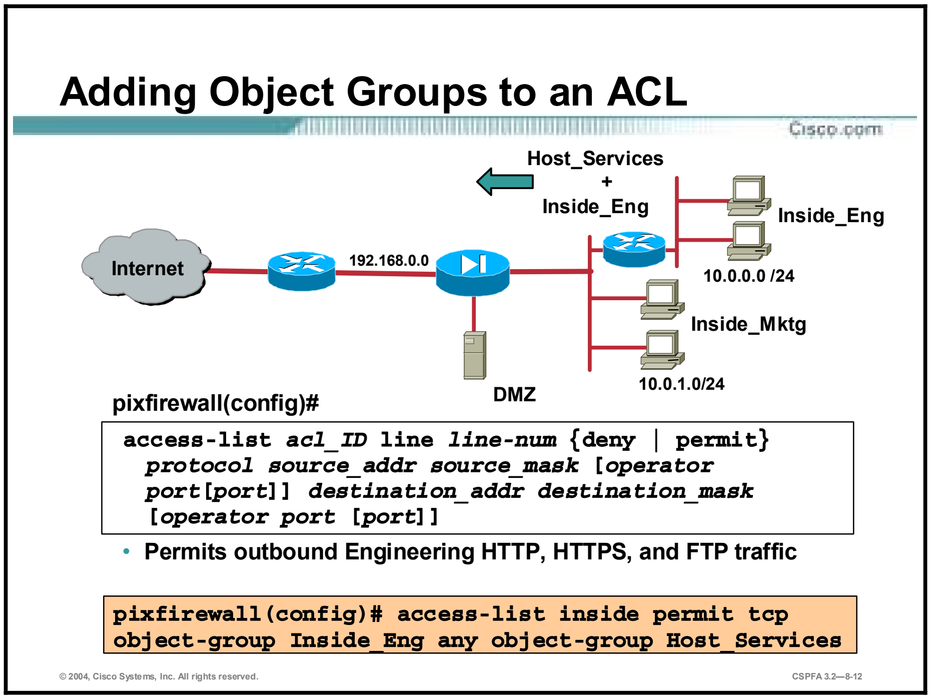
port-object eq *service*

port-object range *begin_service end_service*

eq	Keyword indicating "equal to."
service	The decimal number or name of a TCP or UDP port for a particular service object.
range	Keyword indicating that the range parameters follow.
begin_service	The decimal number or name of a TCP or UDP port that is the beginning value for a range of services.
end_service	The decimal number or name of a TCP or UDP port that is the ending value for a range of services.

In the figure, the administrator wants each Inside_Eng host to have outbound HTTP, HTTPS, and FTP protocol capabilities. An object-group, Host_Services, is defined. Individual protocols, HTTP, HTTPS, and FTP, are added in the subcommand mode.

Note The protocol type of a service group object and the protocol type of the ACE to which it is associated must match. For example, if the service group Host_Service is created for tcp services, this object can only be associated with an ACE (permit or deny) that also refers to tcp services: pixfirewall(config)# access-list inside permit **tcp** object-group Inside_Eng any **object-group Host_Services.**



Adding Object Groups to an ACL

pixfirewall(config)#

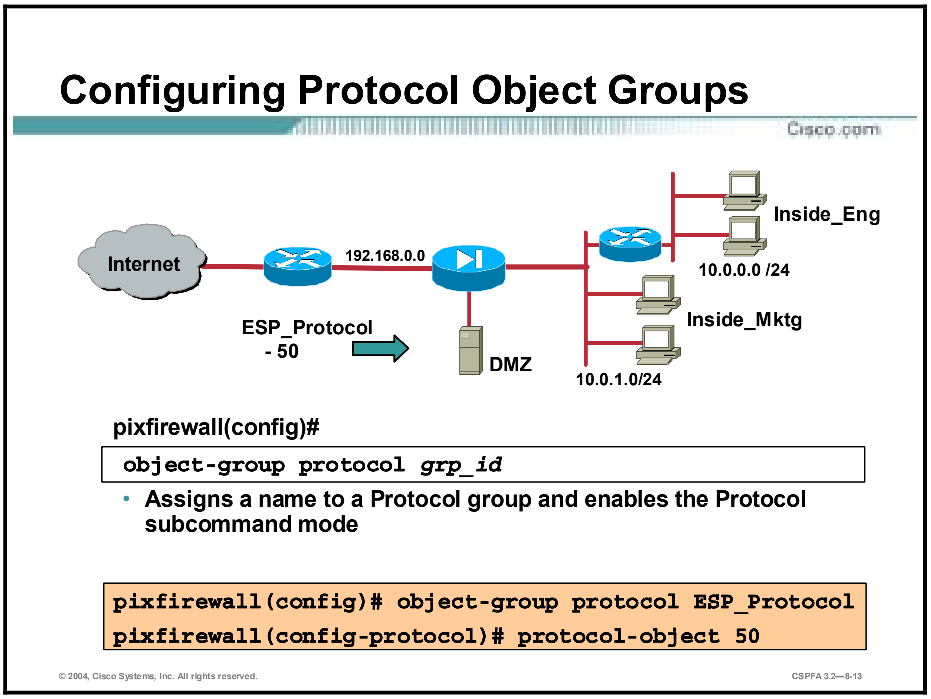
```
access-list acl_ID line line-num {deny | permit}
  protocol source_addr source_mask [operator
  port[port]] destination_addr destination_mask
  [operator port [port]]
```

- Permits outbound Engineering HTTP, HTTPS, and FTP traffic

```
pixfirewall(config)# access-list inside permit tcp
object-group Inside_Eng any object-group Host_Services
```

CSPFA 3.2—8-12

The last step is to add the object groups to an ACL. In the figure, Inside_Eng and Host_Services are defined within the ACL statement. Hosts within the Inside_Eng group can access outbound any host with the protocols defined within the Host_Services object group.



Configuring Protocol Object Groups

pixfirewall(config)#

```
object-group protocol grp_id
```

- Assigns a name to a Protocol group and enables the Protocol subcommand mode

```
pixfirewall(config)# object-group protocol ESP_Protocol
pixfirewall(config-protocol)# protocol-object 50
```

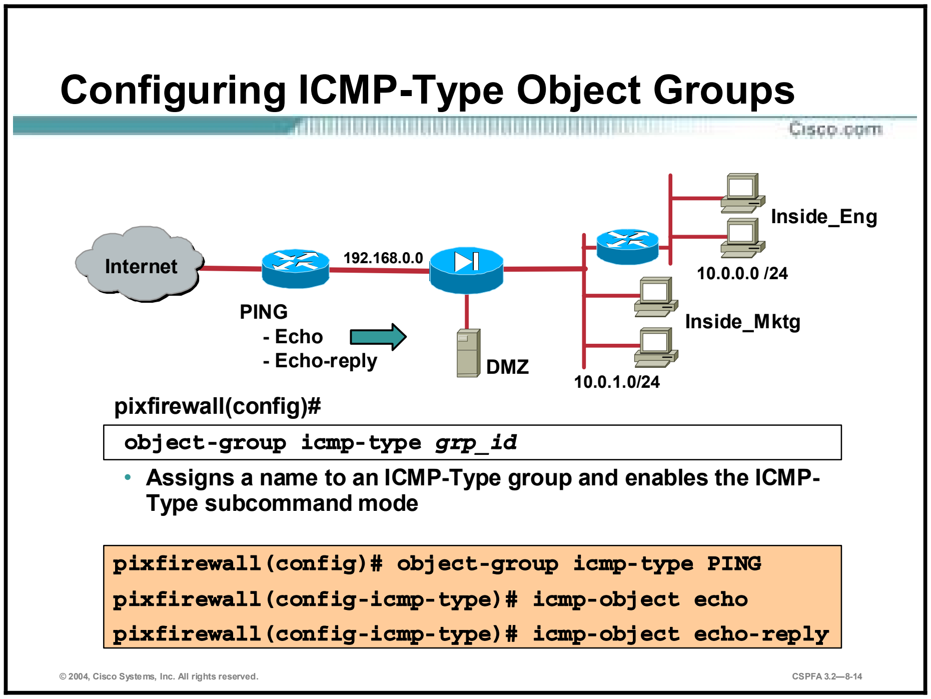
CSPFA 3.2—8-13

To configure a Protocol object group, first enter the **object-group protocol** command to name the Protocol object and enable the Protocol object subcommand mode. Once inside the subcommand mode, you can use the **protocol-object** command to add a protocol to the current protocol object group.

The syntax for the **protocol-object** command is as follows:

protocol-object ***protocol***

protocol	The numeric identifier of the specific IP protocol (1 to 254) or a literal keyword identifier (icmp, tcp, or udp). If you wish to include all IP protocols, use the keyword ip.



Configuring ICMP-Type Object Groups

Cisco.com

pixfirewall(config)#

```
object-group icmp-type grp_id
```

- Assigns a name to an ICMP-Type group and enables the ICMP-Type subcommand mode

```
pixfirewall(config)# object-group icmp-type PING
pixfirewall(config-icmp-type)# icmp-object echo
pixfirewall(config-icmp-type)# icmp-object echo-reply
```

CSPFA 3.2—8-14

To configure an ICMP-Type object group, first enter the **object-group icmp-type** command to name the ICMP-Type object and enable the ICMP-Type subcommand mode. Once inside the subcommand mode, you can use the **icmp-object** command to add an ICMP-Type to the object group.

The syntax for the **icmp-object** command is as follows:

icmp-object ***icmp-type***

icmp-type	Numeric value or name of an ICMP message type.

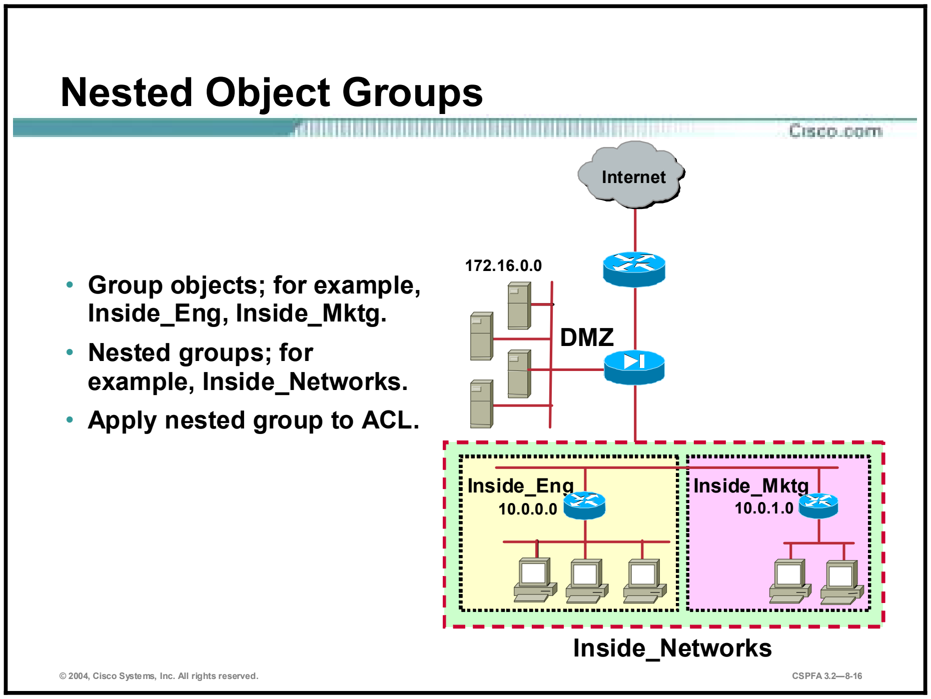
The following are valid ICMP message types:

- 0—echo-reply
- 3—destination-unreachable
- 4—source-quench
- 5—redirect
- 6—alternate-address
- 8—echo
- 9—router-advertisement
- 10—router-solicitation
- 11—time-exceeded
- 12—parameter-problem
- 13—timestamp-request
- 14—timestamp-reply

- 15—information-request
- 16—information-reply
- 17—address-mask-request
- 18—address-mask-reply
- 31—conversion-error
- 32—mobile-redirect

Nested Object Groups

This topic explains how to configure and use nested object groups.



Nested Object Groups

- Group objects; for example, Inside_Eng, Inside_Mktg.
- Nested groups; for example, Inside_Networks.
- Apply nested group to ACL.

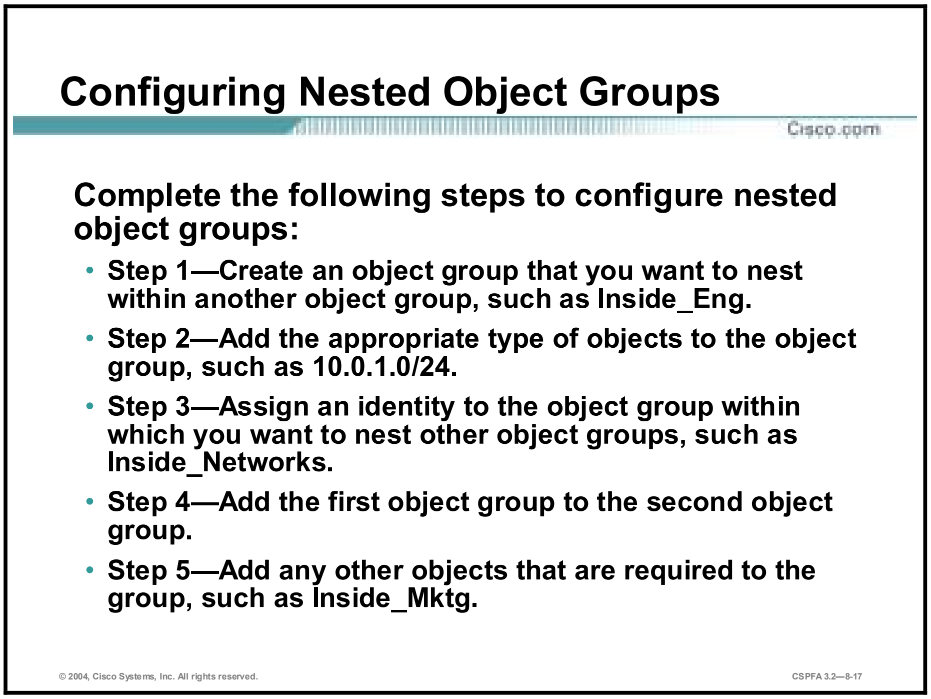
CSPFA 3.2—8-16

In the first topic of this lesson, you learned how to group objects to simplify the task of creating and applying ACLs. It is also possible to group objects within a nested group. An object can be a member of a group. For object groups to be nested, they must be of the same type, for example, all networks/hosts, protocols, or services. In the figure, for example, the administrator configured hosts from the 10.0.0.0/24 network to form the Inside_Eng object group. The administrator added hosts from the 10.0.1.0/24 network to form the Inside_Mktg object group. For some ACLs, the administrator found it advantageous to combine the Inside_Eng and Inside_Mktg object groups to form the nested object group Inside_Networks and apply the nested object group, Inside_Networks to selected ACLs. Hierarchical object grouping can achieve greater flexibility and modularity for specifying access rules.

The **group-object** command enables you to construct hierarchical, or nested, object groups. The difference in object groups and group objects is as follows:

- An object group is group consisting of objects.
- A group object is an object in a nested group and is itself a group.

Duplicated objects are allowed in an object group if it is due to the inclusion of group objects. For example, if object 1 is in both group A and group B, you can define a group C which includes both A and B. You cannot, however, include a group object, which causes the group hierarchy to become circular. For example, you cannot have group A include group B and also have group B include group A.



Configuring Nested Object Groups

Complete the following steps to configure nested object groups:

- **Step 1—Create an object group that you want to nest within another object group, such as Inside_Eng.**
- **Step 2—Add the appropriate type of objects to the object group, such as 10.0.1.0/24.**
- **Step 3—Assign an identity to the object group within which you want to nest other object groups, such as Inside_Networks.**
- **Step 4—Add the first object group to the second object group.**
- **Step 5—Add any other objects that are required to the group, such as Inside_Mktg.**

CSPFA 3.2—8-17

Complete the following steps to configure nested object groups:

Step 1 Assign a group identity to the object group that you want to nest within another object group:

```
pixfirewall(config)# object-group network Inside_Eng
```

Step 2 Add the appropriate type of objects to the object group:

```
pixfirewall(config-network)# network-object 10.0.1.0/24
pixfirewall(config-network)# network-object 10.0.2.0/24
pixfirewall(config-network)# network-object 10.0.3.0/24
```

Step 3 Create the object group within which you want to nest another object group:

```
pixfirewall(config)# object-group network Inside_Networks
```

Step 4 Add the first object group to the group that will contain it:

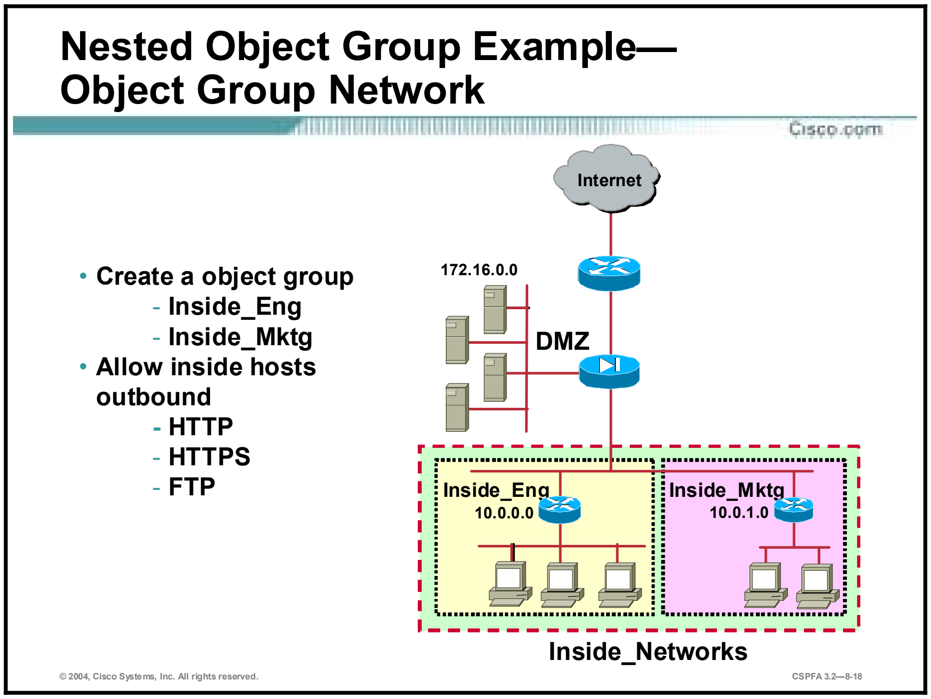
```
pixfirewall(config-network)# group-object Inside_Eng
```

Step 5 Add any other objects that are required to the group:

```
pixfirewall(config-network)# network-object 10.0.4.0/24
```

The resulting configuration of Inside_Networks in this example is equivalent to the following:

```
pixfirewall(config-network)# network-object 10.0.1.0/24
pixfirewall(config-network)# network-object 10.0.2.0/24
pixfirewall(config-network)# network-object 10.0.3.0/24
pixfirewall(config-network)# network-object 10.0.4.0/24
```

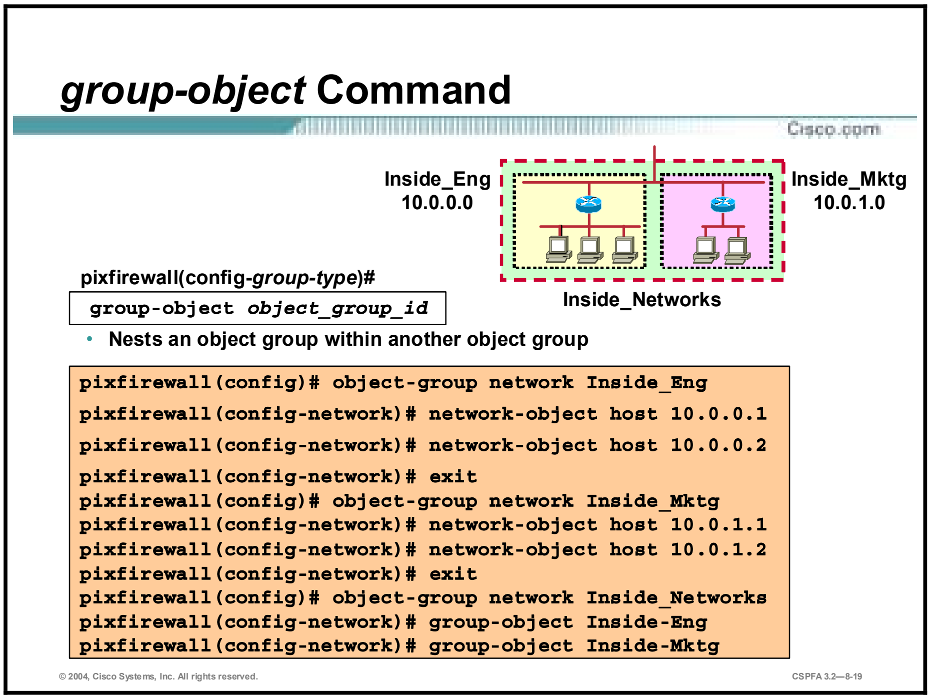


Nested Object Group Example— Object Group Network

- Create a object group
  - Inside_Eng
  - Inside_Mktg
- Allow inside hosts outbound
  - HTTP
  - HTTPS
  - FTP

CSPFA 3.2—8-18

In the example in the figure, the administrator wants to enable selected hosts to establish HTTP, HTTPS, and FTP outbound sessions. The administrator configures an object group for selected Inside_Eng and Inside_Mktg hosts. These two groups are nested inside another group, Inside_Networks. To ease configuration, the three protocols are grouped. The next pages discuss the configuration.



group-object Command

Cisco.com
Inside_Eng
10.0.0.0
Inside_Mktg
10.0.1.0
Inside_Networks

pixfirewall(config-*group-type*)#

```
group-object object_group_id
```

- Nests an object group within another object group

```
pixfirewall(config)# object-group network Inside_Eng
pixfirewall(config-network)# network-object host 10.0.0.1
pixfirewall(config-network)# network-object host 10.0.0.2
pixfirewall(config-network)# exit
pixfirewall(config)# object-group network Inside_Mktg
pixfirewall(config-network)# network-object host 10.0.1.1
pixfirewall(config-network)# network-object host 10.0.1.2
pixfirewall(config-network)# exit
pixfirewall(config)# object-group network Inside_Networks
pixfirewall(config-network)# group-object Inside-Eng
pixfirewall(config-network)# group-object Inside-Mktg
```

CSPFA 3.2—8-19

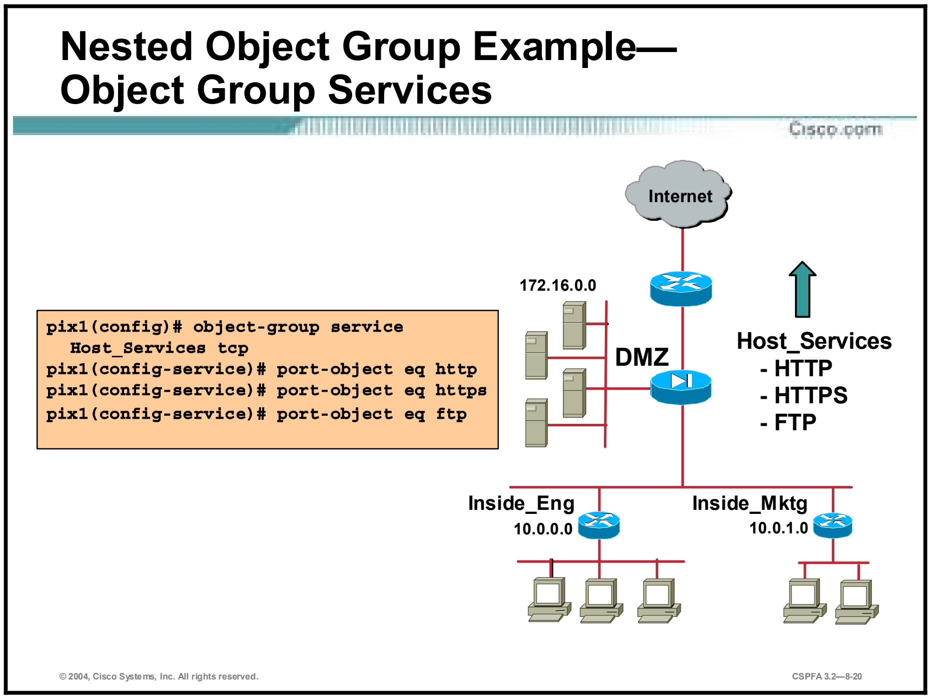
The **group-object** command is a subcommand of the **object-group** command and is used to nest an object group within another object group. For object groups to be nested, they must be of the same type. For example, you can group two or more Network object groups together, but you cannot group a Protocol group and a Network group together.

The syntax for the **group-object** command is as follows:

group-object ***object_group_id***

object_group_id	The group_id of an existing object group. The type of the object groups, both child and parent, must be the same.

In the example in the figure, hosts 10.0.0.1 and 10.0.0.2 are grouped into the Inside_Eng object group. Hosts 10.0.1.1 and 10.0.1.2 are grouped into the Inside_Mktg object group. These two object groups are combined into one nested group called Inside_Networks.

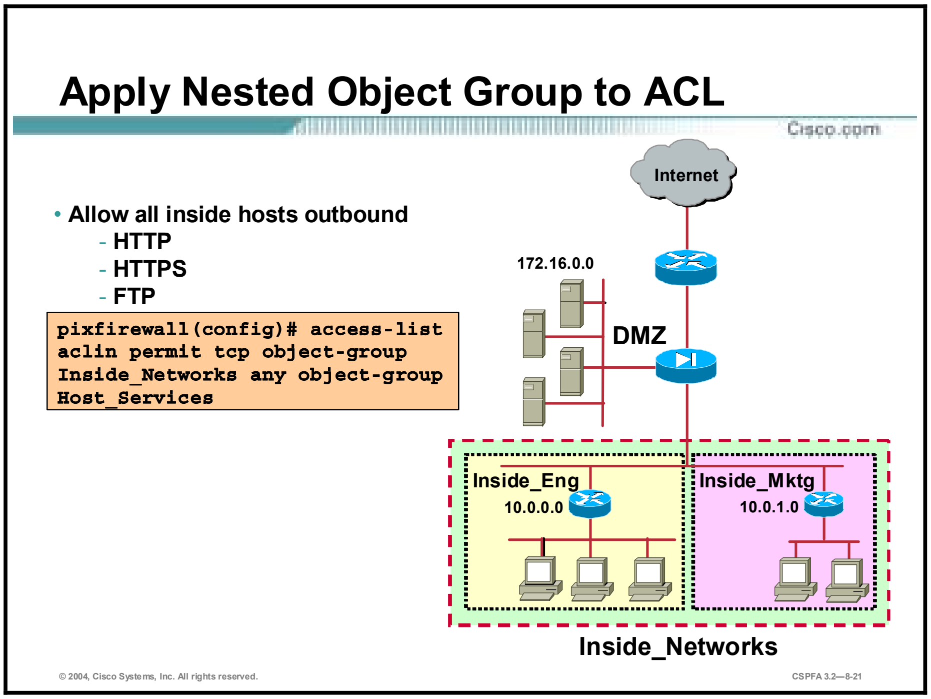


Nested Object Group Example— Object Group Services

```
pix1(config)# object-group service
  Host_Services tcp
pix1(config-service)# port-object eq http
pix1(config-service)# port-object eq https
pix1(config-service)# port-object eq ftp
```

CSPFA 3.2—8-20

In the example in the figure, three protocols are combined into one Services object group. All the hosts in Inside_Eng and Inside_Mktg have the same protocol privileges. It is easier to group the protocols than add them to ACLs individually.



Apply Nested Object Group to ACL

- Allow all inside hosts outbound
  - HTTP
  - HTTPS
  - FTP

```
pixfirewall(config)# access-list
aclin permit tcp object-group
Inside_Networks any object-group
Host_Services
```

CSPFA 3.2—8-21

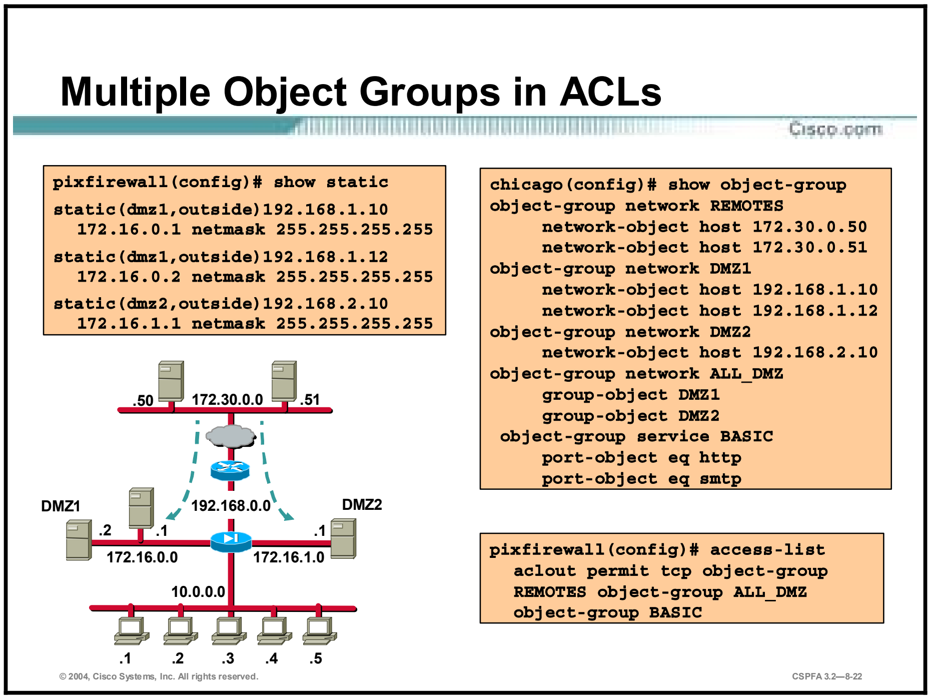
You must use the keyword object-group before the object group name to use object groups in your ACLs. An object group cannot be removed or emptied if it is currently being used in an access list command. In the figure, all hosts in the Inside_Networks nested group are permitted to access any host via the services in the Host_Services group.

When used with object grouping, the syntax for the **access-list** commands is as follows:

access-list *acl_ID* {**deny** | **permit**} **object-group** *protocol_obj_grp_id* **object-group** *network_obj_grp_id* [**object-group** *service_obj_grp_id*] **object-group** *network_obj_grp_id* [**object-group** *service_obj_grp_id*]

access-list *acl_ID* {**deny** | **permit**} **icmp object-group** *network_obj_grp_id* **object-group** *network_obj_grp_id* [*icmp_type* | **object-group** *icmp_type_obj_grp_id*]

acl_ID	Name of an access list. You can use either a name or a number.
deny	When used with the **access-group** command, the deny option prohibits a packet from traversing the PIX Firewall.
permit	When used with the **access-group** command, the permit option selects a packet to traverse the PIX Firewall.
object-group	Specifies an object group.
protocol_obj_grp_id	Name of an existing protocol object group.
network_obj_grp_id	Name of an existing network object group.
service_obj_grp_id	Name of an existing service object group.
icmp	Specifies that the access lists permits or denies icmp.
icmp_obj_grp_id	Name of an existing icmp-type object group.



Multiple Object Groups in ACLs

```
pixfirewall(config)# show static
static(dmz1,outside)192.168.1.10
  172.16.0.1 netmask 255.255.255.255
static(dmz1,outside)192.168.1.12
  172.16.0.2 netmask 255.255.255.255
static(dmz2,outside)192.168.2.10
  172.16.1.1 netmask 255.255.255.255
```

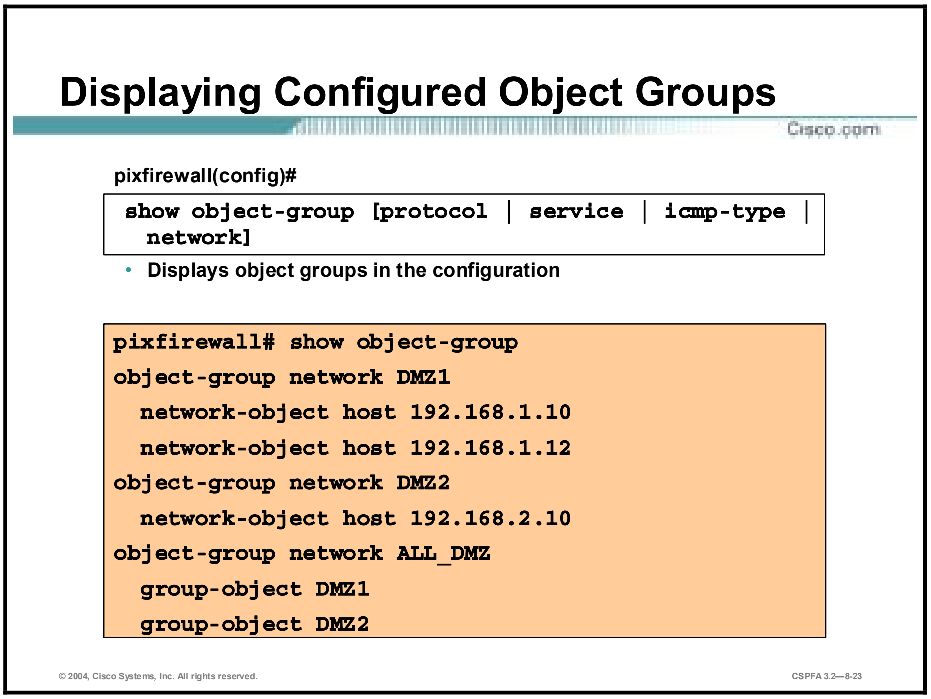
```
chicago(config)# show object-group
object-group network REMOTES
     network-object host 172.30.0.50
     network-object host 172.30.0.51
object-group network DMZ1
     network-object host 192.168.1.10
     network-object host 192.168.1.12
object-group network DMZ2
     network-object host 192.168.2.10
object-group network ALL_DMZ
     group-object DMZ1
     group-object DMZ2
 object-group service BASIC
     port-object eq http
     port-object eq smtp
```

```
pixfirewall(config)# access-list
  aclout permit tcp object-group
  REMOTES object-group ALL_DMZ
  object-group BASIC
```

CSPFA 3.2—8-22

In the figure, object groups are configured so that one ACL entry enables remote hosts 172.30.0.50 and 172.30.0.51 to initiate HTTP and SMTP connections to DMZ1 and DMZ2 hosts in the ALL_DMZ nested group. With object grouping, one ACL entry is required. Without object grouping the following ACL entries would be required.

```
access-list outside permit tcp host 172.30.0.50 host 192.168.1.10 eq http

access-list outside permit tcp host 172.30.0.50 host 192.168.1.10 eq smtp

access-list outside permit tcp host 172.30.0.51 host 192.168.1.10 eq http

access-list outside permit tcp host 172.30.0.51 host 192.168.1.10 eq smtp

access-list outside permit tcp host 172.30.0.50 host 192.168.1.12 eq http

access-list outside permit tcp host 172.30.0.50 host 192.168.1.12 eq smtp

access-list outside permit tcp host 172.30.0.51 host 192.168.1.12 eq http

access-list outside permit tcp host 172.30.0.51 host 192.168.1.12 eq smtp

access-list outside permit tcp host 172.30.0.50 host 192.168.2.10 eq http

access-list outside permit tcp host 172.30.0.50 host 192.168.2.10 eq smtp

access-list outside permit tcp host 172.30.0.51 host 192.168.2.10 eq http

access-list outside permit tcp host 172.30.0.51 host 192.168.2.10 eq smtp
```



Displaying Configured Object Groups

Cisco.com

pixfirewall(config)#

```
show object-group [protocol | service | icmp-type |
  network]
```

- Displays object groups in the configuration

```
pixfirewall# show object-group
object-group network DMZ1
  network-object host 192.168.1.10
  network-object host 192.168.1.12
object-group network DMZ2
  network-object host 192.168.2.10
object-group network ALL_DMZ
  group-object DMZ1
  group-object DMZ2
```

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—8-23

Use the **show object-group** command to display a list of the currently configured object groups. The PIX Firewall displays defined object groups by their grp_id when the **show object-group id** ***grp_id*** command form is entered, and by group type when the **show object-group** command is entered with the protocol, service, icmp-type, or network option. When you enter **show object-group** without a parameter, all defined object groups are shown.

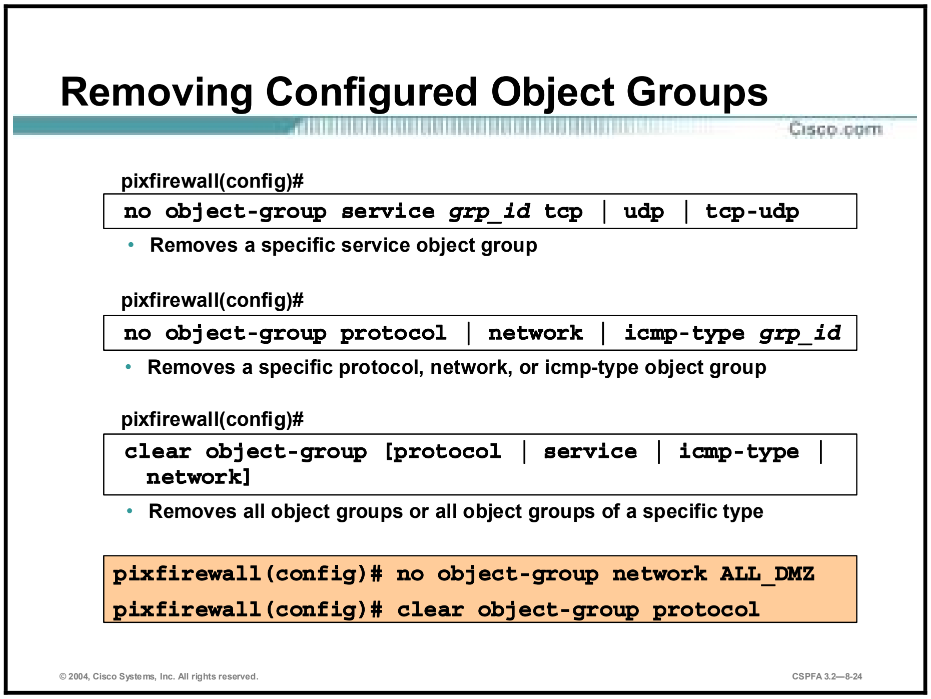
The syntax for the **show object-group** command is as follows:

show object-group [protocol | service | icmp-type | network]

show object-group id ***grp_id***

protocol	Instructs the PIX Firewall to display protocol object groups.
service	Instructs the PIX Firewall to display service object groups.
icmp-type	Instructs the PIX Firewall to display icmp-type object groups.
network	Instructs the PIX Firewall to display network object groups.
id	Instructs the PIX Firewall to display the object group specified by object_grp_id.
grp_id	The name of a previously defined object group.

Note The **show config** and **write** commands display ACLs as configured with the object group names. However, the **show access-list** command displays the ACEs expanded into individual statements within their object groupings.



Removing Configured Object Groups

pixfirewall(config)#

```
no object-group service grp_id tcp | udp | tcp-udp
```

- Removes a specific service object group

pixfirewall(config)#

```
no object-group protocol | network | icmp-type grp_id
```

- Removes a specific protocol, network, or icmp-type object group

pixfirewall(config)#

```
clear object-group [protocol | service | icmp-type |
  network]
```

- Removes all object groups or all object groups of a specific type

```
pixfirewall(config)# no object-group network ALL_DMZ
pixfirewall(config)# clear object-group protocol
```

CSPFA 3.2—8-24

Any object group command can be removed with its no form. Use the **no object-group** command to remove a specific object group.

The syntax for the **no object-group** commands is as follows:

no object-group service ***grp_id*** {**tcp** | **udp** | **tcp-udp**}

no object-group { **protocol** | **network** | **icmp-type**} ***grp_id***

service	Instructs the PIX Firewall to remove service object groups.
grp_id	The name of a previously defined object group that you want to remove.
tcp	Specifies that the service object group contains ports that are used for TCP only.
udp	Specifies that the service object group contains ports that are used for UDP only.
tcp-udp	Specifies that the service object group contains ports that are used for both TCP and UDP.
protocol	Instructs the PIX Firewall to remove protocol object groups.
network	Instructs the PIX Firewall to remove network object groups.
icmp-type	Instructs the PIX Firewall to remove icmp-type object groups.

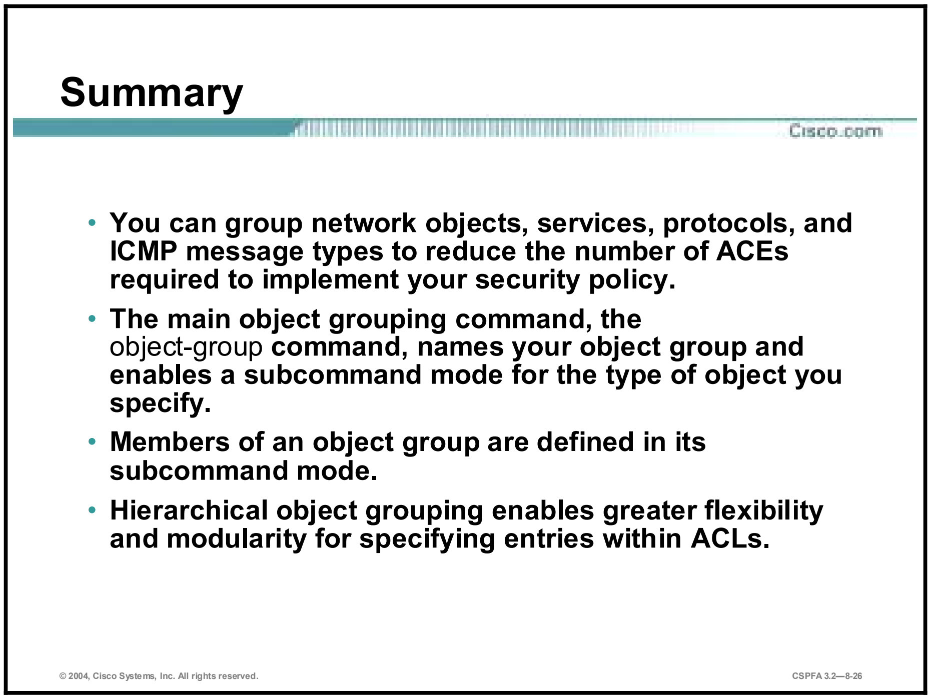
The **clear object-group** command can be used to remove all object groups or all object groups of a specific type. When entered without a parameter, the **clear object-group** command removes all defined object groups that are not being used in a command. Using the protocol, service, icmp-type, or network parameter removes all defined object groups that are not being used in a command for that group type only. An object group cannot be removed if it is part of an active ACL and its removal would result in the ACL becoming incomplete or invalid.

The syntax for the **clear object-group** command is as follows:

clear object-group *[grp_type]*

Summary

This topic summarizes what you have learned in this lesson.



Summary

- You can group network objects, services, protocols, and ICMP message types to reduce the number of ACEs required to implement your security policy.
- The main object grouping command, the object-group command, names your object group and enables a subcommand mode for the type of object you specify.
- Members of an object group are defined in its subcommand mode.
- Hierarchical object grouping enables greater flexibility and modularity for specifying entries within ACLs.

CSPFA 3.2—8-26

Lab Exercise—Configure Object Groups

Complete the following lab exercise to practice what you learned in this lesson.

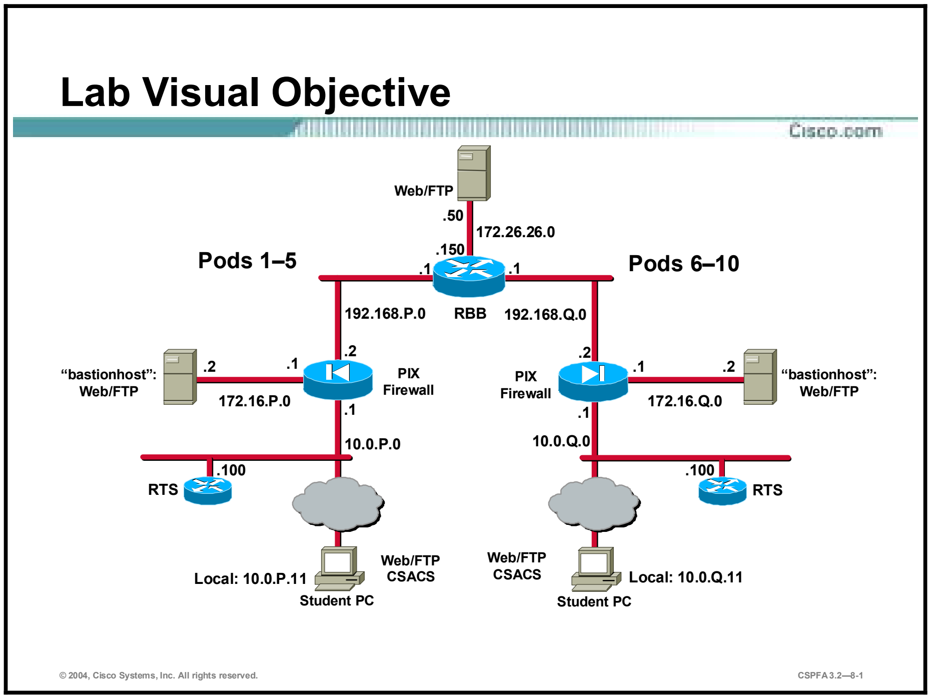
Objectives

In this lab exercise you will complete the following tasks:

- Configure a service object group.
- Configure an ICMP-Type object group.
- Configure a nested server object group.
- Configure an inbound ACL with object groups.
- Configure web and ICMP access to the inside host.
- Test and verify the inbound ACL.

Visual Objectives

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—8-1

Task 1—Configure a Service Object Group

Complete the following steps to configure a service group containing HTTP and FTP:

Step 1 Create a TCP service group named MYSERVICES. This step assigns a name to the group and enables the service object subcommand mode:

```
pixP(config)# object-group service MYSERVICES tcp
```

Step 2 Add HTTP and FTP to the service object group:

```
pixP(config-service)# port-object eq http
pixP(config-service)# port-object eq ftp
```

Step 3 Return to configuration mode:

```
pixP(config-network)# exit
```

Step 4 Verify that the object group has been configured successfully:

```
pixP(config)#show object-group
object-group service MYSERVICES tcp
  port-object eq www
  port-object eq ftp
```

Task 2—Configure an ICMP-Type Object Group

Complete the following steps to configure an ICMP-Type group:

Step 1 Create an ICMP-Type object group named PING to assign a name to the group and enable the ICMP-Type subcommand mode:

```
pixP(config)# object-group icmp-type PING
```

Step 2 Add ICMP echo to the ICMP-Type object group:

```
pixP(config-icmp-type)# icmp-object echo
```

Step 3 Add ICMP echo-reply to the ICMP-Type object group:

```
pixP(config-icmp-type)# icmp-object echo-reply
```

Step 4 Add ICMP unreachable messages to the ICMP-Type object group:

```
pixP(config-icmp-type)# icmp-object unreachable
```

Step 5 Return to configuration mode:

```
pixP(config-icmp-type)# exit
```

Step 6 Verify that the object group has been configured successfully:

```
pixP(config)# show object-group
object-group service MYSERVICES tcp
  port-object eq www
  port-object eq ftp
object-group icmp-type PING
  icmp-object echo
  icmp-object echo-reply
  icmp-object unreachable
```

Task 3—Configure a Nested Server Object Group

Complete the following steps to nest an object group within another object group:

Step 1 Create a network object group named FTPSERVERS:

```
pixP(config)# object-group network FTPSERVERS
```

Step 2 Add your bastion host to the object group:

```
pixP(config-network)# network-object host 192.168.P.11
```

(where P = pod number)

Step 3 Return to configuration mode:

```
pixP(config-network)# exit
```

Step 4 Create a network object group named ALLSERVERS:

```
pixP(config)# object-group network ALLSERVERS
```

Step 5 Nest the FTPSERVERS group within the ALLSERVERS group:

```
pixP(config-network)# group-object FTPSERVERS
```

Step 6 Add the following servers to the ALLSERVERS group:

- 192.168.P.10
- 192.168.P.6
- 192.168.P.7

```
pixP(config-network)# network-object host 192.168.P.10
pixP(config-network)# network-object host 192.168.P.6
pixP(config-network)# network-object host 192.168.P.7
```

(where P = pod number)

Step 7 Verify that the object group has been configured successfully:

```
pixP(config-network)# show object-group
object-group service MYSERVICES tcp
  port-object eq www
  port-object eq ftp
object-group icmp-type PING
  icmp-object echo
  icmp-object echo-reply
  icmp-object unreachable
object-group network FTPSERVERS
  network-object host 192.168.P.11
object-group network ALLSERVERS
  group-object FTPSERVERS
  network-object host 192.168.P.10
  network-object host 192.168.P.6
  network-object host 192.168.P.7
```

(where P = pod number)

Task 4—Configure an Inbound ACL with Object Groups

Complete the following steps to configure an inbound ACL to do the following:

- Allow inbound web traffic from a peer pod’s network to your bastion host.
- Allow inbound ftp traffic from a peer pod’s internal host to your bastion host.

Step 1 Remove the ACLs you configured in the previous lab exercise. Notice that entering the **clear access-list** command takes you back to configuration mode.

```
pixP(config-network)# clear access-list
```

Step 2 Verify that all ACLs have been removed:

```
pixP(config)# show access-list
```

Step 3 Test web access to your peer pod’s bastion host by completing the following sub-steps. You should be unable to access your peer’s bastion host.

1. Open a web browser on your student PC.

2. Use the web browser to access the bastion host of your peer pod group by entering **http://192.168.Q.11**.

(where Q = peer pod number)

Step 4 Test web access to the inside host of your peer's pod by completing the following sub-steps. You should be unable to access your peer's inside host.

1. Open a web browser on your student PC.

2. Use the web browser to access the inside host of your peer pod group by entering **http://192.168.Q.10**.

(where Q = peer pod number)

Step 5 From your FTP client, test FTP access to your peer pod's bastion host. You should be unable to access your peer's bastion host via FTP.

```
Start>Run>ftp 192.168.Q.11
```

(where Q = peer pod number)

Step 6 Use the MYSERVICES group to create an ACL permitting inbound web and FTP access to the bastion host:

```
pixP(config)# access-list ACLIN permit tcp 192.168.Q.0 255.255.255.0
object-group FTPSERVERS object-group MYSERVICES
```

(where Q = peer pod number)

Step 7 Bind the ACL to the outside interface:

```
pixP(config)# access-group ACLIN in interface outside
```

Step 8 Have a peer ping your inside host. The ping should fail.

```
C:\>ping 192.168.Q.10
Pinging 192.168.Q.10 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
```

(where Q = peer pod number)

Step 9 Have a peer ping your bastion host. The ping should fail.

```
C:\>ping 192.168.Q.11
Pinging 192.168.Q.11 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
```

(where Q = peer pod number)

Step 10 Test web access to your peer pod's bastion host by completing the following sub-steps. You should now be able to access your peer's bastion host.

1. Open a web browser on your student PC.

2. Use the web browser to access the bastion host of your peer pod group by entering **http://192.168.Q.11**.

(where Q = peer pod number)

Step 11 Test web access to your peer pod's inside host by completing the following sub-steps. You should still be unable to access your peer pod's inside host.

1. Open a web browser on the client PC.

2. Use the web browser to access the inside host of your peer pod group by entering **http://192.168.Q.10**.

(where Q = peer pod number)

Step 12 From your FTP client, test FTP access to your peer pod's bastion host. You should now be able to access your peer's bastion host via FTP.

```
Start>Run>ftp 192.168.Q.11
```

(where Q = peer pod number)

Step 13 From your FTP client, test FTP access to your peer pod's inside hosts. You should still be unable to access your peer's inside host via FTP.

```
Start>Run>ftp 192.168.Q.10
```

(where Q = peer pod number)

Task 5—Configure Web and ICMP Access to the Inside Host

Complete the following steps to configure ACLIN to do the following:

- Permit inbound web and ICMP traffic to all hosts behind the PIX Firewall.
- Deny all other traffic from the Internet.

Step 1 Use a network hosts group to add an ACL entry permitting web traffic to all hosts behind the PIX Firewall:

```
pixP(config)# access-list ACLIN permit tcp any object-group ALLSERVERS
eq www
```

Step 2 Permit ICMP traffic to all hosts behind the PIX Firewall:

```
pixP(config)# access-list ACLIN permit icmp any any object-group PING
```

Step 3 Deny all other traffic from the Internet:

```
pixP(config)# access-list ACLIN deny ip any any
```

Step 4 Bind the ACL to the outside interface:

```
pixP(config)# access-group ACLIN in interface outside
```

Step 5 Create an ACL to permit echo replies to the inside host from your bastion host:

```
pixP(config)# access-list ACLDMZ permit icmp any any object-group PING
```

Step 6 Bind the ACL to the DMZ interface:

```
pixP(config)# access-group ACLDMZ in interface dmz
```

Step 7 Display the ACLs and observe the hit counts:

```
pixP(config)# show access-list
access-list ACLIN; 10 elements
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 object-
group FTPSERVERS object-group MYSERVICES
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq www(hitcnt=2)
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq ftp(hitcnt=1)
access-list ACLIN line 2 permit tcp any object-group ALLSERVERS eq www
access-list ACLIN line 2 permit tcp any host 192.168.P.11 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.10 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.6 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.7 eq www
(hitcnt=0)
access-list ACLIN line 3 permit icmp any any object-group PING
access-list ACLIN line 3 permit icmp any any echo (hitcnt=0)
access-list ACLIN line 3 permit icmp any any echo-reply (hitcnt=0)
access-list ACLIN line 3 permit icmp any any unreachable (hitcnt=0)
access-list ACLIN line 4 deny ip any any (hitcnt=0)
access-list ACLDMZ; 3 elements
access-list ACLDMZ line 1 permit icmp any any object-group PING
access-list ACLDMZ line 1 permit icmp any any echo (hitcnt=0)
access-list ACLDMZ line 1 permit icmp any any echo-reply (hitcnt=0)
access-list ACLDMZ line 1 permit icmp any any unreachable (hitcnt=0)
```

(where P=pod number, and Q = peer pod number)

Task 6—Test and Verify the Inbound ACL

Complete the following steps to test your inbound ACL:

Step 1 Have a peer inside host ping your inside host:

```
C:\>ping 192.168.Q.10
Pinging 192.168.Q.10 with 32 bytes of data:
Reply from 192.168.Q.10: bytes=32 time<10ms TTL=128
Reply from 192.168.Q.10: bytes=32 time<10ms TTL=128
Reply from 192.168.Q.10: bytes=32 time<10ms TTL=128
Reply from 192.168.Q.10: bytes=32 time<10ms TTL=128
```

(where Q = peer pod number)

Step 2 Have a peer inside host ping your bastion host:

```
C:\>ping 192.168.Q.11
Pinging 192.168.Q.11 with 32 bytes of data:
Reply from 192.168.Q.11: bytes=32 time<10ms TTL=128
Reply from 192.168.Q.11: bytes=32 time<10ms TTL=128
Reply from 192.168.Q.11: bytes=32 time<10ms TTL=128
Reply from 192.168.Q.11: bytes=32 time<10ms TTL=128
```

(where Q = peer pod number)

Step 3 From your student PC, ping your bastion host:

```
C:\>ping 172.16.P.2
Pinging 172.16.P.2 with 32 bytes of data:
Reply from 172.16.P.2: bytes=32 time<10ms TTL=128
Reply from 172.16.P.2: bytes=32 time<10ms TTL=128
Reply from 172.16.P.2: bytes=32 time<10ms TTL=128
Reply from 172.16.P.2: bytes=32 time<10ms TTL=128
```

(where P = pod number)

Step 4 From your student PC, ping the super server:

```
C:\>ping 172.26.26.50
Pinging 172.26.26.50 with 32 bytes of data:
Reply from 172.26.26.50: bytes=32 time<10ms TTL=128
Reply from 172.26.26.50: bytes=32 time<10ms TTL=128
Reply from 172.26.26.50: bytes=32 time<10ms TTL=128
Reply from 172.26.26.50: bytes=32 time<10ms TTL=128
```

Step 5 Test web access to your peer pod's bastion host by completing the following sub-steps. You should be able to access your peer's bastion host.

1. Open a web browser on your student PC.

2. Use the web browser to access the bastion host of your peer pod group by entering **http://192.168.Q.11**.
(where Q = peer pod number)

Step 6 Test web access to your peer pod's inside host by completing the following sub-steps. You should now be able to access your peer pod's inside host.

1. Open a web browser on the client PC.

2. Use the web browser to access the inside host of your peer pod group by entering **http://192.168.Q.10**.

(where Q = peer pod number)

Step 7 From your FTP client, test FTP access to your peer pod's bastion host. You should be able to access your peer's bastion host via FTP.

```
Start>Run>ftp 192.168.Q.11
```

(where Q = peer pod number)

Step 8 From your FTP client, test FTP access to your peer pod's inside host. You should still be unable to access your peer's inside host via FTP.

```
Start>Run>ftp 192.168.Q.10
```

(where Q = peer pod number)

Step 9 Display the ACLs again and observe the hit counts:

```
pixP(config)# show access-list

access-list ACLIN; 10 elements

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 object-
group FTPSERVERS object-group MYSERVICES

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq www (hitcnt=4)

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq ftp (hitcnt=2)

access-list ACLIN line 2 permit tcp any object-group ALLSERVERS eq www

access-list ACLIN line 2 permit tcp any host 192.168.P.11 eq www
(hitcnt=0)

access-list ACLIN line 2 permit tcp any host 192.168.P.10 eq www
(hitcnt=2)

access-list ACLIN line 2 permit tcp any host 192.168.P.6 eq www
(hitcnt=0)

access-list ACLIN line 2 permit tcp any host 192.168.P.7 eq www
(hitcnt=0)

access-list ACLIN line 3 permit icmp any any object-group PING

access-list ACLIN line 3 permit icmp any any echo (hitcnt=12)

access-list ACLIN line 3 permit icmp any any echo-reply (hitcnt=4)

access-list ACLIN line 3 permit icmp any any unreachable (hitcnt=0)

access-list ACLIN line 4 deny ip any any (hitcnt=3)

access-list ACLDMZ; 3 elements

access-list ACLDMZ line 1 permit icmp any any object-group PING

access-list ACLDMZ line 1 permit icmp any any echo (hitcnt=0)

access-list ACLDMZ line 1 permit icmp any any echo-reply (hitcnt=8)

access-list ACLDMZ line 1 permit icmp any any unreachable (hitcnt=0)
```

9

Advanced Protocol Handling

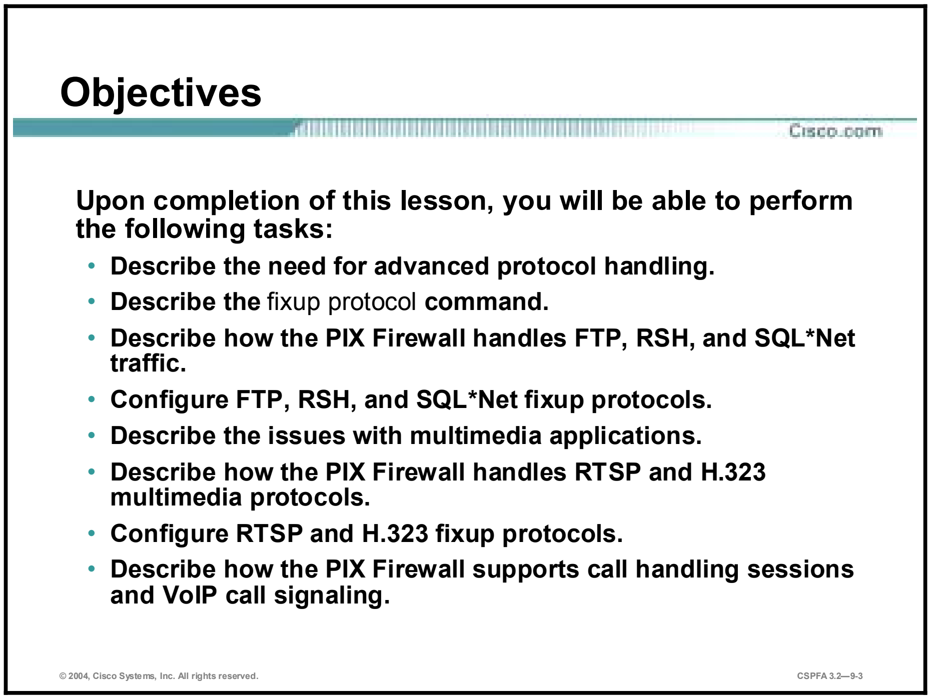
Overview

This lesson includes the following topics:

- Objectives
- Advanced protocols
- Multimedia support
- Summary
- Lab exercise

Objectives

This topic lists the lesson’s objectives.



Objectives

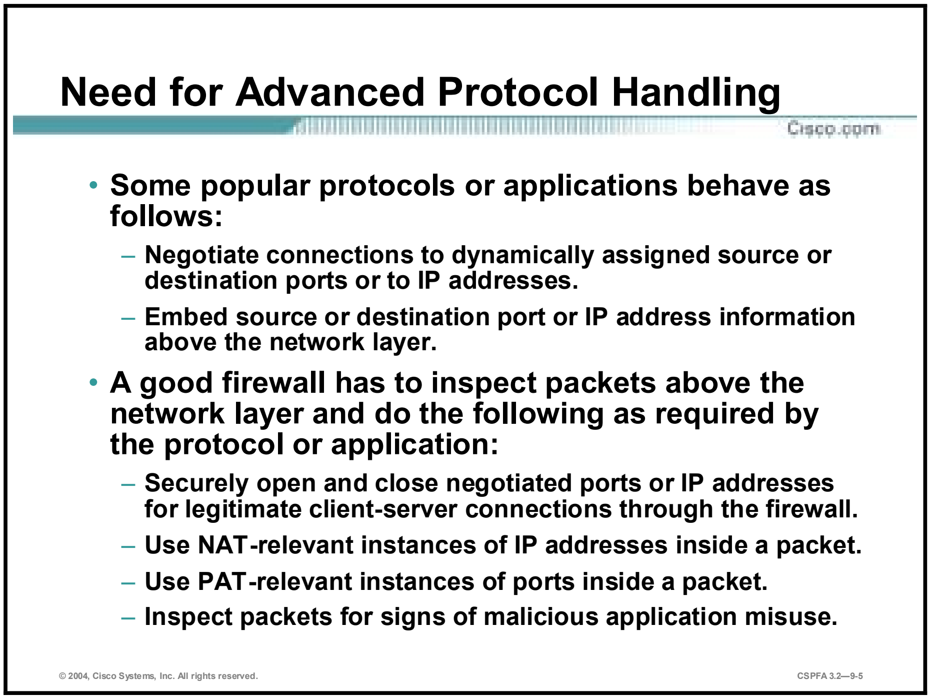
Upon completion of this lesson, you will be able to perform the following tasks:

- **Describe the need for advanced protocol handling.**
- **Describe the** fixup protocol **command.**
- **Describe how the PIX Firewall handles FTP, RSH, and SQL*Net traffic.**
- **Configure FTP, RSH, and SQL*Net fixup protocols.**
- **Describe the issues with multimedia applications.**
- **Describe how the PIX Firewall handles RTSP and H.323 multimedia protocols.**
- **Configure RTSP and H.323 fixup protocols.**
- **Describe how the PIX Firewall supports call handling sessions and VoIP call signaling.**

CSPFA 3.2—9-3

Advanced Protocols

This topic discusses the configuration and handling of the FTP, remote shell (RSH), and SQL protocols.



Need for Advanced Protocol Handling

- Some popular protocols or applications behave as follows:
  - Negotiate connections to dynamically assigned source or destination ports or to IP addresses.
  - Embed source or destination port or IP address information above the network layer.
- A good firewall has to inspect packets above the network layer and do the following as required by the protocol or application:
  - Securely open and close negotiated ports or IP addresses for legitimate client-server connections through the firewall.
  - Use NAT-relevant instances of IP addresses inside a packet.
  - Use PAT-relevant instances of ports inside a packet.
  - Inspect packets for signs of malicious application misuse.

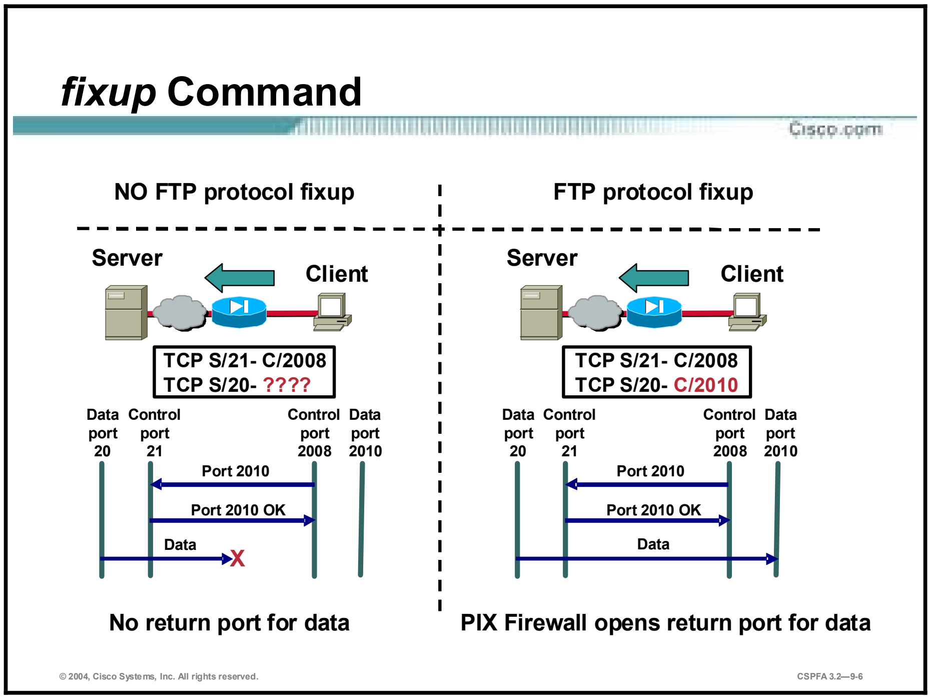
CSPFA 3.2—9-5

Today, corporations that use the Internet for business transactions want to keep their internal networks secure from potential threats. These corporations usually implement firewalls as part of their network defense strategy. Firewalls can help protect their networks; some firewalls may cause problems as well. For example, applications such as FTP, HTTP, multimedia, and SQL*Net require their communications protocols to dynamically negotiate source or destination ports or IP addresses. Some firewalls cannot participate in these dynamic protocol negotiations, resulting in either the complete blockage of these corporate services or the need to preconfigure static “holes” in the firewall to allow these services.

A good firewall has to inspect packets above the network layer and do the following as required by the protocol or application:

- Securely open and close negotiated ports or IP addresses for legitimate client-server connections through the firewall.
- Use Network Address Translation (NAT)-relevant instances of an IP address inside a packet.
- Use Port Address Translation (PAT)-relevant instances of ports inside a packet.
- Inspect packets for signs of malicious application misuse.

You can configure the Cisco PIX Firewall to inspect the required protocols, or applications, and permit them to traverse the PIX Firewall with dynamic, stateful adjustments to the firewall’s security policy. This enables the corporation’s networks to remain secure while still being able to continue conducting day-to-day business.



fixup Command

Cisco.com

NO FTP protocol fixup

Server

Client

TCP S/21- C/2008
TCP S/20- ????

Data port 20

Control port 21

Control port 2008

Data port 2010

Port 2010

Port 2010 OK

Data

X

No return port for data

FTP protocol fixup

Server

Client

TCP S/21- C/2008
TCP S/20- C/2010

Data port 20

Control port 21

Control port 2008

Data port 2010

Port 2010

Port 2010 OK

Data

PIX Firewall opens return port for data

CSPFA 3.2—9-6

The Adaptive Security Algorithm (ASA), used by the PIX Firewall for stateful application inspection, ensures the secure use of applications and services. Some applications require special handling by the PIX Firewall application inspection function. Applications that require special application inspection functions are those that embed IP addressing information in the user data packet or that open secondary channels on dynamically assigned ports.

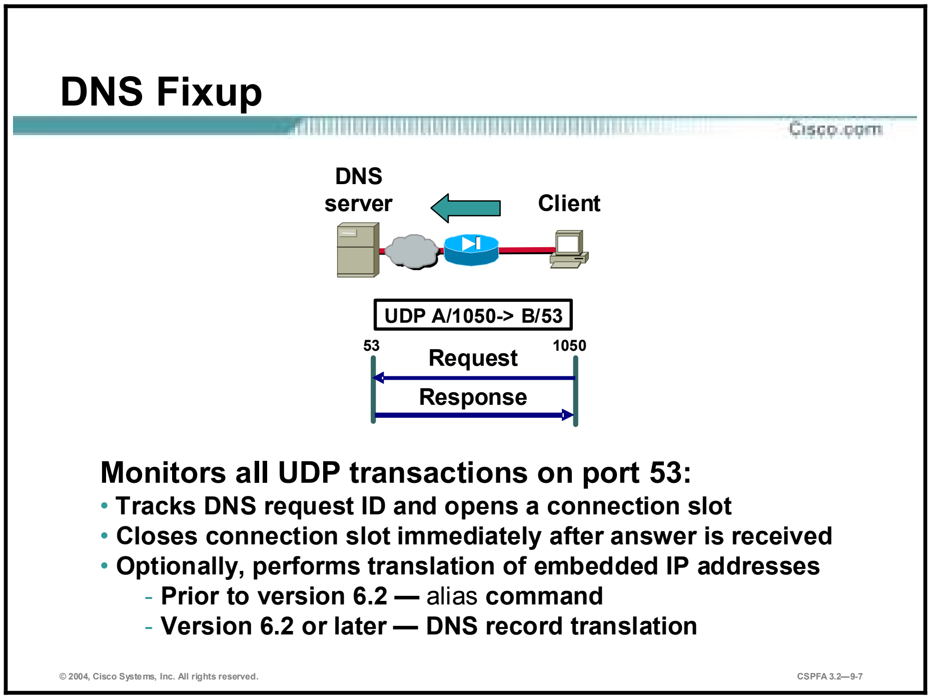
The application inspection function works with NAT to help identify the location of embedded addressing information. This allows NAT to translate these embedded addresses and to update any checksum or other fields that are affected by the translation.

The application inspection function also monitors sessions to determine the port numbers for secondary channels. Many protocols open secondary TCP or UDP ports. The initial session on a well-known port is used to negotiate dynamically assigned port numbers. The application inspection function monitors these sessions, identifies the dynamic port assignments, and permits data exchange on these ports for the duration of the specific session. In the example in the figure, the FTP client is shown in active mode opening a control channel between its port 2008 and the FTP server port 21. When data is to be exchanged, the FTP client alerts the FTP server through the control channel that it expects the data to be delivered back from FTP server port 20 to its port 2010. If FTP fixup is not enabled, the return data from FTP server port 20 to FTP client port 2010 is blocked by the PIX Firewall. With FTP fixup enabled, however, the PIX Firewall inspects the FTP control channel to recognize that the data channel will be established to the new FTP client port 2010 and temporarily creates an opening for the data channel traffic for the life of the session.

The **fixup** command enables you to enable or disable the use of a service or protocol throughout the PIX Firewall. The ports that you specify are those that the PIX Firewall listens on for each respective service. You can change the port value for each service except RSH.

For the most part, there is no reason to change these port numbers. But in special circumstances you may have a service listening on a nonstandard port number. For example, you could have an FTP server listening on port 5000. The PIX Firewall would not recognize that port 5000 is being used for FTP and will block the returned FTP data connection from the server. This problem could be resolved by adding port 5000 to the **fixup protocol** command: **fixup protocol** ***ftp 5000***. This command enables the PIX Firewall to recognize that connections to port 5000 should be treated in the same manner as connections to port 20.

By default, the PIX Firewall is configured to fix up the following protocols: FTP, Simple Mail Transfer Protocol (SMTP), HTTP, RSH, Real-Time Streaming Protocol (RTSP), SQL*Net, H.323, Internet Locator Service (ILS), Skinny Client Control Protocol (SCCP), Session Initiation Protocol (SIP), Computer Telephony Interface Quick Buffer Encoding (CTIQBE), and Media Gateway Control Protocol (MGCP).



DNS Fixup

DNS server

Client

UDP A/1050-> B/53

53

1050

Request

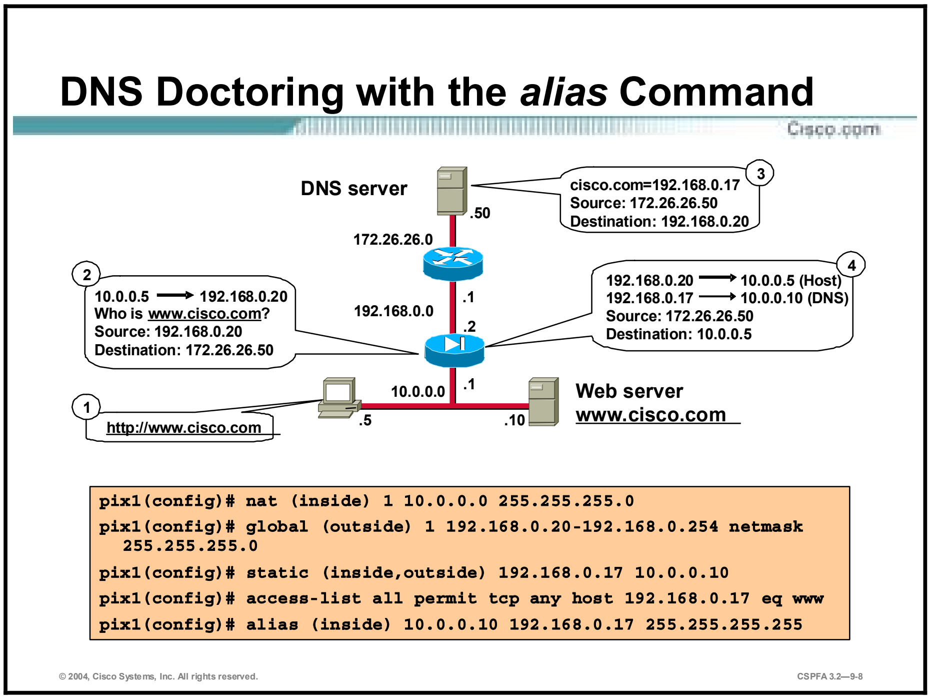
Response

Monitors all UDP transactions on port 53:

- **Tracks DNS request ID and opens a connection slot**
- **Closes connection slot immediately after answer is received**
- **Optionally, performs translation of embedded IP addresses**
  - **Prior to version 6.2 —** alias **command**
  - **Version 6.2 or later — DNS record translation**

CSPFA 3.2—9-7

The PIX Firewall knows that Domain Name System (DNS) queries are a one-request, one-answer conversation, so the connection slot is released immediately after a DNS answer is received. When the DNS A record is returned, the PIX Firewall applies address translation not only to the destination address, but also to the embedded IP address of the web server. This address is contained in the user data portion of the DNS reply packet. As a result, a web client on the inside network gets the address it needs to connect to the web server on the inside network. Prior to PIX Firewall release 6.2, the PIX Firewall translated the embedded IP address with the help of the **alias** command. In PIX Firewall release 6.2, PIX Firewall has full support for NAT of embedded IP addresses within a DNS response packet. There is more on DNS doctoring later in this lesson.



DNS Doctoring with the *alias* Command

```
pix1(config)# nat (inside) 1 10.0.0.0 255.255.255.0
pix1(config)# global (outside) 1 192.168.0.20-192.168.0.254 netmask
  255.255.255.0
pix1(config)# static (inside,outside) 192.168.0.17 10.0.0.10
pix1(config)# access-list all permit tcp any host 192.168.0.17 eq www
pix1(config)# alias (inside) 10.0.0.10 192.168.0.17 255.255.255.255
```

CSPFA 3.2—9-8

The **alias** command translates one IP address into another. One of the main uses of this command is DNS doctoring, which is translating the IP address embedded in a DNS response.

The example in the figure reveals how DNS doctoring is helpful and shows how to use the **alias** command for this purpose. The internal web server in the example in the figure, the web server shown for www.cisco.com, has an IP address of 10.0.0.10. Hosts on the 10.0.0.0 network seeking to access the web server by its domain name must resolve the name by using the DNS server on the outside interface of the PIX Firewall. This presents a problem that can be solved by using the **alias** command. To gain a better understanding of this problem and its solution, follow this sequence of events as an inside host attempts to access the web server by its hostname:

Step 1 A user at host 10.0.0.5 attempts to access www.cisco.com via the user's web browser.

Step 2 The only way the IP address of www.cisco.com can be located is through the DNS server on the outside of the PIX Firewall, so the host sends the packet to the PIX Firewall.

Step 3 Because NAT is configured for the inside network, the PIX Firewall translates the source address from 10.0.0.5 to 192.168.0.20 and forwards the message requesting the IP address of www.cisco.com to the DNS server.

Step 4 The DNS server locates its A record for www.cisco.com and sends its DNS reply to 192.168.0.20. The reply contains the response, www.cisco.com = 192.168.0.17.

Step 5 When the PIX Firewall receives the DNS response, the NAT function translates 192.168.0.20 back to 10.0.0.5, and the alias function translates the DNS reply message address of 192.168.0.17 to 10.0.0.10, and then forwards the packet to the originating host. From the doctored IP address with the DNS response, the originating host knows that the web server is on its own network and that it can access it directly. If the **alias** command did not notify the PIX Firewall to translate the address embedded in the response, the originating host would send the next packet back through the PIX Firewall, thinking that www.cisco.com resides on the outside.

The following steps were taken to provide this capability:

Step 1 A **static** command was configured to map the web server's internal address to 192.168.0.17, a globally routable IP address. This is necessary because the DNS server has a static mapping of www.cisco.com to 192.168.0.17. For the DNS server to resolve the name to the IP address, the IP address must always be the same. Furthermore, for users on the Internet to access the web server on the inside, a static and access control list (ACL) combination must be configured.

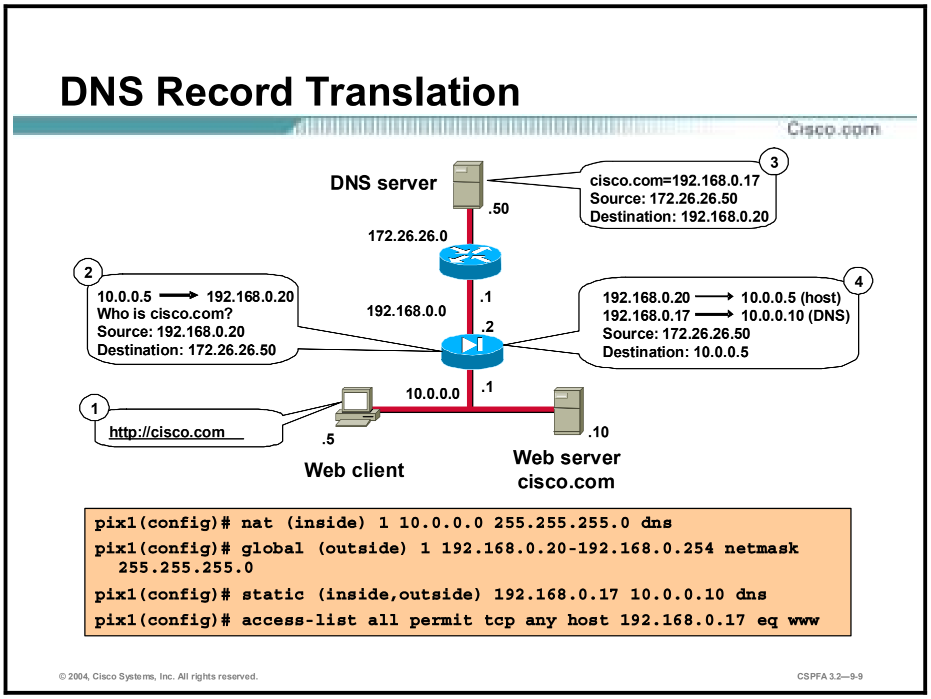
Step 2 An ACL was configured to give anyone on the Internet access to the web server on port 80.

Step 3 DNS doctoring was configured. This enables the PIX Firewall to watch for DNS replies that contain 192.168.0.17, and then replace the 192.168.0.17 address with 10.0.0.10.

When used for DNS doctoring, the **alias** command reads similar to the following:

"If a DNS packet destined for foreign_network_address is returned to the PIX Firewall, alter the DNS packet by changing the foreign network address to the dnat_network_address."

Note There must be an A record in the DNS zone file for the foreign network address in the **alias** command.



DNS Record Translation

1. http://cisco.com

2. 10.0.0.5 ⟶ 192.168.0.20
Who is cisco.com?
Source: 192.168.0.20
Destination: 172.26.26.50

3. cisco.com=192.168.0.17
Source: 172.26.26.50
Destination: 192.168.0.20

4. 192.168.0.20 ⟶ 10.0.0.5 (host)
192.168.0.17 ⟶ 10.0.0.10 (DNS)
Source: 172.26.26.50
Destination: 10.0.0.5

```
pix1(config)# nat (inside) 1 10.0.0.0 255.255.255.0 dns
pix1(config)# global (outside) 1 192.168.0.20-192.168.0.254 netmask
  255.255.255.0
pix1(config)# static (inside,outside) 192.168.0.17 10.0.0.10 dns
pix1(config)# access-list all permit tcp any host 192.168.0.17 eq www
```

CSPFA 3.2—9-9

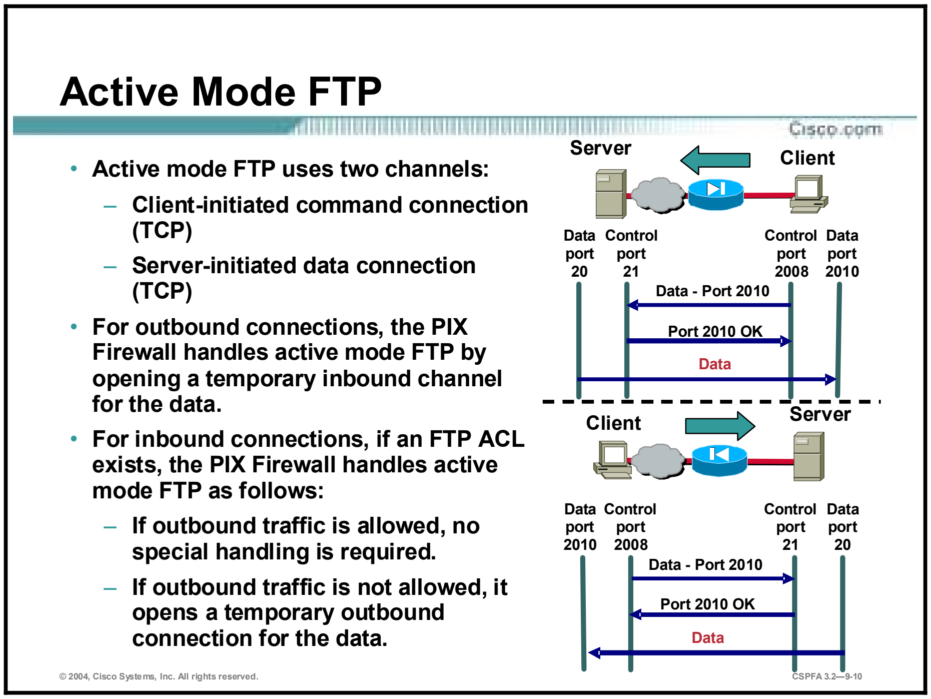
Cisco PIX Firewall Software Version 6.2 introduces full support for NAT of DNS messages originating from either inside (more secure) or outside (less secure) interfaces.

This means that if a client on an inside network requests DNS resolution of an inside address from a DNS server on an outside interface, the DNS A record is translated correctly. It is no longer necessary to use the **alias** command to perform DNS doctoring. The PIX Firewall translates the DNS A record on behalf of the **alias** command.

In the figure, the client on the inside network issues an HTTP request to server 10.0.0.10, using its hostname cisco.com. The PIX Firewall translates the web client's nonroutable source address in the IP header and forwards the request to the DNS server on its outside interface. When the DNS A record is returned, the PIX Firewall applies address translation not only to the destination address, but also to the embedded IP address of the web server. This address is contained in the user data portion of the DNS reply packet. As a result, the web client on the inside network gets the address it needs to connect to the web server on the inside network. NAT of DNS messages is implemented in both the **nat** and **static** commands:

nat [(*interface*)] *id address* [*netmask* [outside] [dns] [norandomseq] [timeout *hh:mm:ss*] [*connection_limit* [*em_limit*]]]

static [(*prenat-interface, postnat-interface*)] {*mapped_address*| interface} *real_address* [dns] [netmask *mask*] [norandomseq] [*connection_limit* [*em_limit*]]



Active Mode FTP

- Active mode FTP uses two channels:
  - Client-initiated command connection (TCP)
  - Server-initiated data connection (TCP)
- For outbound connections, the PIX Firewall handles active mode FTP by opening a temporary inbound channel for the data.
- For inbound connections, if an FTP ACL exists, the PIX Firewall handles active mode FTP as follows:
  - If outbound traffic is allowed, no special handling is required.
  - If outbound traffic is not allowed, it opens a temporary outbound connection for the data.

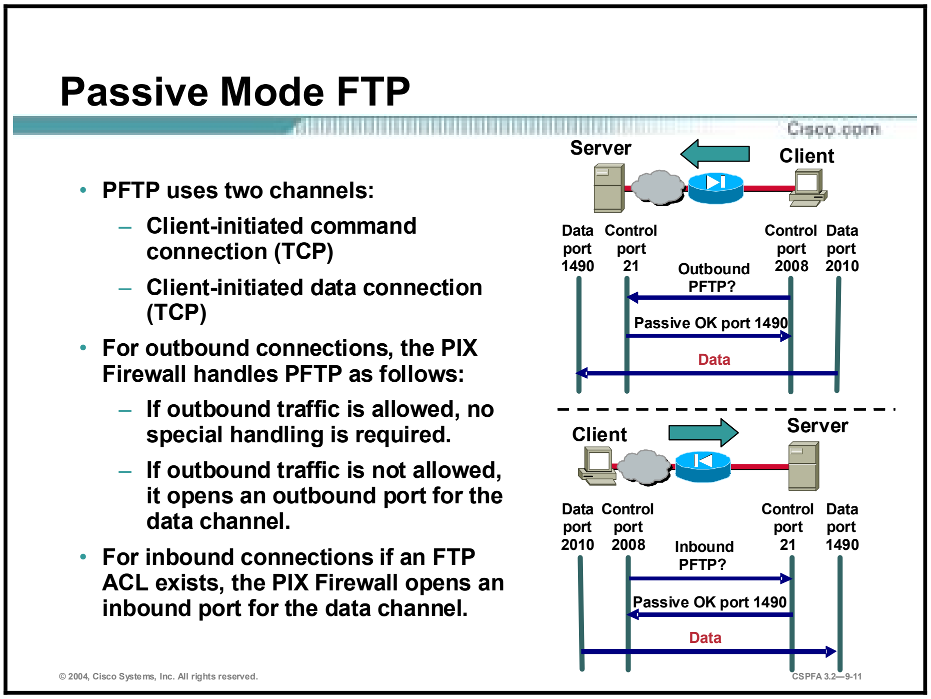
CSPFA 3.2—9-10

Active mode FTP uses two channels for communications. When a client starts an FTP connection, it opens a TCP channel from one of its high-order ports to port 21 on the server. This is referred to as the command channel. When the client requests data from the server, it tells the server to send the data to a given high-order port. The server acknowledges the request and initiates a connection from its own port 20 to the high-order port that the client requested. This is referred to as the data channel.

Because the server initiates the connection to the requested port on the client, it was difficult in the past to have firewalls allow this data channel to the client without permanently opening port 20 connections from outside servers to inside clients for outbound FTP connections. This created a potential vulnerability by exposing clients on the inside of the firewall. Protocol fixups have resolved this problem.

For FTP traffic, the PIX Firewall behaves in the following manner:

- Outbound connections—When the client requests data, the PIX Firewall opens a temporary inbound opening for the data channel from the server. This opening is torn down after the data is sent.
- Inbound connections:
  - If an access control list (ACL) exists allowing inbound connections to an FTP server, and if all outbound TCP traffic is implicitly allowed, no special handling is required because the server initiates the data channel from the inside.
  - If an ACL exists allowing inbound connections to an FTP server, and if all outbound TCP traffic is not implicitly allowed, the PIX Firewall opens a temporary opening for the data channel from the server. This opening is torn down after the data is sent.



Passive Mode FTP

- PFTP uses two channels:
  - Client-initiated command connection (TCP)
  - Client-initiated data connection (TCP)
- For outbound connections, the PIX Firewall handles PFTP as follows:
  - If outbound traffic is allowed, no special handling is required.
  - If outbound traffic is not allowed, it opens an outbound port for the data channel.
- For inbound connections if an FTP ACL exists, the PIX Firewall opens an inbound port for the data channel.

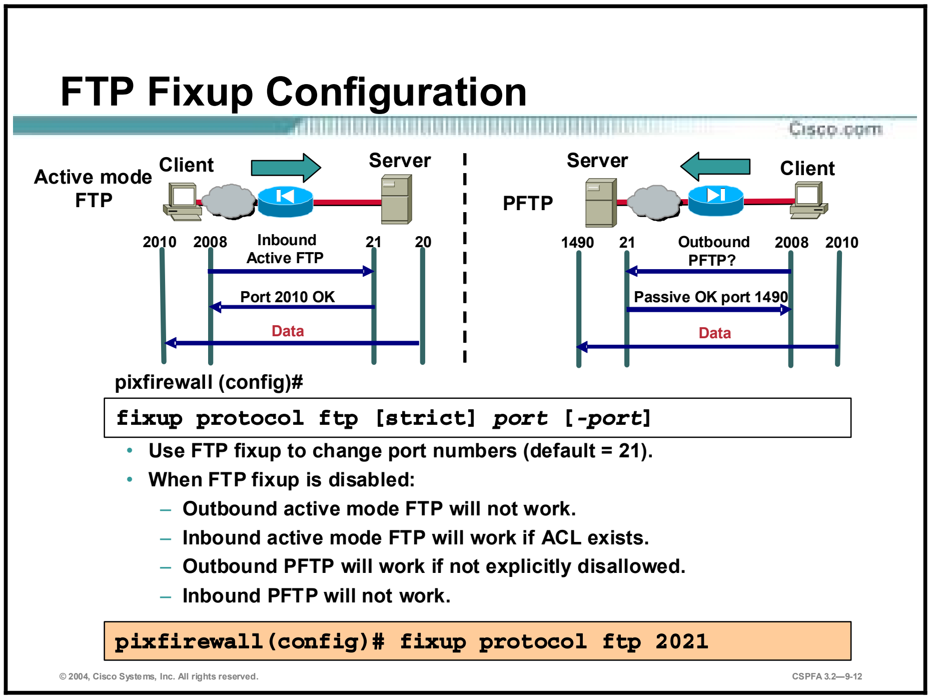
CSPFA 3.2—9-11

Passive mode FTP (PFTP) also uses two channels for communications. The command channel works the same as in a active mode FTP connection, but the data channel setup works differently. When the client requests data from the server, it asks the server if it accepts PFTP connections. If the server accepts PFTP connections, it sends the client a high-order port number to use for the data channel. The client then initiates the data connection from its own high-order port to the port that the server sent.

Because the client initiates both the command and data connections, early firewalls could easily support outbound connections without exposing inside clients to attack. Inbound connections, however, proved more of a challenge. The FTP protocol fixup resolved this issue.

For PFTP traffic, the PIX Firewall behaves in the following manner:

- Outbound connections:
  - If all outbound TCP traffic is implicitly allowed, no special handling is required because the client initiates both the command and data channels from the inside.
  - If all outbound TCP traffic is not implicitly allowed, the PIX Firewall opens a temporary opening for the data channel from the client. This opening is torn down after the data is sent.
- Inbound connections—If an ACL exists allowing inbound connections to a PFTP server, when the client requests data, the PIX Firewall opens a temporary inbound opening for the data channel initiated by client. This opening is torn down after the data is sent.



FTP Fixup Configuration

pixfirewall (config)#

```
fixup protocol ftp [strict] port [-port]
```

- Use FTP fixup to change port numbers (default = 21).
- When FTP fixup is disabled:
  - Outbound active mode FTP will not work.
  - Inbound active mode FTP will work if ACL exists.
  - Outbound PFTP will work if not explicitly disallowed.
  - Inbound PFTP will not work.

```
pixfirewall(config)# fixup protocol ftp 2021
```

CSPFA 3.2—9-12

By default, the PIX Firewall inspects port 21 connections for FTP traffic. If you have FTP servers using ports other than port 21, you need to use the **fixup protocol ftp** command to have the PIX Firewall inspect these other ports for FTP traffic.

The **fixup protocol ftp** command causes the PIX Firewall to do the following for FTP traffic on the indicated port:

- Perform NAT or PAT in packet payload.
- Dynamically create openings for FTP data connections.
- Log FTP commands (when Syslog is enabled).

The strict option to the **fixup protocol ftp** command prevents web browsers from sending embedded commands in FTP requests. Each FTP command must be acknowledged before a new command is allowed. Connections sending embedded commands are dropped.

Use the **no** form of the command to disable the inspection of traffic on the indicated port for FTP connections. If the **fixup protocol ftp** command is not enabled for a given port, then:

- Outbound active mode FTP will *not* work properly on that port.
- Outbound PFTP will work properly on that port as long as outbound traffic is not explicitly disallowed.
- Inbound active mode FTP will work properly on that port if an ACL to the inside server exists.
- Inbound PFTP will *not* work properly on that port.

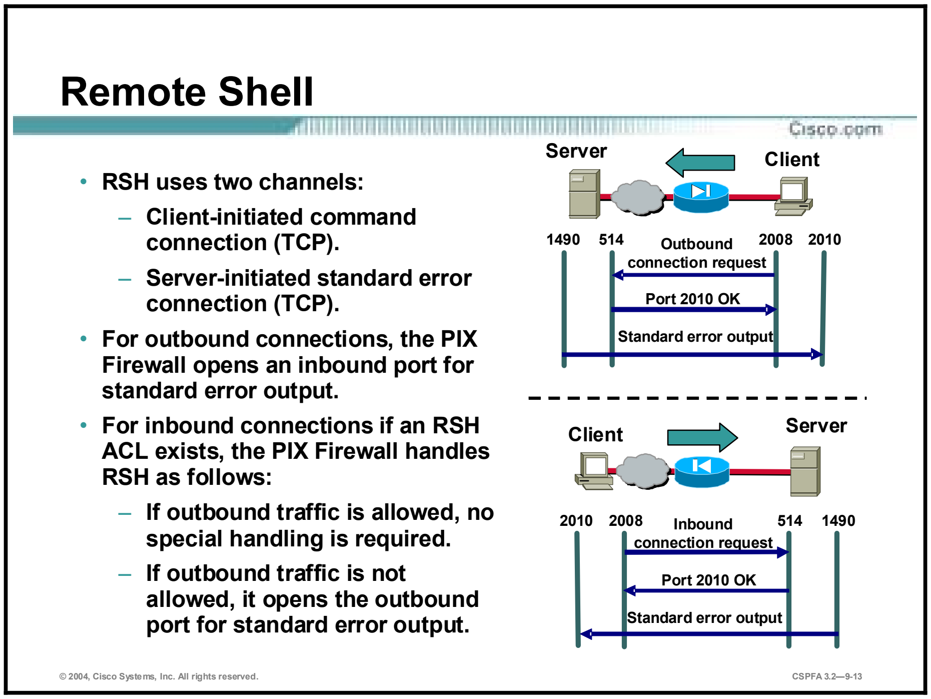
Using the **no fixup protocol ftp** command without any arguments causes the PIX Firewall to clear all previous **fixup protocol ftp** assignments and set port 21 back as the default.

The syntax for the **fixup protocol ftp** command is as follows:

fixup protocol ftp [strict] ***port*** **[-*port*]**

no fixup protocol ftp [*strict*] [*port* [-*port*]]

strict	Prevents web browsers from sending embedded commands in FTP requests.
***port* [-*port*]**	Single port or port range for which the PIX Firewall will inspect FTP connections.



Remote Shell

Cisco.com

- RSH uses two channels:
  - Client-initiated command connection (TCP).
  - Server-initiated standard error connection (TCP).
- For outbound connections, the PIX Firewall opens an inbound port for standard error output.
- For inbound connections if an RSH ACL exists, the PIX Firewall handles RSH as follows:
  - If outbound traffic is allowed, no special handling is required.
  - If outbound traffic is not allowed, it opens the outbound port for standard error output.

Server

Client

1490 514 Outbound connection request 2008 2010

Port 2010 OK

Standard error output

Client

Server

2010 2008 Inbound connection request 514 1490

Port 2010 OK

Standard error output

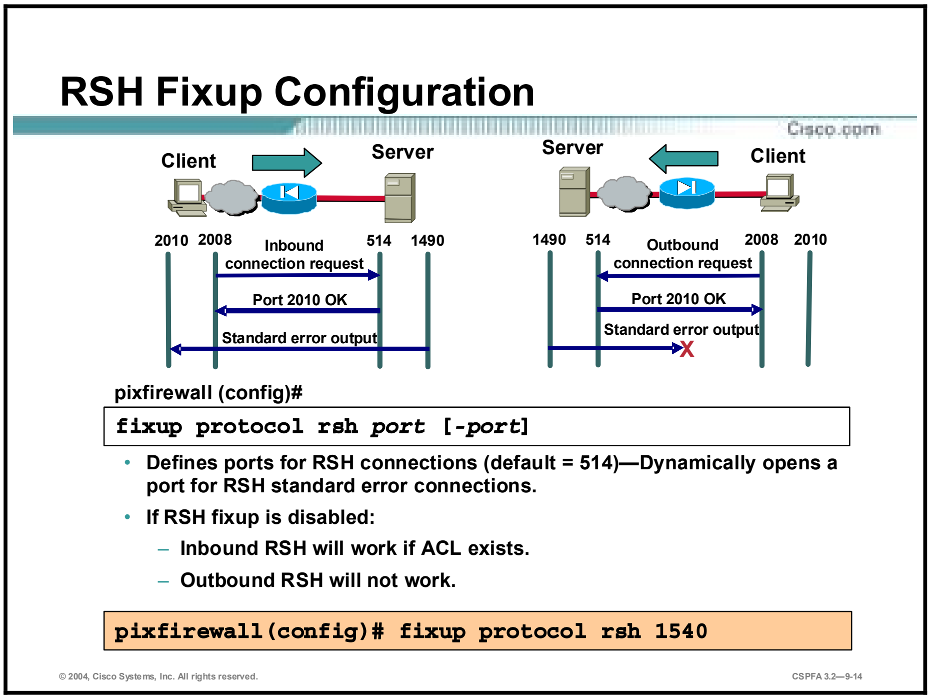
© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—9-13

Remote shell (RSH) uses two channels for communications. When a client first starts an RSH connection, it opens a TCP channel from one of its high-order ports to port 514 on the server. The server opens another channel for standard error output to the client.

For RSH traffic, the PIX Firewall behaves in the following manner:

- Outbound connections—When standard error messages are sent from the server, the PIX Firewall opens a temporary inbound opening for this channel. This opening is torn down when no longer needed.
- Inbound connections:
  - If an ACL exists allowing inbound connections to an RSH server, and if all outbound TCP traffic is implicitly allowed, no special handling is required because the server initiates the standard error channel from the inside.
  - If an ACL exists allowing inbound connections to an RSH server, and if all outbound TCP traffic is *not* implicitly allowed, the PIX Firewall opens a temporary opening for the standard error channel from the server. This opening is torn down after the messages are sent.



RSH Fixup Configuration

pixfirewall (config)#

```
fixup protocol rsh port [-port]
```

- Defines ports for RSH connections (default = 514)—Dynamically opens a port for RSH standard error connections.
- If RSH fixup is disabled:
  - Inbound RSH will work if ACL exists.
  - Outbound RSH will not work.

```
pixfirewall(config)# fixup protocol rsh 1540
```

CSPFA 3.2—9-14

By default, the PIX Firewall inspects port 514 connections for RSH traffic. If you have RSH servers using ports other than port 514, you need to use the **fixup protocol rsh** command to have the PIX Firewall inspect these other ports for RSH traffic.

The **fixup protocol rsh** command causes the PIX Firewall to dynamically create openings for RSH standard error connections for RSH traffic on the indicated port.

Use the **no** form of the command to disable the inspection of traffic on the indicated port for RSH connections. If the **fixup protocol rsh** command is not enabled for a given port, then:

- Outbound RSH will *not* work properly on that port.
- Inbound RSH will work properly on that port if an ACL to the inside server exists.

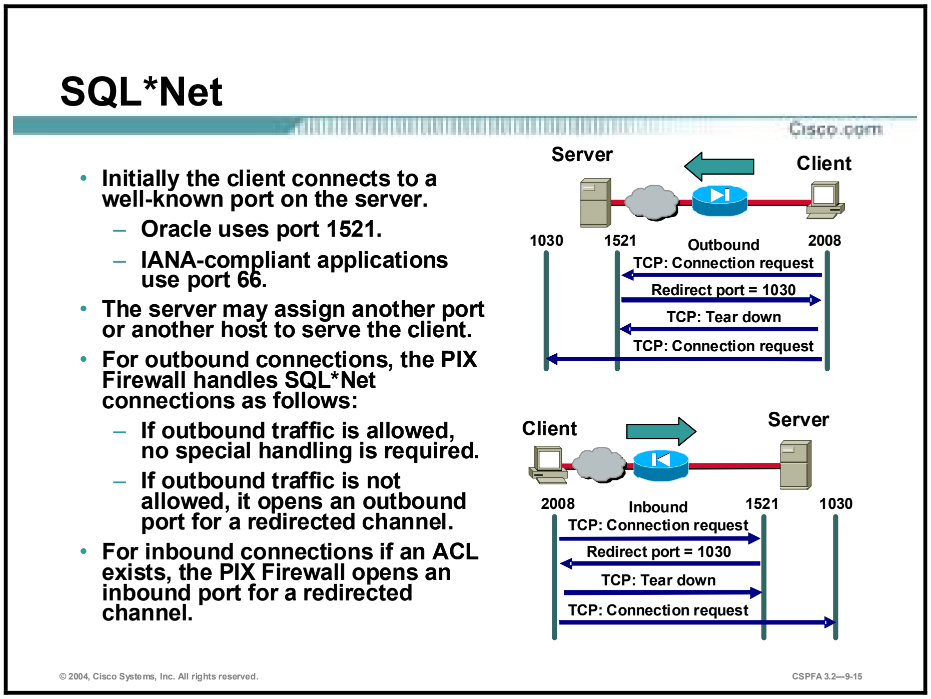
Using the **no fixup protocol rsh** command without any arguments causes the PIX Firewall to clear all previous **fixup protocol rsh** assignments and set port 514 back as the default.

The syntax for the **fixup protocol rsh** command is as follows:

fixup protocol rsh *port* [-*port*]

no fixup protocol rsh [*port* [-*port*]]

***port* [-*port*]**	Single port or range of ports for which the PIX Firewall inspects RSH connections.



SQL*Net

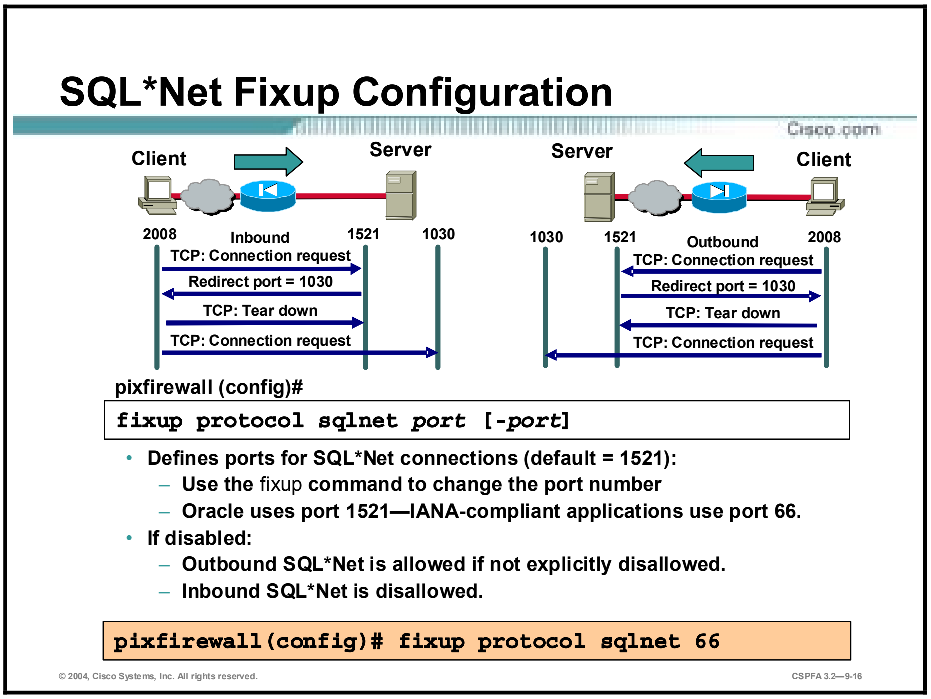
- Initially the client connects to a well-known port on the server.
  - Oracle uses port 1521.
  - IANA-compliant applications use port 66.
- The server may assign another port or another host to serve the client.
- For outbound connections, the PIX Firewall handles SQL*Net connections as follows:
  - If outbound traffic is allowed, no special handling is required.
  - If outbound traffic is not allowed, it opens an outbound port for a redirected channel.
- For inbound connections if an ACL exists, the PIX Firewall opens an inbound port for a redirected channel.

CSPFA 3.2—9-15

SQL*Net only uses one channel for communications but it could be redirected to a different port, and even more commonly to a different secondary server altogether. When a client starts an SQL*Net connection, it opens a standard TCP channel from one of its high-order ports to port 1521 on the server. The server then proceeds to redirect the client to a different port or IP address. The client tears down the initial connection and establishes the second connection.

For SQL*Net traffic, the PIX Firewall behaves in the following manner:

- Outbound connections:
  - If all outbound TCP traffic is implicitly allowed, no special handling is required because the client initiates all TCP connections from the inside.
  - If all outbound TCP traffic is *not* implicitly allowed, the PIX Firewall opens an ACL for the redirected channel between the server and the client.
- Inbound connections—If an ACL exists allowing inbound connections to an SQL*Net server, the PIX Firewall opens an inbound opening for the redirected channel.



SQL*Net Fixup Configuration

pixfirewall (config)#

```
fixup protocol sqlnet port [-port]
```

- Defines ports for SQL*Net connections (default = 1521):
  - Use the fixup command to change the port number
  - Oracle uses port 1521—IANA-compliant applications use port 66.
- If disabled:
  - Outbound SQL*Net is allowed if not explicitly disallowed.
  - Inbound SQL*Net is disallowed.

```
pixfirewall(config)# fixup protocol sqlnet 66
```

CSPFA 3.2—9-16

By default, the PIX Firewall inspects port 1521 connections for SQL*Net traffic. If you have SQL*Net servers using ports other than port 1521, you must use the **fixup protocol sqlnet** command to have the PIX Firewall inspect these other ports for SQL*Net traffic.

The **fixup protocol sqlnet** command causes the PIX Firewall to do the following for SQL*Net traffic on the indicated port:

- Perform NAT in packet payload.
- Dynamically create openings for SQL*Net redirected connections.

Use the **no** form of the command to disable the inspection of traffic on the indicated port for SQL*Net connections. If the **fixup protocol sqlnet** command is not enabled for a given port, then:

- Outbound SQL*Net will work properly on that port as long as outbound traffic is not explicitly disallowed.
- Inbound SQL*Net will *not* work properly on that port.

Using the **no fixup protocol sqlnet** command without any arguments causes the PIX Firewall to clear all previous **fixup protocol sqlnet** assignments and set port 1521 back as the default.

The syntax for the **fixup protocol sqlnet** command is as follows:

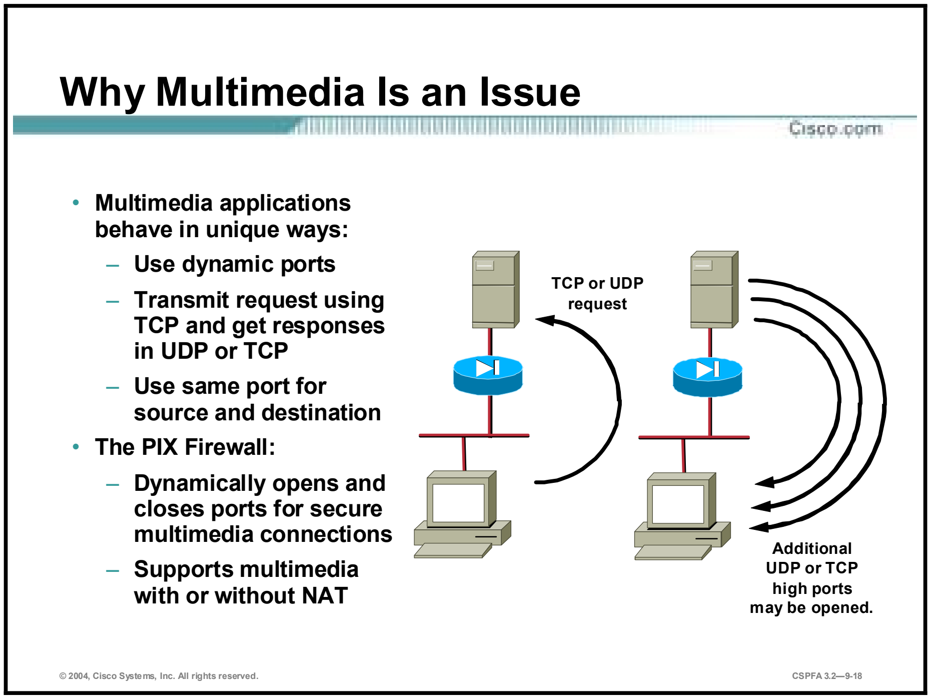
fixup protocol sqlnet[***port*****[-*****port*****]]**

no fixup protocol sqlnet [***port*****[-*****port*****]]**

***port*[-*port*]**	Single port or port range that the PIX Firewall inspects for SQL*Net connections.

Multimedia Support

This topic discusses multimedia: advantages and application supports, H.323 support, and important multimedia configurations.



Why Multimedia Is an Issue

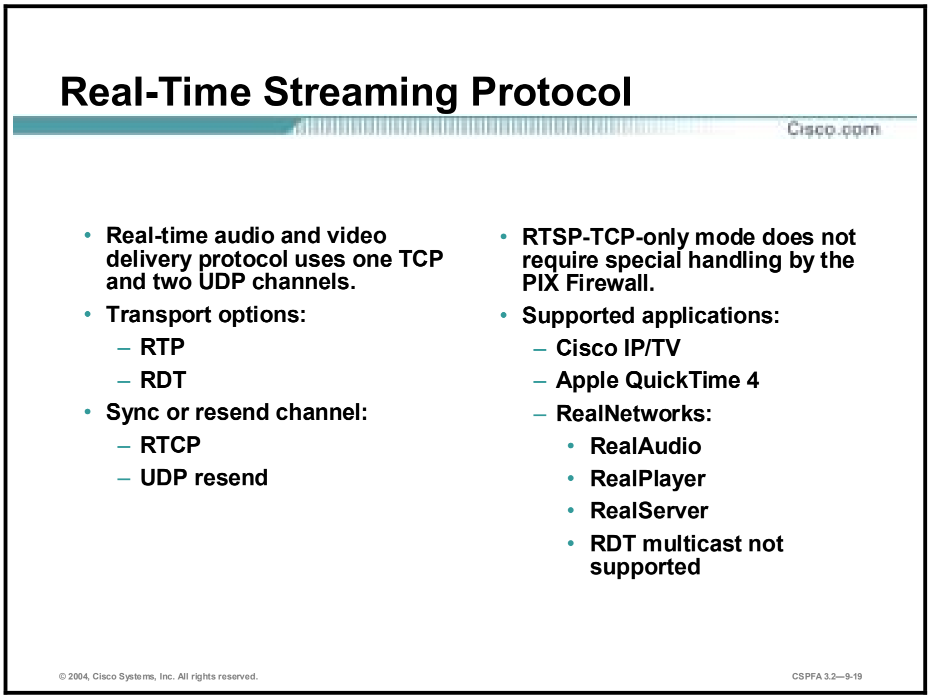
- Multimedia applications behave in unique ways:
  - Use dynamic ports
  - Transmit request using TCP and get responses in UDP or TCP
  - Use same port for source and destination
- The PIX Firewall:
  - Dynamically opens and closes ports for secure multimedia connections
  - Supports multimedia with or without NAT

Multimedia applications may transmit requests on TCP, get responses on UDP or TCP, use dynamic ports, use the same port for source and destination, and so on. Every application behaves in a different way. Implementing support for all multimedia applications using a single secure method is very difficult. Two examples of multimedia applications follow:

- RealAudio—Sends the originating request to TCP port 7070. The RealAudio server replies with multiple UDP streams anywhere from UDP port 6970 through 7170 on the client machine.
- CUseeMe client—Sends the originating request from TCP port 7649 to TCP port 7648. The CUseeMe datagram is unique in that it includes the legitimate IP address in the header as well as in the payload, and sends responses from UDP port 7648 to UDP port 7648.

The PIX Firewall dynamically opens and closes UDP ports for secure multimedia connections. You do not need to open a large range of ports, which creates a security risk, or have to reconfigure any application clients.

Also, the PIX Firewall supports multimedia with or without NAT. Many firewalls that cannot support multimedia with NAT limit multimedia usage to only registered users, or require exposure of inside IP addresses to the Internet. Lack of support for multimedia with NAT often forces multimedia vendors to join proprietary alliances with firewall vendors to accomplish compatibility for their applications.



Real-Time Streaming Protocol

Cisco.com

- Real-time audio and video delivery protocol uses one TCP and two UDP channels.
- Transport options:
  - RTP
  - RDT
- Sync or resend channel:
  - RTCP
  - UDP resend
- RTSP-TCP-only mode does not require special handling by the PIX Firewall.
- Supported applications:
  - Cisco IP/TV
  - Apple QuickTime 4
  - RealNetworks:
    - RealAudio
    - RealPlayer
    - RealServer
    - RDT multicast not supported

CSPFA 3.2—9-19

Real-Time Streaming Protocol (RTSP) is a real-time audio and video delivery control protocol used by many popular multimedia applications. It uses one TCP channel and multiple UDP channels. The TCP channel is the control channel and is used to negotiate the UDP delivery channels depending on the transport mode, Real-Time Transport Protocol (RTP) or Session Description Protocol (SDP) that is configured on the client. RTSP applications use the well-known port 554, usually TCP, and rarely UDP. The PIX Firewall supports only TCP.

The first UDP channel is the data connection and may use one of the following transport modes:

- RTP
- RealNetworks Real Data Transport (RDT)

The second UDP channel is a data connection feedback channel, and it may use one of the following modes:

- RTCP
- UDP Resend

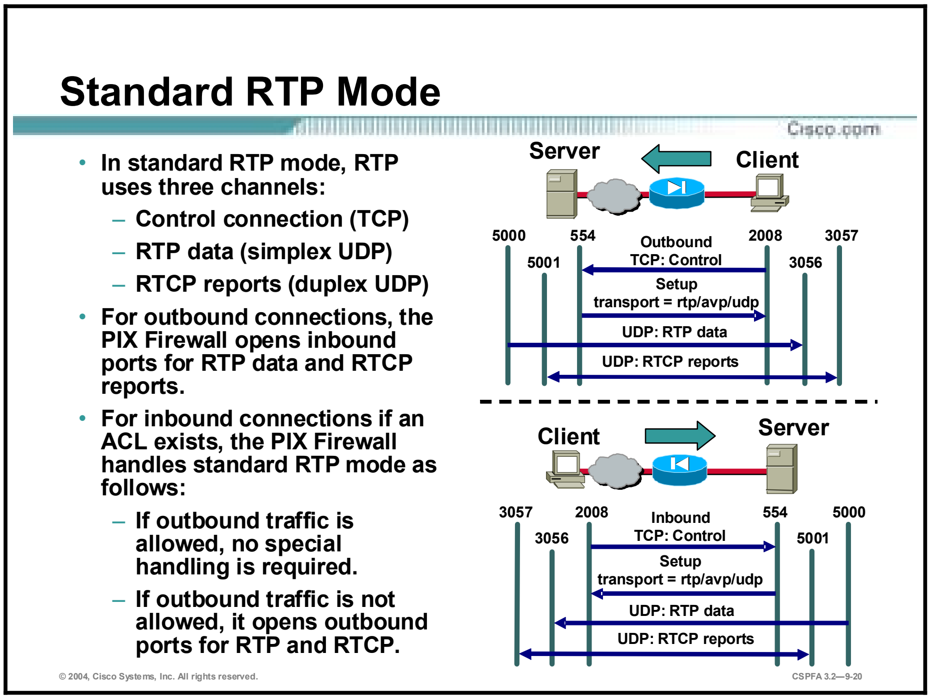
RTSP supports a TCP-only mode. This mode contains only one TCP connection, which is used as the control and data channels. Because this mode contains only one constant standard TCP connection, no special handling by the PIX Firewall is required.

The following are RTSP applications supported by the PIX Firewall:

- Cisco IP/TV
- Apple QuickTime 4
- RealNetworks
  - RealAudio
  - RealPlayer

— RealServer

Note RealNetworks RDT multicast is not supported.



Standard RTP Mode

- In standard RTP mode, RTP uses three channels:
  - Control connection (TCP)
  - RTP data (simplex UDP)
  - RTCP reports (duplex UDP)
- For outbound connections, the PIX Firewall opens inbound ports for RTP data and RTCP reports.
- For inbound connections if an ACL exists, the PIX Firewall handles standard RTP mode as follows:
  - If outbound traffic is allowed, no special handling is required.
  - If outbound traffic is not allowed, it opens outbound ports for RTP and RTCP.

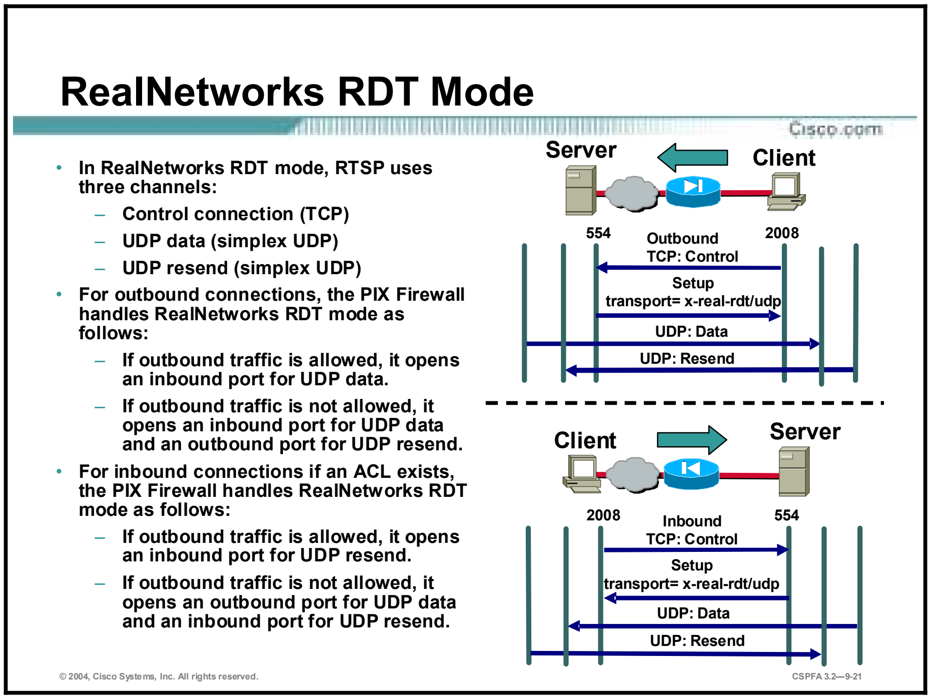
CSPFA 3.2—9-20

In standard RTP mode, the following three channels are used by RTSP:

- TCP control channel—Standard TCP connection initiated from the client to the server.
- RTP data channel—Simplex (unidirectional) UDP session used for media delivery using the RTP packet format from the server to the client. The client's port is always an even numbered port.
- RTCP reports—Duplex (bidirectional) UDP session used to provide synchronization information to the client and packet loss information to the server. The RTCP port is always the next consecutive port from the RTP data port.

For standard RTP mode RTSP traffic, the PIX Firewall behaves in the following manner:

- Outbound connections—After the client and the server negotiate the transport mode and the ports to use for the sessions, the PIX Firewall opens temporary inbound dynamic openings for the RTP data channel and RTCP report channel from the server.
- Inbound connections:
  - If an ACL exists allowing inbound connections to an RTSP server, and if all outbound UDP traffic is implicitly allowed, no special handling is required since the server initiates the data and report channel from the inside.
  - If an ACL exists allowing inbound connections to an RTSP server, and if all outbound TCP traffic is *not* implicitly allowed, the PIX Firewall opens temporary dynamic openings for the data and report channels from the server.



RealNetworks RDT Mode

- In RealNetworks RDT mode, RTSP uses three channels:
  - Control connection (TCP)
  - UDP data (simplex UDP)
  - UDP resend (simplex UDP)
- For outbound connections, the PIX Firewall handles RealNetworks RDT mode as follows:
  - If outbound traffic is allowed, it opens an inbound port for UDP data.
  - If outbound traffic is not allowed, it opens an inbound port for UDP data and an outbound port for UDP resend.
- For inbound connections if an ACL exists, the PIX Firewall handles RealNetworks RDT mode as follows:
  - If outbound traffic is allowed, it opens an inbound port for UDP resend.
  - If outbound traffic is not allowed, it opens an outbound port for UDP data and an inbound port for UDP resend.

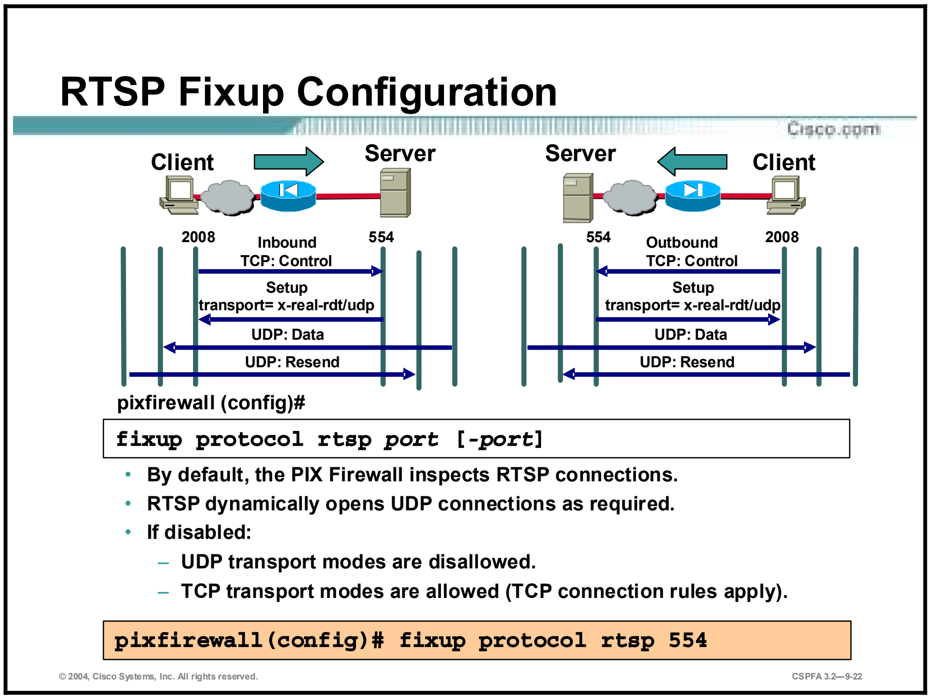
CSPFA 3.2—9-21

In RealNetworks RDT mode, the following three channels are used by RTSP:

- TCP control channel—Standard TCP connection initiated from the client to the server.
- UDP data channel—Simplex (unidirectional) UDP session used for media delivery using the standard UDP packet format from the server to the client.
- UDP resend—Simplex (unidirectional) UDP session used for the client to request that the server resend lost data packets.

For RealNetworks RDT mode RTSP traffic, the PIX Firewall behaves in the following manner:

- Outbound connections:
  - If outbound UDP traffic is implicitly allowed, and after the client and the server negotiate the transport mode and the ports to use for the session, the PIX Firewall opens temporary inbound openings for the UDP data channel from the server.
  - If outbound UDP traffic is *not* implicitly allowed, and after the client and the server negotiate the transport mode and the ports to use for the session, the PIX Firewall opens a temporary inbound opening for the UDP data channel from the server and a temporary outbound opening for the UDP resend channel from the client.
- Inbound connections:
  - If an ACL exists allowing inbound connections to an RTSP server, and if all outbound UDP traffic is implicitly allowed, the PIX Firewall opens a temporary inbound opening for the UDP resend from the client.
  - If an ACL exists allowing inbound connections to an RTSP server, and if all outbound TCP traffic is *not* implicitly allowed, the PIX Firewall opens temporary opening for the UDP data and UDP resend channels from the server and client, respectively.



RTSP Fixup Configuration

pixfirewall (config)#

```
fixup protocol rtsp port [-port]
```

- By default, the PIX Firewall inspects RTSP connections.
- RTSP dynamically opens UDP connections as required.
- If disabled:
  - UDP transport modes are disallowed.
  - TCP transport modes are allowed (TCP connection rules apply).

```
pixfirewall(config)# fixup protocol rtsp 554
```

CSPFA 3.2—9-22

By default, the PIX Firewall inspects for RTSP connections. The **fixup protocol rtsp** command causes the PIX Firewall to dynamically create dynamic openings for RTSP UDP channels for RTSP traffic on the indicated port. Use the **no** form of the command to disable the inspection of traffic on the indicated port for RTSP connections. If the **fixup protocol rtsp** command is not enabled for a given port, then neither outbound nor inbound RTSP will work properly on that port.

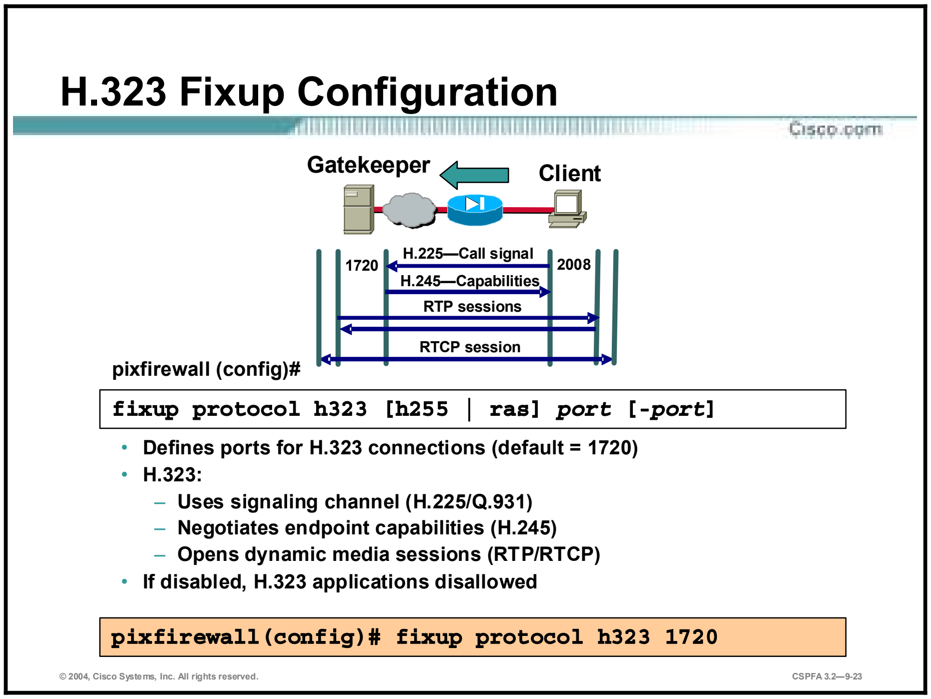
Using the **no fixup protocol rtsp** command without any arguments causes the PIX Firewall to clear all previous **fixup protocol rtsp** assignments.

The syntax for the **fixup protocol rtsp** command is as follows:

fixup protocol rtsp *port* [-*port*]

no fixup protocol rtsp [*port* [-*port*]]

***port* [-*port*]**	Port or port range for which the PIX Firewall inspects RTSP connections.



H.323 Fixup Configuration

pixfirewall (config)#

```
fixup protocol h323 [h255 | ras] port [-port]
```

- Defines ports for H.323 connections (default = 1720)
- H.323:
  - Uses signaling channel (H.225/Q.931)
  - Negotiates endpoint capabilities (H.245)
  - Opens dynamic media sessions (RTP/RTCP)
- If disabled, H.323 applications disallowed

```
pixfirewall(config)# fixup protocol h323 1720
```

CSPFA 3.2—9-23

H.323 is more complicated than other traditional protocols because it uses two TCP connections and four to six UDP sessions for a single “call.” (Only one of the TCP connections goes to a well-known port; all the other ports are negotiated and are temporary.) Furthermore, the content of the streams is far more difficult for firewalls to understand than with many other protocols because H.323 encodes packets using Abstract Syntax Notation, or ASN.1.

The call signaling function uses H.225 call signaling to establish a connection between two H.323 endpoints. In systems that do not have a gatekeeper, the call signaling channel is opened between the two endpoints involved in the call. In systems that contain a gatekeeper, the call signaling channel is opened between the endpoints and the gatekeeper or between the endpoints themselves as chosen by the gatekeeper. The PIX Firewall dynamically allocates the H.245 connection based on the inspection of the H.225 messages.

The H.245 control function uses the H.245 control channel to carry end-to-end control messages governing operations of the H.323 entity, including capabilities exchange, opening and closing logical channels that carry the audiovisual and data information, mode preferences, and so on. The endpoint establishes one H.245 control channel for each call. The endpoints can establish multiple multimedia logical channels using RTP and RTCP. Within each H.245 message, the H.323 endpoints exchange port numbers that are used for subsequent UDP media streams. The H.323 fixup application inspects the H.245 messages to identify these ports and dynamically creates connections for the media exchange. RTP uses the negotiated port number, while RTCP uses the next-higher port number.

The H.323 control channel handles H.225, H.245, and H.323. H.323 inspection uses the following ports.

- 1718—Gatekeeper discovery UDP port
- 1719—Registration, admission, and status (RAS) UDP port
- 1720—TCP control port

The two major functions of H.323 inspection are as follows:

- Perform NAT on the necessary embedded IP version 4 (IPv4) addresses in the H.225 and H.245 messages. Because H.323 messages are encoded in packed encoding rules (PER) format, the PIX Firewall uses an ASN.1 decoder to decode the H.323 messages.
- Dynamically allocate the negotiated H.245 and RTP/RTCP connections.

The PIX Firewall administrator must open an ACL for the well-known H.323 port 1720 for the H.225 call signaling.

The syntax for the **fixup protocol h323** command is as follows:

fixup protocol h323 [h225 | ras] ***port*** **[-*port*]**

no fixup protocol h323 [h225 | ras] [*port* [-*port*]]

h225	Specifies the use of H.225, which is the ITU standard that governs H.225.0 session establishment and packetization, with H.323. H.225.0 actually describes several different protocols: RAS, use of Q.931, and use of RTP.
ras	Specifies the use of RAS with H.323 to enable dissimilar communication devices to communicate with each other. H.323 defines a common set of codecs, call setup and negotiating procedures, and basic data transport methods.
***port*[-*port*]**	Single port or port range that the PIX Firewall inspects for H.323 connections.

By default, the PIX Firewall inspects port 1720 connections for H.323 traffic. Use the **fixup protocol h323** command to change the default port. Use the **no fixup protocol h323** command without any arguments to clear all **fixup protocol 323** command assignments.

Supported H.323 applications are as follows:

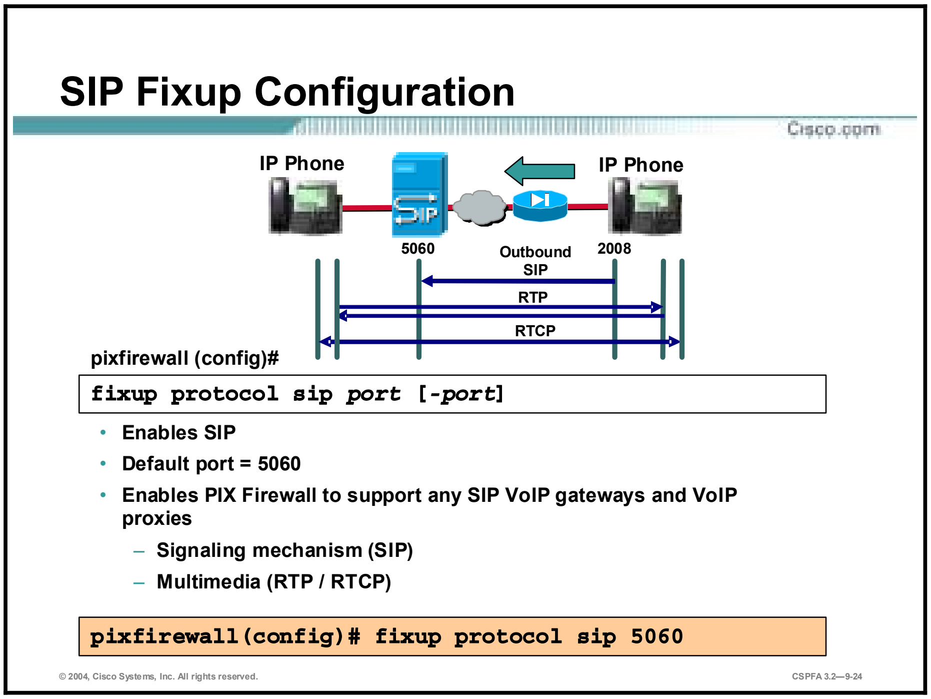
- Cisco Multimedia Conference Manager
- Microsoft NetMeeting
- Intel Video Phone
- Intel InternetPhone
- CUseeMe Networks:
  - MeetingPoint
  - CUseeMe Pro
- VocalTec Communications:
  - Internet Phone
  - Gatekeeper

Cisco PIX Firewall Software Versions 5.2 and higher support H.323 version 2. H.323 supports H.323 Voice over IP (VoIP) gateways and VoIP gatekeepers. H.323 version 2 adds the following functionality to the PIX Firewall:

- Fast Connect or Fast Start procedure for faster call setup. Fast Connect uses only one TCP connection, and RAS uses a single UDP connection for registration, admissions, and status.

- H.245 tunneling for resource conservation, call synchronization, and reduced set up time.

Cisco PIX Firewall Software Version 6.3 supports H.323 versions 3 and 4.



SIP Fixup Configuration

pixfirewall (config)#

```
fixup protocol sip port [-port]
```

- Enables SIP
- Default port = 5060
- Enables PIX Firewall to support any SIP VoIP gateways and VoIP proxies
  - Signaling mechanism (SIP)
  - Multimedia (RTP / RTCP)

```
pixfirewall(config)# fixup protocol sip 5060
```

CSPFA 3.2—9-24

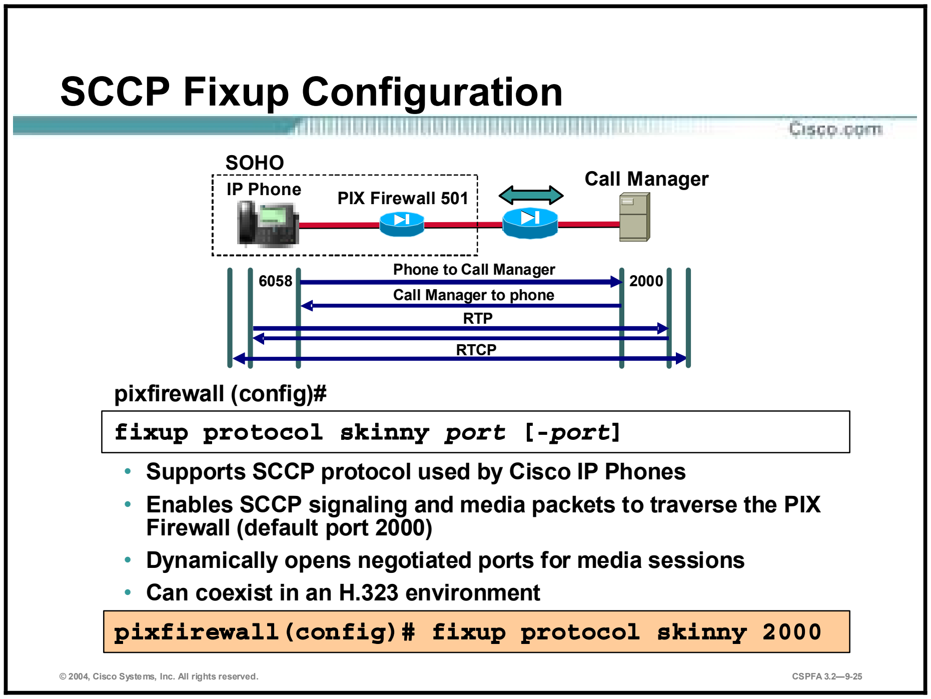
SIP is an application-layer control protocol used to set up and tear down multimedia sessions. These multimedia sessions include Internet telephony and similar applications. SIP uses RTP for media transport and RTCP for providing a Quality of Service (QoS) feedback loop. Using SIP, the PIX Firewall can support any SIP VoIP gateways and VoIP proxy servers.

To support SIP calls through the PIX Firewall, signaling messages for the media connection addresses, media ports, and embryonic connections for the media must be inspected, because while the signaling is sent over a well-known destination port (UDP/TCP 5060), the media streams are dynamically allocated. The **fixup protocol sip** command can be used to enable or disable SIP support. SIP is a text-based protocol and contains IP addresses throughout the text. With the SIP fixup enabled, the PIX Firewall inspects the packets, and both NAT and PAT are supported.

SIP support is enabled by default on port 5060. The **show conn state sip** command can be used to display all active SIP connections.

The **timeout** command with the **sip media** option modifies the duration for the SIP media inactivity timer. When this time elapses, SIP connections with RTP/RTCP expire. The **timeout** command with the sip option modifies the duration for the SIP inactivity timer. When this time elapses, the port used by the SIP service closes.

Note The PIX Firewall also supports SIP proxies.



SCCP Fixup Configuration

Cisco.com
SOHO
IP Phone
PIX Firewall 501
Call Manager
6058
Phone to Call Manager
2000
Call Manager to phone
RTP
RTCP

pixfirewall (config)#

```
fixup protocol skinny port [-port]
```

- Supports SCCP protocol used by Cisco IP Phones
- Enables SCCP signaling and media packets to traverse the PIX Firewall (default port 2000)
- Dynamically opens negotiated ports for media sessions
- Can coexist in an H.323 environment

```
pixfirewall(config)# fixup protocol skinny 2000
```

CSPFA 3.2—9-25

In Cisco PIX Firewall Software Versions 6.0 and higher, the PIX Firewall application handling supports SCCP, used by Cisco IP Phones for VoIP call signaling. SCCP defines the set of messages that is needed for a Cisco IP Phone to communicate with the Cisco Call Manager for call setup. The IP Phone uses a randomly selected TCP port to send and receive SCCP messages. Call Manager listens for SCCP messages at TCP port 2000. SCCP uses RTP and RTCP for media transmissions. The media ports are randomly selected by the IP Phones.

Skinny fixup enables the PIX Firewall to dynamically open negotiated ports for media sessions. An application layer ensures that all SCCP signaling and media packets can traverse the PIX Firewall and interoperate with H.323 terminals. SCCP support allows an IP Phone and Cisco Call Manager to be placed on separate sides of the PIX Firewall.

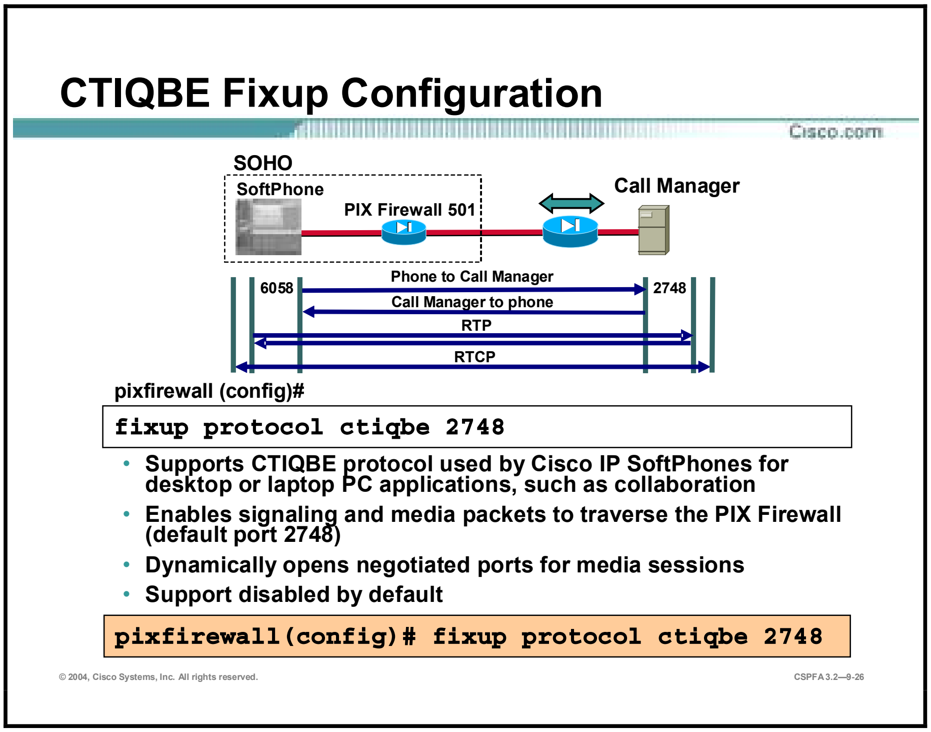
Skinny fixup is enabled by default to listen for SCCP messages on port 2000. Use the **fixup protocol skinny** command to change the default port. Use the **no fixup protocol skinny** command without any arguments to clear all **fixup protocol skinny** assignments.

The syntax for the **fixup protocol skinny** command is as follows:

fixup protocol skinny port [- port]

no fixup protocol skinny [*port* [*-port*]]

***port* [-*port*]**	Port or port range for which the PIX Firewall inspects SCCP connections.



CTIQBE Fixup Configuration

pixfirewall (config)#

```
fixup protocol ctiqbe 2748
```

- Supports CTIQBE protocol used by Cisco IP SoftPhones for desktop or laptop PC applications, such as collaboration
- Enables signaling and media packets to traverse the PIX Firewall (default port 2748)
- Dynamically opens negotiated ports for media sessions
- Support disabled by default

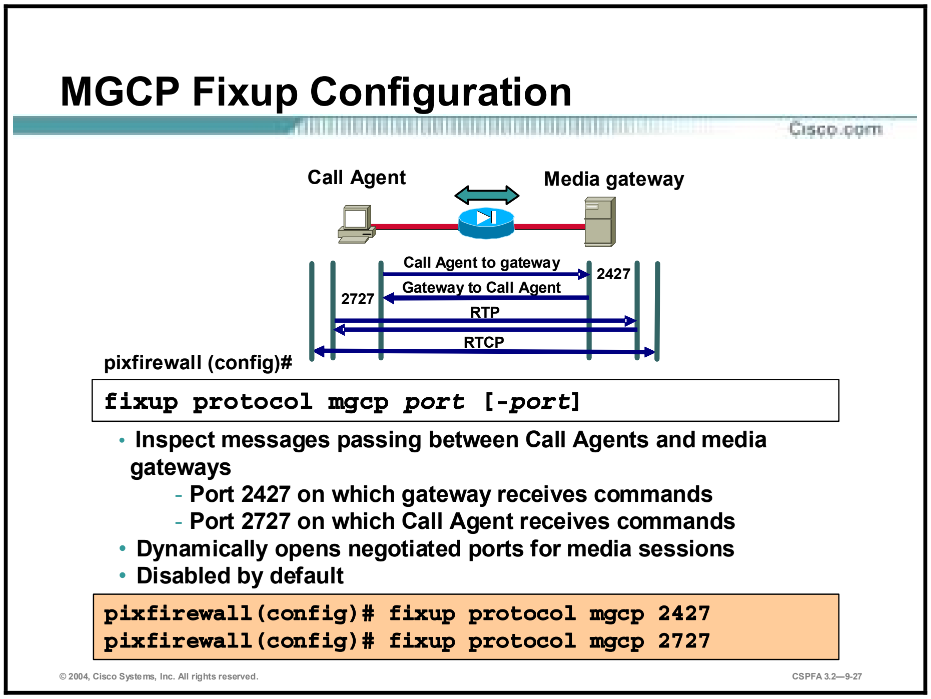
```
pixfirewall(config)# fixup protocol ctiqbe 2748
```

CSPFA 3.2—9-26

The Telephony Application Programming Interface (TAPI) and Java Telephony Application Programming Interface (JTAPI) are used by many Cisco VoIP applications. Cisco PIX Firewall Software Version 6.3 introduces support for a specific protocol, Computer Telephony Interface Quick Buffer Encoding (CTIQBE), which is used by Cisco TAPI Service Provider (TSP) to communicate with Cisco Call Manager. Support for this protocol is disabled by default.

To enable support for this protocol, enter the following command:

fixup protocol ctiqbe 2748



MGCP Fixup Configuration

Cisco.com

Call Agent
Media gateway
Call Agent to gateway
2427
Gateway to Call Agent
2727
RTP
RTCP

pixfirewall (config)#

```
fixup protocol mgcp port [-port]
```

- Inspect messages passing between Call Agents and media gateways
  - Port 2427 on which gateway receives commands
  - Port 2727 on which Call Agent receives commands
- Dynamically opens negotiated ports for media sessions
- Disabled by default

```
pixfirewall(config)# fixup protocol mgcp 2427
pixfirewall(config)# fixup protocol mgcp 2727
```

CSPFA 3.2—9-27

Cisco PIX Firewall Software Version 6.3 introduces support for application inspection of the Media Gateway Control Protocol (MGCP). MGCP is used for controlling media gateways from external call control elements called media gateway controllers or call agents. A media gateway is typically a network element that provides conversion between the audio signals carried on telephone circuits and data packets carried over the Internet or over other packet networks. Examples of media gateways are:

- Trunking gateway—Provides an interface between the telephone network and a VoIP network. Such gateways typically manage a large number of digital circuits.
- Residential gateway—Provides a traditional analog (RJ-11) interface to a VoIP network. Examples of residential gateways include cable modem/cable set-top boxes, xDSL devices, and broadband wireless devices.
- Business gateway—Provides a traditional digital PBX interface or an integrated soft PBX interface to a VoIP network. MGCP messages are transmitted over UDP.

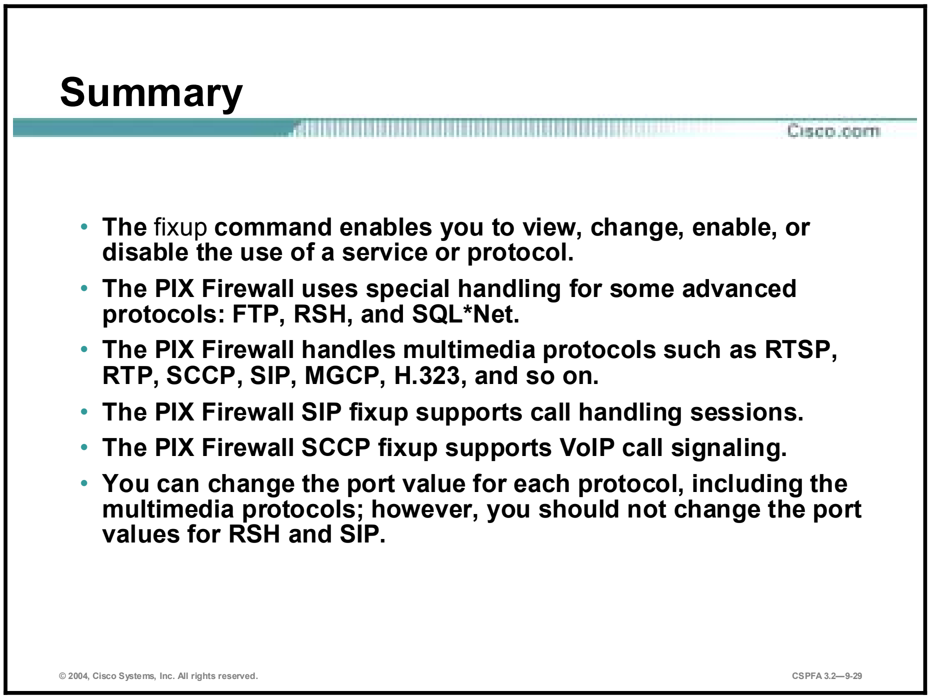
Application inspection for MGCP is disabled by default. To use MGCP, you typically need to configure at least two ports, one on which the gateway receives commands and one for the port on which the call agent receives commands. Normally, a call agent will send commands to port 2427, while a gateway will send commands to port 2727. Audio packets are transmitted over an IP network using RTP. MGCP fixup enables the PIX Firewall to securely open negotiated UDP ports for legitimate media connections through the PIX Firewall. To enable MGCP application inspection for call agents and gateways using the default ports, enter the following commands:

```
fixup protocol mgcp 2427

fixup protocol mgcp 2727
```

Neither NAT nor PAT is supported by PIX Firewall Software Version 6.3 or lower.

Summary

This topic summarizes what you learned in this lesson.



Summary

Cisco.com

- **The** fixup **command enables you to view, change, enable, or disable the use of a service or protocol.**
- **The PIX Firewall uses special handling for some advanced protocols: FTP, RSH, and SQL*Net.**
- **The PIX Firewall handles multimedia protocols such as RTSP, RTP, SCCP, SIP, MGCP, H.323, and so on.**
- **The PIX Firewall SIP fixup supports call handling sessions.**
- **The PIX Firewall SCCP fixup supports VoIP call signaling.**
- **You can change the port value for each protocol, including the multimedia protocols; however, you should not change the port values for RSH and SIP.**

CSPFA 3.2—9-29

Lab Exercise—Configure and Test Advanced Protocol Handling on the Cisco PIX Firewall

Complete the following lab exercise to practice what you have learned in this lesson.

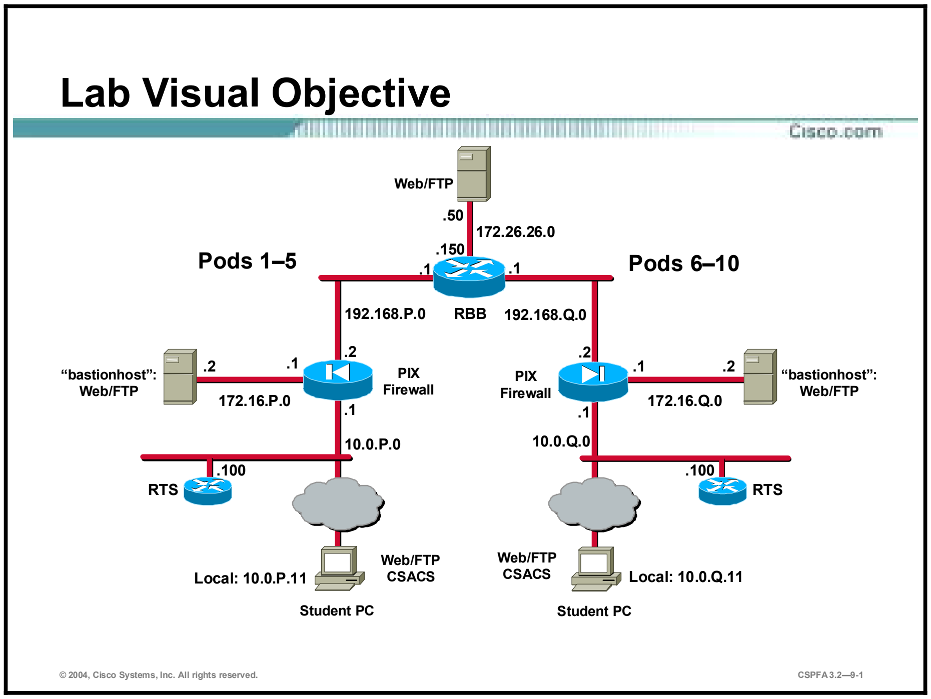
Objectives

In this lab exercise you will complete the following tasks:

- Display the fixup protocol configurations.
- Change the fixup protocol configurations.
- Test the outbound FTP fixup protocol.
- Test the inbound FTP fixup protocol.
- Set the fixup protocols to the default settings.

Visual Objectives

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

Task 1—Display the Fixup Protocol Configurations

Complete the following step and enter the command as directed to see the current configurations of your PIX Firewall:

Step 1 List the fixup protocols that are running on your PIX Firewall:

```
pixP(config)# show fixup protocol
```

(where P = pod number)

Q1) Complete the table with the ports assigned to the fixup protocols:

ftp	
http	
h323 h225	
h323 ras	
ils	
rsh	
rtsp	
smtp	
sqlnet	

sip	
skinny	

Task 2—Change the Fixup Protocol Configurations

Complete the following steps and enter the commands as directed to change some of the current configurations of your PIX Firewall:

Step 1 Disable the following fixup protocols:

```
pixP(config)# no fixup protocol http 80
pixP(config)# no fixup protocol smtp 25
pixP(config)# no fixup protocol h323 h225 1720
pixP(config)# no fixup protocol sqlnet 1521
```

(where P = pod number)

Step 2 Define a range of ports for SQL*Net connections:

```
pixP(config)# fixup protocol sqlnet 66-76
```

(where P = pod number)

Step 3 Verify the fixup protocol settings using the **show fixup protocol** command:

```
pixP(config)# show fixup protocol
fixup protocol ftp 21
fixup protocol h323 ras 1718-1719
fixup protocol ils 389
fixup protocol rsh 514
fixup protocol rtsp 554
fixup protocol sip 5060
fixup protocol sip udp 5060
fixup protocol skinny 2000
no fixup protocol http 80
no fixup protocol smtp 25
no fixup protocol h323 h225 1720
no fixup protocol sqlnet 1521
fixup protocol sqlnet 66-76
```

(where P = pod number)

Task 3—Test the Outbound FTP Fixup Protocol

Complete the following steps and enter the commands as directed to test the outbound FTP fixup protocol:

Step 1 Enable console logging on your PIX Firewall:

```
pixP(config)# logging console debug
```

(where P = pod number)

Step 2 FTP to the backbone server from your student PC using the Windows FTP client:

```
C:\> ftp 172.26.26.50
User (172.26.26.50:(none)): anonymous
Password: user@
```

Step 3 Do a directory listing at the FTP prompt:

```
ftp> dir
```

Q2) What logging messages were generated on your PIX Firewall console?

A) ____

Step 4 Quit your FTP session:

```
ftp> quit
```

Step 5 Turn off the FTP fixup protocol on your PIX Firewall:

```
pixP(config)# no fixup protocol ftp
```

(where P = pod number)

Step 6 Again, ftp to the backbone server from your student PC using the Windows FTP client:

```
C:\> ftp 172.26.26.50
User (172.26.26.50:(none)): anonymous
Password: user@
```

Q3) Were you able to log into the server? Why or why not?

A) ____

Step 7 Do a directory listing at the FTP prompt:

```
ftp> dir
```

Q4) Were you able to see a file listing? Why or why not?

A) ____

Step 8 Quit your FTP session:

```
ftp> quit
```

Note: If the FTP client is hung, press **Ctrl+C** until you break back to the C:\ prompt or close the command prompt window.

Step 9 Set the browser for passive ftp. Ftp to the backbone server from your student PC using your web browser. To do this, enter the following in the URL field:

```
ftp://172.26.26.50
```

Q5) Were you able to connect? Why or why not?

A) ____

Q6) Were you able to see a file listing? Why or why not?

A) ____

Step 10 Close your web browser.

Task 4—Test the Inbound FTP Fixup Protocol

Complete the following steps and enter the commands as directed to test the inbound FTP fixup protocol:

Step 1 Re-enable the FTP fixup protocol on your PIX Firewall:

```
pixP(config)# fixup protocol ftp 21
```

(where P = pod number)

Step 2 Ftp to a peer pod's bastion host from your student PC using your web browser. To do this, enter the following in the URL field:

```
ftp://192.168.Q.11
```

(where Q = peer pod number)

Note: The instructor assigns the peer pod number.

Q7) What logging messages were generated on your PIX Firewall console?

A) ____

Step 3 Close your web browser.

Step 4 Turn off the FTP fixup protocol on your PIX Firewall:

```
pixP(config)# no fixup protocol ftp
```

(where P = pod number)

Step 5 Ftp to a peer pod's bastion host from your student PC using your web browser. To do this, enter the following in the URL field:

```
ftp://192.168.Q.11
```

(where Q = peer pod number)

Note: The instructor assigns the peer pod number.

Q8) Were you able to connect to the peer pod's inside FTP server? Why or why not?

A) ____

Task 5—Set the Fixup Protocols to the Default Settings

Complete the following steps and enter the commands as directed to set all fixups to the factory default:

Step 1 Set all fixup protocols to the factory defaults:

```
pixP(config)# clear fixup
```

(where P = pod number)

Step 2 Verify the fixup protocol settings:

```
pixP(config)# show fixup protocol
fixup protocol ftp 21
fixup protocol http 80
fixup protocol h323 h225 1720
fixup protocol h323 ras 1718-1719
fixup protocol ils 389
fixup protocol rsh 514
fixup protocol rtsp 554
fixup protocol smtp 25
fixup protocol sqlnet 1521
fixup protocol sip 5060
fixup protocol sip udp 5060
fixup protocol skinny 2000
```

(where P = pod number)

Answers

Q1) In the spaces provided, write the ports assigned to all the fixup protocols:

ftp	21
http	80
h323 h225	1720
h323 ras	1718–1719
ils	389
rsh	514
rtsp	554
smtp	25
sqlnet	1521
sip	5060
skinny	2000

Q2) What logging messages were generated on your PIX Firewall console?

A) 302013: Built outbound TCP connection 83 for outside:172.26.26.50/21 (172.26.26.50/21) to inside:10.1.P.11/1063 (192.168.P.10/1063).
(where P = pod number)

B) 302013: Built outbound TCP connection 84 for outside:172.26.26.50/20 (172.26.26.50/20) to inside:10.1.P.11/1064 (192.168.P.10/1064).
(where P = pod number)

C) 302014: Teardown TCP connection 84 for outside:172.26.26.50/20 to inside:10.1.P.11/1064 duration 0:00:01 bytes 363 TCP FINs.
(where P = pod number)

Q3) Were you able to log into the server? Why or why not?

A) Yes. Outbound connections are allowed, and only the command channel is set up at this point.

Q4) Were you able to see a file listing? Why or why not?

A) No. A dir command causes the FTP server to open a data connection back to the client. Without the FTP fixup, the PIX Firewall does not allow this data connection from the outside.

Q5) Were you able to connect? Why or why not?

A) Yes. Outbound connections are allowed.

Q6) Were you able to see a file listing? Why or why not?

A) Yes. The web browser uses passive mode FTP, so the data channel is initiated from the inside and therefore is allowed by the PIX default policy.

Q7) What logging messages were generated on your PIX Firewall console?

A) 302013: Built outbound TCP connection 62 for outside:192.168.Q.11/21 (192.168.Q.11/21) to inside:10.1.P.11/1076 (192.168.P.10/1076).
(where P = pod number, and Q = peer pod number)

B) 302013: Built outbound TCP connection 63 for outside:192.168.Q.11/2004 (192.168.Q.11/2004) to inside:10.1.P.11/1077 (192.168.P.10/1077).
(where P = pod number, and Q = peer pod number)

C) 302014: Teardown TCP connection 63 for outside:192.168.Q.11/2004 to inside:10.1.P.11/1077 duration 0:00:01 bytes 363 TCP FINs.
(where P = pod number, and Q = peer pod number)

Q8) Were you able to connect to the peer pod’s inside FTP server? Why or why not?

A) No. Both the control and data channel are initiated from the outside.

10

Attack Guards, Intrusion Detection, and Shunning

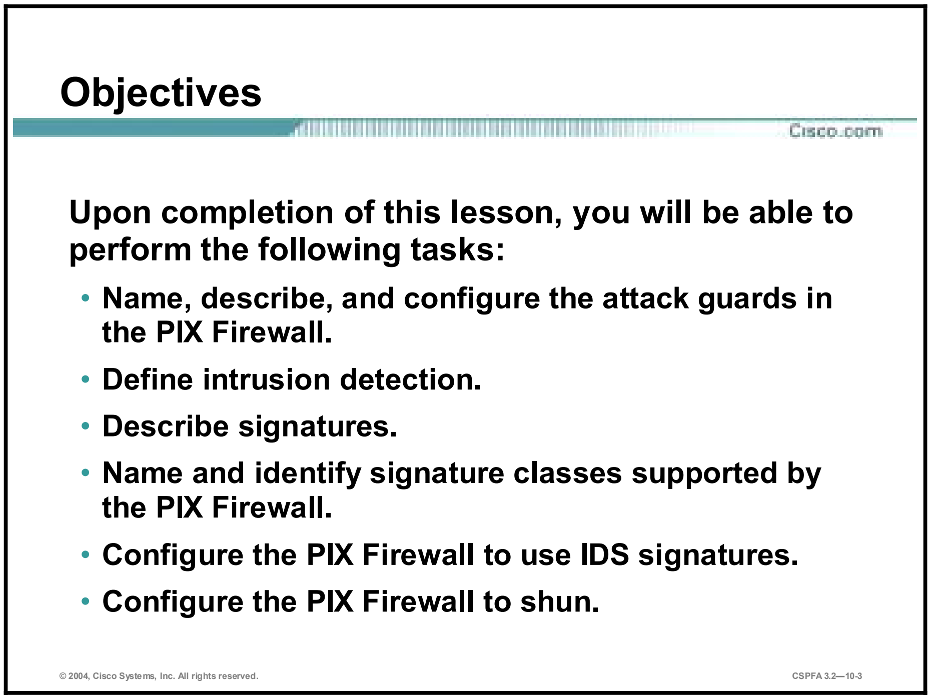
Overview

This lesson includes the following topics:

- Objectives
- Attack guards
- Intrusion detection
- Shunning
- Summary
- Lab exercise

Objectives

This topic lists the lesson's objectives.



Objectives

Cisco.com

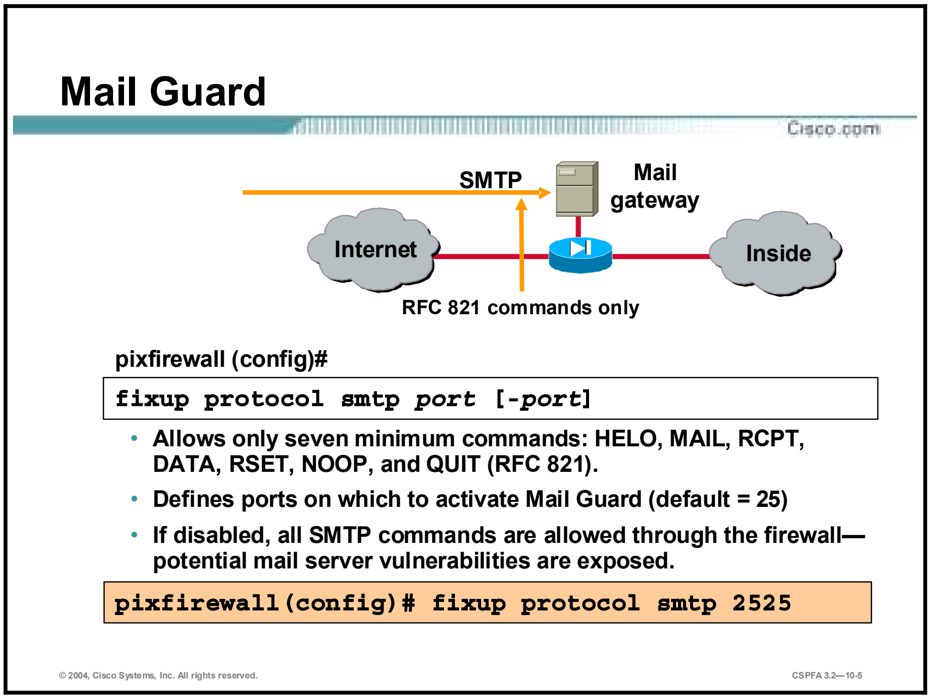
Upon completion of this lesson, you will be able to perform the following tasks:

- **Name, describe, and configure the attack guards in the PIX Firewall.**
- **Define intrusion detection.**
- **Describe signatures.**
- **Name and identify signature classes supported by the PIX Firewall.**
- **Configure the PIX Firewall to use IDS signatures.**
- **Configure the PIX Firewall to shun.**

CSPFA 3.2—10-3

Attack Guards

This topic discusses the guards put in place to protect against attacks by e-mail; Domain Name System (DNS); fragmentation; authentication, authorization, and accounting (AAA); and SYN floods.



Mail Guard

pixfirewall (config)#

```
fixup protocol smtp port [-port]
```

- Allows only seven minimum commands: HELO, MAIL, RCPT, DATA, RSET, NOOP, and QUIT (RFC 821).
- Defines ports on which to activate Mail Guard (default = 25)
- If disabled, all SMTP commands are allowed through the firewall—potential mail server vulnerabilities are exposed.

```
pixfirewall(config)# fixup protocol smtp 2525
```

CSPFA 3.2—10-5

Mail Guard provides a safe conduit for Simple Mail Transfer Protocol (SMTP) connections from the outside to an inside e-mail server. Mail Guard enables a mail server to be deployed within the internal network without it being exposed to known security problems with some mail server implementations.

When configured, Mail Guard allows only seven SMTP commands as specified in RFC 821 section 4.5.1: HELO, MAIL, RCPT, DATA, RSET, NOOP, and QUIT. Other commands, such as KILL, WIZ, and so forth, are intercepted by the PIX Firewall and are never sent to the mail server inside the network. The PIX Firewall responds with an OK even to denied commands, so that attackers will not know that their attempts are being thwarted.

By default, the Cisco Secure PIX Firewall inspects port 25 connections for SMTP traffic. If you have SMTP servers using ports other than port 25, you must use the **fixup protocol smtp** command to have the PIX Firewall inspect these other ports for SMTP traffic.

Use the **no fixup protocol smtp** command to disable the inspection of traffic on the indicated port for SMTP connections. If the **fixup protocol smtp** command is not enabled for a given port, then potential mail server vulnerabilities are exposed.

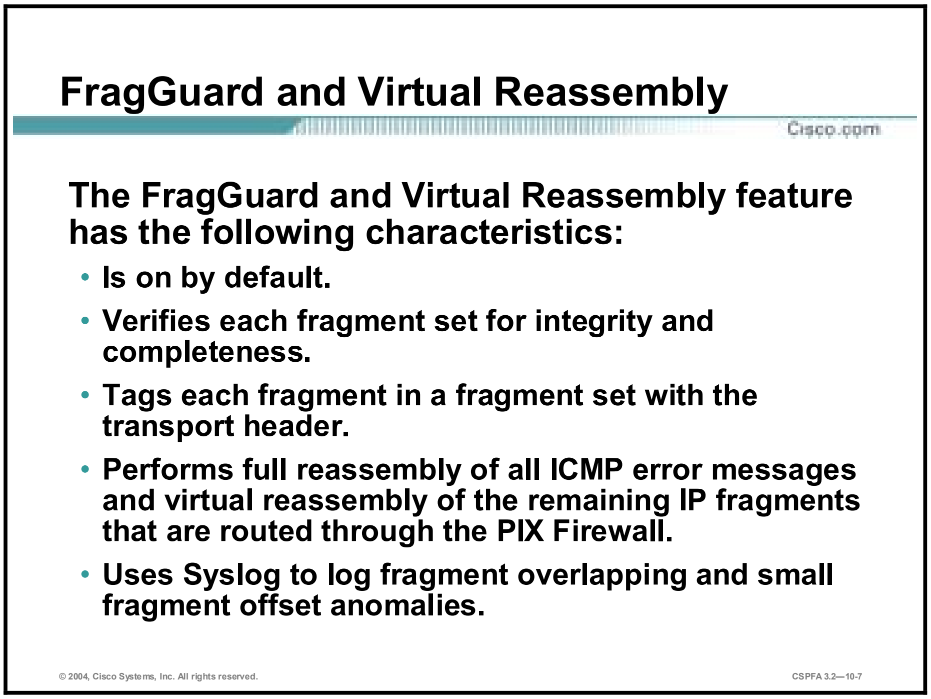
Using the **no fixup protocol smtp** command without any arguments causes the PIX Firewall to clear all previous **fixup protocol smtp** assignments and set port 25 back as the default.

The syntax for the **fixup protocol smtp** command is as follows:

fixup protocol smtp *port*[-*port*]

no fixup protocol smtp[*port*[-*port*]]

***port*[-*port*]**	Single port or port range that the PIX Firewall inspects for SMTP connections.



FragGuard and Virtual Reassembly

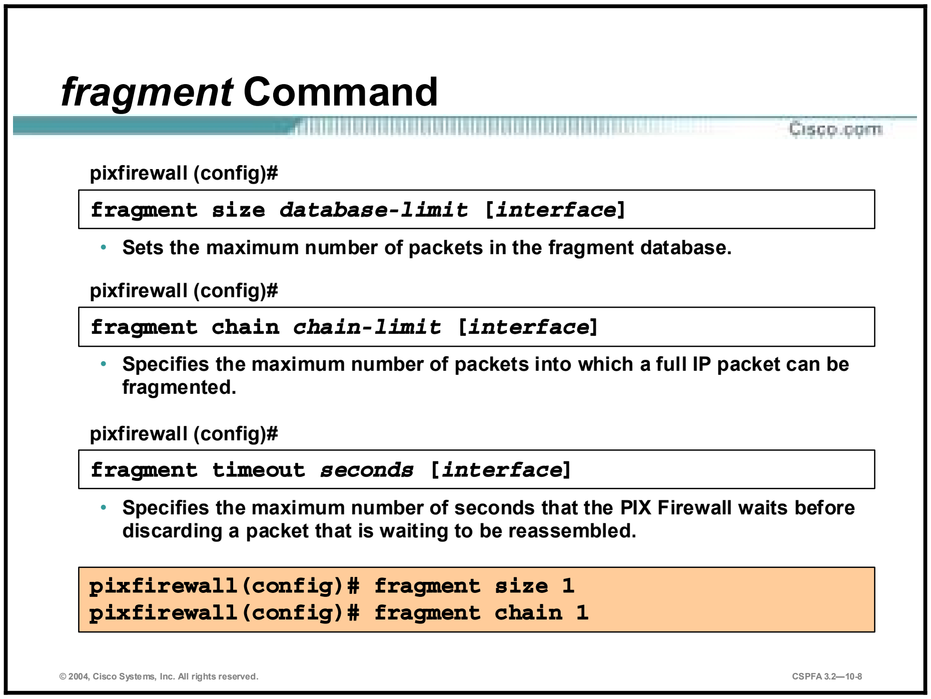
The FragGuard and Virtual Reassembly feature has the following characteristics:

- **Is on by default.**
- **Verifies each fragment set for integrity and completeness.**
- **Tags each fragment in a fragment set with the transport header.**
- **Performs full reassembly of all ICMP error messages and virtual reassembly of the remaining IP fragments that are routed through the PIX Firewall.**
- **Uses Syslog to log fragment overlapping and small fragment offset anomalies.**

CSPFA 3.2—10-7

FragGuard and Virtual Reassembly is a PIX Firewall feature that provides IP fragment protection. Virtual reassembly is the process of gathering a set of IP fragments, verifying integrity and completeness, tagging each fragment in the set with the transport header, and not coalescing the fragments into a full IP packet. Virtual Reassembly provides the benefits of full reassembly by verifying the integrity of each fragment set and tagging it with the transport header. It also minimizes the buffer space that must be reserved for packet reassembly. Full reassembly of packets is expensive in terms of buffer space that must be reserved for collecting and coalescing the fragments. Since coalescing of fragments is not performed with virtual reassembly, no preallocation of the buffer is needed.

FragGuard and Virtual Reassembly perform full reassembly of all ICMP error messages and virtual reassembly of the remaining IP fragments that are routed through the PIX Firewall. They use Syslog to log any fragment overlapping and small fragment offset anomalies, especially those caused by a Teardrop.c attack.



fragment Command

pixfirewall (config)#

```
fragment size database-limit [interface]
```

- Sets the maximum number of packets in the fragment database.

pixfirewall (config)#

```
fragment chain chain-limit [interface]
```

- Specifies the maximum number of packets into which a full IP packet can be fragmented.

pixfirewall (config)#

```
fragment timeout seconds [interface]
```

- Specifies the maximum number of seconds that the PIX Firewall waits before discarding a packet that is waiting to be reassembled.

```
pixfirewall(config)# fragment size 1
pixfirewall(config)# fragment chain 1
```

CSPFA 3.2—10-8

The **fragment** command provides management of packet fragmentation and improves the PIX Firewall's compatibility with the Network File System (NFS), a client/server application that enables a computer user to view and optionally store and update files on a remote computer as though they were on the user's own computer. In general, the default values of the **fragment** command should be used. However, if a large percentage of the network traffic through the PIX Firewall is NFS, additional tuning may be necessary to avoid database overflow. See system log message 209003 for additional information.

You can use the **fragment size** command to set the maximum number of packets in the fragment database. Use the **fragment chain** command to specify the maximum number of packets into which a packet can be fragmented, and use the **fragment timeout** command to specify the maximum number of seconds the PIX Firewall waits after the first fragment is received before discarding a fragment waiting for reassembly. The example in the figure uses the **fragment size** and **fragment chain** commands to disallow all fragments through the PIX Firewall.

In an environment where the maximum transmission unit (MTU) between the NFS server and client is small, such as a WAN interface, the chain option may require additional tuning. In this case, NFS over TCP is highly recommended to improve efficiency.

Setting the *database-limit* of the size option to a large value can make the PIX Firewall more vulnerable to a DoS attack by fragment flooding. Do not set the *database-limit* equal to or greater than the total number of blocks in the PIX Firewall's 1550 or 16384 memory pool. See the **show blocks** command for more details.

The **show fragment** command displays the states of the fragment databases. If the interface name is specified, only the database residing at the specified interface is displayed. The following list explains the output of the **show fragment** command:

- Chain—Maximum fragments for a single packet set by the chain option.
- Timeout—Maximum seconds set by the timeout option.

- Queue—Number of packets currently awaiting reassembly.
- Assemble—Number of packets successfully reassembled.
- Fail—Number of packets that failed to be reassembled.
- Overflow—Number of packets that overflowed the fragment database.

Use the **clear fragment** command to reset the fragment databases and defaults. This causes the PIX Firewall to discard all fragments currently waiting for reassembly, and reset the size, chain, and timeout options to their default values.

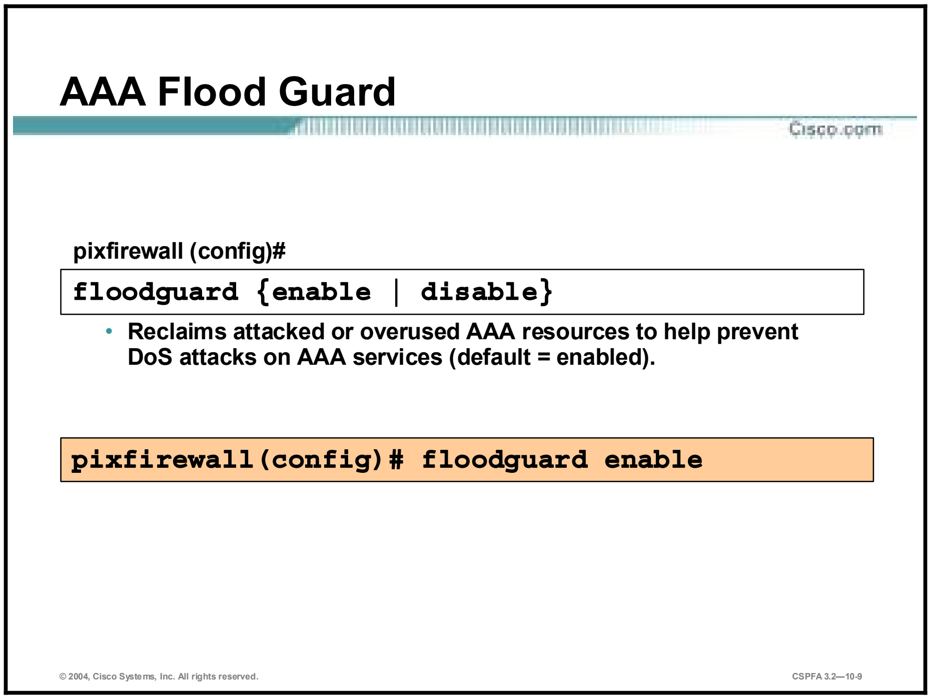
The syntax for the **fragment** commands is as follows:

fragment size ***database-limit*** **[*interface*]**

fragment chain ***chain-limit*** **[*interface*]**

fragment timeout ***seconds*** **[*interface*]**

size *database-limit*	Sets the maximum number of packets in the fragment database. The default is 200. The maximum is 1,000,000 or the total number of blocks.
interface	The PIX Firewall interface. If not specified, the command will apply to all interfaces.
chain *chain-limit*	Specifies the maximum number of packets into which a full IP packet can be fragmented. The default is 24. The maximum is 8200.
timeout *seconds*	Specifies the maximum number of seconds that a packet fragment will wait to be reassembled after the first fragment is received before being discarded. The default is five seconds. The maximum is 30 seconds.



AAA Flood Guard

pixfirewall (config)#

```
floodguard {enable | disable}
```

- Reclaims attacked or overused AAA resources to help prevent DoS attacks on AAA services (default = enabled).

```
pixfirewall(config)# floodguard enable
```

CSPFA 3.2—10-9

The **floodguard** command enables the PIX Firewall to reclaim resources if the user authentication (uauth) subsystem runs out of resources. If an inbound or outbound uauth connection is being attacked or overused, the PIX Firewall actively reclaims TCP resources. When the resources are depleted, the PIX Firewall shows messages indicating that it is out of resources or out of TCP users. If the PIX Firewall uauth subsystem is depleted, TCP user resources in different states are reclaimed, depending on urgency, in the following order:

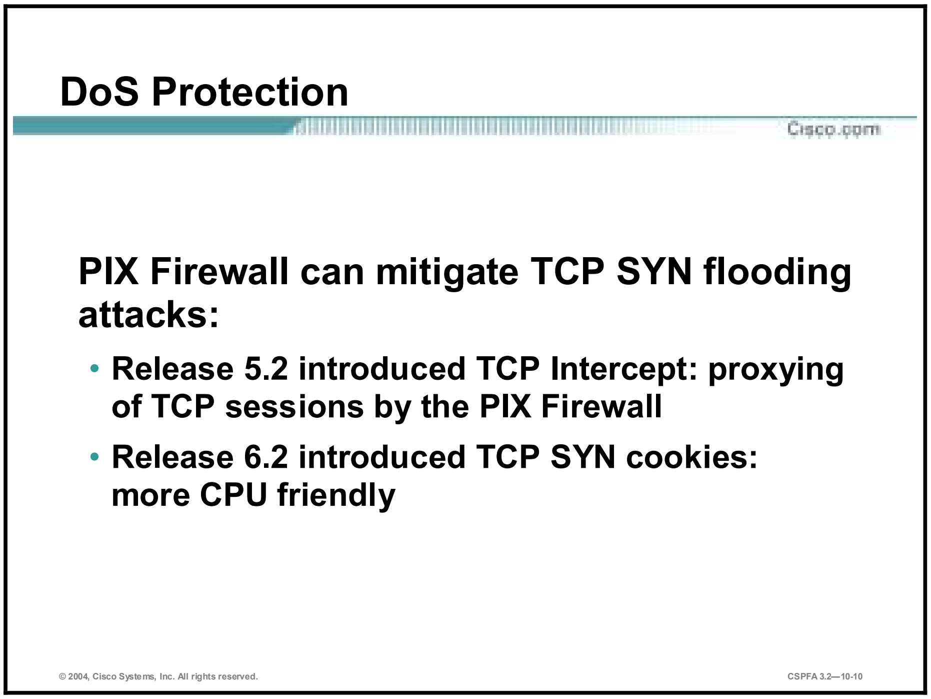
1. Timewait
2. FinWait
3. Embryonic
4. Idle

The **floodguard** command is enabled by default.

The syntax for the **floodguard** command is as follows:

floodguard {enable | disable}

enable	Enables AAA Flood Guard.
disable	Disables AAA Flood Guard.



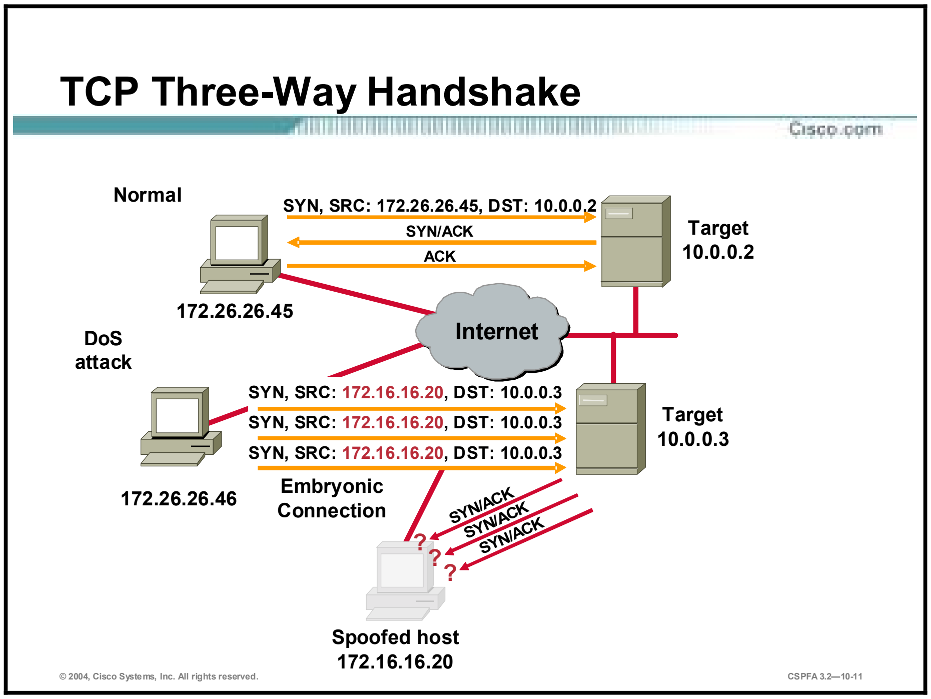
DoS Protection

PIX Firewall can mitigate TCP SYN flooding attacks:

- **Release 5.2 introduced TCP Intercept: proxying of TCP sessions by the PIX Firewall**
- **Release 6.2 introduced TCP SYN cookies: more CPU friendly**

CSPFA 3.2—10-10

Protection against various DoS attacks has increased through newer versions of PIX Firewall operating systems. Beginning in version 5.2, TCP Intercept provided for proxy resets of sessions without any knowledge or interference from the destination station. Version 6.2 introduced SYN cookies, which is another proxy verification tool that the PIX Firewall operating system uses to validate a new session. There is more on TCP Intercept and SYN cookies later in this lesson.

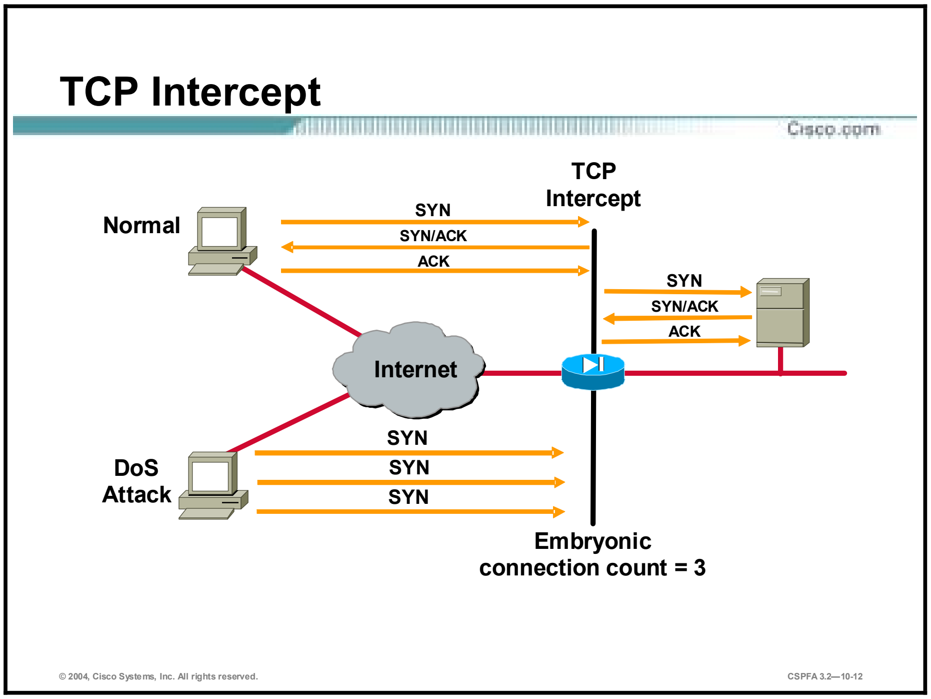


TCP Three-Way Handshake

Cisco.com

CSPFA 3.2—10-11

SYN flood attacks, also known as TCP flood or half-open connections attacks, are common DoS attacks perpetrated against IP servers. The attacker spoofs a nonexistent source IP address and floods the target with SYN packets pretending to come from the spoofed host. SYN packets to a host are the first step in the three-way handshake of a TCP-type connection; therefore, the target responds as expected with SYN-ACK packets destined to the spoofed host or hosts. Because these SYN-ACK packets are sent to hosts that do not exist, the target sits and waits for the corresponding ACK packets that never show up. This causes the target to overflow its port buffer with embryonic (half-open) connections and stop responding to legitimate requests.



TCP Intercept

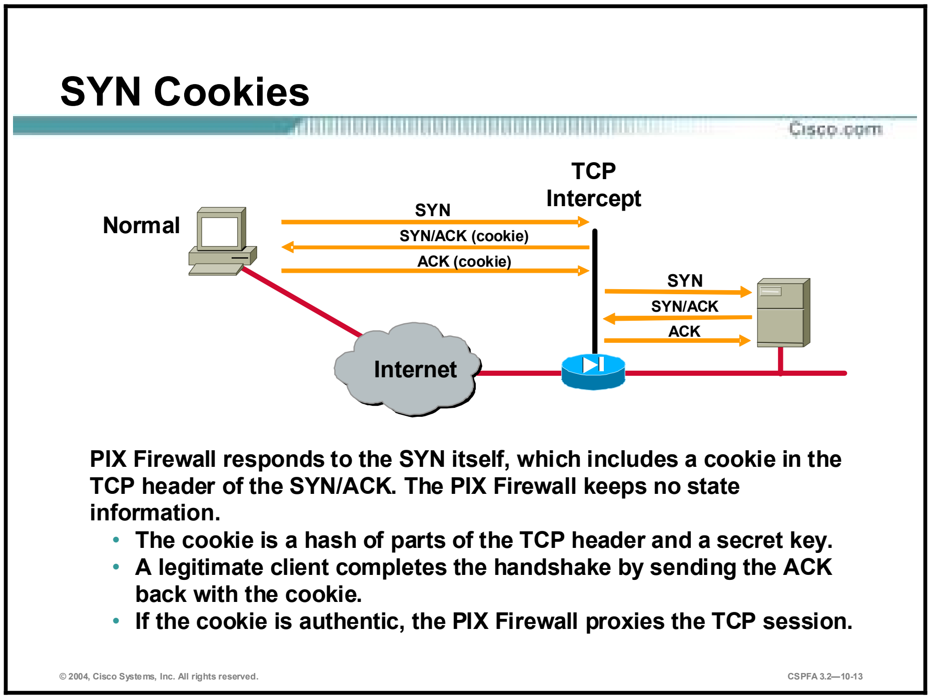
© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—10-12

In PIX Firewall Software Version 5.2, the **static** command’s SYN Flood Guard feature offers an improved mechanism for protecting systems reachable via a static access control list (ACL) from TCP SYN attacks. Previously, if an embryonic connection limit was configured in a **static** command statement, the PIX Firewall simply dropped new connection attempts once the embryonic threshold was reached. This could allow even a modest attack to stop an organization’s web traffic. For **static** command statements without an embryonic connection limit, the PIX Firewall passes all traffic. If the target of an attack has no TCP SYN attack protection or insufficient protection (like most operating systems), its embryonic connection table overloads and all traffic stops.

With the new TCP Intercept feature in versions 5.2 and higher, once the optional embryonic connection limit is reached, and until the embryonic connection count falls below this threshold, every SYN bound for the affected server is intercepted. For each SYN, the PIX Firewall responds on behalf of the server with an empty SYN/ACK segment. The PIX Firewall retains pertinent state information, drops the packet, and waits for the client’s acknowledgement. If the ACK is received, a copy of the client’s SYN segment is sent to the server, and the TCP three-way handshake is performed between the PIX Firewall and the server. Only if this three-way handshake completes will the connection be allowed to resume as normal.

The TCP Intercept feature requires no special configuration. The embryonic connection limits on both the **static** and **nat** commands include the new behavior.



SYN Cookies

PIX Firewall responds to the SYN itself, which includes a cookie in the TCP header of the SYN/ACK. The PIX Firewall keeps no state information.

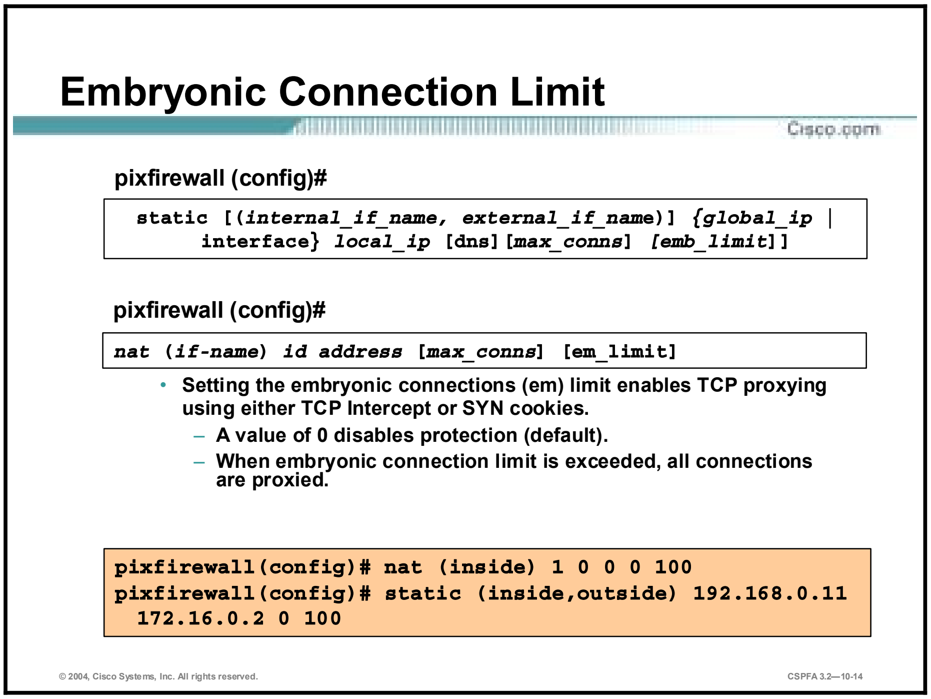
- The cookie is a hash of parts of the TCP header and a secret key.
- A legitimate client completes the handshake by sending the ACK back with the cookie.
- If the cookie is authentic, the PIX Firewall proxies the TCP session.

CSPFA 3.2—10-13

In PIX Firewall Software Version 6.2, SYN cookies were introduced. The SYN cookies feature represents a less CPU-intensive method of verifying incoming TCP sessions for validity. SYN cookies are an implementation of TCP in which servers respond to a TCP SYN request with a cookie. In the original TCP implementation, when a server received a SYN packet, it responded with a SYN-ACK, and entered the half-open state to wait for the ACK that would complete the handshake. Too many half-open connections can result in full buffers.

In the SYN cookies implementation of TCP, when the server receives a SYN packet, it responds with a SYN-ACK packet where the ACK sequence number is calculated from the source address, source port, source sequence number, destination address, destination port, and a secret seed. Then the server releases all state. If an ACK returns from the client, the server can recalculate it to determine if it is a response to a previous SYN-ACK. If so, the server can directly enter the TCP_ESTABLISHED state and open the connection. In this way, the server avoids managing a batch of potentially useless half-open connections.

The PIX Firewall, rather than the protected server, can respond using SYN cookies. This feature replaces TCP Intercept. It is more scalable in terms of performance.



Embryonic Connection Limit

Cisco.com

pixfirewall (config)#

```
static [(internal_if_name, external_if_name)] {global_ip |
     interface} local_ip [dns][max_conns] [emb_limit]]
```

pixfirewall (config)#

```
nat (if-name) id address [max_conns] [em_limit]
```

- Setting the embryonic connections (em) limit enables TCP proxying using either TCP Intercept or SYN cookies.
  - A value of 0 disables protection (default).
  - When embryonic connection limit is exceeded, all connections are proxied.

```
pixfirewall(config)# nat (inside) 1 0 0 0 100
pixfirewall(config)# static (inside,outside) 192.168.0.11
  172.16.0.2 0 100
```

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—10-14

Use the **static** command to limit the number of embryonic connections allowed to the server to protect internal hosts against DoS attacks. Use the ***em_limit*** argument to limit the number of embryonic or half-open connections that the server or servers you are trying to protect can handle. A value of zero disables protection. When the embryonic connection limit value is exceeded, all connections are proxied.

The syntax used in the **static** command for enabling the SYN Flood Guard is as follows:

static [(*internal_if_name, external_if_name*)] {*global_ip* | interface} *local_ip* [dns][netmask *mask*][*max_conns* [*emb_limit*]]

static [(*prenat_interface, postnat_interface*)] *mapped_address* | interface *real_address* [dns] [netmask *mask*] [*connection_limit* [*em_limit*]]

prenat_interface	Usually the inside interface, in which case the translation is applied to the inside address.
postnat_interface	The outside interface when prenat_interface is the inside interface. However, if the outside interface is used for prenat_interface, the translation is applied to the outside address and the postnat_interface is the inside interface.
mapped_address	The address into which real_address is translated.
interface	Specifies that the IP address should be taken from the PIX Firewall's interface. This is useful in Dynamic Host Configuration Protocol (DHCP) scenarios where the outside interface address can change.
real_address	The address to be mapped.
dns	Specifies that DNS replies that match the xlate are translated.

mask	The network mask. The mask pertains to both mapped_address and real_address. For host addresses, always use 255.255.255.255. For network addresses, use the appropriate class mask or subnet mask (for example, for Class A networks, use 255.0.0.0). An example subnet mask is 255.255.255.224.
connection_limit	The maximum connections permitted to the local_ip. The default = 0 (unlimited).
em_limit	The maximum number of embryonic connections permitted to the local_ip. The default = 0 (unlimited).

Use the **nat** command to protect external hosts against DoS attacks and to limit the number of embryonic connections from the external host. Use the ***em_limit*** argument to limit the number of embryonic or half-open connections that the server or servers you are trying to protect can handle.

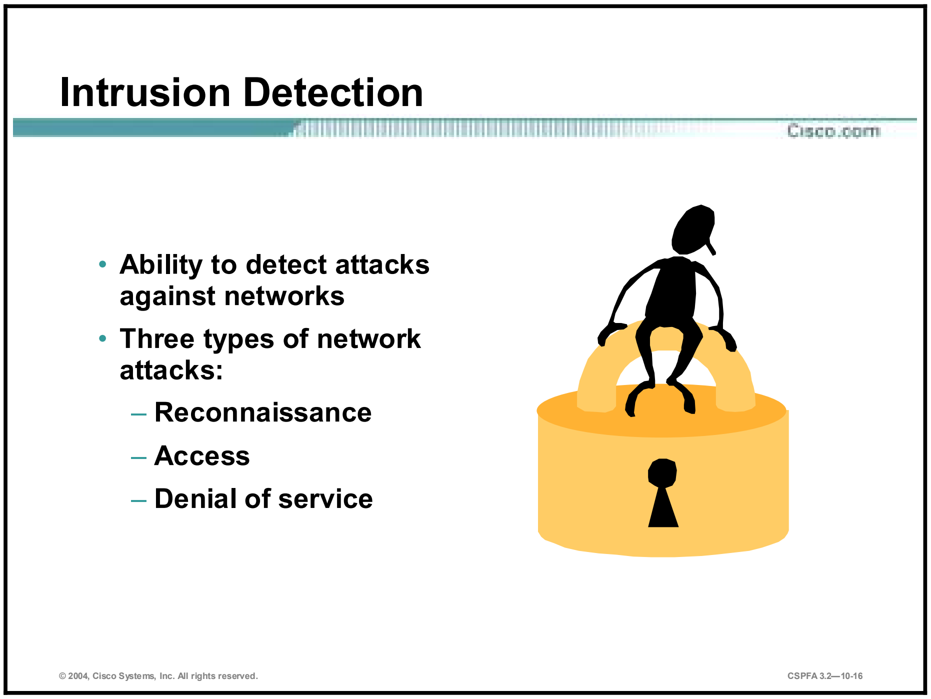
The syntax used in the **nat** command for enabling the SYN Flood Guard is as follows:

nat [(*if_name*)] *id address* [*netmask* [dns] [timeout *hh:mm:ss*] [*conn_limit* [*em_limit*]]]

if_name	The network interface name.
id	A number used for matching with a corresponding global pool of IP addresses. The matching global pool must use the same ID.
address	The IP address of hosts or networks that will be translated to a global pool of IP addresses.
netmask	The network mask for the address.
dns	Specifies that DNS replies that match the xlate are translated.
timeout *hh:mm:ss*	The timeout interval for the translation slot. Timeout occurs only if no TCP or UDP connection is actively using the translation.
conn_limit	The maximum TCP connections permitted from the host you specify.
em_limit	The embryonic connection limit. The default is 0, which means unlimited connections. Set it lower for slower systems, higher for faster systems.

Intrusion Detection

This topic explains the intrusion detection capabilities of the PIX Firewall.



Intrusion Detection

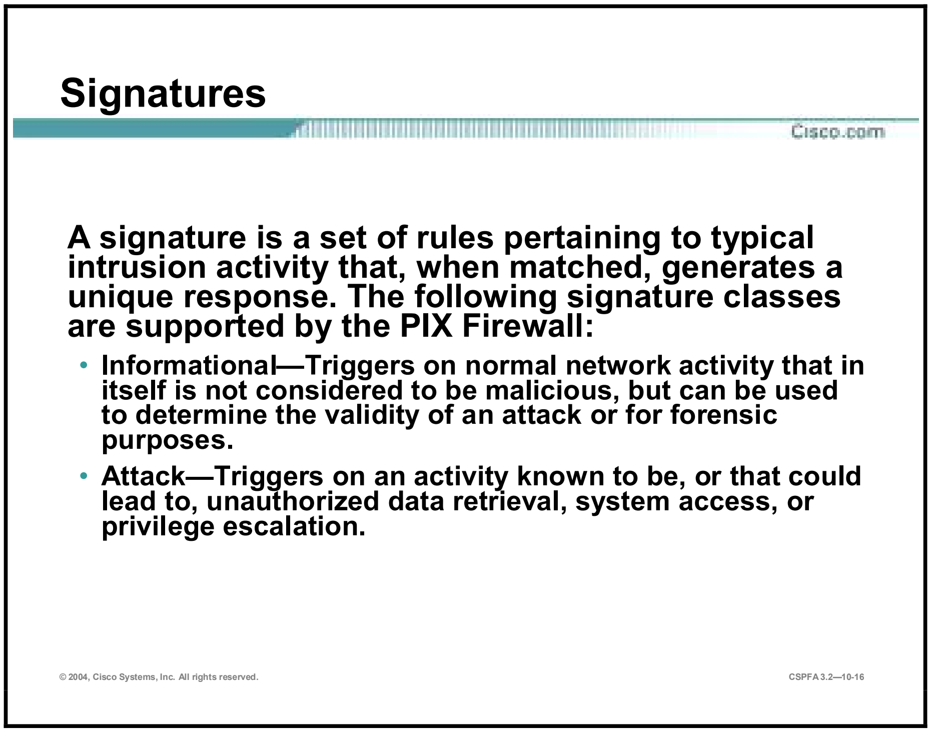
Cisco.com

- Ability to detect attacks against networks
- Three types of network attacks:
  - Reconnaissance
  - Access
  - Denial of service

CSPFA 3.2—10-16

PIX Firewall Software Versions 5.2 and higher have Cisco Intrusion Detection System (IDS) capabilities. Intrusion detection is the ability to detect attacks against your network. There are three types of network attacks:

- Reconnaissance attacks—An intruder is attempting to discover and map systems, services, or vulnerabilities.
- Access attacks—An intruder attacks networks or systems to retrieve data, gain access, or escalate their access privilege.
- Denial of service (DoS) attacks—An intruder attacks your network in such a way that damages or corrupts your computer system, or denies you and others access to your networks, systems, or services.



Signatures

A signature is a set of rules pertaining to typical intrusion activity that, when matched, generates a unique response. The following signature classes are supported by the PIX Firewall:

- Informational—Triggers on normal network activity that in itself is not considered to be malicious, but can be used to determine the validity of an attack or for forensic purposes.
- Attack—Triggers on an activity known to be, or that could lead to, unauthorized data retrieval, system access, or privilege escalation.

CSPFA 3.2—10-16

The PIX Firewall performs intrusion detection by using intrusion detection signatures. A signature is a set of rules pertaining to typical intrusion activity. Highly skilled network engineers research known attacks and vulnerabilities and can develop signatures to detect these attacks and vulnerabilities.

With intrusion detection enabled, the PIX Firewall can detect signatures and generate a response when a set of rules is matched to network activity. It can monitor packets for more than 55 intrusion detection signatures and can be configured to send an alarm to a Syslog server or a server running Cisco's Security Monitor, drop the packet, or reset the TCP connection. The signatures supported by the PIX Firewall are a subset of the signatures supported by the Cisco IDS product family.

The PIX Firewall can detect two different types of signatures: informational signatures and attack signatures. Information class signatures are signatures that are triggered by normal network activity that in itself is not considered to be malicious, but can be used to determine the validity of an attack or for forensics purposes. Attack class signatures are signatures that are triggered by an activity known to be, or that could lead to, unauthorized data retrieval, system access, or privileged escalation.

The table lists examples of the IDS signatures supported by the PIX Firewall.

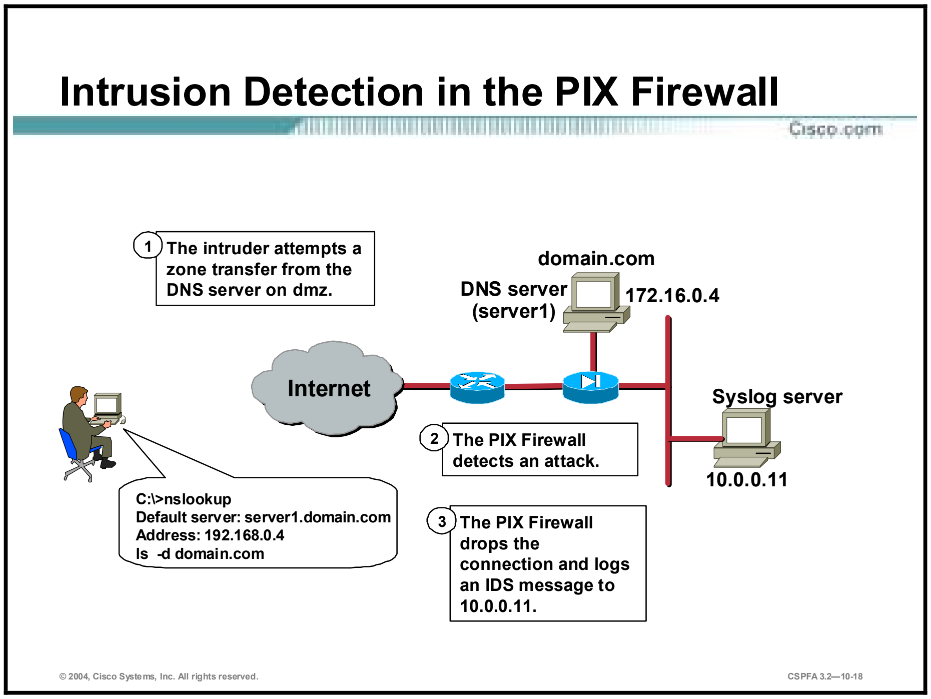
Message #	Signature ID	Signature Title	Signature Type
400000	1000	IP Options-Bad Option List	Informational
400001	1001	IP Options-Record Packet Route	Informational
400002	1002	IP Options-Timestamp	Informational
400003	1003	IP Options-Security	Informational
400007	1100	IP Fragment Attack	Attack

Message #	Signature ID	Signature Title	Signature Type
400010	2000	ICMP Echo Reply	Informational
400011	2001	ICMP Host Unreachable	Informational
400013	2003	ICMP Redirect	Informational
400014	2004	ICMP Echo Request	Informational
400023	2150	Fragmented ICMP Traffic	Attack
400024	2151	Large ICMP Traffic	Attack
400025	2154	Ping of Death Attack	Attack
400032	4051	UDP Snork Attack	Attack
400035	6051	DNS Zone Transfer	Attack
400041	6103	Proxied RPC Request	Attack

IDS Syslog messages all start with %PIX-4-4000nn and have the following format: %PIX-4-4000nn IDS: *sig_num sig_msg* from *ip_addr* to **ip_addr** on interface *int_name*. For example, %PIX-4-400013 IDS:2003 ICMP redirect from 10.4.1.2 to 10.2.1.1 on interface dmz, and %PIX-4-400032 IDS:4051 UDP Snork attack from 10.1.1.1 to 192.168.1.1 on interface outside.

Refer to *System Log Messages for the Cisco Secure PIX Firewall Version 5.2* or *System Log Messages for the Cisco Secure PIX Firewall Version 5.3* for a list of all supported messages. You can view these documents online at the following sites:

- www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_v52/syslog/index.htm
- www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_v53/syslog/index.htm



Intrusion Detection in the PIX Firewall

1 The intruder attempts a zone transfer from the DNS server on dmz.

2 The PIX Firewall detects an attack.

3 The PIX Firewall drops the connection and logs an IDS message to 10.0.0.11.

```
C:\>nslookup
Default server: server1.domain.com
Address: 192.168.0.4
ls  -d domain.com
```

CSPFA 3.2—10-18

Intrusion detection, or auditing, is enabled on the PIX Firewall with the **ip audit** commands. Using the **ip audit** commands, audit policies can be created to specify the traffic that is audited or to designate actions to be taken when a signature is detected. After a policy is created, it can be applied to any PIX Firewall interface.

Each interface can have two policies: one for informational signatures and one for attack signatures. If you want them both to be active simultaneously, they should share the same policy name. When a policy for a given signature class is created and applied to an interface, all supported signatures of that class are monitored unless you disable them with the **ip audit signature disable** command.

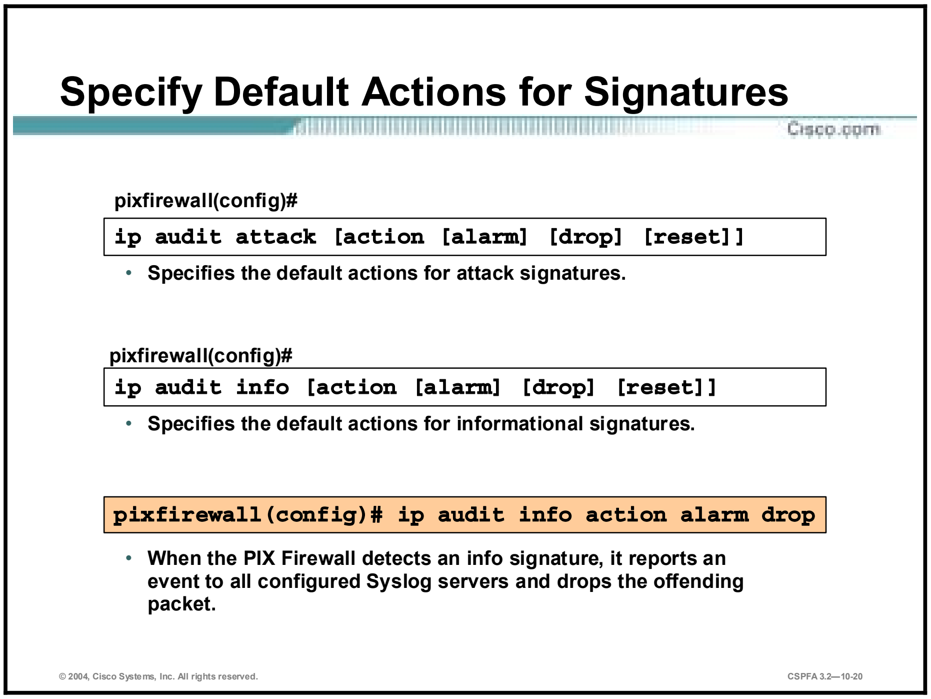
The PIX Firewall supports both inbound and outbound auditing. Auditing is performed by looking at the IP packets as they arrive at an input interface. For example, if an attack policy is applied to the outside interface, attack signatures are triggered when attack traffic arrives at the outside interface in an inward direction, either as inbound traffic or as return traffic from an outbound connection.

In the figure, the PIX Firewall has an attack policy, which contains the alarm and drop actions, applied to its outside interface. Therefore, the following series of events takes place:

Step 1 The intruder attempts to transfer a DNS zone from the DNS server on the DMZ.

Step 2 The PIX Firewall detects an attack.

Step 3 The PIX Firewall drops the connection and sends an IDS Syslog message to the Syslog server at 10.0.0.3.



Specify Default Actions for Signatures

pixfirewall(config)#

```
ip audit attack [action [alarm] [drop] [reset]]
```

- Specifies the default actions for attack signatures.

pixfirewall(config)#

```
ip audit info [action [alarm] [drop] [reset]]
```

- Specifies the default actions for informational signatures.

```
pixfirewall(config)# ip audit info action alarm drop
```

- When the PIX Firewall detects an info signature, it reports an event to all configured Syslog servers and drops the offending packet.

CSPFA 3.2—10-20

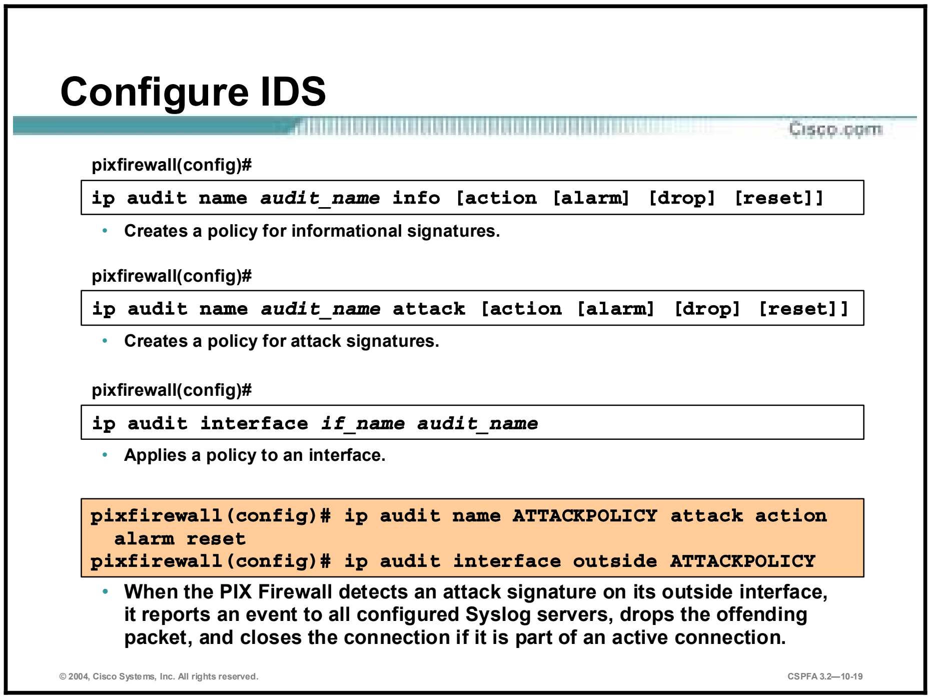
The **ip audit attack** command specifies the default actions to be taken for attack signatures. The **no ip audit attack** command resets the action to be taken for attack signatures to the default action. The **show ip audit attack** command displays the default attack actions. The **ip audit info**, **no ip audit info**, and **show ip audit info** commands perform the same functions for signatures classified as informational. Specify the **ip audit info** command without an action option to cancel event reactions.

The syntax for these **ip audit** commands is as follows:

ip audit attack [*action* [*alarm*] [*drop*] [*reset*]]

ip audit info [*action* [*alarm*] [*drop*] [*reset*]]

audit attack	Specifies the default actions to be taken for attack signatures.
audit info	Specifies the default actions to be taken for informational signatures.
action *actions*	The ***alarm*** option indicates that when a signature match is detected in a packet, the PIX Firewall reports the event to all configured Syslog servers. The ***drop*** option drops the offending packet. The ***reset*** option drops the offending packet and closes the connection if it is part of an active connection. The default is ***alarm***.



Configure IDS

pixfirewall(config)#

```
ip audit name audit_name info [action [alarm] [drop] [reset]]
```

- Creates a policy for informational signatures.

pixfirewall(config)#

```
ip audit name audit_name attack [action [alarm] [drop] [reset]]
```

- Creates a policy for attack signatures.

pixfirewall(config)#

```
ip audit interface if_name audit_name
```

- Applies a policy to an interface.

```
pixfirewall(config)# ip audit name ATTACKPOLICY attack action
  alarm reset
pixfirewall(config)# ip audit interface outside ATTACKPOLICY
```

- When the PIX Firewall detects an attack signature on its outside interface, it reports an event to all configured Syslog servers, drops the offending packet, and closes the connection if it is part of an active connection.

CSPFA 3.2—10-19

Use the **ip audit** command to override the IDS signature defaults. First create a policy with the **ip audit name** command, and then apply the policy to an interface with the **ip audit interface** command.

There are two variations of the **ip audit name** command: **ip audit name info** and **ip audit name attack**. The **ip audit name info** command is used to create policies for signatures classified as informational. All informational signatures, except those disabled or excluded by the **ip audit signature** command, become part of the policy. The **ip audit name attack** command performs the same function for signatures classified as attack signatures.

The **ip audit name** commands also allow you to specify actions to be taken when a signature is triggered. If a policy is defined without actions, the default actions take effect. The default action for both attack and info signatures is alarm.

The **no ip audit name** command can be used to remove an audit policy. The **show ip audit name** command displays audit policies. Use the **no ip audit interface** command to remove a policy from an interface. Use the **show ip audit interface** command to display the interface configuration.

The syntax for these **ip audit** commands is as follows:

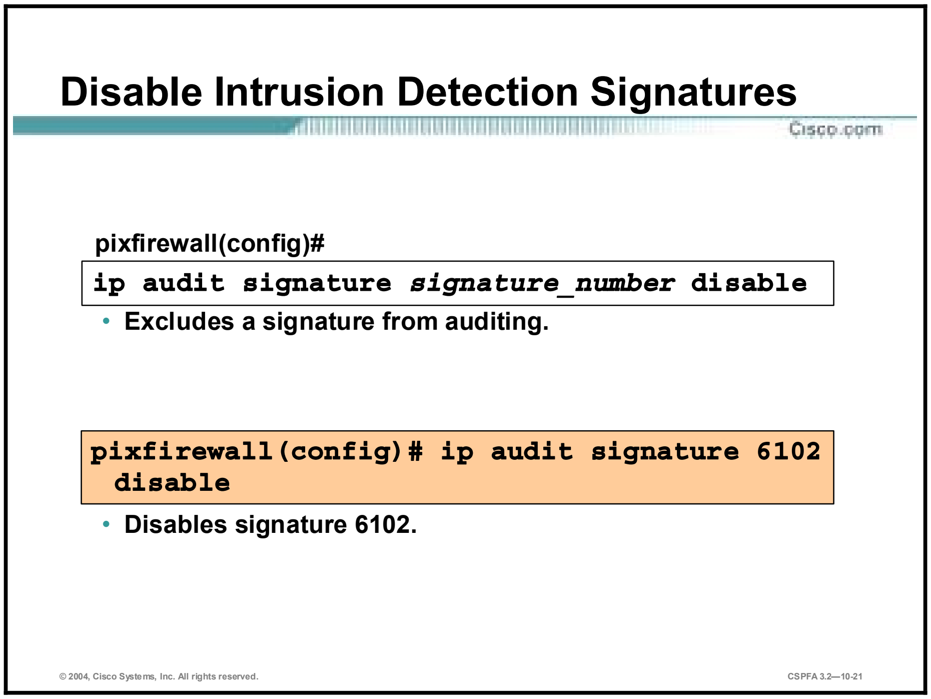
ip audit name ***audit_name*** **info [*****action*** **[*****alarm*****] [*****drop*****] [*****reset*****]]**

ip audit name ***audit_name*** **attack [*****action*** **[*****alarm*****] [*****drop*****] [*****reset*****]]**

ip audit interface ***if_name audit_name***

audit_name	Audits the policy name viewed with the **show ip audit name** command.

action *actions*	The ***alarm*** option indicates that when a signature match is detected in a packet, the PIX Firewall reports the event to all configured Syslog servers. The ***drop*** option drops the offending packet. The ***reset*** option drops the offending packet and closes the connection if it is part of an active connection. The default is ***alarm***.
audit interface	Applies an audit specification or policy (via the **ip audit name** command) to an interface.
if_name	The interface to which the policy is applied.



Disable Intrusion Detection Signatures

pixfirewall(config)#

```
ip audit signature signature_number disable
```

- Excludes a signature from auditing.

```
pixfirewall(config)# ip audit signature 6102
 disable
```

- Disables signature 6102.

CSPFA 3.2—10-21

If you wish to exclude a signature from auditing, use the **ip audit signature disable** command. The **no ip audit signature** command is used to re-enable a signature, and the **show ip audit signature** command displays disabled signatures.

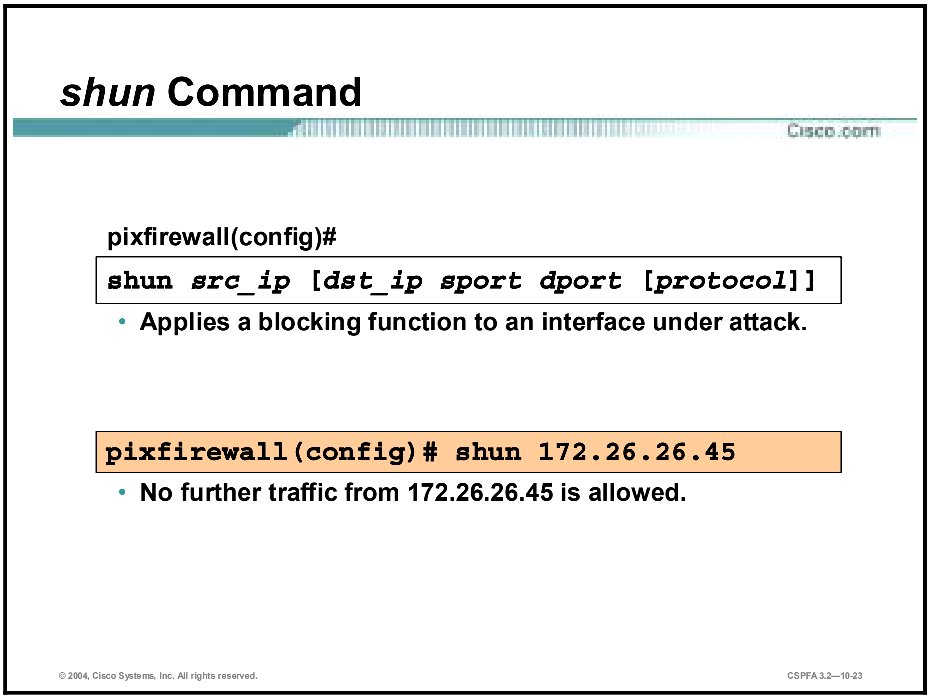
The syntax for the **ip audit signature** command is as follows:

ip audit signature *signature_number* disable

audit signature	Specifies what messages to display, attaches a global policy to a signature, and disables or excludes a signature from auditing.
signature_number	Intrusion detection signature number.

Shunning

This topic explains the PIX Firewall's shunning capabilities.



shun Command

pixfirewall(config)#

```
shun src_ip [dst_ip sport dport [protocol]]
```

- Applies a blocking function to an interface under attack.

```
pixfirewall(config)# shun 172.26.26.45
```

- No further traffic from 172.26.26.45 is allowed.

CSPFA 3.2—10-23

The PIX Firewall's shun feature allows a PIX Firewall, when combined with a Cisco IDS Sensor, to dynamically respond to an attacking host by preventing new connections and disallowing packets from any existing connection. A Cisco IDS device instructs the PIX Firewall to shun sources of traffic when those sources of traffic are determined to be malicious.

The **shun** command, intended for use primarily by a Cisco IDS device, applies a blocking function to an interface receiving an attack. The **shun** command is not interface specific. Traffic from the specified source address is dropped no matter which interface it arrives on. Packets containing the IP source address of the attacking host are dropped and logged until the blocking function is removed manually or by the Cisco IDS master unit. No traffic from the IP source address is allowed to traverse the PIX Firewall, and any remaining connections time out as part of the normal architecture. The blocking function of the **shun** command is applied whether or not a connection with the specified host address is currently active.

The offending host can be inside or outside of the PIX Firewall. If the **shun** command is used only with the source IP address of the host, no further traffic from the offending host is allowed.

The **show shun** command displays all shuns currently enabled in the exact format specified. The **no** form of the **shun** command disables a shun based on src_ip.

Note PIX Firewall shunning is supported in Cisco IDS 3.0.

The **show shun** command displays all shuns currently enabled in the exact format specified.

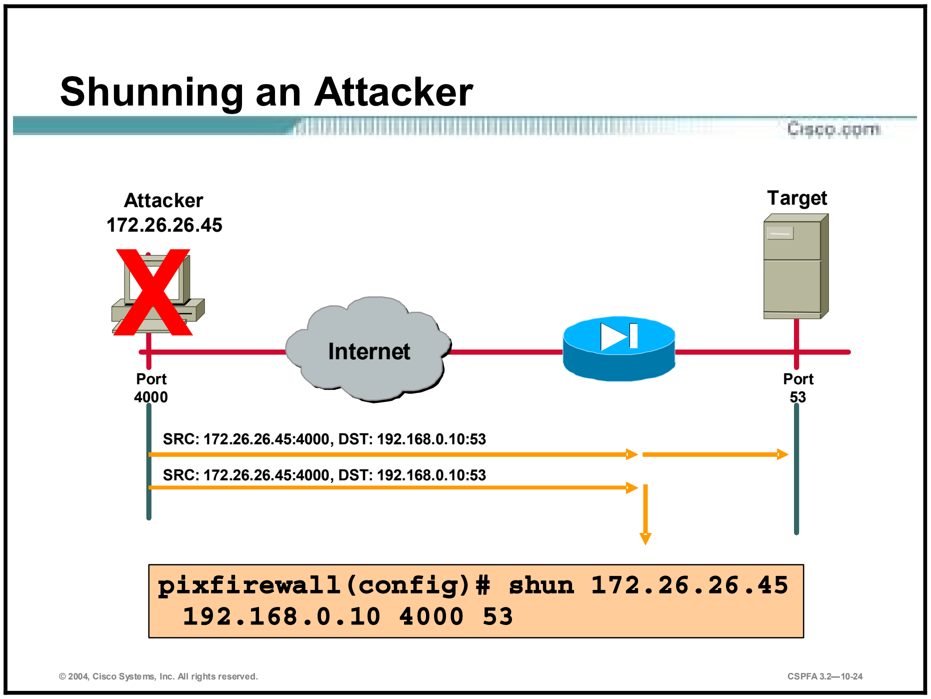
The syntax for the shun command is as follows:

shun ***src_ip*** **[*dst_ip sport dport* [*protocol*]]**

show shun ***src_ip***

clear shun [statistics]

clear	Disables all shuns currently enabled and clears shun statistics. Specifying statistics only clears the counters for that interface.
dport	The destination port of the connection causing the shun.
dst_ip	The address of the of the target host.
protocol	The optional IP protocol, such as UDP or TCP.
sport	The source port of the connection causing the shun.
src_ip	The address of the attacking host.
statistics	Clears only interface counters.



Shunning an Attacker

Cisco.com
Attacker
172.26.26.45
Target
Internet
Port
4000
Port
53
SRC: 172.26.26.45:4000, DST: 192.168.0.10:53
SRC: 172.26.26.45:4000, DST: 192.168.0.10:53

```
pixfirewall(config)# shun 172.26.26.45
  192.168.0.10 4000 53
```

CSPFA 3.2—10-24

In the figure, host 172.26.26.45 has been attempting a DNS zone transfer from host 192.168.0.10 using a source port other than the well-known DNS port of TCP 53. The offending host (172.26.26.45) has made a connection with the victim (192.168.0.10) with TCP. The connection in the PIX Firewall connection table reads as follows:

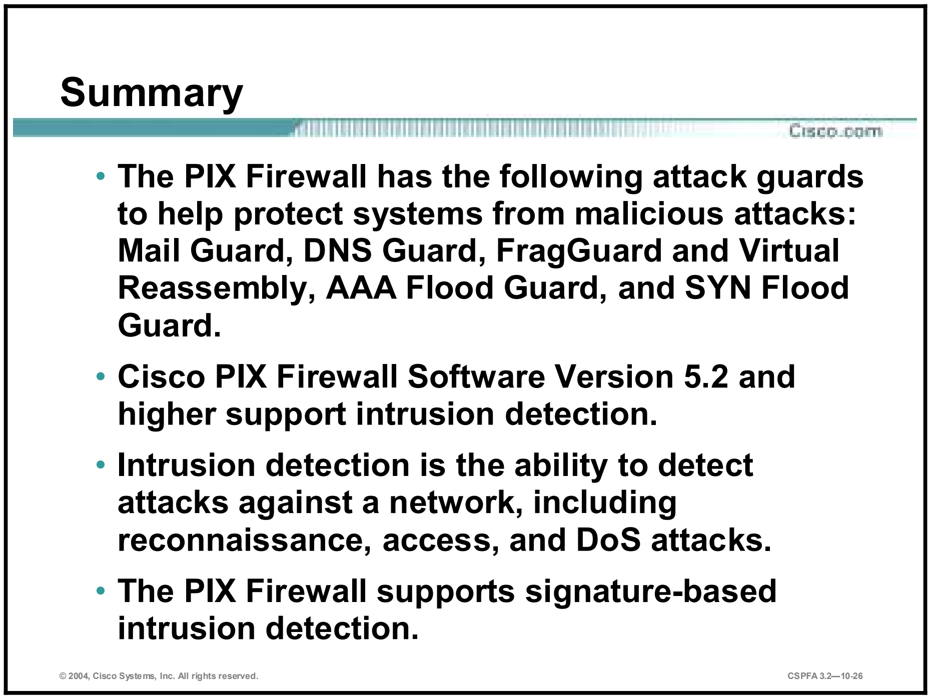
```
172.26.26.45, 4000-> 10.0.0.11 PROT TCP
```

If the **shun** command is applied as shown in the figure, the PIX Firewall deletes the connection from its connection table and prevents packets from 172.26.26.45 from reaching the inside host.

Note	The PIX Firewall configuration contains a static mapping of host 10.0.0.11 to global address 192.168.0.10.

Summary

This topic summarizes what you learned in this lesson.

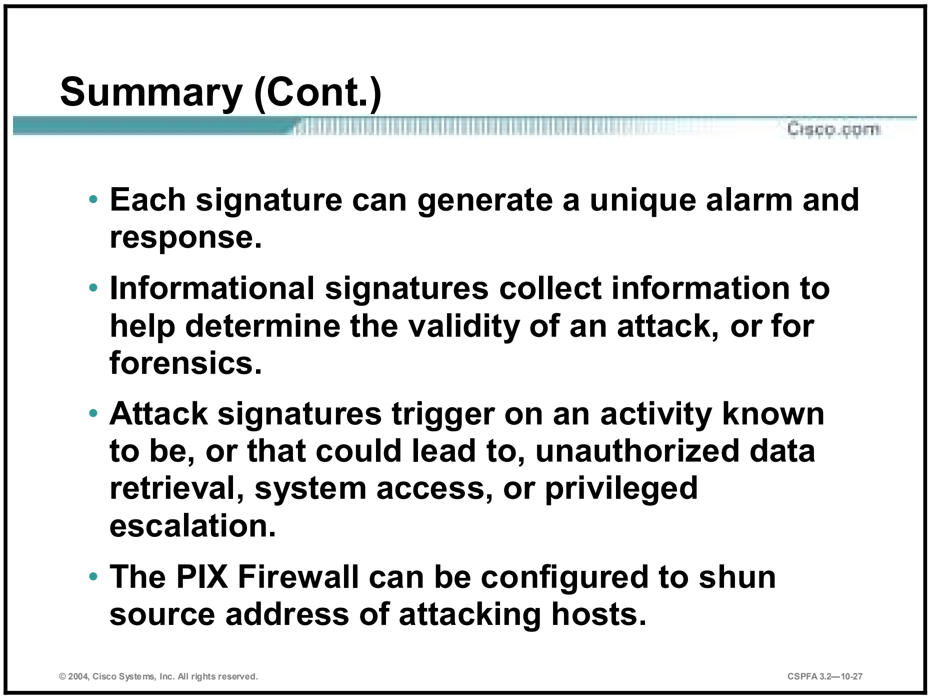


Summary

Cisco.com

- The PIX Firewall has the following attack guards to help protect systems from malicious attacks: Mail Guard, DNS Guard, FragGuard and Virtual Reassembly, AAA Flood Guard, and SYN Flood Guard.
- Cisco PIX Firewall Software Version 5.2 and higher support intrusion detection.
- Intrusion detection is the ability to detect attacks against a network, including reconnaissance, access, and DoS attacks.
- The PIX Firewall supports signature-based intrusion detection.

CSPFA 3.2—10-26



Summary (Cont.)

Cisco.com

- Each signature can generate a unique alarm and response.
- Informational signatures collect information to help determine the validity of an attack, or for forensics.
- Attack signatures trigger on an activity known to be, or that could lead to, unauthorized data retrieval, system access, or privileged escalation.
- The PIX Firewall can be configured to shun source address of attacking hosts.

CSPFA 3.2—10-27

Lab Exercise—Configure Intrusion Detection

Complete the following lab exercises to practice what you have learned in this chapter.

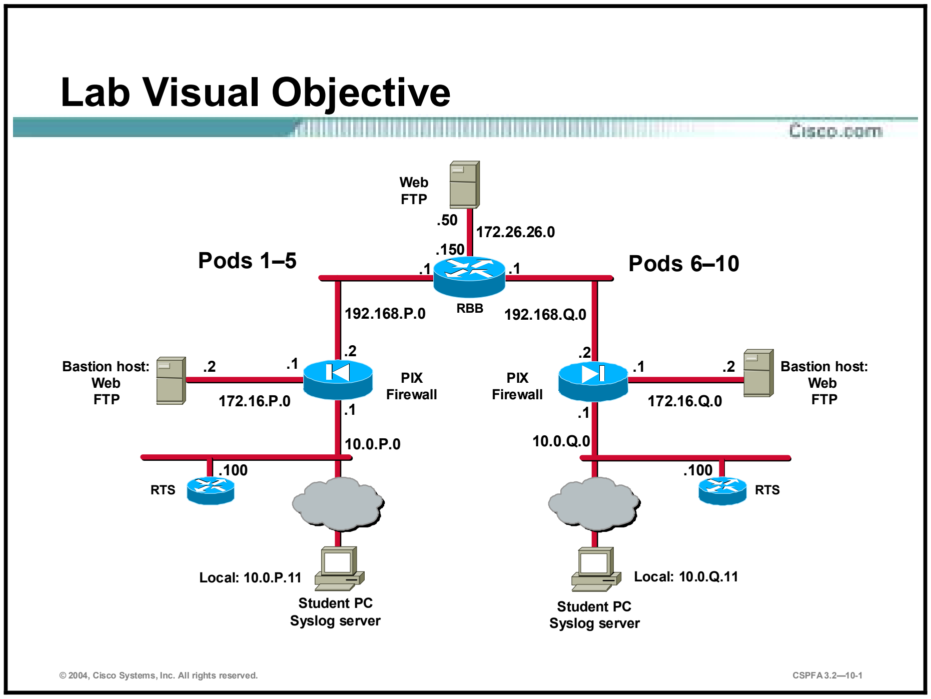
Objectives

In this lab exercise you will complete the following tasks:

- Configure the use of Cisco Intrusion Detection System (IDS) information signatures and send Cisco IDS Syslog output to a Syslog server.
- Configure the use of IDS attack signatures and send Cisco IDS Syslog output to a Syslog server.

Visual Objective

The following illustration displays the lab topology for your classroom environment.



Lab Visual Objective

Task 1—Configure the Use of IDS Information Signatures and Send Cisco IDS Syslog Output to a Syslog Server

Complete the following steps to configure the use of Cisco IDS signatures and to send Cisco IDS Syslog output to a Syslog server:

Step 1 Turn off console logging:

```
pixP(config)# no logging console debug
```

Step 2 Verify that you can ping a peer pod's internal host from your Windows command:

```
C:\>ping 192.168.Q.10
Pinging 192.168.Q.10 with 32 bytes of data:
Reply from 192.168.Q.10: bytes=32 time<10ms TTL=125
Reply from 192.168.Q.10: bytes=32 time<10ms TTL=125
Reply from 192.168.Q.10: bytes=32 time<10ms TTL=125
Reply from 192.168.Q.10: bytes=32 time<10ms TTL=125
```

(where Q = peer pod number)

Step 3 Specify an information signature policy on your PIX Firewall:

```
pixP(config)# ip audit name INFOPOLICY info action alarm reset
```

Step 4 Apply the information signature policy to the outside interface:

```
pixP(config)# ip audit interface outside INFOPOLICY
```

Step 5 Open and minimize the Kiwi Syslog Daemon on your desktop.

Step 6 Return to your Windows command line and attempt to ping your peer pod's internal host. The ping should fail.

```
C:\>ping 192.168.Q.10
Pinging 192.168.Q.10 with 32 bytes of data:


Request timed out.
Request timed out.
Request timed out.
Request timed out.
```

(where Q = peer pod number)

Step 7 Observe the messages that appear on the Kiwi Syslog Daemon display. The log should be similar to the following:

```
%PIX-4-400010: IDS:2000 ICMP echo request from 192.168.Q.10 to
192.168.P.10 on interface outside
%PIX-4-400010: IDS:2000 ICMP echo request from 192.168.Q.10 to
192.168.P.10 on interface outside
%PIX-4-400010: IDS:2000 ICMP echo request from 192.168.Q.10 to
192.168.P.10 on interface outside
%PIX-4-400010: IDS:2000 ICMP echo request from 192.168.Q.10 to
192.168.P.10 on interface outside
```

(where P = pod number, and Q = peer pod number)

Step 8 Remove the information signature policy from the outside interface:

```
pixP(config)# no ip audit interface outside INFOPOLICY
```

Step 9 Remove the audit policy audit_name:

```
pixP(config)# no ip audit name INFOPOLICY
```

Step 10 Verify that the information signature policy has been removed from the outside interface, the default informational actions have been restored, and the ip audit name has been removed:

```
pixP(config)# show ip audit interface
pixP(config)# show ip audit info
ip audit info action alarm
pixP(config)# show ip audit name
```

Task 2—Configure the Use of IDS Attack Signatures and Send Cisco IDS Syslog Output to a Syslog Server

Complete the following steps to configure the use of IDS attack signatures and send IDS Syslog output to a Syslog server:

Step 1 From your Windows NT command line, ping your bastion host with an ICMP packet size of **1000**:

```
C:\>ping -l 1000 172.16.P.2
```

```
Pinging 172.16.P.2 with 1000 bytes of data:


Reply from 172.16.P.2: bytes=1000 time<10ms TTL=128
Reply from 172.16.P.2: bytes=1000 time<10ms TTL=128
Reply from 172.16.P.2: bytes=1000 time<10ms TTL=128
Reply from 172.16.P.2: bytes=1000 time<10ms TTL=128
```

(where P = pod number)

Step 2 Specify an attack policy:

```
pixP(config)# ip audit name ATTACKPOLICY attack action alarm reset
```

Step 3 Apply the attack policy to the inside interface:

```
pixP(config)# ip audit interface inside ATTACKPOLICY
```

Step 4 From your Windows NT command line, ping your bastion host with an ICMP packet size of **10000**:

```
C:\>ping -l 10000 172.16.P.2
Pinging 172.16.P.2 with 10000 bytes of data:


Request timed out.
Request timed out.
Request timed out.
Request timed out.
```

(where P = pod number)

Step 5 Observe the messages that appear on the Kiwi Syslog Daemon display. The log should be similar to the following:

```
%PIX-4-400025: IDS:2154 ICMP ping of death from 10.1.P.11 to
172.16.P.2 on interface inside
%PIX-4-400023: IDS:2150 ICMP fragment from 10.1.P.11 to 172.16.P.2 on
interface inside
%PIX-4-400023: IDS:2150 ICMP fragment from 10.1.P.11 to 172.16.P.2 on
interface inside
%PIX-4-400023: IDS:2150 ICMP fragment from 10.1.P.11 to 172.16.P.2 on
interface inside
%PIX-4-400023: IDS:2150 ICMP fragment from 10.1.P.11 to 172.16.P.2 on
interface inside
%PIX-4-400023: IDS:2150 ICMP fragment from 10.1.P.11 to 172.16.P.2 on
interface inside
%PIX-4-400023: IDS:2150 ICMP fragment from 10.1.P.11 to 172.16.P.2 on
interface inside
%PIX-4-400025: IDS:2154 ICMP ping of death from 10.1.P.11 to
172.16.P.2 on interface inside
%PIX-4-400023: IDS:2150 ICMP fragment from 10.1.P.11 to 172.16.P.2 on
interface inside
%PIX-4-400023: IDS:2150 ICMP fragment from 10.1.P.11 to 172.16.P.2 on
interface inside
```

```
%PIX-4-400023: IDS:2150 ICMP fragment from 10.1.P.11 to 172.16.P.2 on
interface inside

%PIX-4-400023: IDS:2150 ICMP fragment from 10.1.P.11 to 172.16.P.2 on
interface inside

%PIX-4-400023: IDS:2150 ICMP fragment from 10.1.P.11 to 172.16.P.2 on
interface inside

%PIX-4-400023: IDS:2150 ICMP fragment from 10.1.P.11 to 172.16.P.2 on
interface inside
```

(where P = pod number)

Step 6 Remove the attack policy from the inside interface:

```
pixP(config)# no ip audit interface inside ATTACKPOLICY
```

Step 7 Remove the audit policy:

```
pixP(config)# no ip audit name ATTACKPOLICY
```

Step 8 Verify that the attack policy has been removed from the inside interface, the default attack actions have been restored, and the ip audit name has been removed:

```
pixP(config)# show ip audit interface
pixP(config)# show ip audit attack
ip audit attack action alarm
pixP(config)# show ip audit name
```

Step 9 Save your configuration:

```
pixP(config)# write memory
```

Step 10 Turn logging off:

```
pixP(config)# no logging on
```

11

Authentication, Authorization, and Accounting

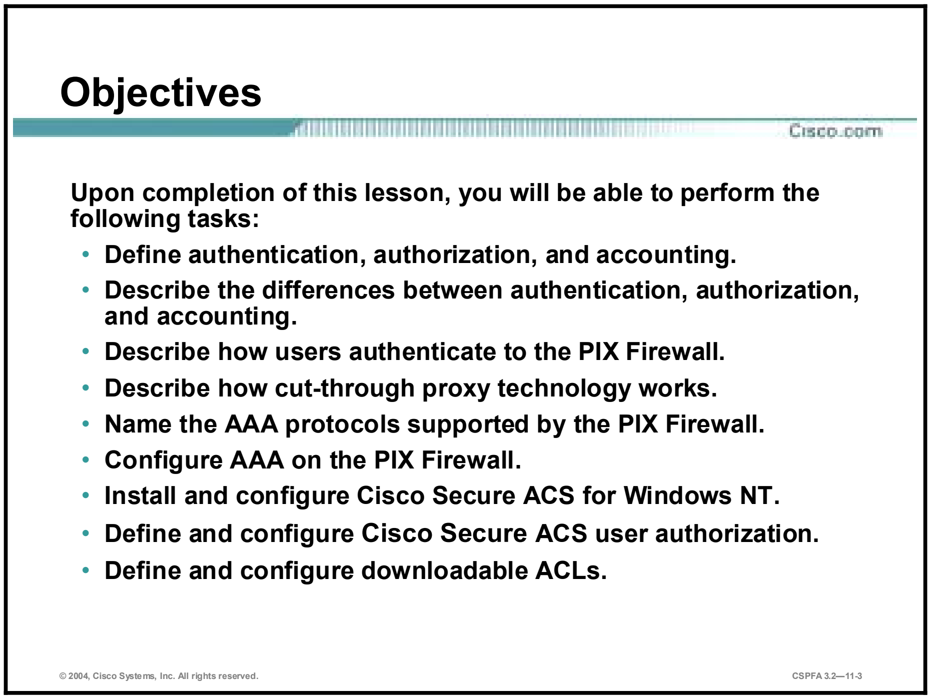
Overview

This lesson includes the following topics:

- Objectives
- Introduction
- Installation of Cisco Secure ACS for Windows 2000
- Authentication configuration
- Authorization configuration
- Downloadable ACLs
- Accounting configuration
- Troubleshooting the AAA configuration
- Summary
- Lab exercise

Objectives

This topic lists the lesson's objectives.



Objectives

Cisco.com

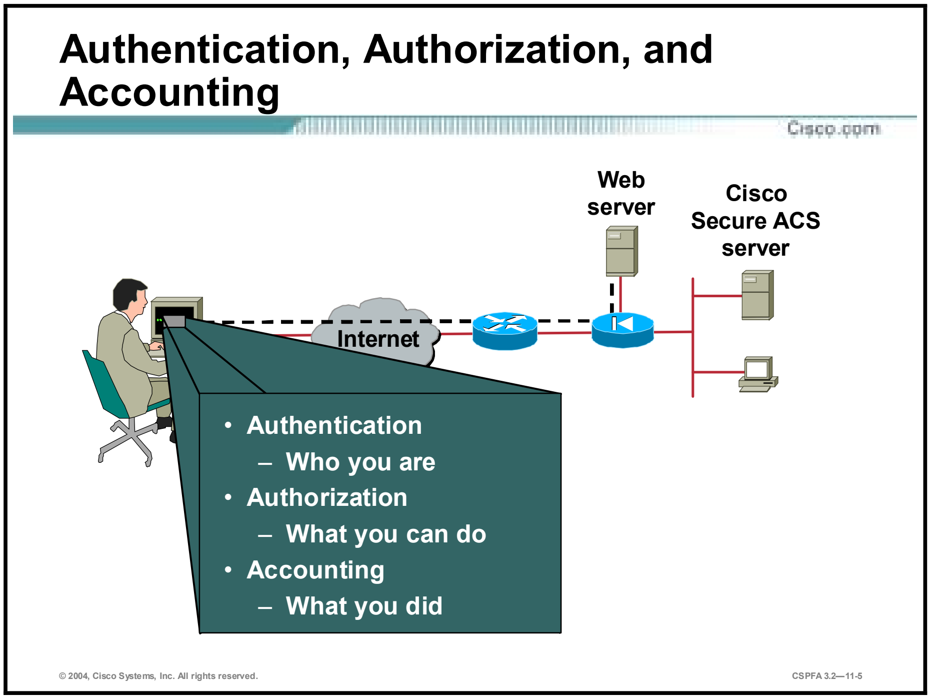
Upon completion of this lesson, you will be able to perform the following tasks:

- **Define authentication, authorization, and accounting.**
- **Describe the differences between authentication, authorization, and accounting.**
- **Describe how users authenticate to the PIX Firewall.**
- **Describe how cut-through proxy technology works.**
- **Name the AAA protocols supported by the PIX Firewall.**
- **Configure AAA on the PIX Firewall.**
- **Install and configure Cisco Secure ACS for Windows NT.**
- **Define and configure Cisco Secure ACS user authorization.**
- **Define and configure downloadable ACLs.**

CSPFA 3.2—11-3

Introduction

This topic introduces the authentication, authorization, and accounting concepts and how the Cisco PIX Firewall supports them.



Authentication, Authorization, and Accounting

Cisco.com

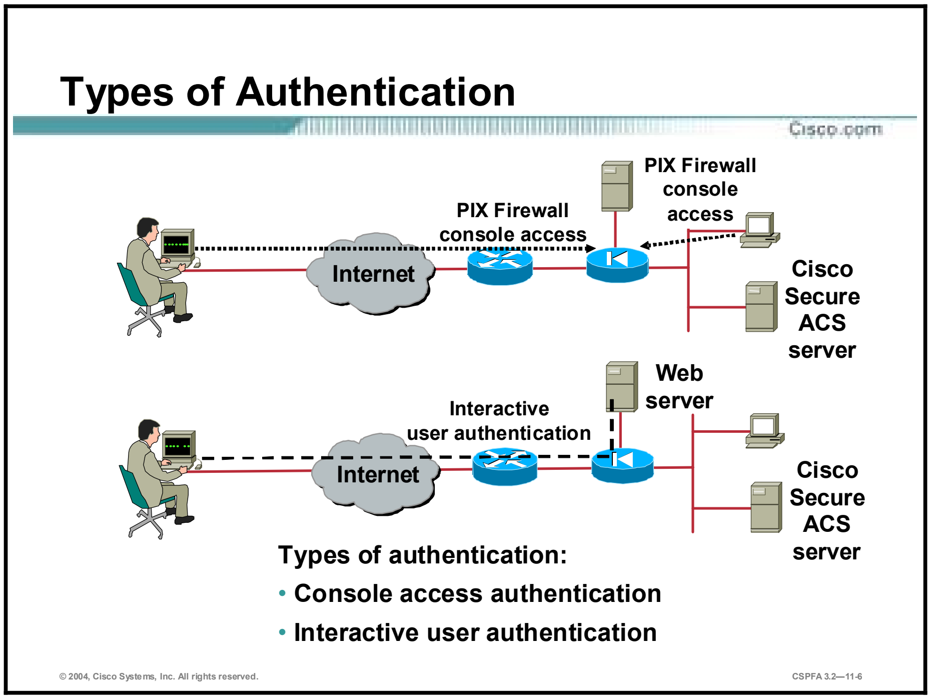
- Authentication
  - Who you are
- Authorization
  - What you can do
- Accounting
  - What you did

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—11-5

Authentication, Authorization, and Accounting (AAA) is used to tell the PIX Firewall who the user is, what the user can do, and what the user did. Authentication is valid without authorization. Authorization is never valid without authentication.

Suppose you have 100 users and you want only six of these users to perform FTP, Telnet, HTTP, or HTTP from outside the network. Tell the PIX Firewall to authenticate inbound traffic, and give all six users identifications on the Terminal Access Controller Access Control System Plus (TACACS+) or Remote Authentication Dial-In User Service (RADIUS) AAA server. With simple authentication, these six users are authenticated with a username and password, and then permitted access to the network. The other 94 users cannot access the network. The PIX Firewall prompts users for their username and password, and then passes their username and password to the TACACS+ or RADIUS AAA server. Depending on the response, the PIX Firewall opens or denies the connection.

Suppose one of these users, “baduser,” is “technically challenged.” You want to allow “baduser” to perform HTTP, but not Telnet, to the network. This means you must add authorization, that is, you must authorize what users can do in addition to authenticating who they are. When you add authorization to the PIX Firewall, it first sends the “technically challenged” user’s username and password to the AAA server, then sends an authorization request telling the AAA server which command “baduser” is trying to use. With the server set up properly, “baduser” is allowed to perform HTTP but is not allowed to perform Telnet.



Types of Authentication

Cisco.com
PIX Firewall console access
PIX Firewall console access
Internet
Cisco Secure ACS server
Web server
Interactive user authentication
Internet
Cisco Secure ACS server

Types of authentication:

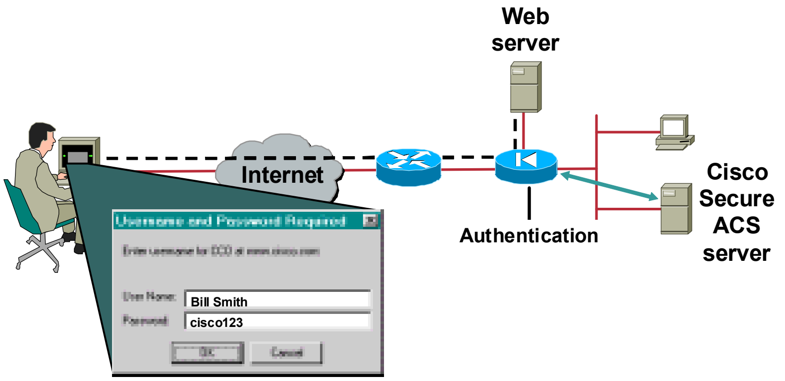
- Console access authentication
- Interactive user authentication

CSPFA 3.2—11-6

Two types of authentication are available, console access authentication and interactive user authentication. Console access enables the administrator to require authentication verification to access the PIX Firewall. The access authentication service options available are as follows: *enable, serial, ssh, HTTP*, and *telnet*. In the example in the figure, a remote administrator is attempting to access the PIX Firewall via Secure Shell (SSH) from the user's home office while a local administrator is attempting to access the PIX Firewall via Telnet. Both must be authenticated before they are permitted to access the PIX Firewall.

For interactive user authentication, the PIX Firewall can be configured to require user authentication for a session across the PIX Firewall, as specified in the **aaa authentication** command. Only Telnet, FTP, HTTPS, and HTTP sessions can be intercepted to authenticate users. In the example in the figure, a remote user is attempting an HTTP session with the web server. If the user is authenticated by the PIX Firewall, the HTTP session to the web server is connected, cut-through. The PIX Firewall then shifts the session flow and all traffic flows directly between the server and the client while maintaining session state information.

Types of Interactive User Authentication



User Name: Bill Smith
Password: cisco123

Types of interactive user authentication

- **Telnet**
- **FTP**
- **HTTP**
- **HTTPS**

CSPFA 3.2—11-7

To authenticate a user, only FTP, Telnet, and HTTP sessions can be intercepted. More information on the three authentication sessions is as follows:

- Telnet—You get a prompt generated by the PIX Firewall. You have up to four chances to log in. If the username or password fails after the fourth attempt, the PIX Firewall drops the connection. If authentication and authorization are successful, the destination server prompts you for a username and password.
- FTP—You get a prompt from the FTP program. If you enter an incorrect password, the connection is dropped immediately. If the username or password on the authentication database differs from the username or password on the remote host which you are accessing via FTP, enter the username and password in the following formats:
  - aaa_username@remote_username
  - aaa_password@remote_password

  The PIX Firewall sends the aaa_username and aaa_password that you have specified to the AAA server, and if authentication and authorization are successful, the remote_ username and remote_ password specified are passed to the destination FTP server.

Note Some FTP GUIs do not display challenge values.

- HTTP—You see a window generated by the web browser. If you enter an incorrect password, you are prompted again. If the username or password on the authentication database differs from the username or password on the remote host that you are using HTTP to access, enter the username and password in the following formats:
  - aaa_username@remote_username
  - aaa_password@remote_password

The PIX Firewall sends the aaa_username and aaa_password that you have specified to the AAA server, and if authentication and authorization are successful, the remote_username and remote_password specified are passed to the destination HTTP server.

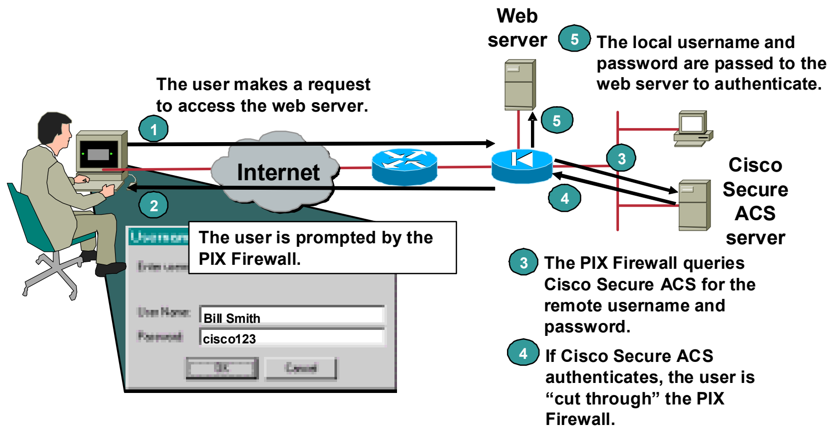
Keep in mind that browsers cache usernames and passwords. If you believe that the PIX Firewall should be timing out an HTTP connection but it is not, reauthentication may actually be taking place, with the web browser sending the cached username and password back to the PIX Firewall. The Syslog service will show this phenomenon. If Telnet and FTP seem to work normally, but HTTP connections do not, this is usually why.

The PIX Firewall supports authentication usernames up to 127 characters, and passwords of up to 63 characters. A password or username may not contain an "at" (@) character as part of the password or username string.

Note If PIX Firewalls are in tandem, Telnet authentication works in the same way as a single PIX Firewall, but FTP and HTTP authentication have additional complexity because you have to enter each password and username with an additional "at" (@) character and password or username for each in-tandem PIX Firewall.

Note Once authenticated with HTTP, a user never has to reauthenticate no matter how low the PIX Firewall uauth timeout is set. This is because the browser caches the "Authorization: Basic=Uuhjksdkfhk==" string in every subsequent connection to that particular site. This can only be cleared when the user exits all instances of the web browser and restarts. Flushing the cache is of no use.

Cut-Through Proxy Operation



1 The user makes a request to access the web server.

2 The user is prompted by the PIX Firewall.

User Name: Bill Smith
Password: cisco123

3 The PIX Firewall queries Cisco Secure ACS for the remote username and password.

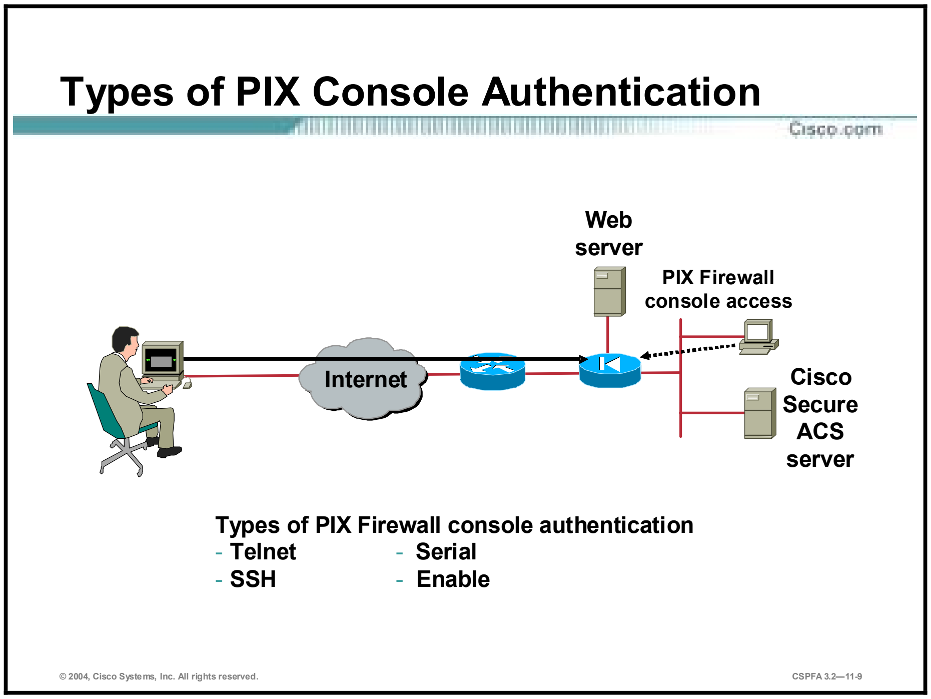
4 If Cisco Secure ACS authenticates, the user is "cut through" the PIX Firewall.

5 The local username and password are passed to the web server to authenticate.

CSPFA 3.2—11-8

The PIX Firewall gains dramatic performance advantages because of the cut-through proxy: a method of transparently verifying the identity of users at the firewall and permitting or denying access to any TCP- or UDP-based application. This method eliminates the price and performance impact that UNIX system-based firewalls impose in similar configurations, and leverages the authentication and authorization services of the Cisco Secure Access Control Server (Cisco Secure ACS).

The PIX Firewall's cut-through proxy challenges a user initially at the application layer, and then authenticates against standard TACACS+ or RADIUS databases. After the policy is checked, the PIX Firewall shifts the session flow, and all traffic flows directly and quickly between the server and the client while maintaining session state information.



Types of PIX Console Authentication

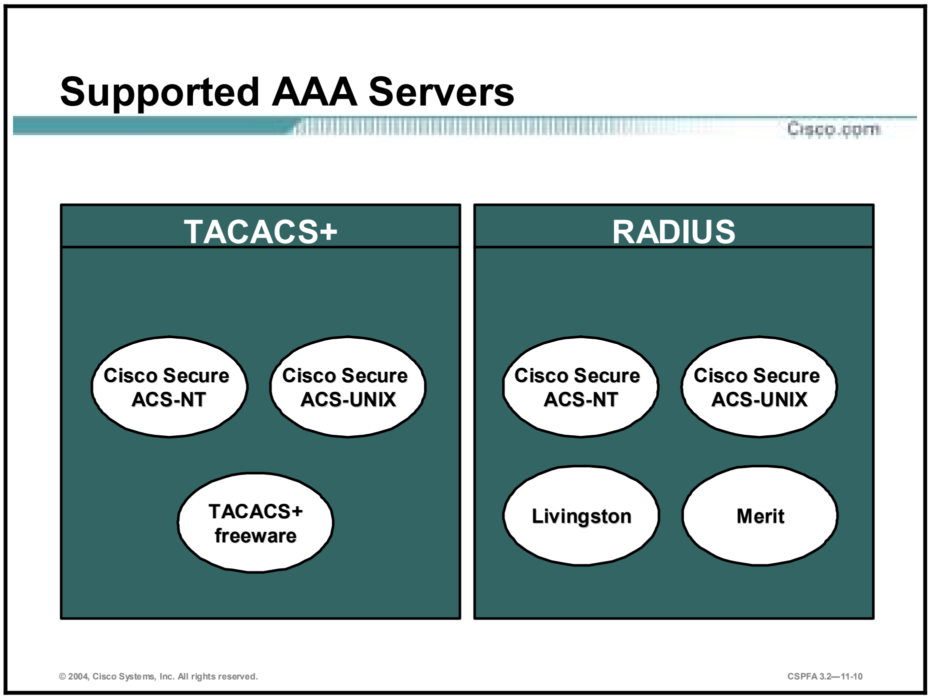
Types of PIX Firewall console authentication

- Telnet
- SSH
- Serial
- Enable

CSPFA 3.2—11-9

The **aaa authentication serial console** command enables you to require authentication verification to access the PIX Firewall unit's console. Authenticated access to the PIX Firewall console involves different types of prompts, depending on the option you choose with the **aaa authentication** [**serial** | **enable** | **telnet** | **ssh**] **console** command. While the **enable** and **ssh** options allow three tries before stopping access attempts with an "access denied" message, both the **serial** and **telnet** options cause the user to be prompted continually until that user successfully logs in. The **serial** option requests a username and password before the first command-line prompt on the serial console connection. The **telnet** option forces you to specify a username and password before the first command-line prompt of a Telnet console connection. The **enable** option requests a username and password before accessing privileged mode for serial, Telnet, or SSH connections. The **ssh** option requests a username and password before the first command-line prompt on the SSH console connection.

Telnet access to the PIX Firewall console is available from any internal interface, and from the outside interface with IPSec configured, and requires previous use of the **telnet** command. SSH access to the PIX Firewall console is also available from any interface without IPSec configured, and requires previous use of the **ssh** command.



Supported AAA Servers

Cisco.com

TACACS+

Cisco Secure ACS-NT

Cisco Secure ACS-UNIX

TACACS+ freeware

RADIUS

Cisco Secure ACS-NT

Cisco Secure ACS-UNIX

Livingston

Merit

© 2004, Cisco Systems, Inc. All rights reserved.

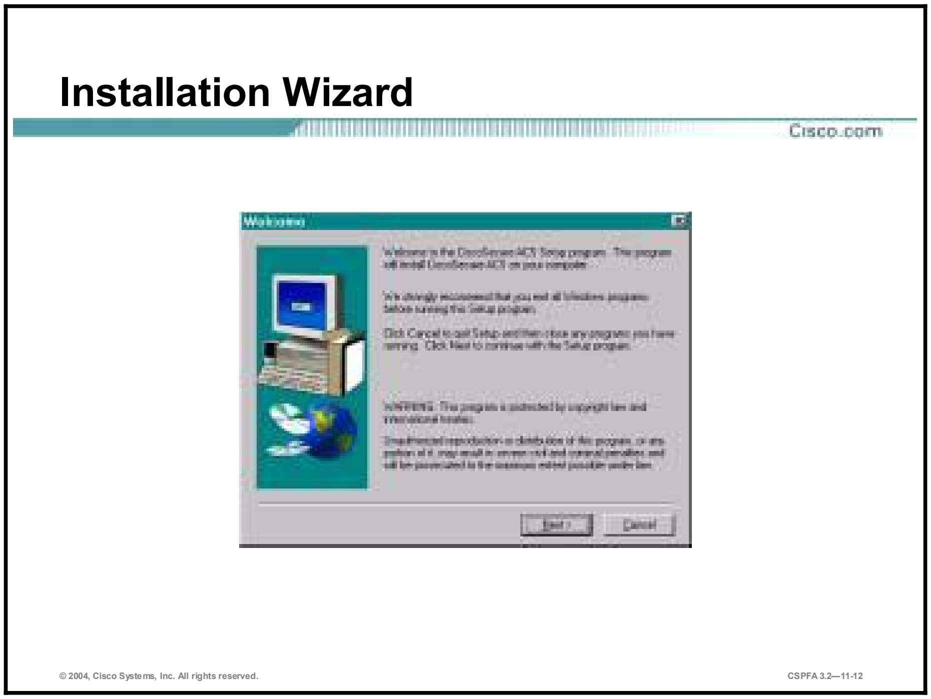
CSPFA 3.2—11-10

The PIX Firewall supports the following AAA protocols and servers:

- Terminal Access Controller Access Control System Plus (TACACS+)
  - Cisco Secure ACS for Windows NT
  - Cisco Secure ACS for UNIX
  - TACACS+ freeware
- Remote Authentication Dial-In User Service (RADIUS)
  - Cisco Secure ACS for Windows NT
  - Cisco Secure ACS for UNIX
  - Livingston
  - Merit

Installation of Cisco Secure ACS for Windows NT

This topic explains how to install the Cisco Secure ACS for Windows NT.



Installation Wizard

Cisco.com

Welcome to the CiscoSecure ACS Setup program. This program will install CiscoSecure ACS on your computer.

We strongly recommend that you exit all Windows programs before running this Setup program.

Click Cancel to quit Setup and then close any programs you have running. Click Next to continue with the Setup program.

WARNING: This program is protected by copyright law and international treaties.

Unauthorized reproduction or distribution of this program, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted to the maximum extent possible under law.

CSPFA 3.2—11-12

Note	Close all Windows programs before you run the setup program.

Complete the following steps to start installation of Cisco Secure ACS for Windows NT:

Step 1 Log in as the local system administrator to the machine on which you are installing Cisco Secure ACS.

Step 2 Insert the Cisco Secure ACS CD-ROM into your CD-ROM drive. The Installation window opens.

Step 3 Click **Install**. The Software License Agreement window opens.

Step 4 Read the Software License Agreement. Click Accept to agree to the licensing terms and conditions. The Welcome window opens.

Step 5 Click **Next**. The Before You Begin window opens.

Step 6 Verify that each condition is met, and then select the check box for each item. Click **Next**.

Step 7 Click **Next**. (Click Explain for more information on the listed items. If any condition is not met, click **Cancel** to exit the program.)

Step 8 If all conditions are met, click **Next** to continue.

Note	If this is a new installation, skip to Step 11.

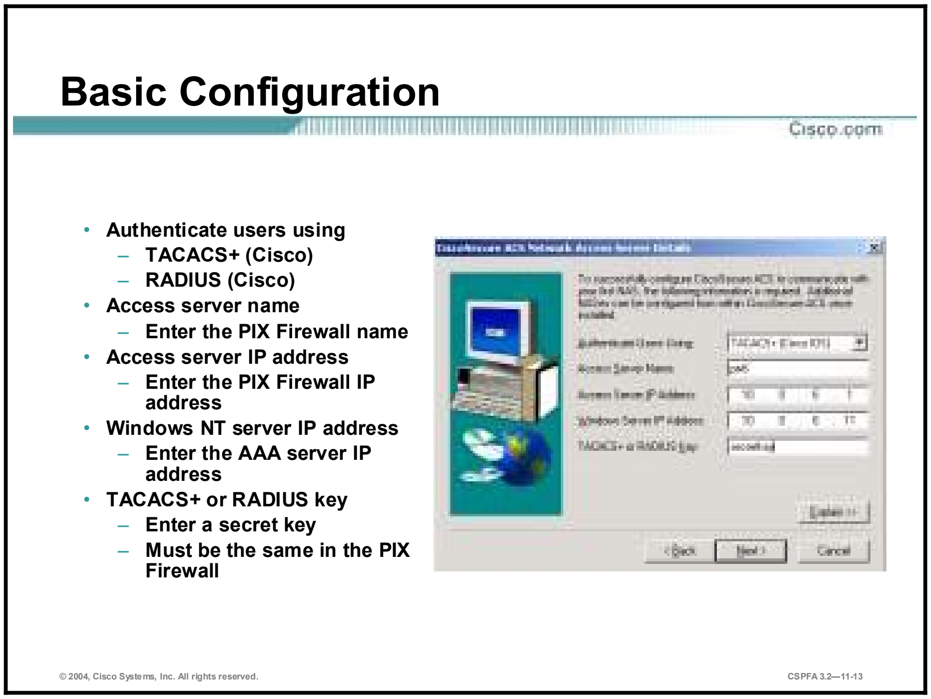
Step 9 (Optional.) If Cisco Secure ACS is already installed, the Previous Installation window opens. You are prompted to remove the previous version and save the existing database information. Click **Yes, keep existing database** and click **Next** to keep the existing data. To use a new database, deselect the check box and click **Next**. If you selected the check box, the setup program backs up the existing database information and removes the old files. When the files are removed, click **OK**.

Step 10 If Setup finds an existing configuration, a prompt asks whether you want to import the configuration. Click **Yes, import configuration** and click **Next** to keep the existing configuration. Deselect the check box and click **Next** to use a new configuration. The Choose Destination Location window opens.

Step 11 Click **Next** to install the software in the default directory. Click **Browse** and enter the directory to use to use a different directory. If the directory does not exist, you are prompted to create one. Click **Yes**. The Authentication Database Configuration window opens.

Step 12 Click the radio button for the authentication databases to be used by Cisco Secure ACS. Select the **CSACS Database only** option (the default). Also select the **Windows NT User Database** option. If you select the first option, Cisco Secure ACS will use only the Cisco Secure ACS database for authentication; if you select the second option, Cisco Secure ACS will check both databases.

Step 13 (Optional.) Click the **Yes, reference "Grant dialin permission to user"** setting to limit dial-in access to only those users you specified in the Windows NT User Manager. Click **Next**. The Network Access Server Details window opens.



Basic Configuration

- Authenticate users using
  - TACACS+ (Cisco)
  - RADIUS (Cisco)
- Access server name
  - Enter the PIX Firewall name
- Access server IP address
  - Enter the PIX Firewall IP address
- Windows NT server IP address
  - Enter the AAA server IP address
- TACACS+ or RADIUS key
  - Enter a secret key
  - Must be the same in the PIX Firewall

CiscoSecure ACS Network Access Server Details

To successfully configure CiscoSecure ACS to communicate with your first NAS, the following information is required. Additional NASes can be configured from within CiscoSecure ACS once installed.

Authenticate Users Using: TACACS+ (Cisco IOS)

Access Server Name: pix6

Access Server IP Address: 10 . 0 . 0 . 1

Windows Server IP Address: 10 . 0 . 0 . 11

TACACS+ or RADIUS Key: secretkey

CSPFA 3.2—11-13

Step 14 Complete the following information:

- Authenticate Users Using—Type of security protocol to be used. TACACS+ (Cisco) is the default.
- Access Server Name—Name of the network access server (NAS) that will be using the Cisco Secure ACS services.
- Access Server IP Address—IP address of the NAS that will be using the Cisco Secure ACS services.
- Windows Server IP Address—IP address of this Windows NT server.
- TACACS+ or RADIUS Key—Shared secret of the NAS and Cisco Secure ACS. These passwords must be identical to ensure proper function and communication between the NAS and Cisco Secure ACS. Shared secrets are case sensitive. Setup installs the Cisco Secure ACS files and updates the Registry. Click **Next**. The Interface Configuration window opens.

Step 15 The Interface Configuration options are disabled by default. Select the check boxes to enable any or all of the options listed. Click Next. The Active Service Monitoring window opens.

Note Configuration options for these items are displayed in the Cisco Secure ACS interface only if they are enabled. You can disable or enable any or all of these and additional options after installation in the Interface Configuration: Advanced Options window.

Step 16 To enable the Cisco Secure ACS monitoring service, CSMon, select the **Enable Log-in Monitoring** check box, then select a script to execute when the login process fails the test:

- No Remedial Action—Leave Cisco Secure ACS operating as is.
- Reboot—Reboot the system on which Cisco Secure ACS is running.
- Restart All—(Default.) Restart all Cisco Secure ACS services.

- Restart RADIUS/TACACS+—Restart only RADIUS, TACACS+, or both protocols.

You can also develop your own scripts to be executed if there is a system failure. See the online documentation for more information.

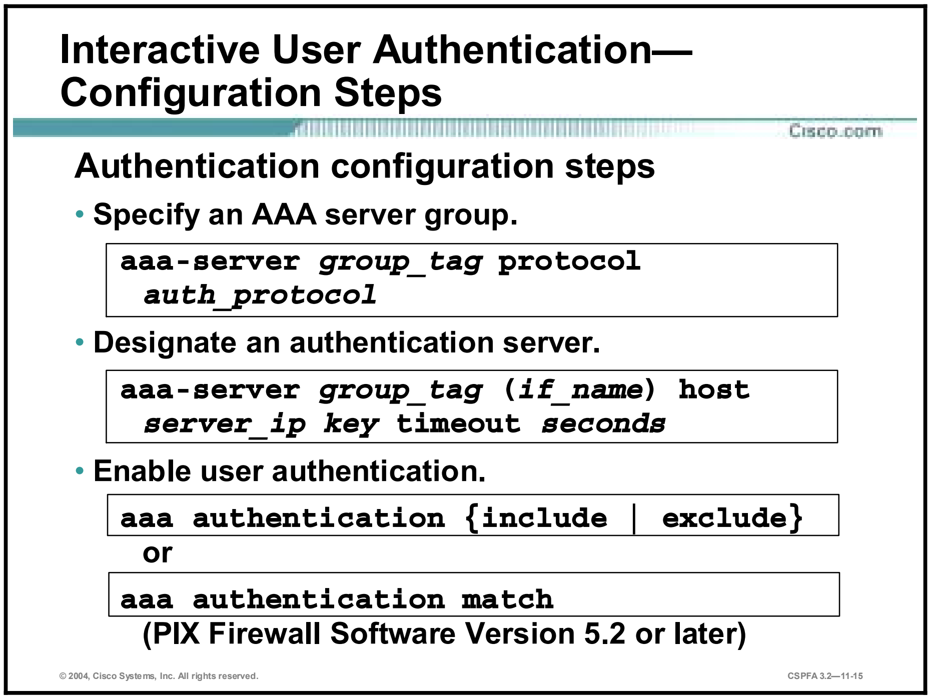
Step 17 To have Cisco Secure ACS generate an e-mail message when administrator events occur, select the **Enable Mail Notifications** check box, and then enter the following information:

- SMTP Mail Server—The name and domain of the sending mail server (for example, server1.company.com).
- Mail account to notify—The complete e-mail address of the intended recipient (for example, msmith@company.com).

Step 18 Click **Next**. The CSACS Service Initiation window opens. If you do not want to configure an NAS from Setup, click Next. To configure a single NAS, click **Yes, I want to configure Cisco IOS now**, and then click **Next**.

Authentication Configuration

This topic discusses how to configure authentication on the PIX Firewall.



Interactive User Authentication—Configuration Steps

Cisco.com

Authentication configuration steps

- Specify an AAA server group.

```
aaa-server group_tag protocol
  auth_protocol
```

- Designate an authentication server.

```
aaa-server group_tag (if_name) host
  server_ip key timeout seconds
```

- Enable user authentication.

```
aaa authentication {include | exclude}
```

or

```
aaa authentication match
```

(PIX Firewall Software Version 5.2 or later)

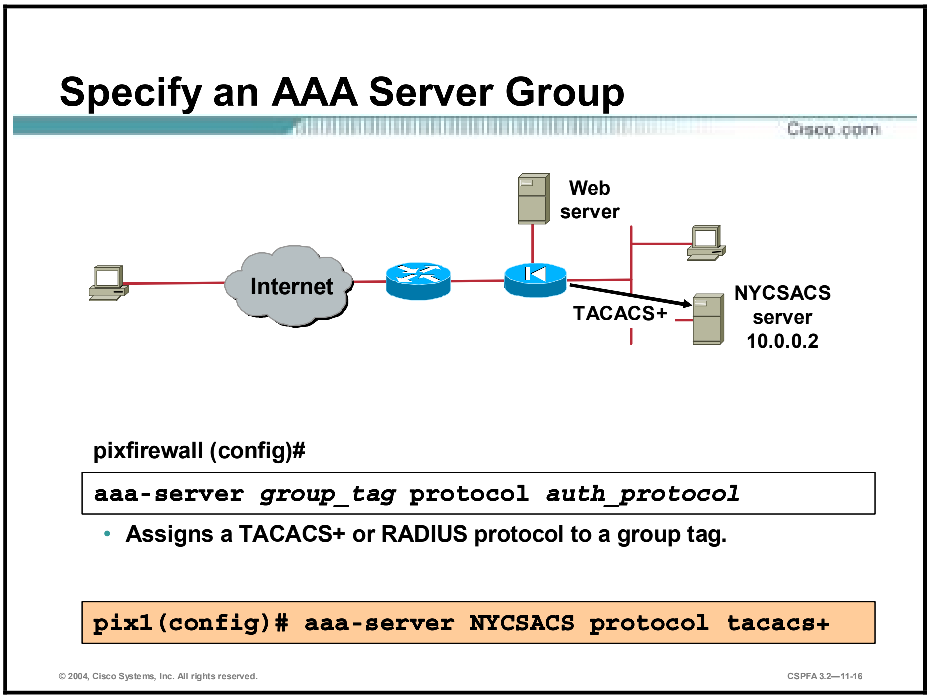
CSPFA 3.2—11-15

Configuring interactive user authentication is a three-step process. The three steps are as follows:

Step 1 Specify an AAA server group. The administrator defines a group name and the authentication protocol.

Step 2 Designate an authentication server. The administrator defines the location of the AAA server and a key.

Step 3 Enable user authentication. The administrator defines a rule to specify which traffic to include or exclude from authentication.



Specify an AAA Server Group

pixfirewall (config)#

```
aaa-server group_tag protocol auth_protocol
```

- Assigns a TACACS+ or RADIUS protocol to a group tag.

```
pix1(config)# aaa-server NYCSACS protocol tacacs+
```

CSPFA 3.2—11-16

Use the **aaa-server** command to specify AAA server groups. The PIX Firewall enables you to define separate groups of TACACS+ or RADIUS servers for specifying different types of traffic, such as a TACACS+ server for inbound traffic and another for outbound traffic. The **aaa** command references the group tag to direct authentication, authorization, or accounting traffic to the appropriate AAA server.

You can have up to 14 tag groups, and each group can have up to 14 AAA servers, for a total of up to 196 TACACS+ or RADIUS servers. When a user logs in, the servers are accessed one at a time, starting with the first server you specify in the tag group, until a server responds.

The default configuration provides the following two aaa-server protocols:

- aaa-server TACACS+ protocol tacacs+
- aaa-server RADIUS protocol radius

Note If you are upgrading from a previous version of the PIX Firewall and have aaa command statements in your configuration, using the default server groups enables you to maintain backward compatibility with the aaa command statements in your configuration.

Note The previous server type option at the end of the aaa authentication and aaa accounting commands has been replaced with the aaa-server group tag. Backward compatibility with previous versions is maintained by the inclusion of two default protocols for TACACS+ and RADIUS.

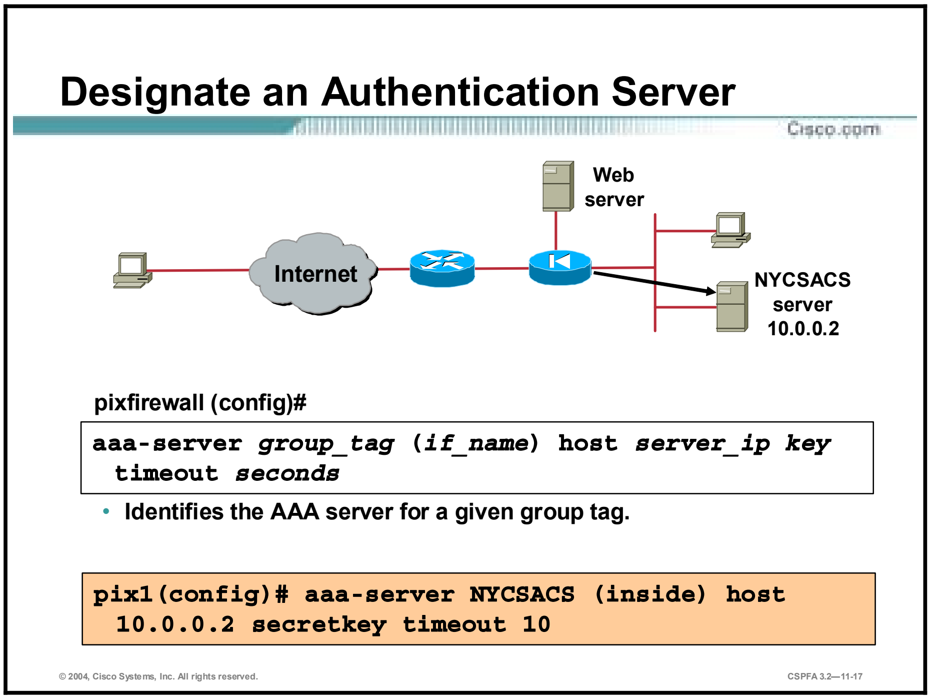
Note The PIX Firewall listens for RADIUS on ports 1645 and 1646. If your RADIUS server uses ports 1812 and 1813, you will need to reconfigure it to listen on ports 1645 and 1646.

The syntaxes for the **aaa-server** commands are as follows:

aaa-server *group_tag* protocol *auth_protocol*

clear aaa-server [*group_tag*]

group_tag	An alphanumeric string that is the name of the server group. Use the group_tag in the **aaa** command to associate **aaa authentication**, **aaa authorization**, and **aaa accounting** command statements with an AAA server.
protocol *auth_protocol*	The type of AAA server, either TACACS+ or RADIUS.



Designate an Authentication Server

pixfirewall (config)#

```
aaa-server group_tag (if_name) host server_ip key
  timeout seconds
```

- Identifies the AAA server for a given group tag.

```
pix1(config)# aaa-server NYCSACS (inside) host
  10.0.0.2 secretkey timeout 10
```

CSPFA 3.2—11-17

The next step is to define the AAA server. The administrator defines the following attributes for the AAA server:

- Group tag—Name of the server group. There can be up to 14 groups.
- Interface—Name of the interface on which the AAA server resides.
- IP address—IP address of the AAA server.
- Secret key—Key used to encrypt data between the PIX Firewall and the AAA server. The key must match between the PIX Firewall and the AAA server.
- Timeout—Time after which the PIX Firewall gives up on the request to the primary AAA server.

In the example in the figure, there is an AAA server that belongs to the NYCSACS group. It is located out the inside interface and has an IP address of 10.0.0.2. The encryption key is "secretkey," and the request timeout is 10 seconds.

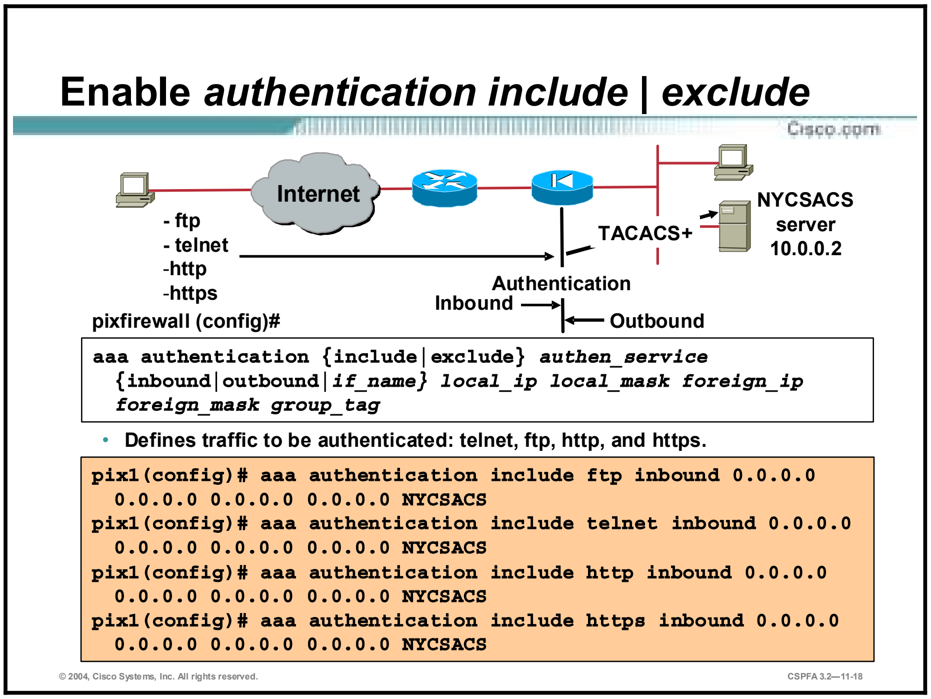
The syntaxes for the **aaa-server** commands are as follows:

aaa-server ***group_tag*** **(*if_name*) host** ***server_ip key*** **timeout** ***seconds***

no aaa-server ***group_tag*** **(*if_name*) host** ***server_ip key*** **timeout** ***seconds***

group_tag	A case-sensitive alphanumeric string that is the name of the server group. Use the group_tag in the **aaa** command to associate **aaa authentication**, **aaa authorization**, and **aaa accounting** command statements with an AAA server.
if_name	The interface name on the side where the AAA server resides.
host *server_ip*	The IP address of the TACACS+ or RADIUS server.

key	A case-sensitive, alphanumeric keyword of up to 127 characters that is the same value as the key on the TACACS+ server. Any characters entered past the limit of 127 are ignored. The key is used for encrypting data between the client and server. The key must be the same on both the client and server systems. Spaces are not permitted in the key, but other special characters are allowed. If a key is not specified, encryption does not occur.
timeout *seconds*	A retransmit timer that specifies the duration of the period during which the PIX Firewall retries access. The PIX Firewall retries access to the AAA server four times, each time for the length of this period, before choosing the next AAA server. The default is 5 seconds. The maximum time is 30 seconds. For example, if the timeout value is 10 seconds, the PIX Firewall retransmits for 10 seconds and if no acknowledgment is received, tries three times more, for a total of 40 seconds, to retransmit data before the next AAA server is selected.



Enable *authentication include | exclude*

```
aaa authentication {include|exclude} authen_service
  {inbound|outbound|if_name} local_ip local_mask foreign_ip
  foreign_mask group_tag
```

- Defines traffic to be authenticated: telnet, ftp, http, and https.

```
pix1(config)# aaa authentication include ftp inbound 0.0.0.0
  0.0.0.0 0.0.0.0 0.0.0.0 NYCSACS
pix1(config)# aaa authentication include telnet inbound 0.0.0.0
  0.0.0.0 0.0.0.0 0.0.0.0 NYCSACS
pix1(config)# aaa authentication include http inbound 0.0.0.0
  0.0.0.0 0.0.0.0 0.0.0.0 NYCSACS
pix1(config)# aaa authentication include https inbound 0.0.0.0
  0.0.0.0 0.0.0.0 0.0.0.0 NYCSACS
```

CSPFA 3.2—11-18

The **aaa authentication** command enables or disables user authentication services. When you start a connection via Telnet, FTP, HTTP, or HTTPS, you are prompted for a username and password. An AAA server, designated previously with the **aaa-server** command, verifies whether the username and password are correct. If they are correct, the PIX Firewall's cut-through proxy permits further traffic between the initiating host and the target host.

The **aaa authentication** command is not intended to mandate your security policy. The AAA servers determine whether a user can or cannot access the system, which services can be accessed, and which IP addresses the user can access. The PIX Firewall interacts with Telnet, FTP, HTTP, and HTTPS to display the prompts for logging. You can specify that a range of hosts, or a single host, be authenticated. You can set the aaa command to a specific host and let the aaa server authenticate the user. You can set the local IP address in the aaa authentication command to 0 to mean all hosts, and let the authentication server decide which hosts are authenticated. The aaa server enforces the PIX Firewall AAA authentication policy.

For each IP address, one **aaa authentication** command is permitted for inbound connections and one for outbound connections. The PIX Firewall permits only one authentication type per network. For example, if one network connects through the PIX Firewall using TACACS+ for authentication, another network connecting through the PIX Firewall can authenticate with RADIUS, but a single network cannot authenticate with both TACACS+ and RADIUS. In the example in the figure, any inbound FTP, HTTP, HTTPS, and Telnet session is intercepted by the PIX Firewall and authenticated by the AAA server. An AAA server from the NYCSACS group verifies the authentication username and password. Once authenticated, the session is cut through the PIX Firewall.

Note For cut-through proxy authentication, the administrator can also use the local PIX Firewall user authentication database by specifying group tag local.

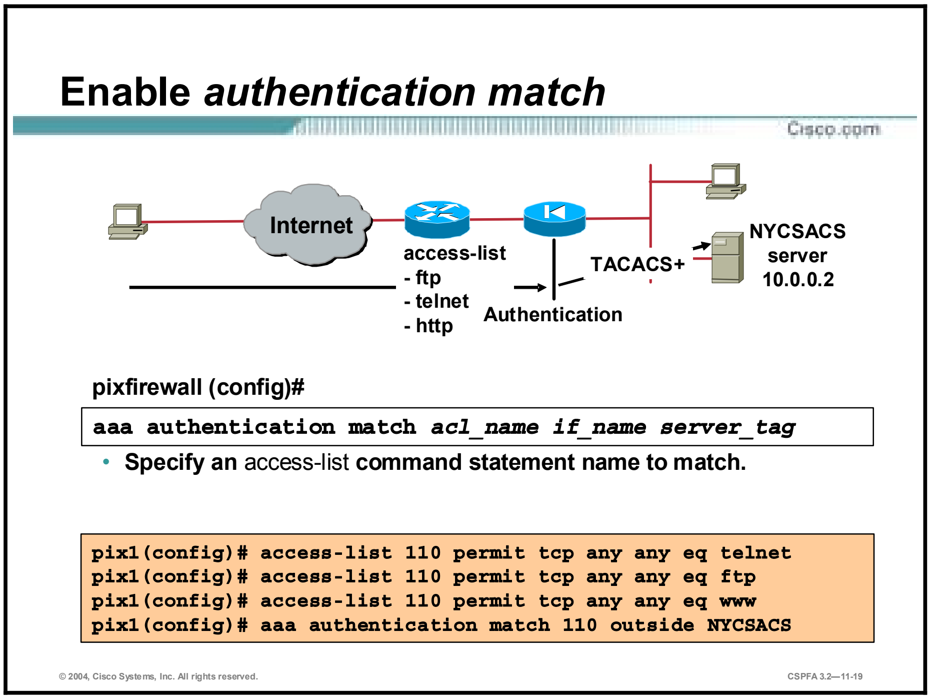
The syntaxes for the **aaa authentication** commands are as follows:

aaa authentication include | exclude ***authen_service*** **inbound | outbound |** ***if_name local_ip local_mask foreign_ip foreign_mask group_tag***

no aaa authentication [include | exclude ***authen_service*** **inbound | outbound |** ***if_name local_ip local_mask foreign_ip foreign_mask group_tag*****]**

clear aaa [authentication include | exclude ***authen_service*** **inbound | outbound |** ***if_name local_ip local_mask foreign_ip foreign_mask group_tag*****]**

include	Creates a new rule with the specified service to include.
exclude	Creates an exception to a stated rule by excluding the specified service from authentication to the specified host. The **exclude** parameter improves the former **except** option by enabling the user to specify a port to exclude to a specific host or hosts.
authen_service	The services that require user authentication before they are let through the firewall. Use *any*, *ftp*, *http*, *https*, or *telnet*.
inbound	Authenticates inbound connections. Inbound means the connection originates on the outside interface and is being directed to the inside or any other perimeter interface.
outbound	Authenticate outbound connections. Outbound means the connection originates on the inside and is being directed to the outside or any other perimeter interface.
if_name	Interface name from which users require authentication. Use *if_name* in combination with the *local_ip* address and the *foreign_ip* address to determine where access is sought and from where. The *local_ip* address is always on the interface with the highest security level and *foreign_ip* address is always on the lowest.
local_ip	The IP address of the host or network of hosts that you want to be authenticated. You can set this address to 0 to mean all hosts and to let the authentication server decide which hosts are authenticated.
local_mask	Network mask of *local_ip*. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host.
foreign_ip	The IP address of the hosts you want to access the *local_ip* address. Use 0 to mean all hosts.
foreign_mask	Network mask of *foreign_ip*. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host.
group_tag	The group tag set with the **aaa-server** command. To use the local PIX Firewall user authentication database, enter LOCAL for this parameter.



Enable *authentication match*

pixfirewall (config)#

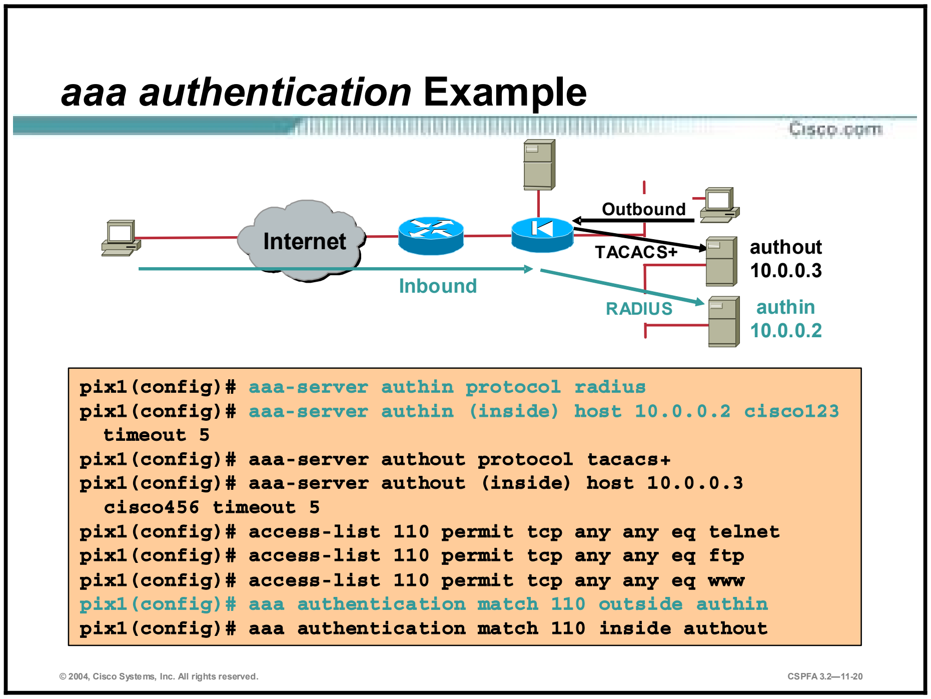
```
aaa authentication match acl_name if_name server_tag
```

- Specify an access-list command statement name to match.

```
pix1(config)# access-list 110 permit tcp any any eq telnet
pix1(config)# access-list 110 permit tcp any any eq ftp
pix1(config)# access-list 110 permit tcp any any eq www
pix1(config)# aaa authentication match 110 outside NYCSACS
```

CSPFA 3.2—11-19

Authentication can also be performed on the successful match of an access control list (ACL) entry. This gives the administrator complete control of the cut-through authentication performed by the PIX Firewall. It can be as selective or permissive as necessary. In the example in the figure, there are three ACL entries. The entries permit Telnet, FTP, and HTTP from any host to any host. The PIX Firewall will intercept any sessions matching the ACL entries received on the outside interface. The sessions are verified by an AAA server belonging to the NYCSACS server group.



aaa authentication Example

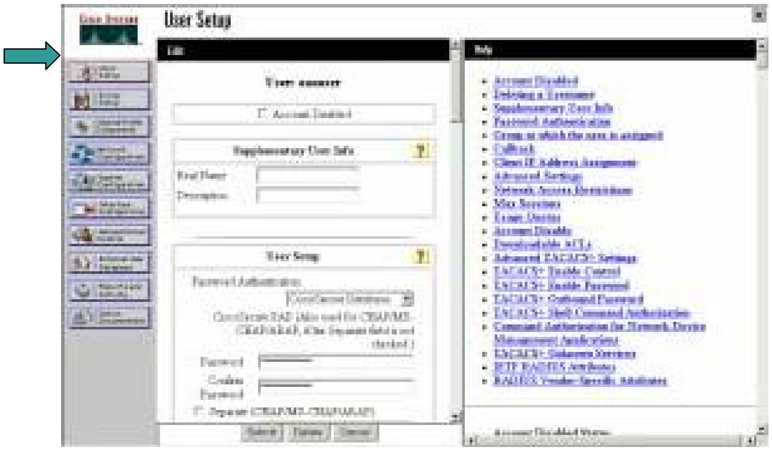
```
pix1(config)# aaa-server authin protocol radius
pix1(config)# aaa-server authin (inside) host 10.0.0.2 cisco123
  timeout 5
pix1(config)# aaa-server authout protocol tacacs+
pix1(config)# aaa-server authout (inside) host 10.0.0.3
  cisco456 timeout 5
pix1(config)# access-list 110 permit tcp any any eq telnet
pix1(config)# access-list 110 permit tcp any any eq ftp
pix1(config)# access-list 110 permit tcp any any eq www
pix1(config)# aaa authentication match 110 outside authin
pix1(config)# aaa authentication match 110 inside authout
```

CSPFA 3.2—11-20

In the example in the figure, there are two traffic flows, inbound and outbound. Any inbound traffic that matches the ACLs is authenticated by the authin AAA server using the RADIUS protocol. Any outbound traffic matching the ACL is authenticated by the authout AAA server using the TACACS+ protocol.

How to Add Users to Cisco Secure ACS

Cisco.com



CSPFA 3.2—11-21

Complete the following steps to add users to the Cisco Secure ACS:

Step 1 Click **User Setup** from the navigation bar. The Select window opens.

Step 2 Enter a name in the User field.

Note The username can contain up to 32 characters. Names cannot contain the following special characters: #, ?, ", *, >, and <. Leading and trailing spaces are not allowed.

Step 3 Click **Add/Edit**. The Edit window opens. The username being added or edited appears at the top of the window.

The Edit window contains the following sections:

- Account Disabled
- Supplementary User Info
- User Setup
- Account Disable

Account Disabled

If you need to disable an account, select the Account Disabled check box in the Account Disabled section to deny access for this user.

Note You must click Submit to have this action take effect.

Supplementary User Info

In this section, you can enter supplemental information to appear in each user profile. The fields shown in the following bullet items are available by default; however, you can insert additional fields by clicking Interface Configuration in the navigation bar and then clicking User Data Configuration (configuring supplemental information is optional):

- Real Name—If the username is not the user’s real name, enter the real name here.
- Description—Enter a detailed description of the user.

User Setup

In the User Setup group box, you can edit or enter the following information for the user as applicable:

- Password Authentication—From the drop-down menu, choose a database to use for username and password authentication. You can select the Windows NT user database or the Cisco Secure database. The Windows NT option authenticates a user with an existing account in the Windows NT User Database located on the same machine as the Cisco Secure ACS server. The Cisco Secure Database option authenticates a user from the local Cisco Secure ACS database. If you select this database, enter and confirm the Password Authentication Protocol (PAP) password to be used. The Separate CHAP/MS-CHAP/ARAP option is not used with the PIX Firewall.

Note The Password and Confirm Password fields are required for all authentication methods except for all third-party user databases.

- Group to which the user is assigned—From the Group to which the user is assigned drop-down menu, choose the group to which to assign the user. The user inherits the attributes and operations assigned to the group. By default, users are assigned to the Default Group. Users who authenticate via the Unknown User method and who are not found in an existing group are also assigned to the Default Group.
- Callback—This information is not used with the PIX Firewall.
- Client IP Address Assignment—This information is not used with PIX Firewall.

Account Disable

The Account Disable group box can be used to define the circumstances under which the user’s account will become disabled.

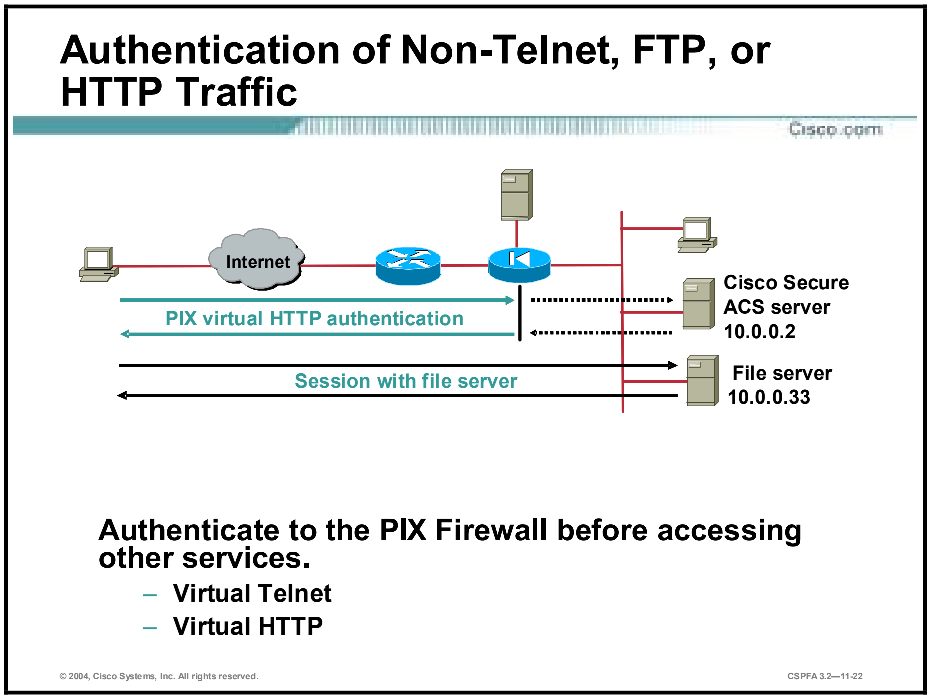
Note This action is not to be confused with account expiration due to password aging. Password aging is defined for groups only, not for individual users.

- Never radio button—Select to keep the user’s account always enabled. This is the default.
- Disable account if radio button—Select to disable the account under the circumstances you specify in the following fields:
  - Date exceeds—From the drop-down menu, choose the month, date, and year on which to disable the account. The default is 30 days after the user is added.

    - Failed attempts exceed—Select the check box and enter the number of consecutive unsuccessful login attempts to allow before disabling the account. The default is 5.
    - Failed attempts since last successful login—This counter shows the number of unsuccessful login attempts since the last time this user logged in successfully.
- Reset current failed attempts count on submit—If an account is disabled because the failed attempts count has been exceeded, select this check box and click **Submit** to reset the failed attempts counter to 0 and reinstate the account.

If you are using the Windows NT user database, this expiration information is in addition to the information in the Windows NT user account. Changes here do not alter settings configured in Windows NT.

Step 4 When you have finished configuring all user information, click **Submit**.



Authentication of Non-Telnet, FTP, or HTTP Traffic

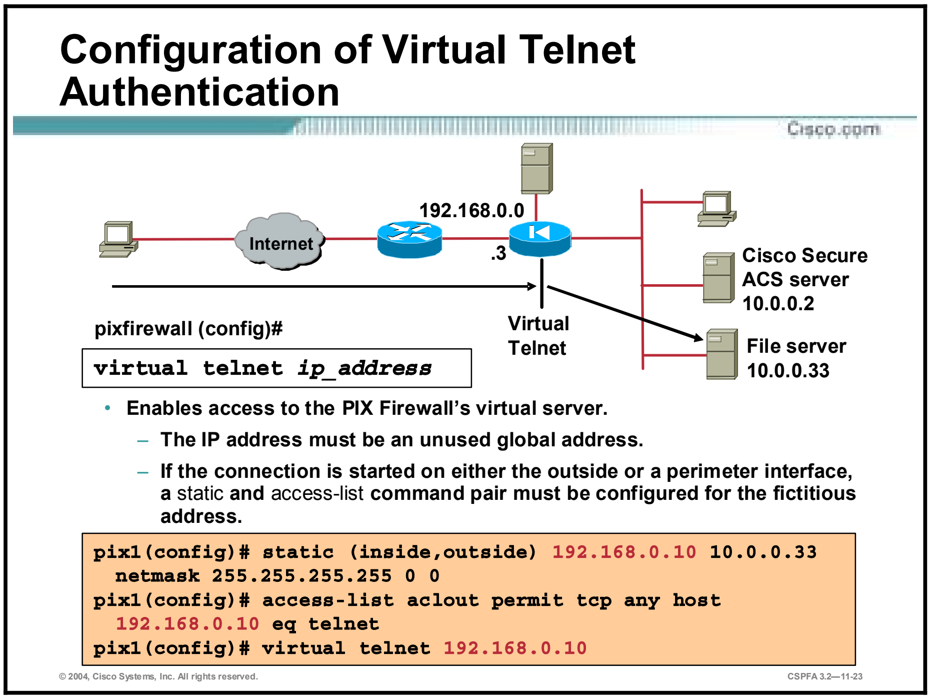
Authenticate to the PIX Firewall before accessing other services.

- **Virtual Telnet**
- **Virtual HTTP**

CSPFA 3.2—11-22

The PIX Firewall authenticates users via Telnet, FTP, HTTP, or HTTPS. But what if users need to access a Microsoft file server on port 139 or a Cisco IP/TV server for instance? How will they be authenticated? Whenever users are required to authenticate to access services other than by Telnet, FTP, HTTP, or HTTPS, they need to do one of the following:

- Option 1—Authenticate first by accessing a Telnet, FTP, HTTP, or HTTPS server before accessing other services.
- Option 2 Authenticate to the PIX Firewall virtual Telnet service before accessing other services. When there are no Telnet, FTP, HTTP, or HTTPS servers with which to authenticate, or just to simplify authentication for the user, the PIX Firewall allows a virtual Telnet authentication option. This option permits the user to authenticate directly with the PIX Firewall using the virtual Telnet IP address.



Configuration of Virtual Telnet Authentication

pixfirewall (config)#

```
virtual telnet ip_address
```

- Enables access to the PIX Firewall’s virtual server.
  - The IP address must be an unused global address.
  - If the connection is started on either the outside or a perimeter interface, a static and access-list command pair must be configured for the fictitious address.

```
pix1(config)# static (inside,outside) 192.168.0.10 10.0.0.33
  netmask 255.255.255.255 0 0
pix1(config)# access-list aclout permit tcp any host
  192.168.0.10 eq telnet
pix1(config)# virtual telnet 192.168.0.10
```

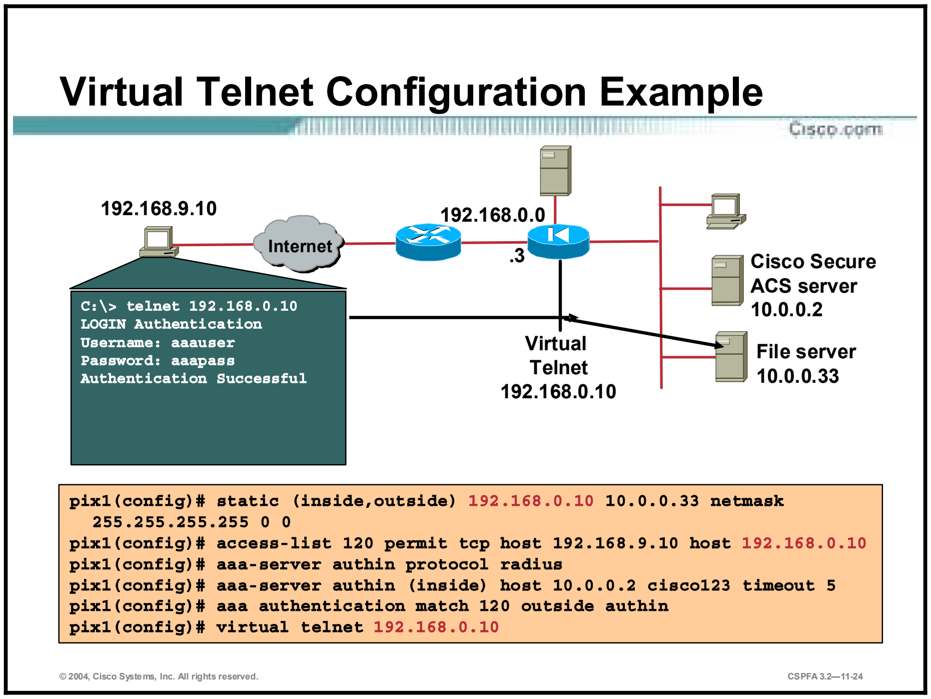
CSPFA 3.2—11-23

When you are using virtual Telnet to authenticate inbound clients, the IP address must be an unused global address. When you are using virtual Telnet to authenticate outbound clients, the IP address must be an unused global address routed directly to the PIX Firewall. In the example in the figure, the file server IP address is statically translated to a global address of 192.168.0.10. To be authenticated, the outside user establishes a virtual Telnet session to 192.168.0.10. The PIX Firewall will intercept the session and authenticate the user. After authentication, the session is cut through to the file server.

The syntax for the **virtual telnet** command is as follows:

virtual telnet ***ip_address***

ip_address	Unused global IP address on PIX Firewall, used for Telnet for authentication.



Virtual Telnet Configuration Example

```
C:\> telnet 192.168.0.10
LOGIN Authentication
Username: aaauser
Password: aaapass
Authentication Successful
```

```
pix1(config)# static (inside,outside) 192.168.0.10 10.0.0.33 netmask
  255.255.255.255 0 0
pix1(config)# access-list 120 permit tcp host 192.168.9.10 host 192.168.0.10
pix1(config)# aaa-server authin protocol radius
pix1(config)# aaa-server authin (inside) host 10.0.0.2 cisco123 timeout 5
pix1(config)# aaa authentication match 120 outside authin
pix1(config)# virtual telnet 192.168.0.10
```

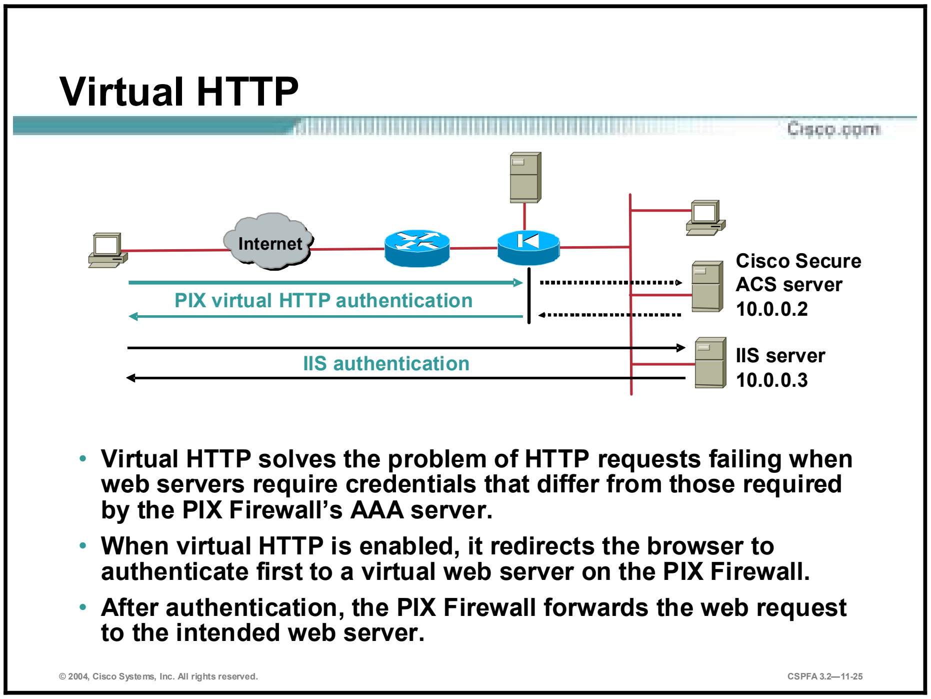
CSPFA 3.2—11-24

The virtual Telnet option provides a way to preauthenticate users who require connections through the PIX Firewall using services or protocols that do not support authentication. The virtual Telnet IP address is used both to authenticate in and authenticate out of the PIX Firewall.

When an unauthenticated user establishes a Telnet session to the virtual IP address, the user is challenged for the username and password, and then authenticated with the TACACS+ or RADIUS server. Once authenticated, the user sees the message "Authentication Successful," and the authentication credentials are cached in the PIX Firewall for the duration of the user authentication (uauth) timeout.

If a user wishes to log out and clear the entry in the PIX Firewall uauth cache, the user can again access the virtual address via Telnet. The user is prompted for a username and password, the PIX Firewall removes the associated credentials from the uauth cache, and the user receives a "Logout Successful" message.

In the figure, the user wants to establish a NetBIOS session on port 139 to access the file server. The user accesses the virtual Telnet address at 192.168.0.10, and is immediately challenged for a username and password before being authenticated with the RADIUS AAA server. Once the user is authenticated, the PIX Firewall allows that user to connect to the file server without reauthentication.



Virtual HTTP

Cisco.com
Internet
PIX virtual HTTP authentication
IIS authentication
Cisco Secure ACS server 10.0.0.2
IIS server 10.0.0.3

- **Virtual HTTP solves the problem of HTTP requests failing when web servers require credentials that differ from those required by the PIX Firewall's AAA server.**
- **When virtual HTTP is enabled, it redirects the browser to authenticate first to a virtual web server on the PIX Firewall.**
- **After authentication, the PIX Firewall forwards the web request to the intended web server.**

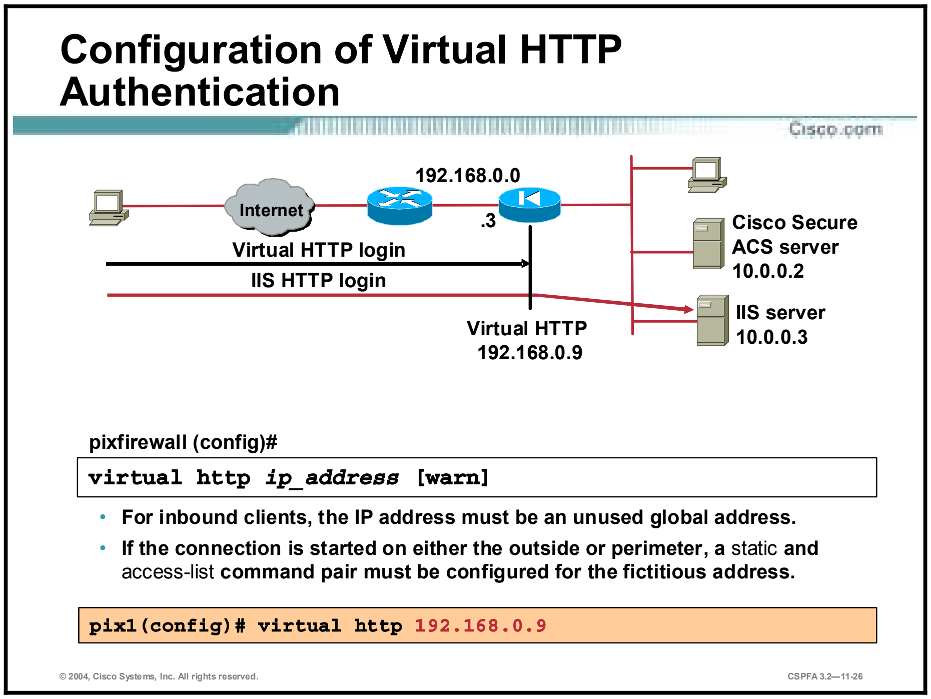
CSPFA 3.2—11-25

With the virtual HTTP option, web browsers work correctly with the PIX Firewall's HTTP authentication. The PIX Firewall assumes that the AAA server database is shared with a web server and automatically provides the AAA server and web server with the same information. The virtual HTTP option works with the PIX Firewall to authenticate the user, separate the AAA server information from the web client's URL request, and direct the web client to the web server. The virtual HTTP option works by redirecting the web browser's initial connection to an IP address, which resides in the PIX Firewall, authenticating the user, then redirecting the browser back to the URL that the user originally requested. This option is so named because it accesses a virtual HTTP server on the PIX Firewall, which in reality does not exist.

This option is especially useful for PIX Firewall interoperability with Microsoft Internet Information Server (IIS), but it is useful for other authentication servers. When using HTTP authentication to a site running Microsoft IIS that has Basic text authentication or Windows NT Challenge/Response authentication enabled, users may be denied access from the Microsoft IIS server because the browser appends the string: "Authorization: Basic=Uuhjksdkfhk==" to the HTTP GET commands. This string contains the PIX Firewall authentication credentials. Windows NT IIS servers respond to the credentials and assume that a Windows NT user is trying to access privileged pages on the server. Unless the PIX Firewall username and password combination is exactly the same as a valid Windows NT username and password combination on the Microsoft IIS server, the HTTP GET command is denied.

To solve this problem, the PIX Firewall redirects the browser's initial connection to its virtual HTTP IP address, authenticates the user, and then redirects the browser to the URL that the user originally requested. Virtual HTTP is transparent to the user; therefore, users enter actual destination URLs in their browsers as they normally would.

Note Do not set the timeout uauth duration to 0 seconds when using the virtual HTTP option. Doing this prevents HTTP connections to the real web server.



Configuration of Virtual HTTP Authentication

pixfirewall (config)#

```
virtual http ip_address [warn]
```

- For inbound clients, the IP address must be an unused global address.
- If the connection is started on either the outside or perimeter, a static and access-list command pair must be configured for the fictitious address.

```
pix1(config)# virtual http 192.168.0.9
```

CSPFA 3.2—11-26

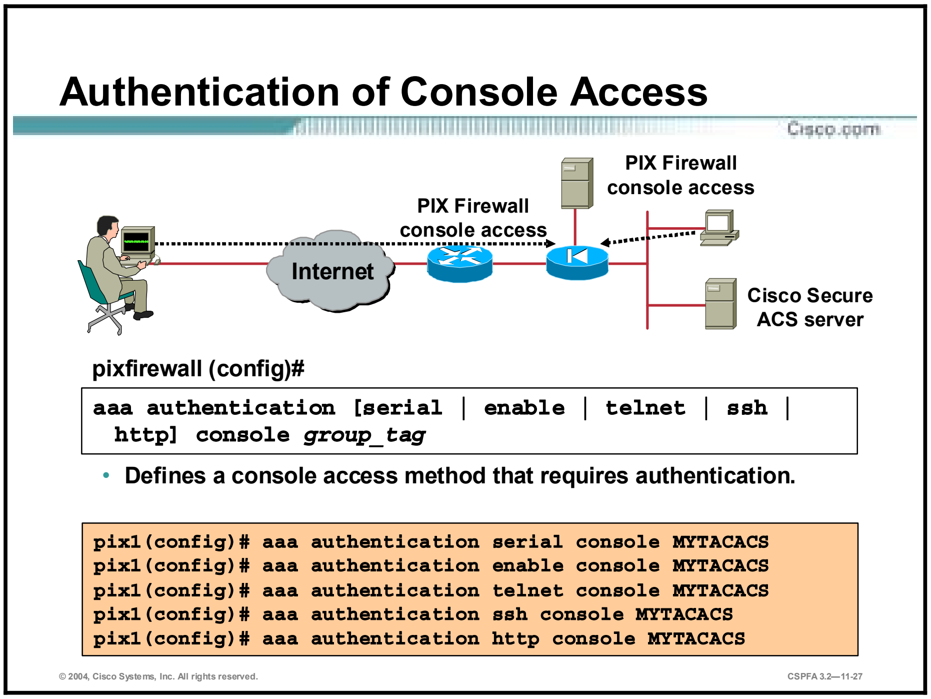
The virtual address identifies the IP address of the virtual HTTP server on the PIX Firewall. For inbound use, the IP address can be any unused global address. Access to this address must be provided by an **access-list** and **static** command pair. To be authenticated, the outside user establishes a virtual HTTP session to 192.168.0.9. The PIX Firewall will intercept the virtual HTTP session and authenticate the user.

The syntax for the **virtual http** command is as follows:

virtual http *ip_address* [warn]

no virtual http *ip_address*

ip_address	The PIX Firewall's network interface IP address.
warn	Informs **virtual http** command users that the command was redirected. This option is applicable only for text-based browsers where the redirect cannot happen automatically.



Authentication of Console Access

Cisco.com
PIX Firewall console access
PIX Firewall console access
Internet
Cisco Secure ACS server

pixfirewall (config)#

```
aaa authentication [serial | enable | telnet | ssh |
  http] console group_tag
```

- Defines a console access method that requires authentication.

```
pix1(config)# aaa authentication serial console MYTACACS
pix1(config)# aaa authentication enable console MYTACACS
pix1(config)# aaa authentication telnet console MYTACACS
pix1(config)# aaa authentication ssh console MYTACACS
pix1(config)# aaa authentication http console MYTACACS
```

CSPFA 3.2—11-27

Use the **aaa authentication console** command to require authentication verification to access the PIX Firewall's console.

Authenticated access to the PIX Firewall console involves different types of prompts, depending on the option you choose with the **aaa authentication console** command:

- **enable** option—Allows three tries before stopping access attempts with an "access denied" message. The **enable** option requests a username and password before accessing privileged mode for serial or Telnet connections.
- **serial** option—Causes the user to be prompted continually until that user successfully logs in. The **serial** option requests a username and password before the first command-line prompt on the serial console connection.
- **ssh** option—Allows three tries before stopping access attempts with a "rejected by server" message. The **ssh** option requests a username and password before the first command-line prompt appears.
- **telnet** option—Causes the user to be prompted continually until that user successfully logs in. The **telnet** option requests a username and password before the first command-line prompt of a Telnet console connection.

The **aaa authentication** command also supports PIX Device Manager (PDM) authentication. By using the **aaa authentication http console** command, you can configure the PIX Firewall to require authentication before allowing PDM to access it. If an **aaa authentication http console** command statement is not defined, you can gain access to the PIX Firewall (via PDM) with no username and the PIX Firewall **enable** password (set with the **password** command). If the **aaa** command is defined but the HTTP authentication request times out, which implies that the AAA server may be down or not available, you can gain access to the PIX Firewall using the username "pix" and the **enable** password.

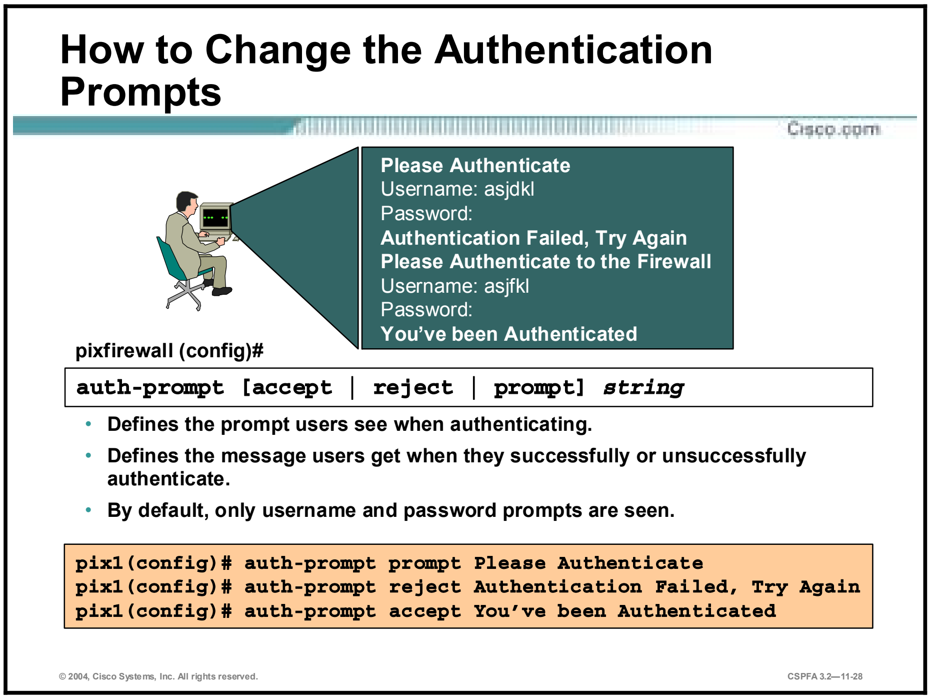
Note The serial console option also logs to a Syslog server changes made to the configuration from the serial console.

The syntaxes for the **aaa authentication console** commands are as follows:

aaa authentication [serial | enable | telnet | ssh | http] console ***group_tag***

no aaa authentication [serial | enable | telnet | ssh | http] console ***group_tag***

serial	Access verification for the PIX Firewall's serial console.
enable	Access verification for the PIX Firewall's privilege mode.
telnet	Access verification for Telnet access to the PIX Firewall console.
ssh	Access verification for SSH access to the PIX Firewall console.
http	Access verification for HTTP access to the PIX Firewall (via PDM).
console	Specifies that access to the PIX Firewall console requires authentication.
group_tag	The group tag set with the **aaa-server** command. To use the local PIX Firewall user authentication database, enter LOCAL for this parameter.



How to Change the Authentication Prompts

Please Authenticate
Username: asjdkl
Password:
Authentication Failed, Try Again
Please Authenticate to the Firewall
Username: asjfkl
Password:
You've been Authenticated

pixfirewall (config)#

```
auth-prompt [accept | reject | prompt] string
```

- **Defines the prompt users see when authenticating.**
- **Defines the message users get when they successfully or unsuccessfully authenticate.**
- **By default, only username and password prompts are seen.**

```
pix1(config)# auth-prompt prompt Please Authenticate
pix1(config)# auth-prompt reject Authentication Failed, Try Again
pix1(config)# auth-prompt accept You've been Authenticated
```

CSPFA 3.2—11-28

Use the **auth-prompt** command to change the AAA challenge text for HTTP, FTP, and Telnet access through the PIX Firewall. This is text that appears above the username and password prompts that you view when logging in.

Note Microsoft Internet Explorer only displays up to 37 characters in an authentication prompt, Netscape Navigator displays up to 120 characters, and Telnet and FTP display up to 235 characters in an authentication prompt.

The syntaxes for the **auth-prompt** commands are as follows:

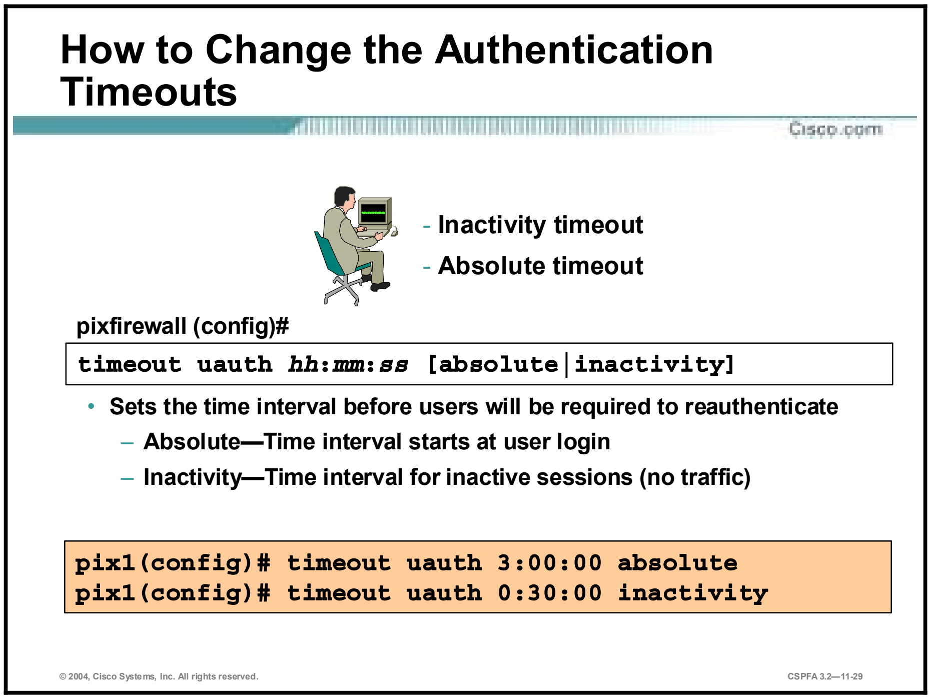
auth-prompt [accept | reject | prompt] ***string***

no auth-prompt [accept | reject | prompt] ***string***

show auth-prompt

clear auth-prompt

accept	If a user authentication via Telnet is accepted, the accept message is displayed.
reject	If a user authentication via Telnet is rejected, the reject message is displayed.
prompt	The AAA challenge prompt string follows this keyword. This keyword is optional for backward compatibility.
string	A string of up to 235 alphanumeric characters. Special characters should not be used; however, spaces and punctuation characters are permitted. Entering a question mark or pressing the Enter key ends the string. (The question mark appears in the string.)



How to Change the Authentication Timeouts

- Inactivity timeout
- Absolute timeout

pixfirewall (config)#

```
timeout uauth hh:mm:ss [absolute|inactivity]
```

- Sets the time interval before users will be required to reauthenticate
  - Absolute—Time interval starts at user login
  - Inactivity—Time interval for inactive sessions (no traffic)

```
pix1(config)# timeout uauth 3:00:00 absolute
pix1(config)# timeout uauth 0:30:00 inactivity
```

CSPFA 3.2—11-29

Use the **timeout uauth** command to specify how long the cache should be kept after the user connections become idle. The timeout command value must be at least two minutes. Use the **clear uauth** command to delete all authorization caches for all users, which causes them to reauthenticate the next time they create a connection.

The inactivity and absolute qualifiers cause users to reauthenticate after either a period of inactivity or an absolute duration. The inactivity timer starts after a connection becomes idle. If a user establishes a new connection before the duration of the inactivity timer, the user is not required to reauthenticate. If a user establishes a new connection after the inactivity timer expires, the user must reauthenticate.

The absolute timer runs continuously, but waits and prompts the user again when the user starts a new connection, such as clicking a link after the absolute timer has elapsed. The user is then prompted to reauthenticate. The absolute timer must be shorter than the xlate timer, otherwise a user could be prompted again after the session has ended.

The inactivity timer gives users the best Internet access because they are not prompted to regularly reauthenticate. Absolute timers provide security and manage the PIX Firewall connections better. Being prompted to reauthenticate regularly helps users manage their use of the resources more efficiently. Also, being prompted minimizes the risk that someone will attempt to continue another user's access after the first user leaves the workstation, such as in a college computer lab.

Both an inactivity timer and an absolute timer can operate at the same time, but you should set the absolute timer duration for a longer period than the inactivity timer. If the absolute timer is set at less than the inactivity timer, the inactivity timer is never invoked. For example, if you set the absolute timer to 10 minutes and the inactivity timer to an hour, the absolute timer prompts the user every 10 minutes, and the inactivity timer will never be started.

If you set the inactivity timer to some duration, but set the absolute timer to 0, users are reauthenticated only after the inactivity time elapses. If you set both timers to 0, users have to reauthenticate on every new connection.

Note Do not set the timeout uauth duration to 0 seconds when using the virtual HTTP option or passive FTP.

The syntaxes for the **timeout uauth** commands are as follows:

timeout uauth *hh:mm:ss* [absolute | inactivity]

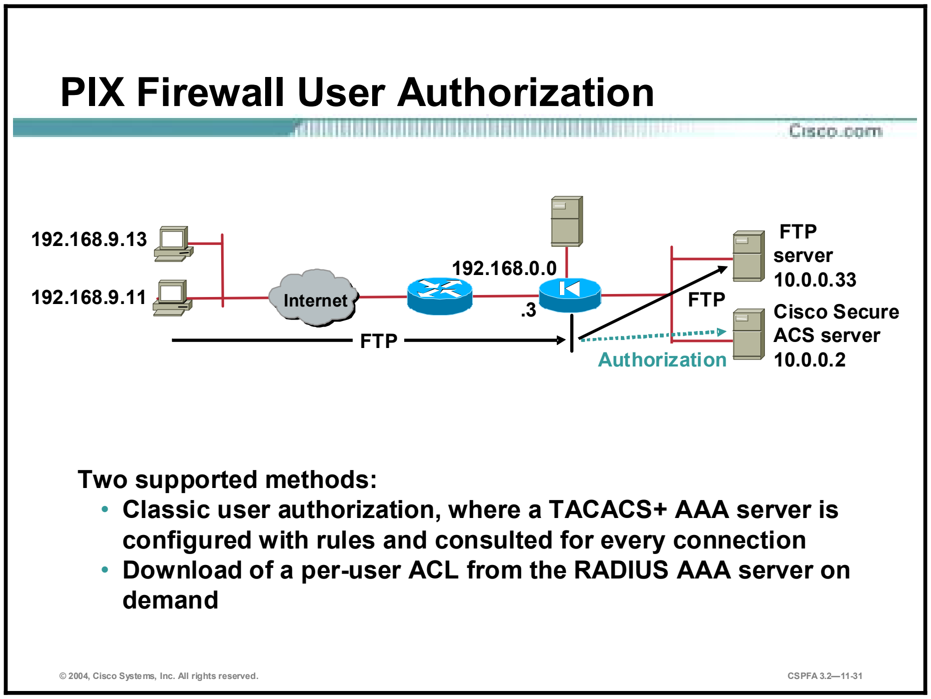
show timeout

clear uauth

uauth *hh:mm:ss*	Duration before the authentication and authorization cache times out and the user has to reauthenticate the next connection. This duration must be shorter than the xlate values. Set to 0 to disable caching.
absolute	Runs the uauth timer continuously, but after timer elapses, waits to reprompt the user until the user starts a new connection (for example, clicking a link in a web browser). To disable absolute, set it to zero (0). The default is 5 minutes.
inactivity	Starts the uauth timer after a connection becomes idle. The default is 0.

Authorization Configuration

This topic discusses the configuration of the PIX Firewall for authorization.



PIX Firewall User Authorization

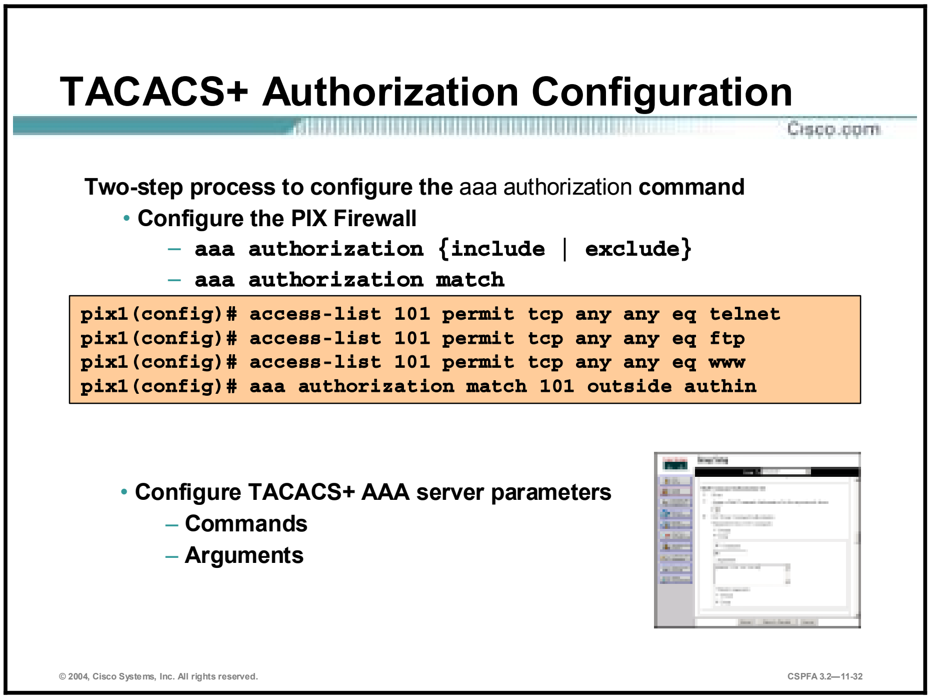
Two supported methods:

- **Classic user authorization, where a TACACS+ AAA server is configured with rules and consulted for every connection**
- **Download of a per-user ACL from the RADIUS AAA server on demand**

CSPFA 3.2—11-31

If you want to allow all authenticated users to perform all operations—HTTP, FTP, and Telnet—through the PIX Firewall, authentication is sufficient and authorization is not needed. But if there is reason to allow only some subset of users or to limit users to certain sites, authorization is needed. The PIX Firewall supports two basic methods of user authorization when you specify per-user access rules in the context of AAA. Two methods are:

- Classic user authorization—The access rules are configured on the TACACS+ AAA server and consulted on demand. With classic authorization, the PIX Firewall is configured with rules specifying which connections need to be authorized by the AAA server. The AAA server is consulted for access rights on demand.
- Download of per-user ACLs—PIX Firewall Software Version 6.2 introduced the ability to store full ACLs on the AAA server and download them to the PIX Firewall as required. Authentication is configured on the PIX Firewall. An ACL is attached to the user or group profile on the AAA server. During the authentication process, an ACL is returned to the PIX Firewall. The ACL is modified based on the source IP address of the authenticated user. This functionality is supported only with RADIUS.



TACACS+ Authorization Configuration

Two-step process to configure the aaa authorization **command**

- **Configure the PIX Firewall**
  - **`aaa authorization {include | exclude}`**
  - **`aaa authorization match`**

```
pix1(config)# access-list 101 permit tcp any any eq telnet
pix1(config)# access-list 101 permit tcp any any eq ftp
pix1(config)# access-list 101 permit tcp any any eq www
pix1(config)# aaa authorization match 101 outside authin
```

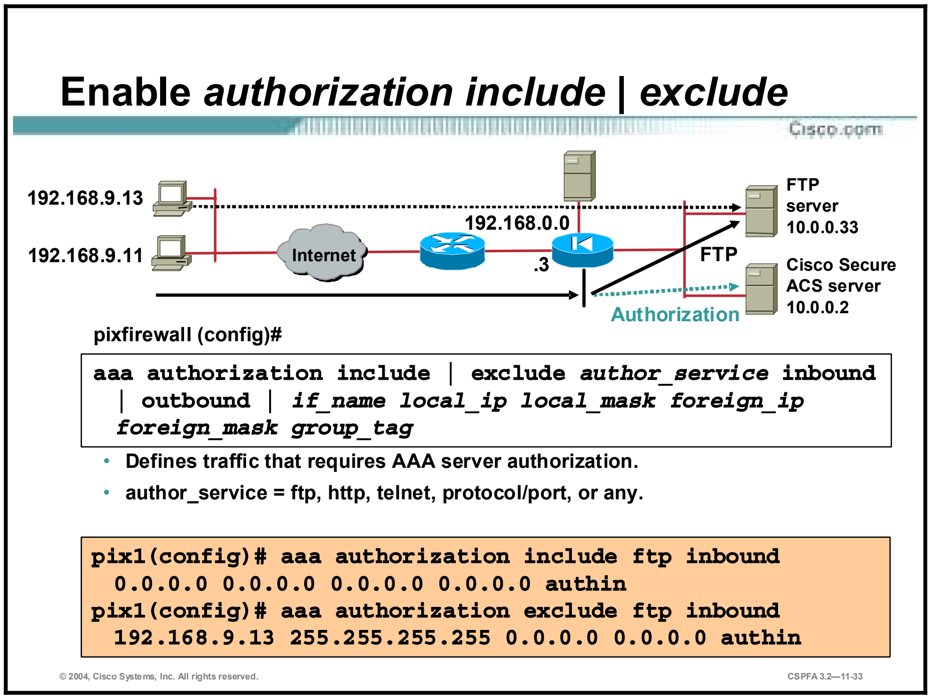
- **Configure TACACS+ AAA server parameters**
  - **Commands**
  - **Arguments**

User authorization is a two-step process, as follows:

Step 1 Configure the PIX Firewall for authorization. The administrator can use the older form of the **aaa authorization {include | exclude}** command or the newer version, the **aaa authorization match** command.

Step 2 Define the TACACS+ AAA server group parameters. The per-group command authorization parameters include commands and arguments.

Note It is assumed that **aaa authentication** configuration was already completed.



Enable *authorization include* | *exclude*

pixfirewall (config)#

```
aaa authorization include | exclude author_service inbound
  | outbound | if_name local_ip local_mask foreign_ip
  foreign_mask group_tag
```

- Defines traffic that requires AAA server authorization.
- author_service = ftp, http, telnet, protocol/port, or any.

```
pix1(config)# aaa authorization include ftp inbound
  0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 authin
pix1(config)# aaa authorization exclude ftp inbound
  192.168.9.13 255.255.255.255 0.0.0.0 0.0.0.0 authin
```

CSPFA 3.2—11-33

The **aaa authorization** command defines which services require authorization. If the traffic matches the source and destination IP addresses and services in the **aaa authorization** command string, the connection must be authorized by the defined TACACS+ AAA server. Services not specified in the command line are authorized implicitly.

In the example in the figure, any outbound FTP sessions must be authorized, with the exception of FTP sessions originating from IP address 192.168.9.13 If authentication passes and authorization is configured, the PIX Firewall forwards the user's session information to the TACACS+ server for authorization. The TACACS+ returns a permit or fail message.

The syntaxes for the **aaa authorization** commands are as follows:

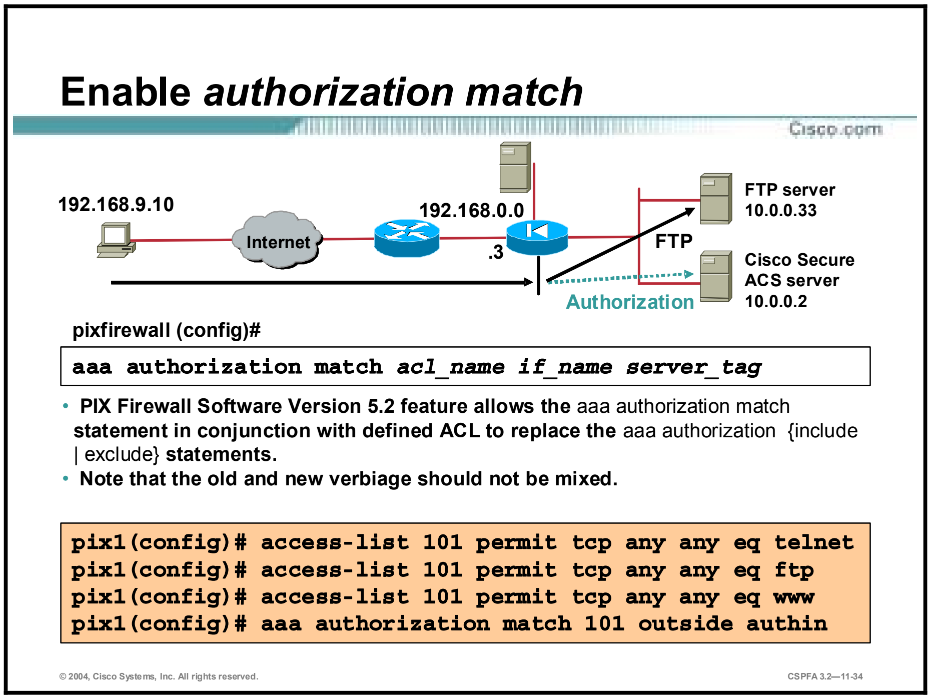
aaa authorization include | **exclude** ***author_service*** **inbound** | **outbound** | ***if_name local_ip local_mask foreign_ip foreign_mask group_tag***

no aaa authorization include | **exclude** ***author_service*** **inbound** | **outbound** | ***if_name local_ip local_mask foreign_ip foreign_mask group_tag***

clear aaa authorization [**include** | **exclude** ***author_service*** **inbound** | **outbound** | ***if_name local_ip local_mask foreign_ip foreign_mask group_tag***]

include *author_service*	The services that require authorization. Use *any, ftp, http, or telnet*. Services not specified are authorized implicitly. Services specified in the **aaa authentication** command do not affect the services, which require authorization.
exclude *author_service*	Creates an exception to a previously stated rule by excluding the specified service from authorization to the specified host. The **exclude** parameter improves the former **except** option by allowing the user to specify a port to exclude for a specific host or hosts.

inbound	Authenticates or authorizes inbound connections. Inbound means the connection originates on the outside interface and is being directed to the inside or any other perimeter interface.
outbound	Authenticates or authorizes outbound connections. Outbound means the connection originates on the inside and is being directed to the outside or any other perimeter interface.
if_name	Interface name from which users require authentication. Use *if_name* in combination with the *local_ip* address and the *foreign_ip* address to determine where access is sought and by whom.
local_ip	The IP address of the host or network of hosts that you want to be authenticated or authorized. You can set this address to 0 to mean all hosts and to let the authentication server decide which hosts are authenticated.
local_mask	Network mask of *local_ip*. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host.
foreign_ip	The IP address of the hosts you want to access the *local_ip* address. Use 0 to mean all hosts.
foreign_mask	Network mask of *foreign_ip*. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host.
group_tag	The group tag set with the **aaa-server** command. Enter LOCAL for the group tag value for local AAA services such as local command authorization using privilege levels, or use the AAA server group tag as defined by the **aaa-server** command.



Enable *authorization match*

pixfirewall (config)#

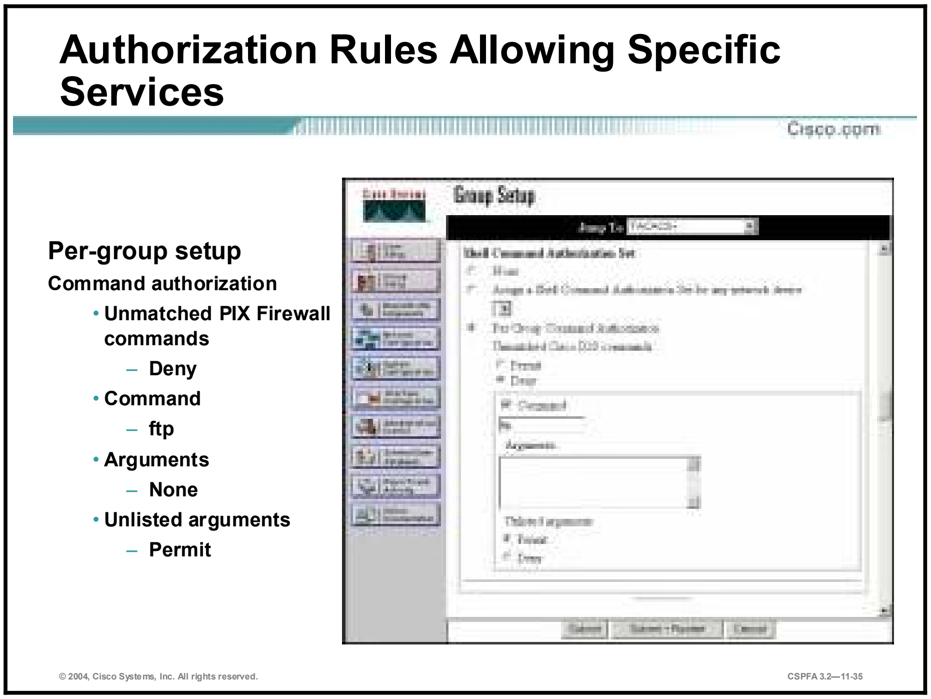
```
aaa authorization match acl_name if_name server_tag
```

- PIX Firewall Software Version 5.2 feature allows the aaa authorization match statement in conjunction with defined ACL to replace the aaa authorization {include | exclude} statements.
- Note that the old and new verbiage should not be mixed.

```
pix1(config)# access-list 101 permit tcp any any eq telnet
pix1(config)# access-list 101 permit tcp any any eq ftp
pix1(config)# access-list 101 permit tcp any any eq www
pix1(config)# aaa authorization match 101 outside authin
```

CSPFA 3.2—11-34

Beginning in PIX Firewall Software Version 5.2, the administrator can define ACLs on the PIX Firewall, then apply them to the **aaa authorization match** command. Any sessions matching the ACL must be authorized by the defined TACACS+ server. In the example in the figure, the three ACL statements are for any-to-any FTP, Telnet, and HTTP traffic. The ACLs are applied to the outside interface. Any traffic matching these characteristics inbound on the outside interface must be authorized by authin TACACS+ server.


Cisco.com
Group Setup
Jump To TACACS+
Shell Command Authorization Set
○ None
○ Assign a Shell Command Authorization Set for any network device
● Per Group Command Authorization
Unmatched Cisco IOS commands
○ Permit
● Deny
☑ Command
ftp
Arguments
Unlisted arguments
● Permit
○ Deny
Submit
Submit + Restart
Cancel

Authorization Rules Allowing Specific Services

Per-group setup

Command authorization

- **Unmatched PIX Firewall commands**
  - **Deny**
- **Command**
  - **ftp**
- **Arguments**
  - **None**
- **Unlisted arguments**
  - **Permit**

CSPFA 3.2—11-35

The **aaa authorization** command defines which traffic to authorize. Clicking the Group Setup button and then selecting the Per Group Command Authorization radio button enables the administrator to permit or deny specific PIX Firewall commands and arguments at the group level. For example, the Executive group might have FTP and HTTP access to all 10.0.9.0/24 servers. The Human Resources group might have FTP and HTTP access to server 10.0.9.6 only. The Per Group Command Authorization option enables the administrator to define authorization for commands, such as FTP, Telnet, and HTTP. It also enables the administrator to define authorization for arguments, such as IP addresses.

To set TACACS+ shell command authorization on a command-by-command basis, select the **Per Group Command Authorization** radio button, then select from the following options:

- Unmatched Cisco IOS commands—To determine how Cisco Secure ACS handles commands that the administrator did not specify in this section, you can choose to either permit or deny, as applicable.
- Command—You can select the Command check box and type the command in the field below it.
- Arguments—For each argument of the command, you can specify whether the argument is to be permitted or denied.
- Unlisted arguments—To permit only those arguments listed, select the Deny option. To allow users to issue all arguments not specifically listed, select the Permit option.

In the example in the figure, a member of a group is authorized to access any IP address via FTP. Complete the following steps to add authorization rules for this specific service in Cisco Secure ACS:

Step 1 Click **Group Setup** from the navigation bar. The Group Setup window opens.

Step 2 Scroll down to the Shell Command Authorization Set area.

Step 3 Select the **Per Group Command Authorization** radio button, which enables the administrator to permit or deny specific command and arguments at the group level.

Step 4 Select the **Deny** radio button, which is found under Unmatched Cisco IOS commands. This step allows only group members to issue FTP commands.

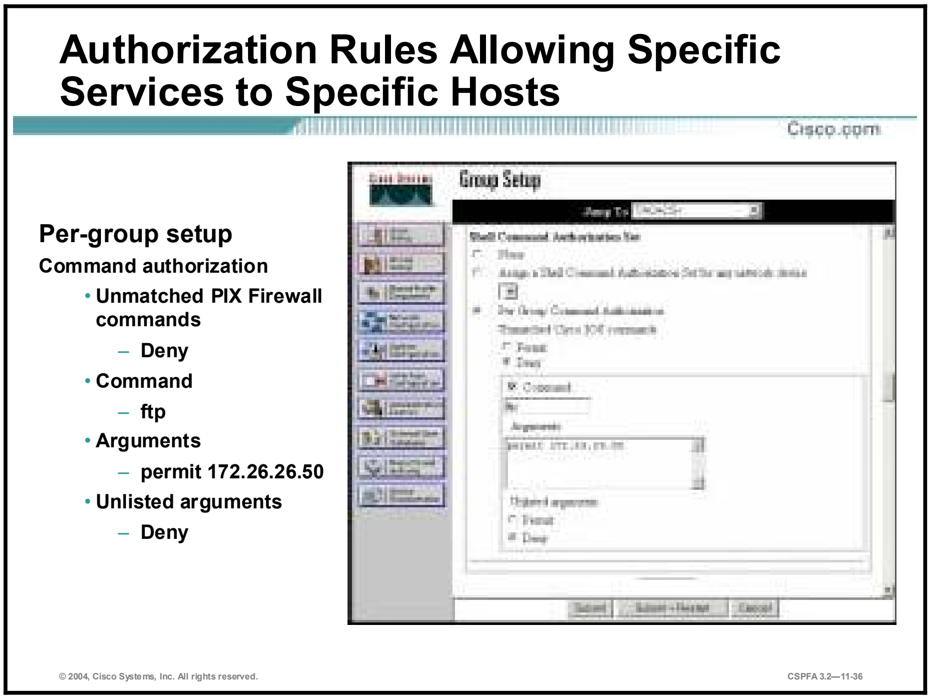
Step 5 Select the **Command** check box.

Step 6 In the command field, enter **FTP**, the allowable service.

Step 7 Leave the Arguments field blank.

Step 8 Select the **Permit** radio button, which is found under Unlisted arguments, to permit the arguments that are not listed.

Step 9 Click the **Submit** button to add more rules, or click the **Submit + Restart** button when you are finished.



Authorization Rules Allowing Specific Services to Specific Hosts

Cisco.com

Per-group setup

Command authorization

- **Unmatched PIX Firewall commands**
  - **Deny**
- **Command**
  - **ftp**
- **Arguments**
  - **permit 172.26.26.50**
- **Unlisted arguments**
  - **Deny**

CSPFA 3.2—11-36

Complete the following steps to add authorization rules for services to specific hosts in Cisco Secure ACS:

Step 1 Click **Group Setup** from the navigation bar. The Group Setup window opens.

Step 2 Scroll down in Group Setup to the Shell Command Authorization Set area.

Step 3 Select the **Per Group Command Authorizatio**n radio button.

Step 4 Select the **Deny** button, which is found under Unmatched Cisco IOS commands. When the administrator selects deny, the user can issue only those commands listed.

Step 5 Select the **Command** check box.

Step 6 In the field below the check box, enter one of the allowable services.

Step 7 In the Arguments field, enter the IP addresses of the host that users are authorized to go to. Use the following format:

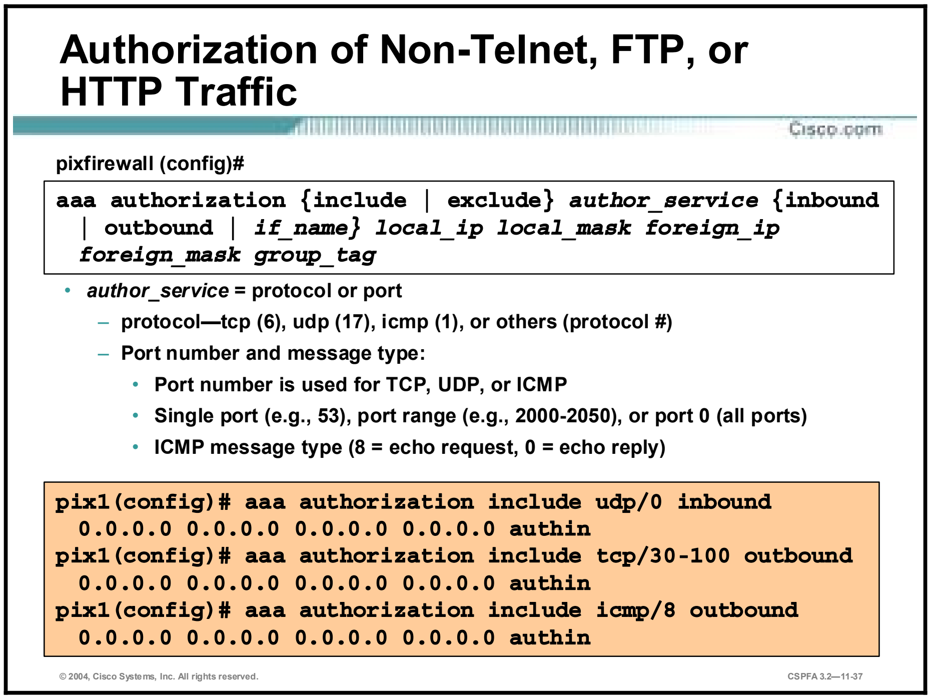
```
permit ip_addr
```

(where *ip_addr* = the IP address of the host)

Step 8 Select the **Deny** radio button, which is found under Unlisted arguments, to deny unlisted arguments.

Step 9 Click the **Submit** button to add more rules, or click the **Submit + Restart** button when you are finished.

In the example in the figure, FTP to 172.26.26.50 is authorized for this group. All other destinations are denied.



Authorization of Non-Telnet, FTP, or HTTP Traffic

pixfirewall (config)#

```
aaa authorization {include | exclude} author_service {inbound
  | outbound | if_name} local_ip local_mask foreign_ip
  foreign_mask group_tag
```

- *author_service* = protocol or port
  - protocol—tcp (6), udp (17), icmp (1), or others (protocol #)
  - Port number and message type:
    - Port number is used for TCP, UDP, or ICMP
    - Single port (e.g., 53), port range (e.g., 2000-2050), or port 0 (all ports)
    - ICMP message type (8 = echo request, 0 = echo reply)

```
pix1(config)# aaa authorization include udp/0 inbound
  0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 authin
pix1(config)# aaa authorization include tcp/30-100 outbound
  0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 authin
pix1(config)# aaa authorization include icmp/8 outbound
  0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 authin
```

CSPFA 3.2—11-37

The syntaxes for the **aaa authorization** of non-Telnet, non-FTP, or non-HTTP commands are as follows:

aaa authorization {include | exclude} *author_service* **{inbound | outbound |** *if_name***}** *local_ip local_mask foreign_ip foreign_mask group_tag*

no aaa authorization [{include | exclude} *author_service* **{inbound | outbound |** *if_name***}** *local_ip local_mask foreign_ip foreign_mask group_tag*

clear aaa [authorization [{include | exclude} *author_service* **{inbound | outbound |** *if_name***}** *local_ip local_mask foreign_ip foreign_mask group_tag***]]**

include *author_service*	The services that require authorization. Use a protocol and port number. Services not specified are authorized implicitly. Services specified in the **aaa authentication** command do not affect the services that require authorization. Use port number of 0 to specify all ports.
exclude *author_service*	Creates an exception to a previously stated rule by excluding the specified service from authorization to the specified host or networks.
inbound	Authenticates or authorizes inbound connections. Inbound means the connection originates on the outside interface and is being directed to the inside or any other perimeter interface.
outbound	Authenticates or authorizes outbound connections. Outbound means the connection originates on the inside and is being directed to the outside or any other perimeter interface.
if_name	Interface name from which users require authentication. Use *if_name* in combination with the *local_ip* address and the *foreign_ip* address to determine where access is sought and from whom.

local_ip	The IP address of the host or network of hosts that you want to be authenticated or authorized. You can set this address to 0 to mean all hosts and to let the authentication server decide which hosts are authenticated.
local_mask	Network mask of *local_ip*. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host.
foreign_ip	The IP address of the hosts you want to access the *local_ip* address. Use 0 to mean all hosts.
foreign_mask	Network mask of *foreign_ip*. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host.
group_tag	The group tag set with the **aaa-server** command.

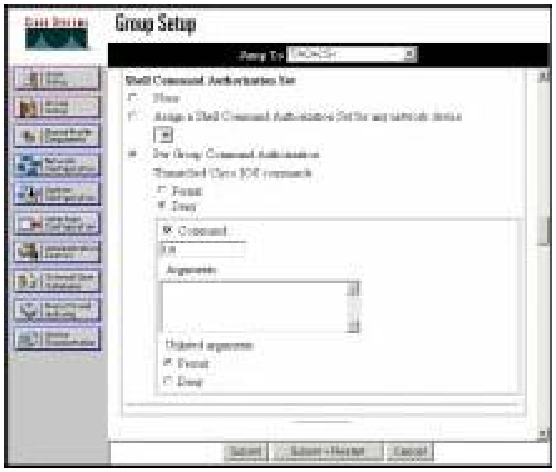
Authorization of Non-Telnet, FTP, or HTTP Traffic on Cisco Secure ACS

Cisco.com

Per-group setup

Command authorization

- **Unmatched PIX commands**
  - **Deny**
- **Command**
  - **1/8**
- **Arguments**
  - **None**
- **Unlisted arguments**
  - **Permit**



Group Setup

Jump To

Shell Command Authorization Set

○ None

○ Assign a Shell Command Authorization Set for any network device

● Per Group Command Authorization

Unmatched Cisco IOS commands

○ Permit

● Deny

☒ Command

1/8

Arguments

Unlisted arguments

● Permit

○ Deny

CSPFA 3.2—11-38

Complete the following steps to add authorization rules for specific non-telnet, FTP, or HTTP services to any host in Cisco Secure ACS:

Step 1 Click **Group Setup** from the navigation bar. The Group Setup window opens.

Step 2 Scroll down in Group Setup to the Shell Command Authorization Set area.

Step 3 Select the **Per Group Command Authorization** radio button.

Step 4 Select the **Deny** radio button, which is found under Unmatched Cisco IOS commands.

Step 5 Select the **Command** check box.

Step 6 In the field below the Command check box, enter an allowable service using the following format:

```
protocol/port
```

(where *protocol* = the protocol number, and *port* = the port number)

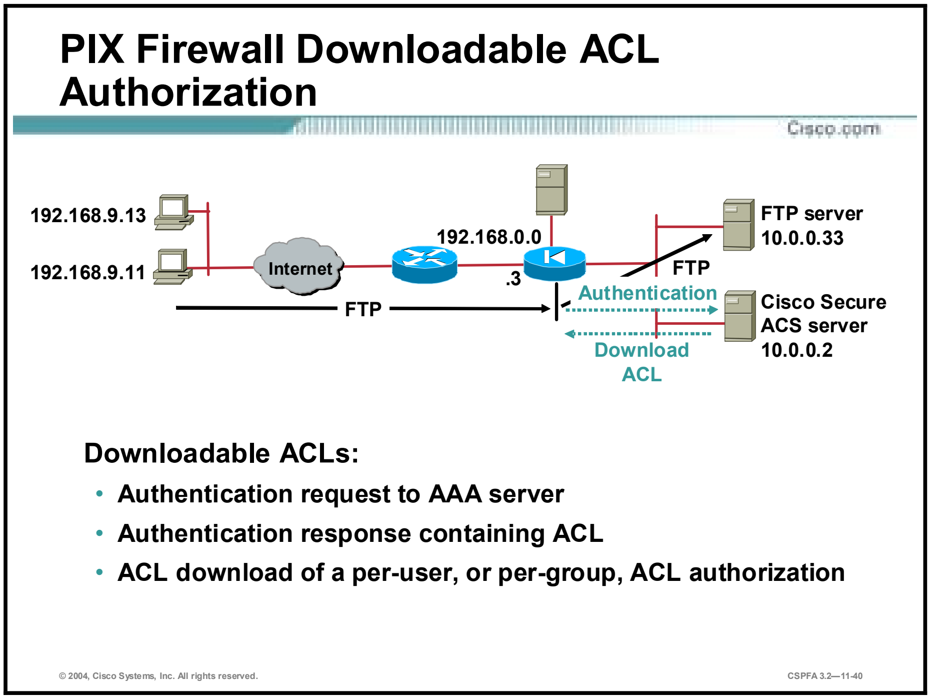
Step 7 Leave the Arguments field blank.

Step 8 Select the **Permit** radio button, which is found under Unlisted arguments.

Step 9 Click the **Submit** radio button to add more rules, or click the **Submit + Restart** radio button when you are finished.

Downloadable ACLs

This topic describes the advantages of downloadable ACLs and explains how to configure them.



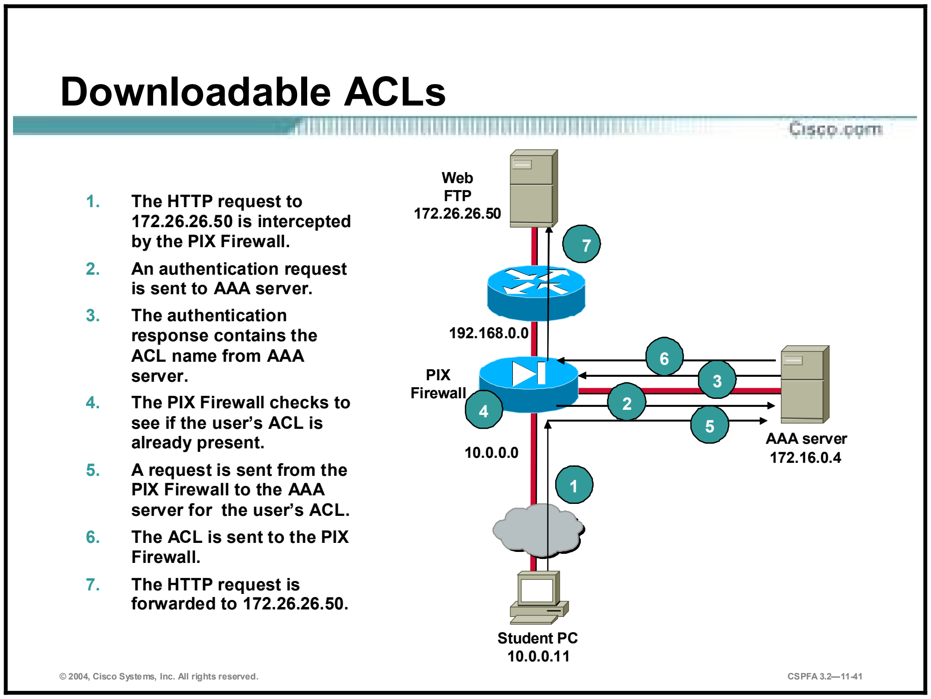
PIX Firewall Downloadable ACL Authorization

Downloadable ACLs:

- **Authentication request to AAA server**
- **Authentication response containing ACL**
- **ACL download of a per-user, or per-group, ACL authorization**

CSPFA 3.2—11-40

PIX Firewall Software Version 6.2 introduced the ability to store full ACLs on the AAA server, and download them to the PIX Firewall as needed. The AAA server is configured with an "ACL" attribute for an individual user or a group. The ACL is returned as a part of the authentication phase. Only authentication needs to be configured on the PIX Firewall, and an ACL attached to the user, or group, profile on the AAA server. The PIX Firewall supports per-user or per-group ACL authorization, so that a user is authorized to do only what is permitted in the user's individual or group ACL entries.



Downloadable ACLs

Cisco.com

1. The HTTP request to 172.26.26.50 is intercepted by the PIX Firewall.
2. An authentication request is sent to AAA server.
3. The authentication response contains the ACL name from AAA server.
4. The PIX Firewall checks to see if the user's ACL is already present.
5. A request is sent from the PIX Firewall to the AAA server for the user's ACL.
6. The ACL is sent to the PIX Firewall.
7. The HTTP request is forwarded to 172.26.26.50.

CSPFA 3.2—11-41

Downloadable ACLs enable you to enter an ACL once, in Cisco Secure ACS, and then load that ACL to any number of PIX Firewalls. Downloadable ACLs work in conjunction with ACLs that are configured directly on the PIX Firewall and applied to its interfaces. Neither type of ACL takes precedence over the other. In order to pass through the PIX Firewall, traffic must be permitted by both the interface ACL and the dynamic ACL if both are applicable. If either ACL denies the traffic, the traffic is prohibited.

Downloadable ACLs are applied to the interface from which the user is prompted to authenticate. They expire when the uauth timer expires and can be removed by entering the **clear uauth** command.

Note Downloadable ACLs are supported with RADIUS only. They are not supported with TACACS+.

As shown in the figure, the following sequence of events takes place when named downloadable ACLs are configured and a user attempts to establish a connection through the PIX Firewall:

1. The user initiates a connection to the web server at 172.26.26.50. The application connection request is intercepted by the PIX Firewall, which then interacts with the user to obtain the username and password.

2. The PIX Firewall builds a RADIUS request containing the user identification and password, and sends it to the AAA server.

3. The AAA server authenticates the user and retrieves from its configuration database the ACL name associated with the user. The AAA server then builds a RADIUS response packet containing the ACL name and sends it to the PIX Firewall.

4. The PIX Firewall checks to see if it already has the named ACL. A downloadable ACL is not downloaded again as long as it exists on the PIX Firewall. Furthermore, to keep ACLs

synchronized between a PIX Firewall and an AAA server, the AAA server downloads a version identification to the PIX Firewall along with the ACL name. This practice enables the PIX Firewall to determine whether it needs to request an updated ACL.

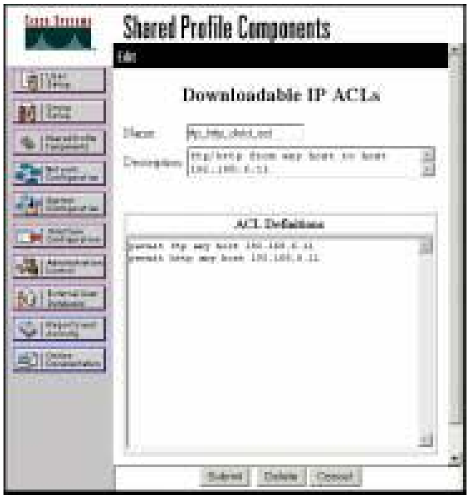
5. If the named ACL is not present, the PIX Firewall uses the name as a user identification and a null password to build a RADIUS access request. The PIX Firewall then sends the RADIUS access request to the AAA server.
6. The AAA server retrieves from its configuration database the ACL associated with the ACL name. The AAA server then builds a RADIUS response packet containing the ACL and sends it to the PIX Firewall.
7. The PIX Firewall extracts the ACL, adds it to its configuration, and forwards the connection request to the application server. The user then connects and interacts with the application server.

The downloaded ACL appears on the PIX Firewall as shown below. The ACL name is the name for the ACL as defined in the Shared Profile Component (SPC), and 3b5385f7 is a unique version identification.

```
access-list #ACSACL#-PIX-acs_ten_acl-3b5385f7 permit ftp any host
172.26.26.50

access-list #ACSACL#-PIX-acs_ten_acl-3b5385f7 permit http any host
172.26.26.50
```

Configuring Downloadable ACLs in Cisco Secure ACS

Cisco.com



Name: ftp_http_dwld_acl

Description: ftp/http from any host to host 192.168.6.11

```
permit ftp any host 192.168.6.11
permit http any host 192.168.6.11
```

CSPFA 3.2—11-42

There are two methods of configuring downloadable ACLs on the AAA server. The first method, downloading named ACLs, is to configure the Shared Profile Components (SPC) to include both the ACL name and the actual ACL and then configure a user, or group, authentication profile to include the SPC. . If you configure a downloadable ACL as a named SPC, you can apply that ACL to any number of Cisco Secure ACS user, or group, profiles. This method should be used when there are frequent requests for downloading a large ACL.

The second method is to configure on an AAA server a user authentication profile that includes the actual PIX Firewall ACL. In this case, the ACL is not identified by a name. You must define each ACL entry in the user profile. This method should be used when there are not frequent requests for the same ACL. For instructions on downloading ACLs without names, refer to the documentation on Cisco.com.

Complete the following steps on the AAA server to configure named downloadable ACLs:

Step 1 Choose **Interface Configuration>Advanced Options** from the main Cisco Secure ACS window to enable the Downloadable ACLs option. Within the Advanced Options group box, select the following:

- **User-Level Downloadable ACLs**
- **Group-Level Downloadable ACLs**

Step 2 Select **Downloadable IP ACLs** from the Shared Profile Components menu item.

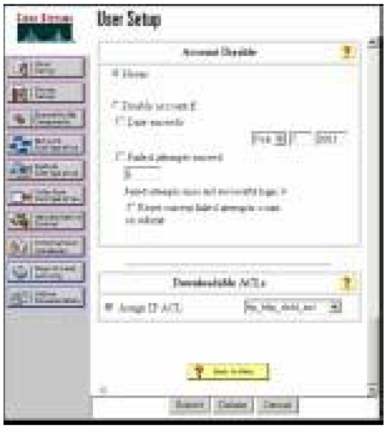
Step 3 Click **Add** to add an ACL definition. Enter the name, description, and the actual definitions for the ACL.

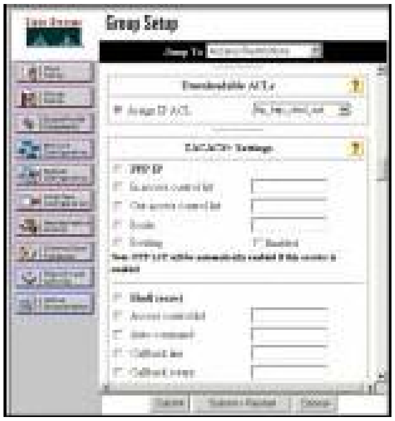
The ACL definition consists of one or more PIX Firewall **access-list** command statements, with each statement on a separate line. Each statement must be entered without the **access-list** keyword and the *acl_ID* argument for the ACL. The rest of the command line must conform to the syntax and semantics rules of the PIX Firewall **access-list** command. A PIX Firewall

Syslog message is logged if there is an error in a downloaded **access-list** command. When you have finished specifying the ACL, click **Submit**.

Assigning the ACL to the User or Group

Cisco.com



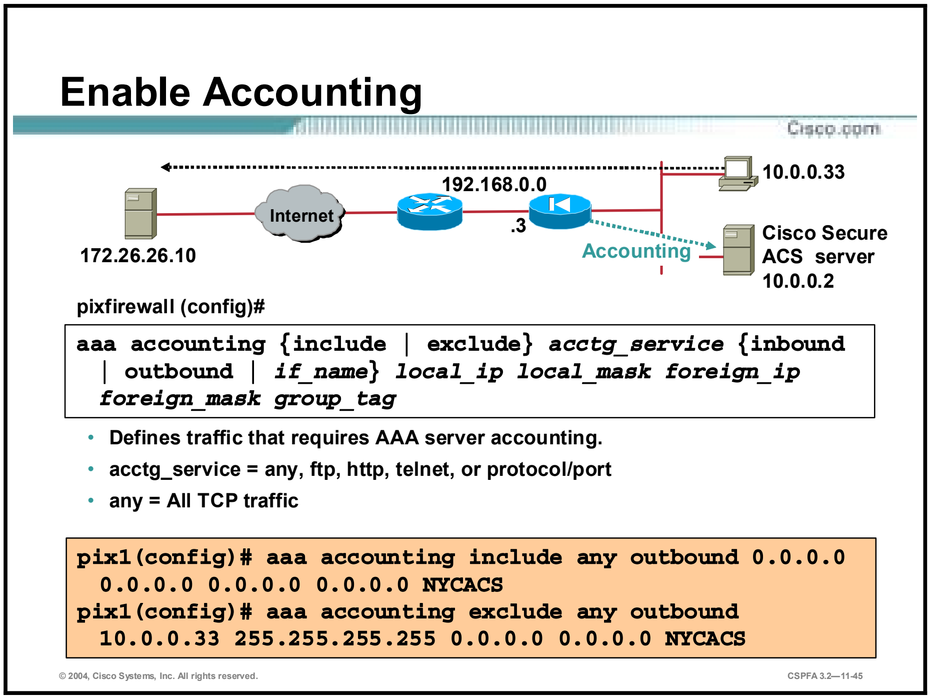


CSPFA 3.2—11-43

After you have configured the downloadable ACL, configure a Cisco Secure ACS user or a group through User Setup or Group Setup to include the defined ACL in the user or group settings.

Accounting Configuration

This topic demonstrates how to enable and configure accounting for all services, specific services, or no services.



Enable Accounting

pixfirewall (config)#

```
aaa accounting {include | exclude} acctg_service {inbound
  | outbound | if_name} local_ip local_mask foreign_ip
  foreign_mask group_tag
```

- Defines traffic that requires AAA server accounting.
- acctg_service = any, ftp, http, telnet, or protocol/port
- any = All TCP traffic

```
pix1(config)# aaa accounting include any outbound 0.0.0.0
  0.0.0.0 0.0.0.0 0.0.0.0 NYCACS
pix1(config)# aaa accounting exclude any outbound
  10.0.0.33 255.255.255.255 0.0.0.0 0.0.0.0 NYCACS
```

CSPFA 3.2—11-45

User accounting records can be kept on a designated AAA server. Accounting information is sent only to the active server in a server group. Traffic that is not specified by an **include** statement is not processed. In the example in the figure, accounting records are kept on the AAA server for all outbound connections except for those connections originating from host 10.0.0.33.

The syntaxes for the **aaa accounting** commands are as follows:

aaa accounting{ include | exclude} ***acctg_service*** **{inbound | outbound |** ***if_name*****}** ***local_ip local_mask foreign_ip foreign_mask group_tag***

no aaa accounting {include | exclude} ***authen_service*** **{inbound | outbound |** ***if_name*****}** ***group_tag***

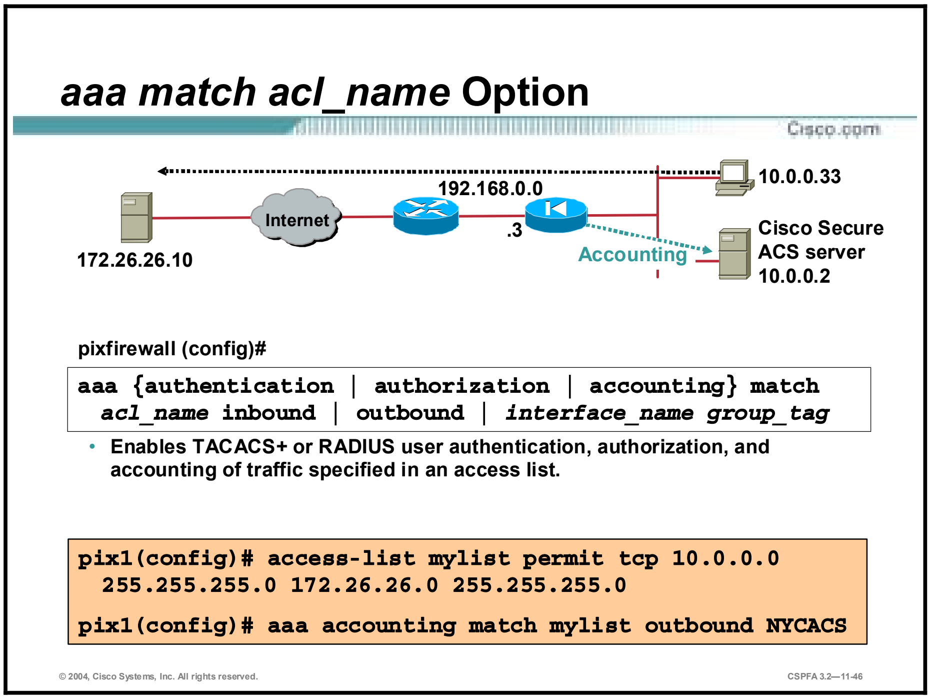
clear aaa [accounting [{include | exclude} ***authen_service*** **{inbound | outbound |** ***if_name*****}** ***group_tag*****]]**

include ***acctg_service***	The accounting service. Accounting is provided for all services, or you can limit it to one or more services. Possible values are *any, ftp, http, telnet,* or *protocol/port.* Use any to provide accounting for all TCP services. To provide accounting for UDP services, use the *protocol/port* form.
exclude ***acctg_service***	Create an exception to a previously stated rule by excluding the specified service from authentication, authorization, or accounting to the specified host. The **exclude** parameter improves the former **except** option by allowing the user to specify a port to exclude to a specific host or hosts.

inbound	Authenticates or authorizes inbound connections. Inbound means the connection originates on the outside interface and is being directed to the inside or any other perimeter interface.
outbound	Authenticates or authorizes outbound connections. Outbound means the connection originates on the inside and is being directed to the outside or any other perimeter interface.
if_name	Interface name from which users require authentication. Use *if_name* in combination with the *local_ip* address and the *foreign_ip* address to determine where access is sought and from whom.
local_ip	The IP address of the host or network of hosts that you want to be authenticated or authorized. You can set this address to 0 to mean all hosts and to let the authentication server decide which hosts are authenticated.
local_mask	Network mask of *local_ip*. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host.
foreign_ip	The IP address of the hosts you want to access the *local_ip* address. Use 0 to mean all hosts.
foreign_mask	Network mask of *foreign_ip*. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host.
group_tag	The group tag set with the **aaa-server** command.

To specify the value of the *acctg_service* argument using the *protocol/port* form, enter the protocol as a number (6 for TCP, 17 for UDP, 1 for Internet Control Message Protocol [ICMP], and so on). The port is the TCP or UDP destination port. A port value of 0 means all ports. If the protocol specified is ICMP, the port is the ICMP type, such as 8 for ICMP echo and 0 for ICMP echo-reply. Examples of the **aaa accounting** command using protocol/port form follow:

- **aaa accounting include 17/53 inbound 0 0 0 0 authin** Enables accounting for Domain Name System (DNS) lookups from the outside interface
- **aaa accounting include 1/0 outbound 0 0 0 0 authout**—Enables accounting of ICMP echo-reply packets arriving at the inside interface from inside hosts
- **aaa accounting include 1/8 outbound 0 0 0 0 authout** Enables accounting only for ICMP echoes (pings) that arrive at the inside interface from an inside host


192.168.0.0
10.0.0.33
Internet
.3
Accounting
172.26.26.10
Cisco Secure
ACS server
10.0.0.2

aaa match acl_name Option

pixfirewall (config)#

```
aaa {authentication | authorization | accounting} match
  acl_name inbound | outbound | interface_name group_tag
```

- Enables TACACS+ or RADIUS user authentication, authorization, and accounting of traffic specified in an access list.

```
pix1(config)# access-list mylist permit tcp 10.0.0.0
  255.255.255.0 172.26.26.0 255.255.255.0
pix1(config)# aaa accounting match mylist outbound NYCACS
```

CSPFA 3.2—11-46

In the PIX Firewall Software Versions 5.2 and higher, the **match** *acl_name* option is available in the **aaa** command. The **aaa** command can take part of its input from an ACL.

In the example in the figure, the ACL mylist permits all TCP traffic from network 10.0.0.0 to network 172.26.26.0. The **match** *acl_name* option in the **aaa** command instructs the PIX Firewall to require authentication when the action the user is trying to perform matches the actions specified in mylist. Therefore, any time a user on the 10.0.0.0 internal network uses any FTP, HTTP, HTTPS, or Telnet applications to access network 172.26.26.0, that user will be required to authenticate. In other words, the command **aaa authentication match mylist outbound authout** is equal to the command **aaa authentication include any outbound 10.0.0.0 255.255.255.0 172.26.26.0 255.255.255.0 authout**.

Traditional **aaa** command configuration and functionality continue to work as in previous versions and are not converted to the ACL format. Hybrid configurations, which are traditional configurations combined with the new ACL configurations, are not recommended.

The syntax for the **aaa authentication | authorization | accounting** command is as follows:

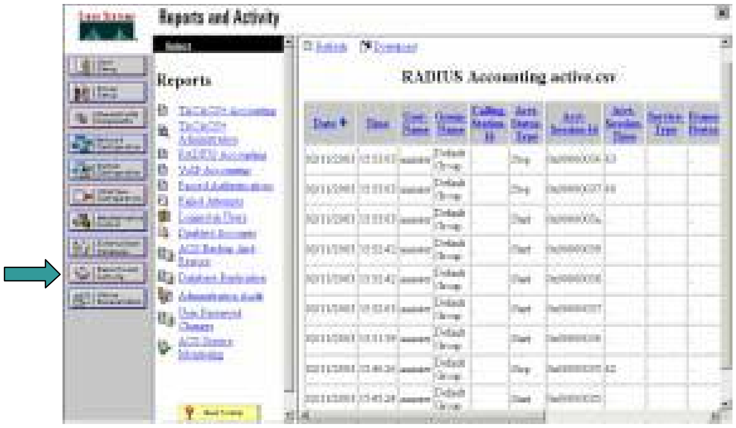
aaa authentication | authorization | accounting match *acl_name* inbound | outbound | *if_name group_tag*

match *acl_name*	Specifies an **access-list** command statement name.
inbound	Authenticates or authorizes inbound connections. Inbound means the connection originates on the outside interface and is being directed to the inside interface.
outbound	Authenticates or authorizes outbound connections. Outbound means the connection originates on the inside and is being directed to the outside interface.
if_name	Interface name from which users require authentication. Use *if_name* in combination with the *local_ip* address and the *foreign_ip* address to determine where access is sought and by whom.

group_tag	The group tab set with the **aaa-server** command. To use the local PIX Firewall user authentication database, enter LOCAL for this parameter.

How to View Accounting Information in Cisco Secure ACS-NT

Cisco.com

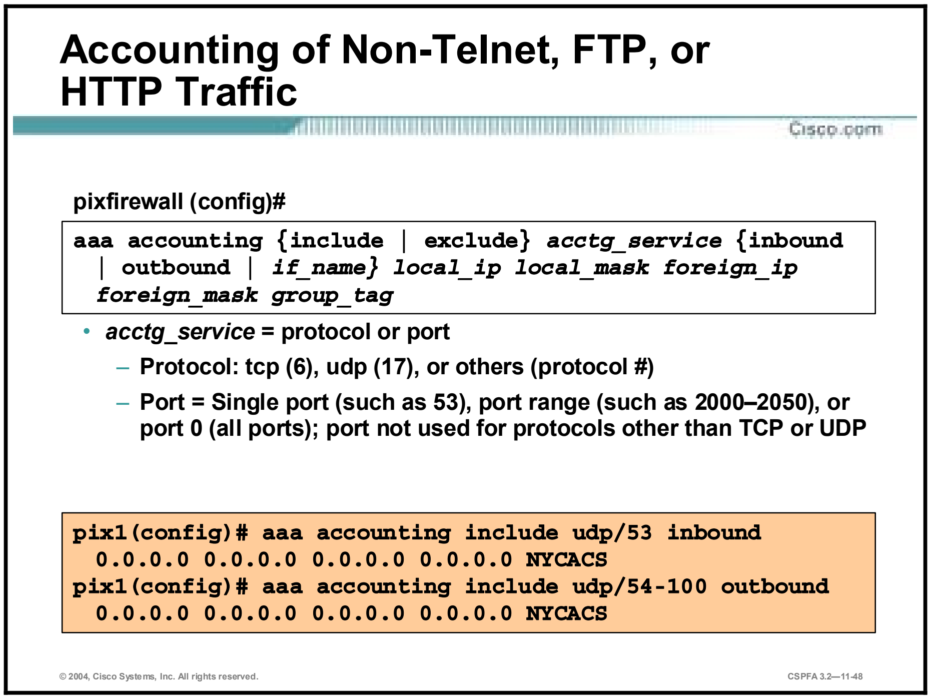


CSPFA 3.2—11-47

Complete the following steps to add accounting rules in Cisco Secure ACS:

Step 1 Click the **Reports and Activity** button in the navigation bar. The Reports and Activity window opens.

Step 2 Click the **RADIUS Accounting** link to display the accounting records.



Accounting of Non-Telnet, FTP, or HTTP Traffic

Cisco.com

pixfirewall (config)#

```
aaa accounting {include | exclude} acctg_service {inbound
  | outbound | if_name} local_ip local_mask foreign_ip
  foreign_mask group_tag
```

- *acctg_service* = protocol or port
  - Protocol: tcp (6), udp (17), or others (protocol #)
  - Port = Single port (such as 53), port range (such as 2000–2050), or port 0 (all ports); port not used for protocols other than TCP or UDP

```
pix1(config)# aaa accounting include udp/53 inbound
  0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 NYCACS
pix1(config)# aaa accounting include udp/54-100 outbound
  0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 NYCACS
```

CSPFA 3.2—11-48

The syntaxes for the **aaa accounting** commands for non-Telnet, FTP, or HTTP traffic are as follows:

aaa accounting include | exclude *acctg_service* **inbound | outbound |** *if_name local_ip local_mask foreign_ip foreign_mask group_tag*

no aaa accounting include | exclude *authen_service* **inbound | outbound |** *if_name group_tag*

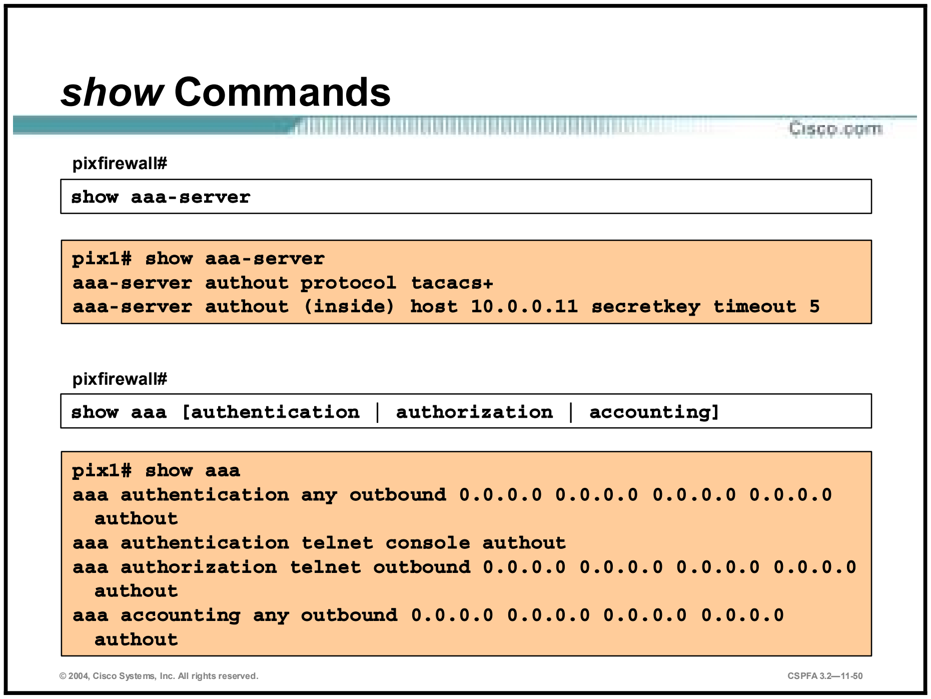
clear aaa [accounting [include | exclude *authen_service* **inbound | outbound |** *if_name group_tag***]]**

include *acctg_service*	The accounting service. Accounting is provided for all services, or you can limit it to one or more services. Possible values are *any, ftp, http*, *https*, *telnet or protocol/port*. Use any to provide accounting for all TCP services. To provide accounting for UDP services, use the protocol/port form.
exclude *acctg_service*	Creates an exception to a previously stated rule by excluding the specified service from authentication, authorization, or accounting to the specified host. The **exclude** parameter improves the former **except** option by enabling the user to specify a port to exclude to a specific host or hosts.
inbound	Authenticates or authorizes inbound connections. Inbound means the connection originates on the outside interface and is being directed to the inside or any other perimeter interface.
outbound	Authenticates or authorizes outbound connections. Outbound means the connection originates on the inside and is being directed to the outside or any other perimeter interface.
if_name	Interface name from which users require authentication. Use *if_name* in combination with the *local_ip* address and the *foreign_ip* address to determine where access is sought and by whom.

local_ip	The IP address of the host or network of hosts that you want to be authenticated or authorized. You can set this address to 0 to mean all hosts and to let the authentication server decide which hosts are authenticated.
local_mask	Network mask of *local_ip*. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host.
foreign_ip	The IP address of the hosts you want to access the *local_ip* address. Use 0 to mean all hosts.
foreign_mask	Network mask of *foreign_ip*. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host.
group_tag	The group tag set with the **aaa-server** command. To use the local PIX Firewall user authentication database, enter LOCAL for this parameter.

Troubleshooting the AAA Configuration

This topic discusses the procedure for verifying the authentication, authorization, and accounting (AAA) configuration.



show Commands

pixfirewall#

```
show aaa-server
```

```
pix1# show aaa-server
aaa-server authout protocol tacacs+
aaa-server authout (inside) host 10.0.0.11 secretkey timeout 5
```

pixfirewall#

```
show aaa [authentication | authorization | accounting]
```

```
pix1# show aaa
aaa authentication any outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
  authout
aaa authentication telnet console authout
aaa authorization telnet outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
  authout
aaa accounting any outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
  authout
```

The syntaxes for the **show aaa-server** and **show aaa** commands are as follows:

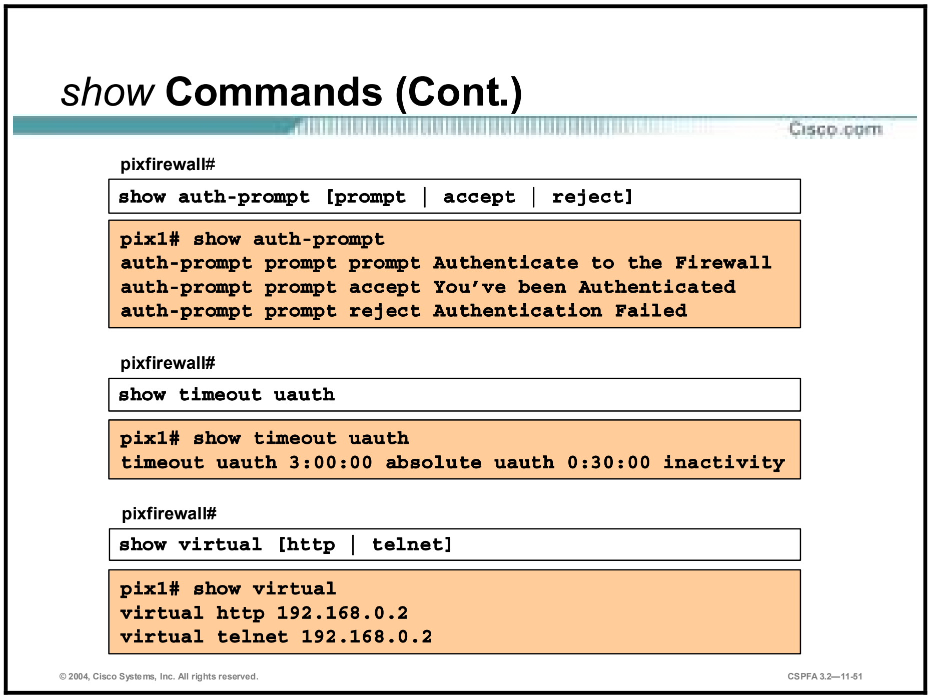
show aaa-server

clear aaa-server [group_tag]

show aaa [authentication | authorization | accounting]

group tag	An alphanumeric string that is the name of the server group.
if_name	The name of the interface on which the server resides.
host *server_ip*	The IP address of the TACACS+ or RADIUS server.
key	A case-sensitive, alphanumeric keyword of up to 127 characters that is the same value as the key on the TACACS+ server. Any characters entered past the limit of 127 are ignored. The key is used between the client and server for encrypting data between them. The key must be the same on both the client and server systems. Spaces are not permitted in the key, but other special characters are.
timeout *seconds*	A retransmit timer that specifies the duration of the period during which the PIX Firewall retries access. The PIX Firewall retries access to the AAA server four times, each time for the length of this period, before choosing the next AAA server. The default is 5 seconds. The maximum time is 30 seconds.
authentication	Displays user authentication, prompts user for username and password, and verifies information with the authentication server.

authorization	Displays TACACS+ user authorization for services. (The PIX Firewall does not support RADIUS authorization.) The authentication server determines which services the user is authorized to access.
accounting	Displays accounting services with the authentication server. Use of this command requires that you previously used the **aaa-server** command to designate an authentication server.



show Commands (Cont.)

pixfirewall#

```
show auth-prompt [prompt | accept | reject]
```

```
pix1# show auth-prompt
auth-prompt prompt prompt Authenticate to the Firewall
auth-prompt prompt accept You've been Authenticated
auth-prompt prompt reject Authentication Failed
```

pixfirewall#

```
show timeout uauth
```

```
pix1# show timeout uauth
timeout uauth 3:00:00 absolute uauth 0:30:00 inactivity
```

pixfirewall#

```
show virtual [http | telnet]
```

```
pix1# show virtual
virtual http 192.168.0.2
virtual telnet 192.168.0.2
```

CSPFA 3.2—11-51

The syntaxes for the **show auth-prompt**, **show timeout uauth**, and the **show virtual** commands are as follows:

show auth-prompt [**prompt** | **accept** | **reject**]

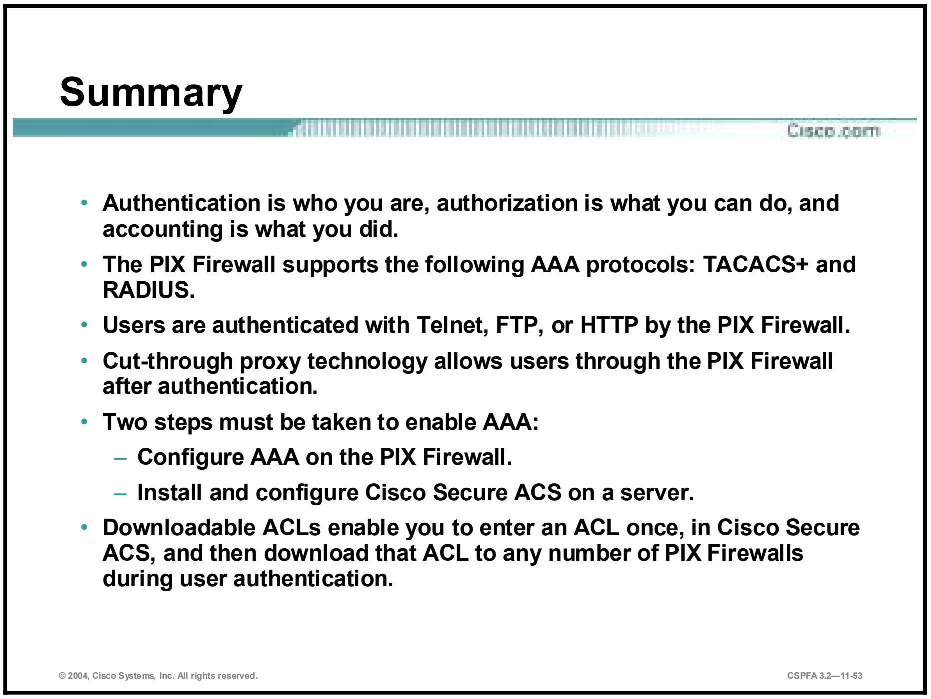
show timeout uauth

show virtual [**http** | **telnet**]

prompt	Displays the prompt users get when authenticating.
accept	Displays the message users get when successfully authenticating.
reject	Displays the message users get when unsuccessfully authenticating.
timeout uauth	Displays the current uauth timer values for all authenticated users.
http	Displays the virtual HTTP configuration.
telnet	Displays the virtual Telnet configuration.

Summary

This topic summarizes what you have learned in this lesson.



Summary

- Authentication is who you are, authorization is what you can do, and accounting is what you did.
- The PIX Firewall supports the following AAA protocols: TACACS+ and RADIUS.
- Users are authenticated with Telnet, FTP, or HTTP by the PIX Firewall.
- Cut-through proxy technology allows users through the PIX Firewall after authentication.
- Two steps must be taken to enable AAA:
  - Configure AAA on the PIX Firewall.
  - Install and configure Cisco Secure ACS on a server.
- Downloadable ACLs enable you to enter an ACL once, in Cisco Secure ACS, and then download that ACL to any number of PIX Firewalls during user authentication.

Lab Exercise—Configure AAA on the PIX Firewall Using Cisco Secure ACS for Windows 2000

Complete the following lab exercise to practice what you learned in this lesson.

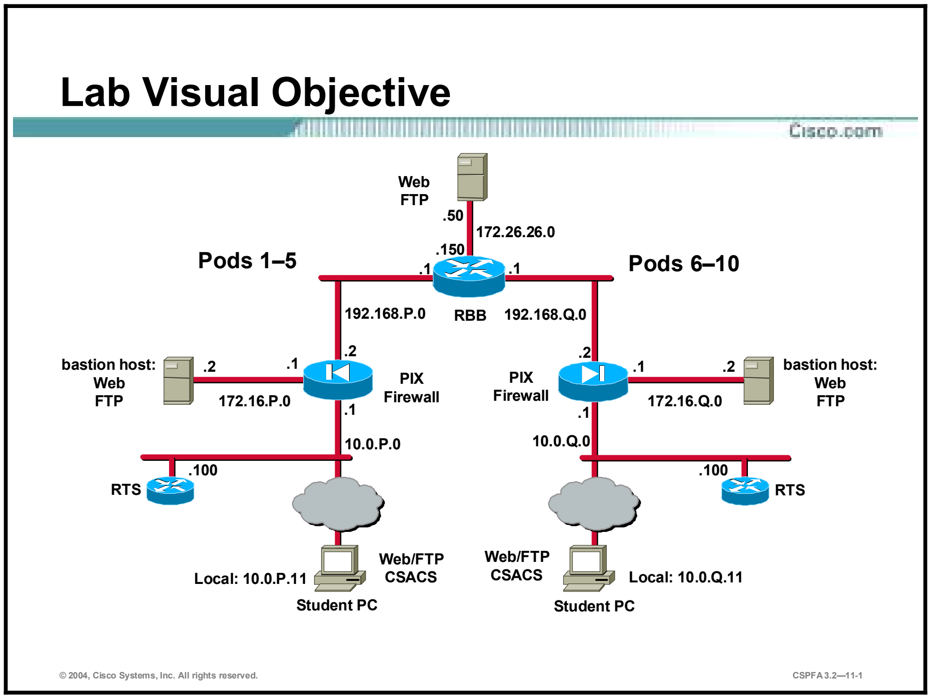
Objectives

In this lab exercise you will complete the following tasks:

- Install the Cisco Secure Access Control Server (ACS) for a Windows 2000 server.
- Add a user to the Cisco Secure ACS database.
- Identify the AAA server and protocol.
- Configure and test inbound authentication.
- Configure and test outbound authentication.
- Configure and test console access authentication.
- Configure and test virtual Telnet authentication.
- Change and test authentication timeouts and prompts.
- Configure ACS to write downloadable ACLs during authentication.
- Test downloadable ACLs with inbound authentication.
- Test downloadable ACLs with outbound authentication.
- Configure and test accounting.

Visual Objective

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—11-1

Task 1—Install the Cisco Secure ACS for a Windows 2000 Server

Complete the following steps to install Cisco Secure ACS on your Windows 2000 server:

Step 1 To install Cisco Secure ACS on your student PC from the files on your hard drive, open the Cisco Secure ACS v3.2 folder on your desktop, and double-click the **setup.exe** program.

Step 2 Click **Accept** to accept the Software License Agreement. The Welcome window opens.

Step 3 Read the Welcome window. Click **Next** to continue. The Before You Begin window opens.

Step 4 Read and then select all four check boxes for the items in the Before You Begin frame. This is a reminder of things you should do prior to installation. Click **Next** to continue. The Choose Destination Location window opens.

Step 5 Use the default installation folder indicated in the Choose Destination Location windows by clicking **Next** to continue. The Authentication Database Configuration windows open.

Step 6 Verify that **Check the Cisco Secure ACS database only** is already selected in the Authentication Database Configuration frame. Click **Next** to continue.

Step 7 Enter the following information in the Cisco Secure ACS Network Access Server Details frame:

- Authenticate users: **RADIUS (Cisco IOS/PIX)**
- Access server name: **pixP**
  (where P = pod number)

- Access server IP address: **10.0.P.1**
  (where P = pod number)
- Windows 2000 Server IP address: **10.0.P.11**
  (where P = pod number)
- TACACS+ or RADIUS key: **secretkey**

Step 8 Click **Next** to start the file installation process.

Step 9 Select all six items displayed in the Advanced Options frame. Click **Next** to continue.

Step 10 Verify that **Enable Log-in Monitoring** is already selected in the Active Service Monitoring frame. Click **Next** to continue.

Step 11 Deselect **Yes, I want to configure IOS software now**.

Step 12 Click **Next** to continue.

Step 13 Verify that the following are already selected in the Cisco Secure ACS Service Initiation window:

- Yes, I want to start the Cisco Secure ACS Service now.
- Yes, I want Setup to launch the Cisco Secure ACS Administrator from my browser following installation.

Step 14 Deselect **Yes, I want to review the Readme file**.

Step 15 Click **Next** to start the Cisco Secure ACS service.

Step 16 Read the Setup Complete window and then click **Finish** to end the installation wizard and start your web browser with Cisco Secure ACS.

Task 2—Add a User to the Cisco Secure ACS Database

Complete the following steps to add a user to the Cisco Secure ACS database in your Windows NT server:

Step 1 The Cisco Secure ACS interface should now be displayed in your web browser. Click **User Setup** to open the User Setup interface.

Step 2 Add a user by entering **aaauser** in the user field.

Step 3 Click **Add/Edit** to go into the user information edit window.

Step 4 Within the User Setup group box, locate the Password and Confirm Password fields directly beneath the following text:

Cisco Secure PAP (Also used for CHAP/MS-CHAP/ARAP, if the Separate field is not checked.)

Step 5 Give the user a password by entering **aaapass** in both the Password and Confirm Password fields.

Step 6 Click **Submit** to add the new user to the Cisco Secure ACS database. Wait for the interface to return to the User Setup main window.

Task 3—Identify the AAA Server and Protocol

Complete the following steps to identify the AAA server and the AAA protocol on the PIX Firewall:

Step 1 Create a group tag called MYRADIUS and assign the TACACS+ protocol to it:

```
pixP(config)# aaa-server MYRADIUS protocol radius
```

Step 2 Assign the Cisco Secure ACS IP address and the encryption key secretkey:

```
pixP(config)# aaa-server MYRADIUS (inside) host insidehost secretkey
```

Step 3 Verify your configuration:

```
pixP(config)# show aaa-server
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server LOCAL protocol local
aaa-server MYRADIUS protocol radius
aaa-server MYRADIUS (inside) host insidehost secretkey timeout 10
```

Task 4—Configure and Test Inbound Authentication

Complete the following steps to enable the use of inbound authentication on the PIX Firewall:

Step 1 Configure the PIX Firewall to require authentication for all inbound traffic:

```
pixP(config)# aaa authentication include any inbound 0 0 0 0 MYRADIUS
```

Step 2 Verify your configuration:

```
pixP(config)# show aaa
aaa authentication include tcp/0 outside 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 MYRADIUS
```

Step 3 Enable console logging of all messages:

```
pixP(config)# logging console debug
```

Note: If your web browser is open, close it. Choose **File-Close** from the web browser's menu.

Step 4 You must now test a peer pod inbound web authentication. Open your web browser, and go to a peer's DMZ web server:

http://192.168.Q.11

(where Q = peer pod number)

Step 5 When the web browser prompts you, enter **aaauser** for the username and **aaapass** for the password. On your PIX Firewall console, you should see the following:

```
109001: Auth start for user '???' from 192.168.Q.10/3463 to
172.16.P.2/80
109001: Auth start for user '???' from 192.168.Q.10/3464 to
172.16.P.2/80
109011: Authen Session Start: user 'aaauser', sid 1
```

```
109005: Authentication succeeded for user 'aaauser' from 172.16.P.2/80
to 192.168.Q.10/3464 on interface outside

302013: Built inbound TCP connection 0 for outside:192.168.Q.10/3464
(192.168.Q.10/3464) to dmz:172.16.P.2/80 (192.168.P.11/80) (aaauser)

304001: 192.168.Q.10 Accessed URL 192.168.P.11:/

302013: Built inbound TCP connection 1 for outside:192.168.Q.10/3465
(192.168.Q.10/3465) to dmz:172.16.P.2/80 (192.168.P.11/80) (aaauser)

304001: 192.168.Q.10 Accessed URL 192.168.P.11:/NETSENSOR.JPG

302014: Teardown TCP connection 1 for outside:192.168.Q.10/3465 to
dmz:172.16.P.2/80 duration 0:01:00 bytes 508 TCP Reset-O (aaauser)

302014: Teardown TCP connection 0 for outside:192.168.Q.10/3464 to
dmz:172.16.P.2/80 duration 0:01:03 bytes 2397 TCP Reset-O (aaauser)
```

(where P = pod number, and Q = peer pod number)

Step 6 After a peer successfully authenticates to your PIX Firewall, display your PIX Firewall authentication statistics:

```
pixP(config)# show uauth
                        Current    Most Seen
Authenticated Users       1          1
Authen In Progress        0          1
user 'aaauser' at 192.168.Q.10, authenticated
   absolute   timeout: 0:05:00
   inactivity timeout: 0:00:00
```

(where Q = peer pod number)

Task 5—Configure and Test Outbound Authentication

Complete the following steps to enable the use of outbound authentication on the PIX Firewall:

Step 1 Configure the PIX Firewall to require authentication for all outbound connections to the super server:

```
pixP(config)# aaa authentication include any outbound 0 0 172.26.26.50
255.255.255.255 MYRADIUS
```

Step 2 Configure the PIX Firewall to require authentication for all outbound connections to the 192.168.P.0 network:

```
pixP(config)# aaa authentication include any outbound 0 0 192.168.P.0
255.255.255.0 MYRADIUS
```

Step 3 Verify your configuration:

```
pixP(config)# show aaa authentication

aaa authentication include tcp/0 outside 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 MYRADIUS

aaa authentication include tcp/0 inside 0.0.0.0 0.0.0.0 172.26.26.50
255.255.255.255 MYRADIUS

aaa authentication include tcp/0 inside 0.0.0.0 0.0.0.0 192.168.P.0
255.255.255.0 MYRADIUS
```

Step 4 Test FTP outbound authentication from your Windows 2000 server:

```
C:\> ftp 172.26.26.50

Connected to 172.26.26.50

220-FTP server : (user ‘aaauser’)

220

User (172.26.26.50:(none)): aaauser@ftpuser

331-Password:

331

Password: aaapass@ftppass

230-220 172.26.26.50 FTP server ready.

331-Password required for ftpuser.

230-User ftpuser logged in.

230

ftp>
```

On your PIX Firewall console, you should see the following:

```
109001: Auth start for user '???' from 10.1.P.11/3142 to
172.26.26.50/21

109011: Authen Session Start: user 'aaauser', sid 13

109005: Authentication succeeded for user 'aaauser' from
10.1.P.11/3142 to 172.26.26.50/21 on interface inside

302013: Built outbound TCP connection 218 for outside:172.26.26.50/21
(172.26.26.50/21) to inside:10.1.P.11/3142 (192.168.P.10/3142)
(aaauser)
```

(where P = pod number)

Step 5 Display authentication statistics on the PIX Firewall:

```
pixP(config)# show uauth

                        Current    Most Seen

Authenticated Users       2           2

Authen In Progress        0           1

user 'aaauser' at insidehost, authenticated

   absolute   timeout: 0:05:00

   inactivity timeout: 0:00:00

user 'aaauser' at 192.168.Q.10, authenticated

   absolute   timeout: 0:05:00

   inactivity timeout: 0:00:00
```

(where Q = peer pod number)

Step 6 Clear the uauth timer:

```
pixP(config)# clear uauth

pixP(config)# show uauth
```

```
                           Current    Most Seen
Authenticated Users           0          2
Authen In Progress            0          2
```

Step 7 If your web browser is open, close it. Choose **File-Exit** from the web browser's menu.

Step 8 Test web outbound authentication. Open your web browser and go to the following URL:

```
http://172.26.26.50
```

Step 9 When you are prompted for a username and password, enter **aaauser** as the username and **aaapass** as the password:

```
User Name: aaauser
Password: aaapass
```

On your PIX Firewall console, you should see the following:

```
109001: Auth start for user '???' from 10.1.P.11/3838 to
172.26.26.50/80

109001: Auth start for user '???' from 10.1.P.11/3840 to
172.26.26.50/80

109011: Authen Session Start: user 'aaauser', sid 4

109005: Authentication succeeded for user 'aaauser' from
10.1.P.11/3840 to 172.26.26.50/80 on interface inside

302013: Built outbound TCP connection 5 for outside:172.26.26.50/80
(172.26.26.50/80) to inside:10.1.P.11/3840 (192.168.P.10/3840)
(aaauser)

304001: aaauser@10.1.P.11 Accessed URL 172.26.26.50:/

302013: Built outbound TCP connection 6 for outside:172.26.26.50/80
(172.26.26.50/80) to inside:10.1.P.11/3841 (192.168.P.10/3841)
(aaauser)

304001: aaauser@10.1.P.11 Accessed URL 172.26.26.50:/NETSENSOR.JPG

302014: Teardown TCP connection 5 for outside:172.26.26.50/80 to
inside:10.1.P.11/3840 duration 0:01:06 bytes 9067 TCP Reset-I
(aaauser)

302014: Teardown TCP connection 6 for outside:172.26.26.50/80 to
inside:10.1.P.11/3841 duration 0:01:03 bytes 13190 TCP Reset-I
(aaauser)
```

(where P = pod number)

Step 10 Display authentication statistics on the PIX Firewall:

```
pixP(config)# show uauth
                            Current    Most Seen
Authenticated Users           1          1
Authen In Progress            0          1
user 'aaauser' at insidehost, authenticated
   absolute   timeout: 0:05:00
   inactivity timeout: 0:00:00
```

Step 11 Close your browser.

Task 6—Configure and Test Console Access Authentication

Complete the following steps to enable console Telnet authentication at the PIX Firewall:

Step 1 Configure the PIX Firewall to require authentication for Telnet console connections:

```
pixP(config)# aaa authentication telnet console MYRADIUS
```

Step 2 Verify your configuration:

```
pixP(config)# show aaa authentication
aaa authentication include tcp/0 outside 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 MYRADIUS
aaa authentication include tcp/0 inside 0.0.0.0 0.0.0.0 172.26.26.50
255.255.255.255 MYRADIUS
aaa authentication include tcp/0 inside 0.0.0.0 0.0.0.0 192.168.P.0
255.255.255.0 MYRADIUS
aaa authentication include tcp/0  0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
MYRADIUS
```

Step 3 Configure the PIX Firewall to allow console Telnet logins:

```
pixP(config)# telnet insidehost 255.255.255.255 inside
```

Step 4 Verify your configuration:

```
pixP(config)# show telnet
insidehost 255.255.255.255 inside
```

Step 5 Clear the uauth timer:

```
pixP(config)# clear uauth
pixP(config)# show uauth
                        Current    Most Seen
Authenticated Users       0           2
Authen In Progress        0           1
```

Step 6 Save your configuration:

```
pixP(config)# write memory
```

(where P = pod number)

Step 7 Telnet to the PIX Firewall console:

```
C:\> telnet 10.0.P.1
Username: aaauser
Password: aaapass
Type help or '?' for a list of available commands.
pixP>
```

On your PIX Firewall console, you should see the following:

```
307002: Permitted Telnet login session from 10.1.P.11
111006: Console Login from aaauser at console
```

(where P = pod number)

Task 7—Configure and Test Virtual Telnet Authentication

Complete the following steps to enable the use of authentication with virtual Telnet on the PIX Firewall:

Step 1 Configure the PIX Firewall to accept authentication to a virtual Telnet service:

```
pixP(config)# virtual telnet 192.168.P.5
```

(where P = pod number)

Step 2 Verify the virtual Telnet configuration:

```
pixP(config)# show virtual telnet
virtual telnet 192.168.P.5
```

(where P = pod number)

Step 3 Clear the uauth timer:

```
pixP(config)# clear uauth
pixP(config)# show uauth
                        Current    Most Seen
Authenticated Users       0          0
Authen In Progress        0          1
```

Step 4 Telnet to the virtual Telnet IP address to authenticate from your Windows 2000 server:

```
C:\> telnet 192.168.P.5
LOGIN Authentication
Username: aaauser
Password: aaapass
Authentication Successful
```

(where P = pod number)

Step 5 Test that you are authenticated. Open your web browser and enter the following in the URL field:

```
http://172.26.26.50
```

You should not be prompted to authenticate.

Step 6 Close your browser.

Step 7 Clear the uauth timer:

```
pixP(config)# clear uauth
pixP(config)# show uauth
                        Current    Most Seen
Authenticated Users       0          1
Authen In Progress        0          1
```

Step 8 Test that you are not authenticated and need to reauthenticate. Open your web browser and enter the following in the URL field:

```
http://172.26.26.50
```

Step 9 When you are prompted, enter **aaauser** for the username and **aaapass** for the password.

Step 10 Close your browser.

Task 8—Change and Test Authentication Timeouts and Prompts

Complete the following steps to change the authentication timeouts and prompts:

Step 1 View the current uauth timeout settings:

```
pixP(config)# show timeout uauth
timeout uauth 0:05:00 absolute
```

Step 2 Set the uauth absolute timeout to 3 hours:

```
pixP(config)# timeout uauth 3 absolute
```

Step 3 Set the uauth inactivity timeout to 30 minutes:

```
pixP(config)# timeout uauth 0:30 inactivity
```

Step 4 Verify the new uauth timeout settings:

```
pixP(config)# show timeout uauth
timeout uauth 3:00:00 absolute uauth 0:30:00 inactivity
```

Step 5 View the current authentication prompt settings:

```
pixP(config)# show auth-prompt
```

Nothing should be displayed.

Step 6 Set the prompt that users get when authenticating:

```
pixP(config)# auth-prompt prompt Please Authenticate
```

Step 7 Set the message that users get when successfully authenticating:

```
pixP(config)# auth-prompt accept You’ve been Authenticated
```

Step 8 Set the message that users get when their authentication is rejected:

```
pixP(config)# auth-prompt reject Authentication Failed, Try Again
```

Step 9 Verify the new prompt settings:

```
pixP(config)# show auth-prompt
auth-prompt prompt Please Authenticate
auth-prompt accept You've been Authenticated
auth-prompt reject Authentication Failed, Try Again
```

Step 10 Clear the uauth timer:

```
pixP(config)# clear uauth
pixP(config)# show uauth
                        Current    Most Seen
Authenticated Users       0           1
Authen In Progress        0           1
```

Step 11 Telnet to the virtual Telnet IP address to test your new authentication prompts. From your Windows 2000 server, enter the following:

```
C:\> telnet 192.168.P.5
LOGIN Authentication
Please Authenticate
Username: aaauser
Password: badpass
Authentication Failed, Try Again
LOGIN Authentication
Please Authenticate
Username: aaauser
Password: aaapass
You've been Authenticated
Authentication Successful
Connection to host lost.
```

(where P = pod number)

Task 9—Configure ACS to Write Downloadable ACLs During Authentication

Complete the following steps to configure ACS to write downloadable ACLs to the PIX Firewall.

Step 1 If your ACS Admin is not already open, launch it by double-clicking the **ACS Admin** icon on your desktop.

Step 2 From the ACS Admin window, click the **Interface Configuration** button. The Interface Configuration window opens.

Step 3 Click **Advanced Options**. The Edit>Advanced Options window opens.

Step 4 Select the check boxes for the following and then click **Submit**. Do not change the other defaults:

- User-Level Downloadable ACLs
- Group-Level Downloadable ACLs

Step 5 Click the **Shared Profile Components** button. The Shared Profile Components window opens.

Step 6 Click **Downloadable IP ACLs**. The Select>Downloadable IP ACLs window opens.

Step 7 Click the **Add** button. The Edit>Downloadable IP ACLs window opens.

Step 8 Enter **RADIUSAUTH** in the Name field.

Step 9 Add the access-list statements below in the ACL Definitions field:

- **permit tcp any host 192.168.P.10**
- **permit tcp any host 192.168.P.11 eq ftp**
- **permit icmp any host 192.168.P.10**

- **deny ip any any**

(where P = pod number)

Step 10 Click **Submit**. The Select>Downloadable ACLs page is displayed with the downloadable ACL highlighted in blue.

Step 11 In ACS Admin, click the **User Setup** button. The User Setup page opens.

Step 12 Click **Find**. The User List is displayed.

Step 13 Click **aaauser.** The Edit>User: aaauser page opens.

Step 14 Scroll down to Downloadable ACLs, and select the check-box for **Assign IP ACL**.

Step 15 Verify that the name of the downloadable ACL you created is displayed in the drop-down box to the right of your checkmark.

Step 16 Click **Submit**.

Task 10—Test Downloadable ACLs with Inbound Authentication

Complete the following steps to test downloadable ACLs with inbound authentication.

Step 1 From your PIX console, clear the uauth cache.

```
pixP(config)# clear uauth

pixP(config)# show uauth

                        Current    Most Seen

Authenticated Users       0           1

Authen In Progress        0           1
```

Step 2 View your existing ACL before testing authorization. Notice that your peer pod is permitted FTP and HTTP access to your bastion host:

```
pixP(config)# show access-list

access-list ACLIN; 10 elements

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 object-
group FTPSERVERS object-group MYSERVICES

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq www(hitcnt=36)

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq ftp(hitcnt=22)

access-list ACLIN line 2 permit tcp any object-group ALLSERVERS eq www

access-list ACLIN line 2 permit tcp any host 192.168.P.11 eq www
(hitcnt=0)

access-list ACLIN line 2 permit tcp any host 192.168.P.10 eq www
(hitcnt=11)

access-list ACLIN line 2 permit tcp any host 192.168.P.6 eq www
(hitcnt=0)

access-list ACLIN line 2 permit tcp any host 192.168.P.7 eq www
(hitcnt=0)

access-list ACLIN line 3 permit icmp any any object-group PING
```

```
access-list ACLIN line 3 permit icmp any any echo (hitcnt=33)
access-list ACLIN line 3 permit icmp any any echo-reply (hitcnt=24)
access-list ACLIN line 3 permit icmp any any unreachable (hitcnt=0)
access-list ACLIN line 4 deny ip any any (hitcnt=5)
access-list ACLDMZ; 3 elements
access-list ACLDMZ line 1 permit icmp any any object-group PING
access-list ACLDMZ line 1 permit icmp any any echo (hitcnt=0)
access-list ACLDMZ line 1 permit icmp any any echo-reply (hitcnt=749)
access-list ACLDMZ line 1 permit icmp any any unreachable (hitcnt=0)
```

(where P = pod number, and Q = peer pod number)

Step 3 Test you peer pod's inbound web authentication and authorization. Open your web browser, and attempt to access your peer's DMZ web server. When prompted to authenticate, enter **aaauser** as the username and **aaapass** as the password. You should be denied access and receive the following error message:

```
Error:  acl authorization denied
http://192.168.Q.11
```

(where Q = peer pod number)

Step 4 View your ACLs again. Notice that the authorization ACL was downloaded to the PIX Firewall.

```
pixP(config)# show access-list
access-list ACLIN; 10 elements
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 object-
group FTPSERVERS object-group MYSERVICES
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq www(hitcnt=42)
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq ftp(hitcnt=22)
access-list ACLIN line 2 permit tcp any object-group ALLSERVERS eq www
access-list ACLIN line 2 permit tcp any host 192.168.P.11 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.10 eq www
(hitcnt=11)
access-list ACLIN line 2 permit tcp any host 192.168.P.6 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.7 eq www
(hitcnt=0)
access-list ACLIN line 3 permit icmp any any object-group PING
access-list ACLIN line 3 permit icmp any any echo (hitcnt=33)
access-list ACLIN line 3 permit icmp any any echo-reply (hitcnt=24)
access-list ACLIN line 3 permit icmp any any unreachable (hitcnt=0)
access-list ACLIN line 4 deny ip any any (hitcnt=5)
access-list ACLDMZ; 3 elements
access-list ACLDMZ line 1 permit icmp any any object-group PING
```

```
access-list ACLDMZ line 1 permit icmp any any echo (hitcnt=0)
access-list ACLDMZ line 1 permit icmp any any echo-reply (hitcnt=749)
access-list ACLDMZ line 1 permit icmp any any unreachable (hitcnt=0)
access-list #ACSACL#-PIX-RADIUSAUTH-3ddb8ab6; 4 elements
access-list #ACSACL#-PIX-RADIUSAUTH-3ddb8ab6 line 1 permit tcp any
host 192.168.P.10 (hitcnt=0)
access-list #ACSACL#-PIX-RADIUSAUTH-3ddb8ab6 line 2 permit tcp any
host 192.168.P.11 eq ftp (hitcnt=0)
access-list #ACSACL#-PIX-RADIUSAUTH-3ddb8ab6 line 3 permit icmp any
host 192.168.P.10 (hitcnt=0)
access-list #ACSACL#-PIX-RADIUSAUTH-3ddb8ab6 line 4 deny ip any any
(hitcnt=6)
```

(where P = pod number, and Q = peer pod number)

Step 5 From the Windows command line on your student PC, initiate an FTP session to your peer pod's bastion host. You should be able to log in as aaauser as follows:

```
C:\>ftp 192.168.Q.11
Connected to 192.168.Q.11.
220-FTP authentication :
220
User (192.168.Q.11:(none)): aaauser@ftpuser
331-Password: aaapass@ftppass
331
Password:
230-230 User logged in
230
ftp> dir
200 PORT command successful.
150 File status OK ; about to open data connection
```

(where Q = peer pod number)

On your PIX Firewall console, you should see output similar to the following:

```
109001: Auth start for user '???' from 192.168.Q.10/2674 to
172.16.P.2/21
109011: Authen Session Start: user 'aaauser', sid 16
109005: Authentication succeeded for user 'aaauser' from 172.16.P.2/21
to 192.168.Q.10/2674 on interface outside
302013: Built inbound TCP connection 199 for outside:192.168.Q.10/2674
(192.168.Q.10/2674) to dmz:172.16.P.2/21 (192.168.P.11/21) (aaauser)
```

(where P = pod number, and Q = peer pod number)

Step 6 View the uauth cache:

```
pixP(config)# show uauth
                        Current    Most Seen
Authenticated Users       1          2
```

```
Authen In Progress        0          1
user 'aaauser' at 192.168.Q.10, authenticated
   access-list #ACSACL#-PIX-RADIUSAUTH-3d9afcb1
   absolute   timeout: 3:00:00
   inactivity timeout: 0:30:00
```

(where Q = peer pod number)

Step 7 View your ACL again. Notice that the hit count on the entry permitting FTP access to your bastion host has incremented:

```
pixP(config)# show access-list
access-list ACLIN; 10 elements
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 object-
group FTPSERVERS object-group MYSERVICES
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq www (hitcnt=45)
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq ftp (hitcnt=26)
access-list ACLIN line 2 permit tcp any object-group ALLSERVERS eq www
access-list ACLIN line 2 permit tcp any host 192.168.P.11 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.10 eq www
(hitcnt=11)
access-list ACLIN line 2 permit tcp any host 192.168.P.6 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.7 eq www
(hitcnt=0)
access-list ACLIN line 3 permit icmp any any object-group PING
access-list ACLIN line 3 permit icmp any any echo (hitcnt=33)
access-list ACLIN line 3 permit icmp any any echo-reply (hitcnt=24)
access-list ACLIN line 3 permit icmp any any unreachable (hitcnt=0)
access-list ACLIN line 4 deny ip any any (hitcnt=5)
access-list ACLDMZ; 3 elements
access-list ACLDMZ line 1 permit icmp any any object-group PING
access-list ACLDMZ line 1 permit icmp any any echo (hitcnt=0)
access-list ACLDMZ line 1 permit icmp any any echo-reply (hitcnt=749)
access-list ACLDMZ line 1 permit icmp any any unreachable (hitcnt=0)
access-list #ACSACL#-PIX-RADIUSAUTH-3ddb8ab6; 4 elements
access-list #ACSACL#-PIX-RADIUSAUTH-3ddb8ab6 line 1 permit tcp any
host 192.168.P.10 (hitcnt=0)
access-list #ACSACL#-PIX-RADIUSAUTH-3ddb8ab6 line 2 permit tcp any
host 192.168.P.11 eq ftp (hitcnt=4)
access-list #ACSACL#-PIX-RADIUSAUTH-3ddb8ab6 line 3 permit icmp any
host 192.168.P.10 (hitcnt=0)
access-list #ACSACL#-PIX-RADIUSAUTH-3ddb8ab6 line 4 deny ip any any
(hitcnt=0)
```

(where P = pod number, and Q = peer pod number)

Step 8 Clear the uauth cache:

```
pixP(config)# clear uauth
```

Step 9 View your ACL again. Notice that the downloadable ACL is gone:

```
pixP(config)# show access-list

access-list ACLIN; 10 elements

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 object-
group FTPSERVERS object-group MYSERVICES

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq www (hitcnt=0)

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq ftp(hitcnt=3)

access-list ACLIN line 2 permit tcp any object-group ALLSERVERS eq www

access-list ACLIN line 2 permit tcp any host 192.168.P.11 eq www
(hitcnt=13)

access-list ACLIN line 2 permit tcp any host 192.168.P.10 eq www
(hitcnt=0)

access-list ACLIN line 2 permit tcp any host 192.168.P.6 eq www
(hitcnt=0)

access-list ACLIN line 2 permit tcp any host 192.168.P.7 eq www
(hitcnt=0)

access-list ACLIN line 3 permit icmp any any object-group PING

access-list ACLIN line 3 permit icmp any any echo (hitcnt=0)

access-list ACLIN line 3 permit icmp any any echo-reply (hitcnt=0)

access-list ACLIN line 3 permit icmp any any unreachable (hitcnt=0)

access-list ACLIN line 4 deny ip any any (hitcnt=0)

access-list ACLDMZ; 3 elements

access-list ACLDMZ line 1 permit icmp any any object-group PING

access-list ACLDMZ line 1 permit icmp any any echo (hitcnt=0)

access-list ACLDMZ line 1 permit icmp any any echo-reply (hitcnt=0)

access-list ACLDMZ line 1 permit icmp any any unreachable (hitcnt=0)
```

(where P = pod number, and Q = peer pod number)

Task 11—Test Downloadable ACLs with Outbound Authentication

Complete the following steps to test downloadable ACLs with outbound authentication:

Step 1 Open your web browser, and attempt to access the super server's web server. When prompted to authenticate, enter the username **aaauser** and the password **aaapass**. You should be denied access because the downloaded ACL RADIUSAUTH is now applied to the inside interface. Authentication is required for outbound access to the super server but RADIUSAUTH does not permit access to the super server:

```
http://172.26.26.50
```

Step 2 View your ACLs. Notice the ACLs that have been downloaded to the PIX Firewall and the hit count on the following entry:

```
access-list #ACSACL#-PIX-Radius_Authorization-3d5c6e76 deny ip any any
  (hitcnt=1)
```

```
pixP(config)# show access-list
access-list ACLIN; 10 elements
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 object-
group FTPSERVERS object-group MYSERVICES
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq www (hitcnt=0)
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq ftp (hitcnt=3)
access-list ACLIN line 2 permit tcp any object-group ALLSERVERS eq www
access-list ACLIN line 2 permit tcp any host 192.168.P.11 eq www
(hitcnt=13)
access-list ACLIN line 2 permit tcp any host 192.168.P.10 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.6 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.7 eq www
(hitcnt=0)
access-list ACLIN line 3 permit icmp any any object-group PING
access-list ACLIN line 3 permit icmp any any echo (hitcnt=0)
access-list ACLIN line 3 permit icmp any any echo-reply (hitcnt=0)
access-list ACLIN line 3 permit icmp any any unreachable (hitcnt=0)
access-list ACLIN line 4 deny ip any any (hitcnt=0)
access-list ACLDMZ; 3 elements
access-list ACLDMZ line 1 permit icmp any any object-group PING
access-list ACLDMZ line 1 permit icmp any any echo (hitcnt=0)
access-list ACLDMZ line 1 permit icmp any any echo-reply (hitcnt=0)
access-list ACLDMZ line 1 permit icmp any any unreachable (hitcnt=0)
access-list #ACSACL#-PIX-RADIUSAUTH-3d936909; 4 elements
access-list #ACSACL#-PIX-RADIUSAUTH-3d936909 line 1 permit tcp any
host 192.168.P.10 (hitcnt=0)
access-list #ACSACL#-PIX-RADIUSAUTH-3d936909 line 2 permit tcp any
host 192.168.P.11 eq ftp (hitcnt=6)
access-list #ACSACL#-PIX-RADIUSAUTH-3d936909 line 3 permit icmp any
host 192.168.P.10 (hitcnt=0)
access-list #ACSACL#-PIX-RADIUSAUTH-3d936909 line 4 deny ip any any
(hitcnt=1)
```

(where P = pod number, and Q = peer pod number)

Step 3 Refresh your browser.

Step 4 View your ACLs again. Notice that the hit count on the following line has incremented:

```
access-list #ACSACL#-PIX-Radius_Authorization-3d5c6e76 deny ip any any
(hitcnt=3)
pixP(config)# show access-list
```

```
access-list ACLIN; 10 elements
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 object-
group FTPSERVERS object-group MYSERVICES
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq www (hitcnt=0)
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq ftp (hitcnt=3)
access-list ACLIN line 2 permit tcp any object-group ALLSERVERS eq www
access-list ACLIN line 2 permit tcp any host 192.168.P.11 eq www
(hitcnt=13)
access-list ACLIN line 2 permit tcp any host 192.168.P.10 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.6 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.7 eq www
(hitcnt=0)
access-list ACLIN line 3 permit icmp any any object-group PING
access-list ACLIN line 3 permit icmp any any echo (hitcnt=0)
access-list ACLIN line 3 permit icmp any any echo-reply (hitcnt=0)
access-list ACLIN line 3 permit icmp any any unreachable (hitcnt=0)
access-list ACLIN line 4 deny ip any any (hitcnt=0)
access-list ACLDMZ; 3 elements
access-list ACLDMZ line 1 permit icmp any any object-group PING
access-list ACLDMZ line 1 permit icmp any any echo (hitcnt=0)
access-list ACLDMZ line 1 permit icmp any any echo-reply (hitcnt=0)
access-list ACLDMZ line 1 permit icmp any any unreachable (hitcnt=0)
access-list #ACSACL#-PIX-RADIUSAUTH-3d936909; 4 elements
access-list #ACSACL#-PIX-RADIUSAUTH-3d936909 line 1 permit tcp any
host 192.168.P.10 (hitcnt=0)
access-list #ACSACL#-PIX-RADIUSAUTH-3d936909 line 2 permit tcp any
host 192.168.P.11 eq ftp (hitcnt=6)
access-list #ACSACL#-PIX-RADIUSAUTH-3d936909 line 3 permit icmp any
host 192.168.P.10 (hitcnt=0)
access-list #ACSACL#-PIX-RADIUSAUTH-3d936909 line 4 deny ip any any
(hitcnt=3)
```

(where P = pod number, and Q = peer pod number)

Step 5 Close your browser.

Step 6 If your ACS Admin is not already open, launch it by double-clicking the **ACS Admin** icon on your desktop.

Step 7 Click the **Shared Profile Components** button. The Shared Profile Components window opens.

Step 8 Click **Downloadable IP ACLs**. The Select>Downloadable IP ACLs window opens.

Step 9 Click **RADIUSAUTH**. The Edit>Downloadable IP ACLs window opens.

Step 10 Insert the access-list statement below at the top of the list of ACL Definitions:

```
permit tcp any host 172.26.26.50 eq www
```

Step 11 Click **Submit**.

Step 12 Return to your telnet session, and clear the uauth cache:

```
pixP(config)# clear uauth
```

Step 13 Open your web browser, and try again to access the super server's web server. When prompted to authenticate, enter the username **aaauser** and the password **aaapass**. You should now be allowed access:

```
http://172.26.26.50
```

Step 14 Close your browser.

Task 12—Configure and Test Accounting

Complete the following steps to enable the use of accounting on the PIX Firewall:

Step 1 Configure the PIX Firewall to perform accounting for all outbound traffic:

```
pixP(config)# aaa accounting include any inbound 0 0 0 0 MYRADIUS
```

Step 2 Verify your configuration:

```
pixP(config)# show aaa accounting
aaa accounting include tcp/0 outside 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
MYRADIUS
```

Step 3 Clear the uauth timer:

```
pixP(config)# clear uauth
pixP(config)# show uauth
                        Current    Most Seen
Authenticated Users       0           1
Authen In Progress        0           1
```

Step 4 Test FTP inbound accounting by accessing your peer pod's bastion host via FTP:

```
C:\> ftp 192.168.Q.11
Connected to 192.168.Q.11
220-Please Authenticate :
220
User (192.168.Q.11:(none)): aaauser@ftpuser
331-Password:
331
Password: aaapass@ftppass
230-220 192.168.Q.11 FTP server ready.
331-Password required for ftpuser
230-User ftpuser logged in.
230
ftp>
```

(where Q = peer pod number)

Step 5 View the accounting records. On Cisco Secure ACS, click **Reports and Activity** to open the Reports and Activity interface.

Step 6 Click the **RADIUS Accounting** link.

Step 7 Click the **RADIUS Accounting active.csv** link to open the accounting records.

Step 8 Disable AAA by entering the following command:

```
pixP(config)# clear aaa
```

Step 9 Remove the aaa-server commands from the configuration:

```
pixP(config)# clear aaa-server
```

Step 10 Turn off the logging:

```
pixP(config)# no logging console debug
```

12

Failover

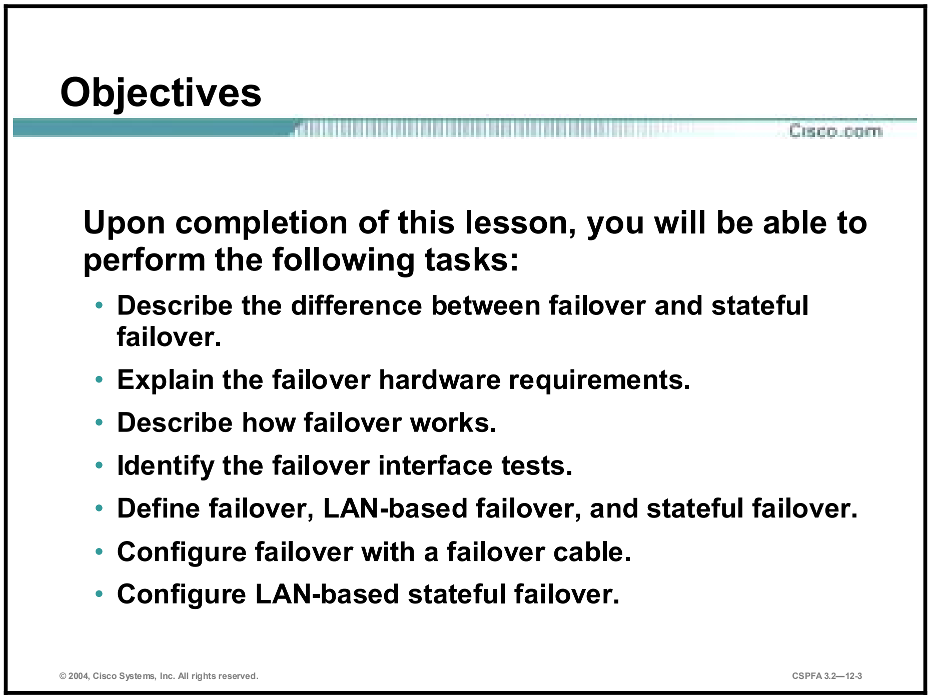
Overview

This lesson includes the following topics:

- Objectives
- Understanding failover
- Serial cable-based failover configuration
- LAN-based failover configuration
- Summary
- Lab exercise

Objectives

This topic lists the lesson’s objectives.



Objectives

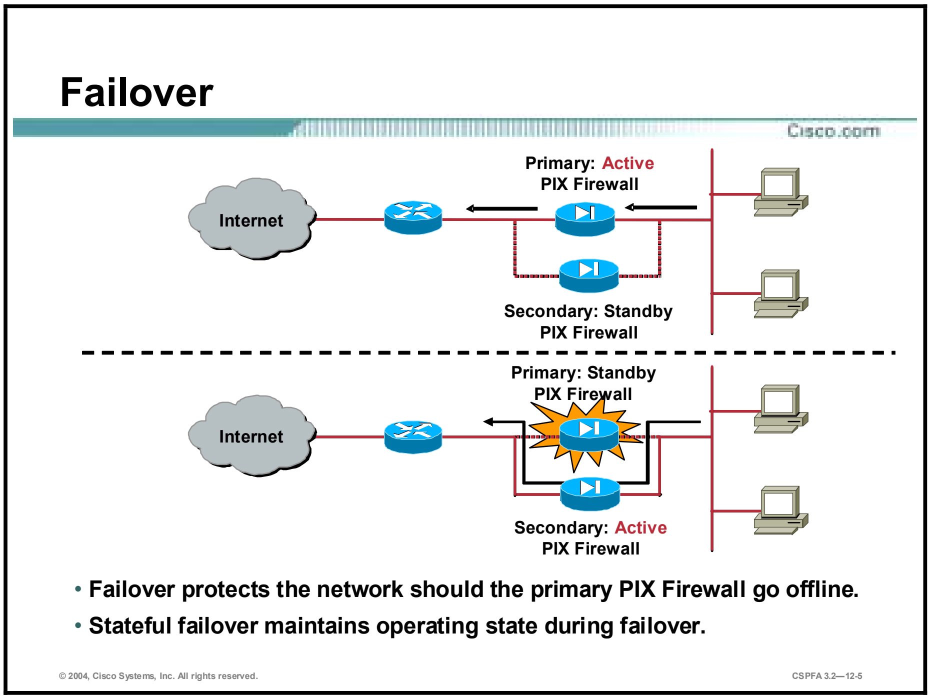
Upon completion of this lesson, you will be able to perform the following tasks:

- **Describe the difference between failover and stateful failover.**
- **Explain the failover hardware requirements.**
- **Describe how failover works.**
- **Identify the failover interface tests.**
- **Define failover, LAN-based failover, and stateful failover.**
- **Configure failover with a failover cable.**
- **Configure LAN-based stateful failover.**

CSPFA 3.2—12-3

Understanding Failover

This topic discusses what failover is and how it works.



Failover

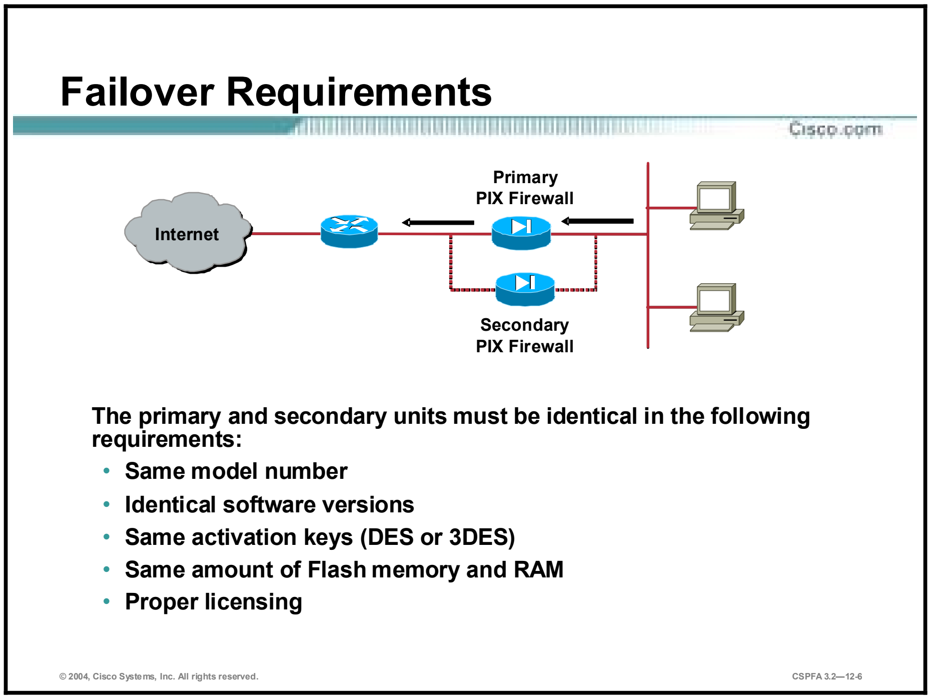
- Failover protects the network should the primary PIX Firewall go offline.
- Stateful failover maintains operating state during failover.

CSPFA 3.2—12-5

The failover function for the Cisco Secure PIX Firewall provides a safeguard in case a PIX Firewall fails. Specifically, when one PIX Firewall fails, another immediately takes its place. In the failover process, there are two PIX Firewalls: the primary PIX Firewall and the secondary PIX Firewall. The primary PIX Firewall functions as the active PIX Firewall, performing normal network functions. The secondary PIX Firewall functions as the standby PIX Firewall, ready to take control should the active PIX Firewall fail to perform. When the primary PIX Firewall fails, the secondary PIX Firewall becomes active while the primary PIX Firewall goes on standby. This entire process is called failover.

A failover occurs when one of the following situations takes place:

- A power-off or a power-down condition occurs on the active PIX Firewall.
- The active PIX Firewall is rebooted.
- A link goes down on the active PIX Firewall for more than 30 seconds.
- The command **failover active** is typed on the standby PIX Firewall, which forces control back to that unit.
- Block memory exhaustion occurs for 15 consecutive seconds or more on the active PIX Firewall.



Failover Requirements

The primary and secondary units must be identical in the following requirements:

- **Same model number**
- **Identical software versions**
- **Same activation keys (DES or 3DES)**
- **Same amount of Flash memory and RAM**
- **Proper licensing**

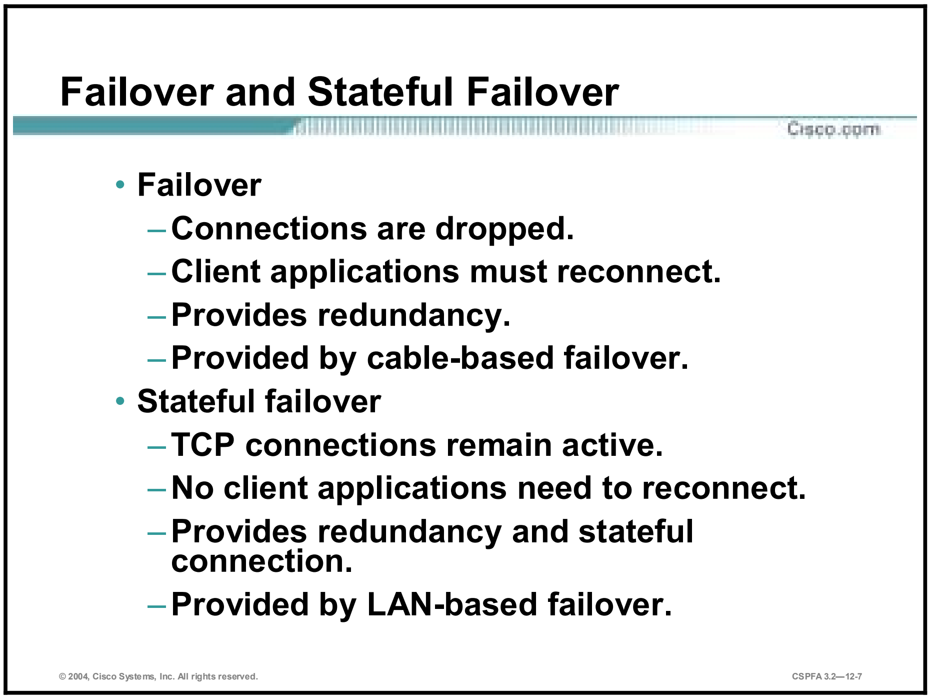
CSPFA 3.2—12-6

The Cisco PIX Firewall 515/515E, 525, and 535 can be used for failover. In order for failover to work, a pair of firewalls must be identical in the following requirements:

- Platform type (a PIX Firewall 515E cannot be used with a PIX Firewall 515)
- Software version
- Activation key (DES or 3DES)
- Flash memory
- Amount of RAM

There is one more factor, licensing. One of the failover units must have an unrestricted (UR) license, while the other can have a failover (FO) or UR license. A restricted license cannot be used for failover, and two units with FO licenses cannot be used in a single failover pair.

Note Neither the PIX Firewall 501 nor the PIX Firewall 506E can be used for failover.



Failover and Stateful Failover

Cisco.com

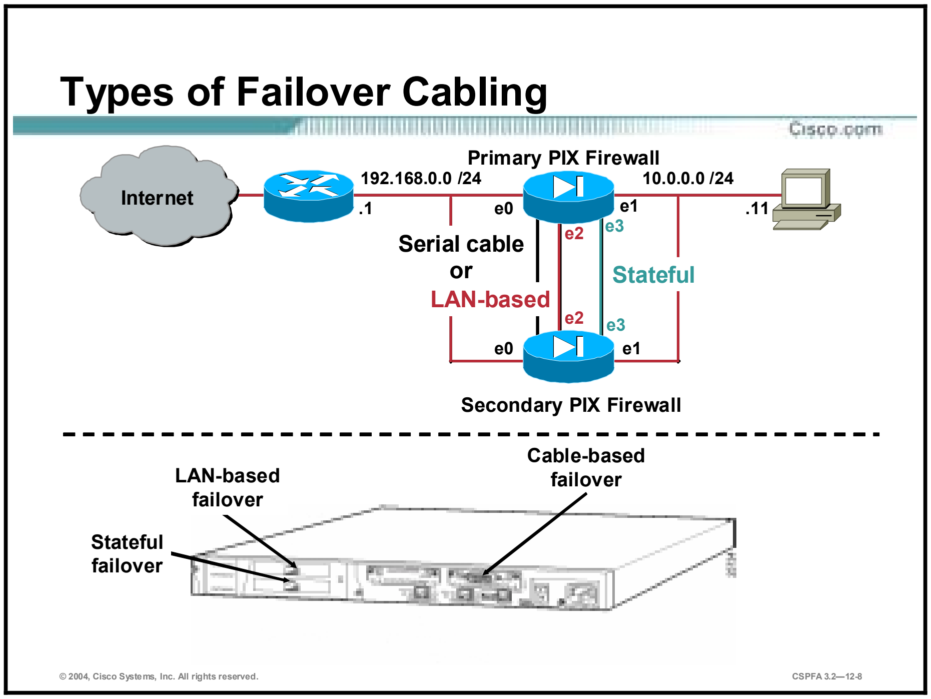
- Failover
  - Connections are dropped.
  - Client applications must reconnect.
  - Provides redundancy.
  - Provided by cable-based failover.
- Stateful failover
  - TCP connections remain active.
  - No client applications need to reconnect.
  - Provides redundancy and stateful connection.
  - Provided by LAN-based failover.

CSPFA 3.2—12-7

As stated earlier in the lesson, failover enables the standby PIX Firewall to take over the duties of the active PIX Firewall when the active PIX Firewall fails. There are two types of failover:

- Failover—When the active PIX Firewall fails and the standby PIX Firewall becomes active, all connections are lost, and client applications must perform a new connection to restart communication through the PIX Firewall. The disconnection happens because the active PIX Firewall does not pass the stateful connection information to the standby PIX Firewall. Failover messages are exchanged over a serial failover cable or a LAN-based failover connection.
- Stateful failover—The stateful failover feature passes per-connection stateful information to the standby unit. After a failover occurs, the same connection information must be available at the new active unit. End-user applications are not required to do a reconnect to keep the same communication session. The state information passed to the standby unit includes the global pool addresses and status, connection and translation information and status, the negotiated H.323 UDP ports, the port allocation bitmap for Port Address Translation (PAT), and other details necessary to let the standby unit take over processing if the primary unit fails.

Depending on the failure, the PIX Firewall switchover takes from 15 to 45 seconds. Applications not handled by stateful failover will then require time to reconnect before the active unit becomes fully functional.



Types of Failover Cabling

© 2004, Cisco Systems, Inc. All rights reserved.

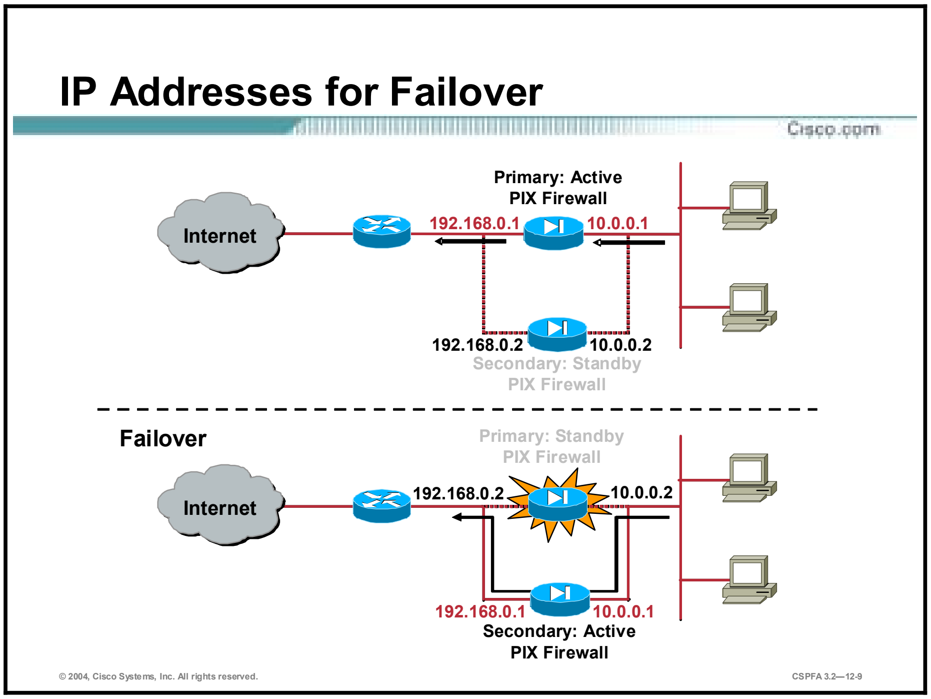
CSPFA 3.2—12-8

The communications identifies the unit as primary or secondary, identifies the power status of the other unit, and serves as a link for various failover communications between the two units. The failover feature in the PIX Firewall monitors failover communication, the power status of the other unit, and hello packets received at each interface. If two consecutive hello packets are not received within a time determined by the failover feature, failover starts testing the interfaces to determine which unit has failed and transfers active control to the secondary unit.

There are three types of failover cabling:

- Serial failover cable—The serial failover cable is a modified RS-232 serial link cable that transfers data at 115 Kbps.
- LAN-based failover cable—PIX Firewall Software Version 6.2 introduced support for LAN-based failover, so a special serial failover cable is no longer required to connect the primary and secondary PIX Firewalls. LAN-based failover overcomes the distance limitations imposed by the six-foot length of the serial failover cable. With LAN-based failover, failover messages are transmitted over Ethernet connections. LAN-based failover provides message encryption and authentication using a manual pre-shared key for added security. LAN-based failover requires an Ethernet connection to be used exclusively for passing failover communications between two PIX Firewall units.
- Stateful cable—The stateful failover cable passes per-connection stateful information to the standby unit. Stateful failover requires an Ethernet interface with a minimum speed of 100 Mbps full duplex to be used exclusively for passing state information between the two PIX Firewall units. The stateful failover interface can be connected to either a 100BASE-TX or 1000BASE-TX full duplex on a dedicated switch or dedicated VLAN of a switch.

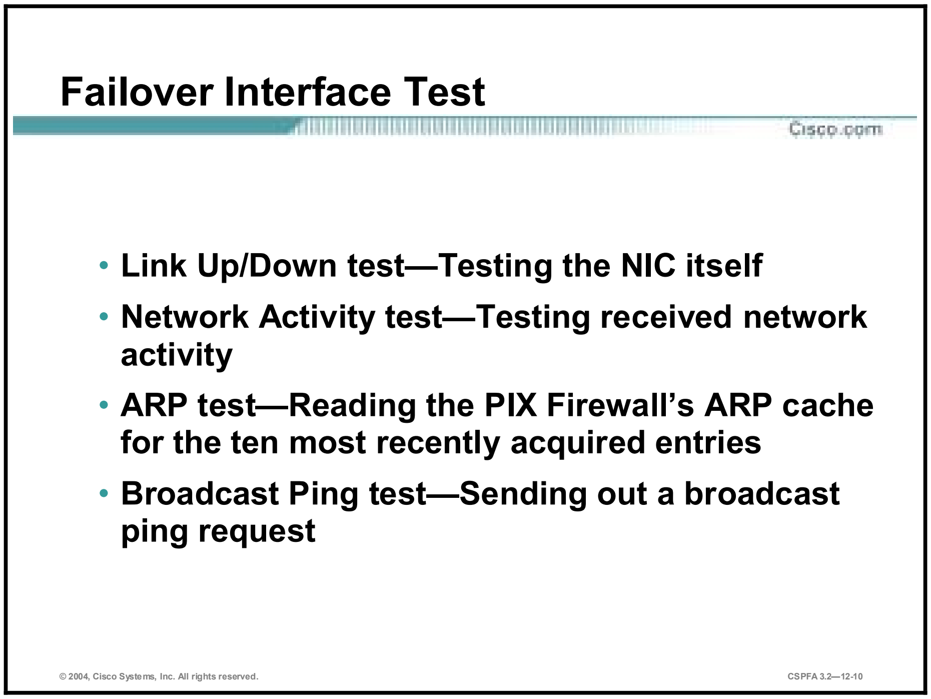
Data is passed over the dedicated interface using IP Protocol 8. No hosts or routers should be on this interface.



IP Addresses for Failover

CSPFA 3.2—12-9

When failover occurs, each unit changes state. The active unit changes to standby state, while the standby unit changes to active state. The unit that becomes active assumes the active IP address along with the MAC address of the unit that was previously active (the primary unit) and begins passing traffic. The unit that is now in standby state assumes the failover IP address and the MAC addresses of the unit that was previously the standby unit. Because network devices see no change in the MAC-to-IP address pairing, no Address Resolution Protocol (ARP) entries change or time out anywhere on the network. In the example in the figure, there are two scenarios, before failover and after failover. Before failover, the primary, or active, PIX Firewall (at the top) has IP addresses of 192.168.0.1 and 10.0.0.1. The secondary, or standby, PIX Firewall (at the bottom) has IP addresses of 192.168.0.2 and 10.0.0.2. After failover, the status and IP and MAC addresses are reversed. The bottom, or secondary, PIX Firewall has assumed the role of the active unit. Its new IP addresses are 192.168.0.1 and 10.0.0.1. The top, primary and new standby, PIX Firewall assumes IP addresses 192.168.0.2 and 10.0.0.2 as well as the standby MAC addresses. The MAC addresses are reversed during failover.



Failover Interface Test

Cisco.com

- Link Up/Down test—Testing the NIC itself
- Network Activity test—Testing received network activity
- ARP test—Reading the PIX Firewall's ARP cache for the ten most recently acquired entries
- Broadcast Ping test—Sending out a broadcast ping request

CSPFA 3.2—12-10

Both the primary and secondary PIX Firewalls send special failover hello packets to each other over all network interfaces and the failover cable every fifteen seconds to make sure that everything is working. When a failure occurs in the active PIX Firewall, and it is not because of a loss of power in the standby PIX Firewall, failover begins a series of tests to determine which PIX Firewall has failed. The purpose of these tests is to generate network traffic to determine which, if either, PIX Firewall has failed.

At the start of each test, each PIX Firewall clears its received packet count for its interfaces. At the conclusion of each test, each PIX Firewall looks to see if it has received any traffic. If it has, the interface is considered operational. If one PIX Firewall receives traffic for a test and the other PIX Firewall does not, the PIX Firewall that did not receive traffic is considered failed. If neither PIX Firewall has received traffic, the tests then continue.

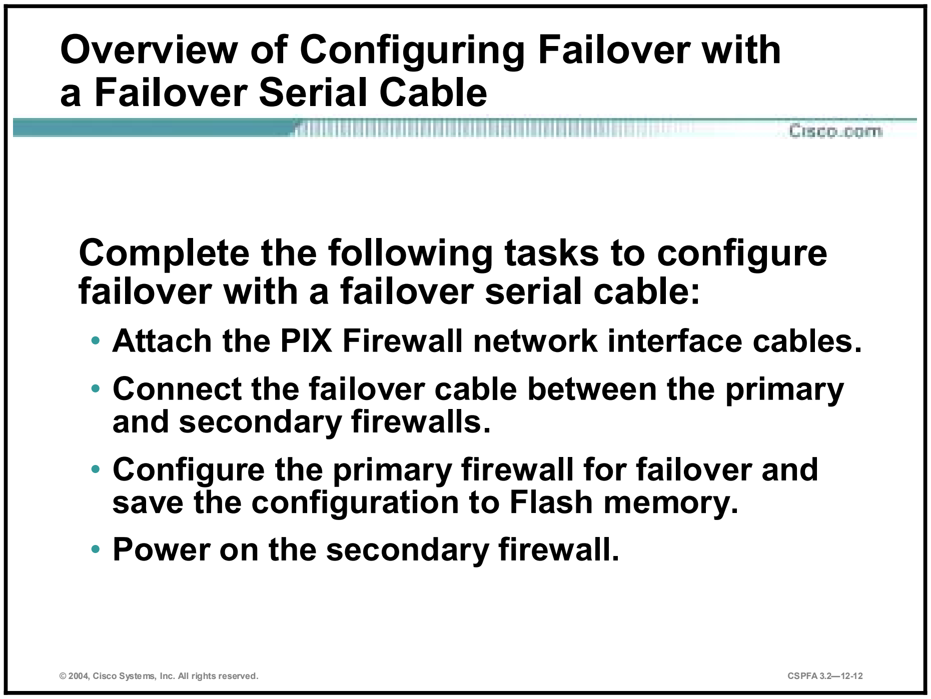
The following are the four different tests used to test for failover:

- LinkUp/Down—This is a test of the NIC itself. If an interface card is not plugged into an operational network, it is considered failed. For example, the hub or switch has failed, has a failed port, or a cable is unplugged. If this test does not find anything, the network activity test begins.
- Network Activity—This is a received network activity test. The PIX Firewall counts all received packets for up to five seconds. If any packets are received at any time during this interval, the interface is considered operational and testing stops. If no traffic is received, the ARP test begins.
- ARP—The ARP test consists of reading the PIX Firewall's ARP cache for the ten most recently acquired entries. The PIX Firewall sends ARP requests one at a time to these machines, attempting to stimulate network traffic. After each request, the PIX Firewall counts all received traffic for up to five seconds. If traffic is received, the interface is considered operational. If no traffic is received, an ARP request is sent to the next machine. If at the end of the list no traffic has been received, the ping test begins.

- Broadcast Ping—The ping test consists of sending out a broadcast ping request. The PIX Firewall then counts all received packets for up to five seconds. If any packets are received at any time during this interval, the interface is considered operational and testing stops. If no traffic is received, the testing starts over again with the ARP test.

Serial Cable-Based Failover Configuration

This topic explains how to configure serial cable-based failover.



Overview of Configuring Failover with a Failover Serial Cable

Cisco.com

Complete the following tasks to configure failover with a failover serial cable:

- **Attach the PIX Firewall network interface cables.**
- **Connect the failover cable between the primary and secondary firewalls.**
- **Configure the primary firewall for failover and save the configuration to Flash memory.**
- **Power on the secondary firewall.**

CSPFA 3.2—12-12

Complete the steps below to configure failover with a serial failover cable. Important details for these steps follow. Before starting this procedure, if you have already powered it on, power off the standby firewall and leave it off until instructed to power it on.

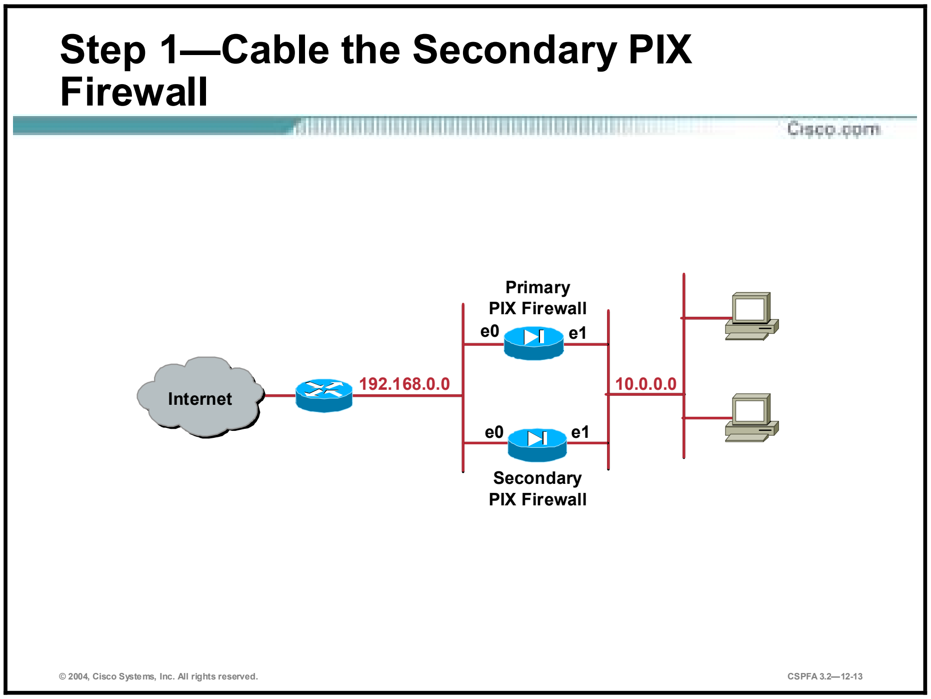
Step 1 Attach a network cable for each network interface you plan to use.

Step 2 Connect the failover cable between the primary PIX Firewall and the secondary PIX Firewall.

Step 3 Configure the following failover parameters on the primary PIX Firewall. When you have finished this configuration, save it to the primary firewall's Flash memory.

1. Failover
2. Failover IP addresses
3. Stateful failover interface (optional, for stateful failover)
4. Failover poll time (optional).

Step 4 Power on the secondary firewall.



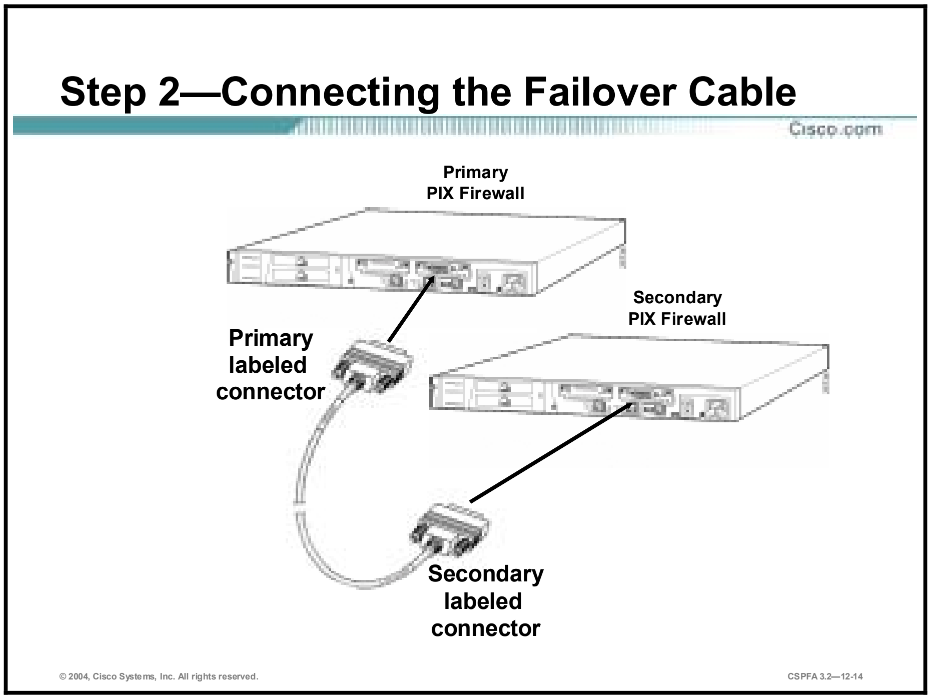
Step 1—Cable the Secondary PIX Firewall

Cisco.com

CSPFA 3.2—12-13

The steps in detail are as follows:

Step 1 After verifying that the secondary PIX Firewall is powered off, attach a network cable for each network interface you plan to use. The IP addresses on the secondary unit are different from the primary unit, but should be in the same subnet for each interface. If you plan to do stateful failover, one of the interfaces must be dedicated to this function.



Step 2—Connecting the Failover Cable

Cisco.com

Primary
PIX Firewall

Secondary
PIX Firewall

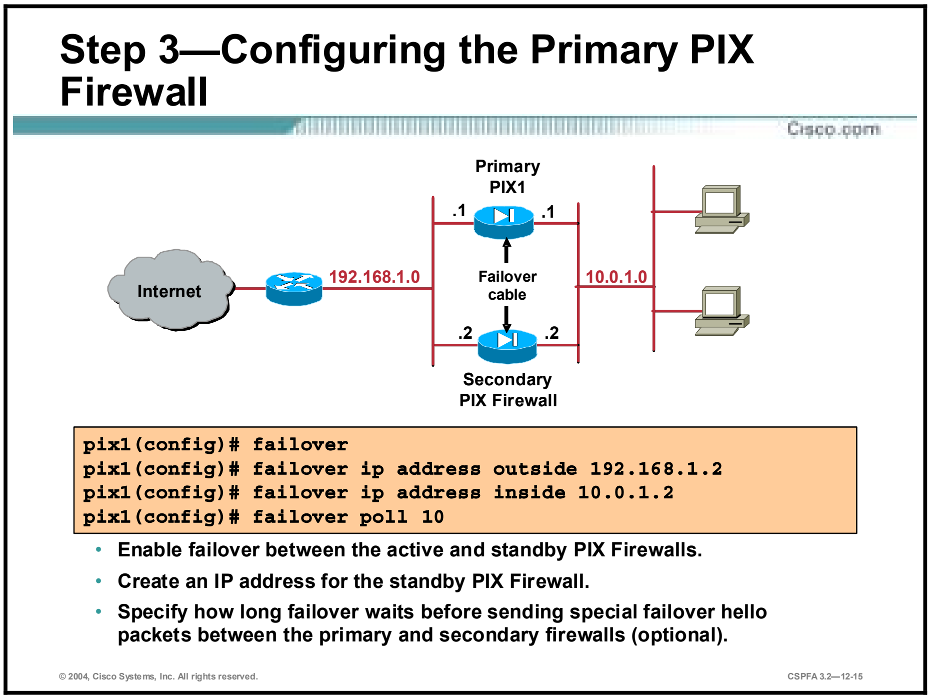
Primary
labeled
connector

Secondary
labeled
connector

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—12-14

Step 2 Connect the failover cable to the primary PIX Firewall, ensuring that the end of the cable marked Primary attaches to the primary firewall and that the end marked Secondary connects to the secondary firewall. Do not power on the secondary firewall.



Step 3—Configuring the Primary PIX Firewall

Cisco.com

```
pix1(config)# failover
pix1(config)# failover ip address outside 192.168.1.2
pix1(config)# failover ip address inside 10.0.1.2
pix1(config)# failover poll 10
```

- Enable failover between the active and standby PIX Firewalls.
- Create an IP address for the standby PIX Firewall.
- Specify how long failover waits before sending special failover hello packets between the primary and secondary firewalls (optional).

CSPFA 3.2—12-15

Step 3 Configure the primary PIX Firewall as follows:

1. Use the **failover** command to enable failover on the primary firewall.

2. Use the **failover ip address** command to enter a failover IP address for each interface. These IP addresses for the standby firewall are different from the active firewall's addresses, but they should be in the same subnet for each interface.

3. If you are configuring stateful failover, use the **failover link** command to specify the name of a dedicated stateful failover interface.

4. If you want failover to occur faster, use the **failover poll** command to set a time shorter than fifteen seconds for the firewalls to exchange hello packets. Set the poll time to a lower value for stateful failover. With a faster poll time, the PIX Firewall can detect failure and trigger failover faster. However, faster detection may cause unnecessary switchovers when the network is temporarily congested or a network card starts slowly. The default failover poll time is fifteen seconds. The minimum value is three seconds, and the maximum is fifteen seconds.

The syntax for the failover configuration commands is as follows:

failover [active]

failover ip address ***if_name ip_address***

failover link [*stateful_if_name*]

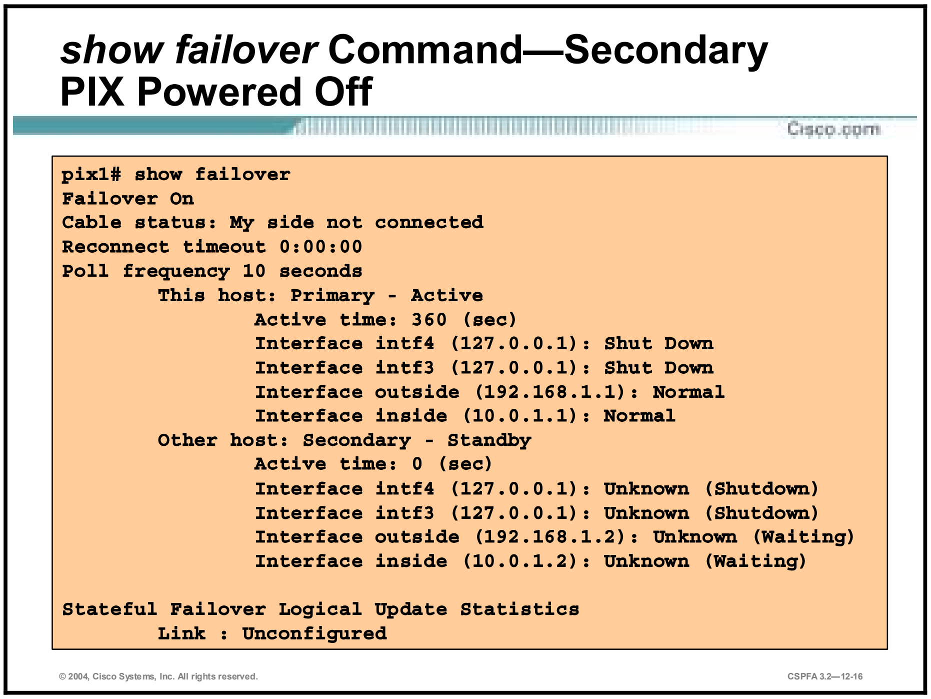
failover poll ***seconds***

active	Makes a PIX Firewall the active PIX Firewall. Use this command when you need to force control of the connection back to the unit you are accessing, such as when you want to switch control back from a PIX Firewall after you have fixed a problem and want to restore service to the primary PIX Firewall.

if_name	Specifies the interface name for the failover IP address.
ip_address	Specifies the IP address used by the standby firewall to communicate with the active firewall. Use this IP address with the **ping** command to check the status of the standby firewall. This address must be on the same network as the system IP address. For example, if the system IP address is 192.159.1.3, you could set the failover IP address to 192.159.1.4.
link	Specifies the interface where a fast LAN link is available for stateful failover.
stateful_if_name	Specifies a dedicated fast LAN link for stateful failover.
poll *seconds*	Specifies how long failover waits before sending special failover "hello" packets between the primary and standby firewalls over all network interfaces and the failover cable. The default is 15 seconds. The minimum value is 3 seconds and the maximum is 15 seconds. Set the time to a lower value for stateful failover. With a faster poll time, the PIX Firewall can detect failure and trigger failover faster. However, faster detection may cause unnecessary switchovers when the network is temporarily congested or a network card starts slowly.

5. Set the clock. The PIX Firewall clock is stored in the CMOS. Specify the **clock set** command on the active PIX Firewall to synchronize the time on both PIX Firewalls.

6. Use the **write memory** command to save the configuration to Flash memory. Configure only the primary firewall.



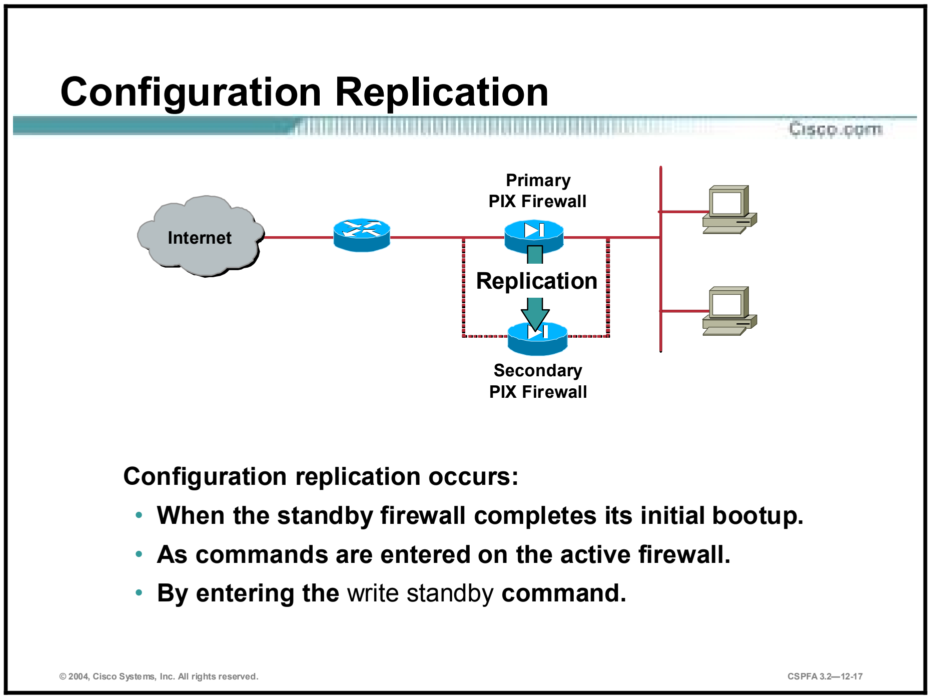
show failover Command—Secondary PIX Powered Off

```
pix1# show failover
Failover On
Cable status: My side not connected
Reconnect timeout 0:00:00
Poll frequency 10 seconds
        This host: Primary - Active
                Active time: 360 (sec)
                Interface intf4 (127.0.0.1): Shut Down
                Interface intf3 (127.0.0.1): Shut Down
                Interface outside (192.168.1.1): Normal
                Interface inside (10.0.1.1): Normal
        Other host: Secondary - Standby
                Active time: 0 (sec)
                Interface intf4 (127.0.0.1): Unknown (Shutdown)
                Interface intf3 (127.0.0.1): Unknown (Shutdown)
                Interface outside (192.168.1.2): Unknown (Waiting)
                Interface inside (10.0.1.2): Unknown (Waiting)

Stateful Failover Logical Update Statistics
        Link : Unconfigured
```

CSPFA 3.2—12-16

The **failover IP address** command specifies the secondary unit's interface addresses. The IP addresses on the secondary unit are different from the primary unit's addresses but should be on the same subnet. In the example in the figure, notice the IP addresses of the primary and secondary units. The outside interface address on the primary unit is 192.168.1.1 while the address of the outside interface of the secondary unit is 192.168.1.2. Use the **show failover** command to view the following:

- Status of failover
- Cable status
- Primary and secondary unit status
- Primary and secondary interface status
- Stateful failover link statistics



Configuration Replication

Configuration replication occurs:

- When the standby firewall completes its initial bootup.
- As commands are entered on the active firewall.
- By entering the write standby command.

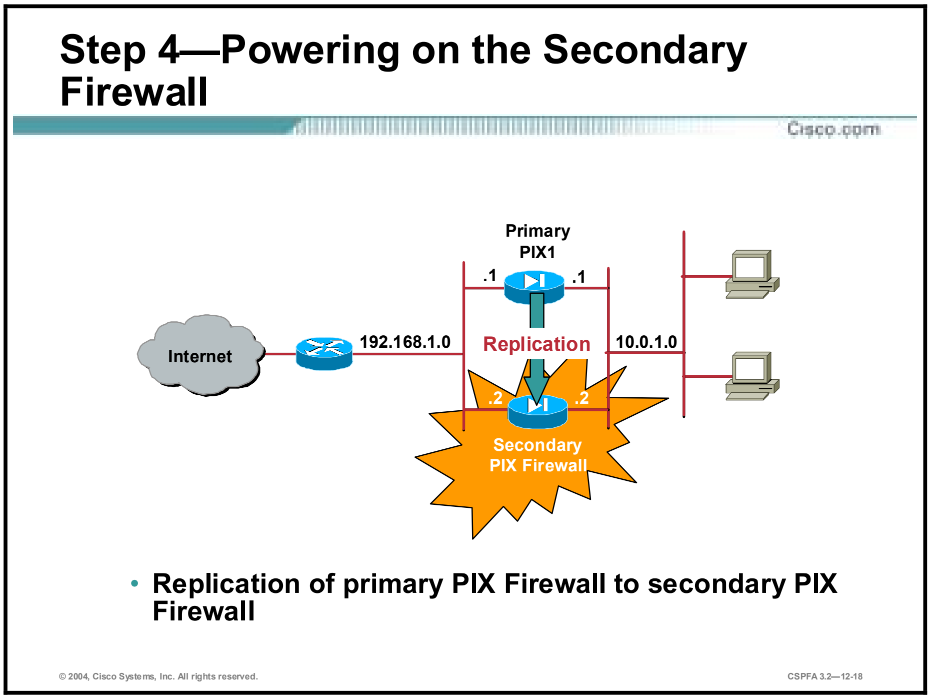
CSPFA 3.2—12-17

Configuration replication is the configuration of the primary PIX Firewall being replicated to the secondary PIX Firewall. To perform configuration replication, both the primary and secondary PIX Firewalls must be exactly the same and run the same software release. The configuration can be replicated from the active PIX Firewall to the standby PIX Firewall in three ways:

- When the standby PIX Firewall completes its initial bootup, the active PIX Firewall replicates its entire configuration to the standby PIX Firewall.
- As commands are entered on the active PIX Firewall, they are sent across the failover cable to the standby PIX Firewall.
- Entering the **write standby** command on the active PIX Firewall forces the entire configuration in memory to be sent to the standby PIX Firewall.

Configuration replication occurs only from memory to memory. Because this is not a permanent place to store configurations, you must use the **write memory** command to write the configuration into Flash memory. If a failover occurs during the replication, the new active PIX Firewall will have only a partial configuration. The newly active PIX Firewall will then reboot itself to recover the configuration from the Flash memory or resynchronize with the new standby PIX Firewall.

When replication starts, the PIX Firewall console displays the message “sync started,” and when replication is complete, displays the message “sync completed.” During replication, information cannot be entered on the PIX Firewall console. Replication can take a long time to complete with standard failover for a large configuration because configuration replication occurs over the failover cable.

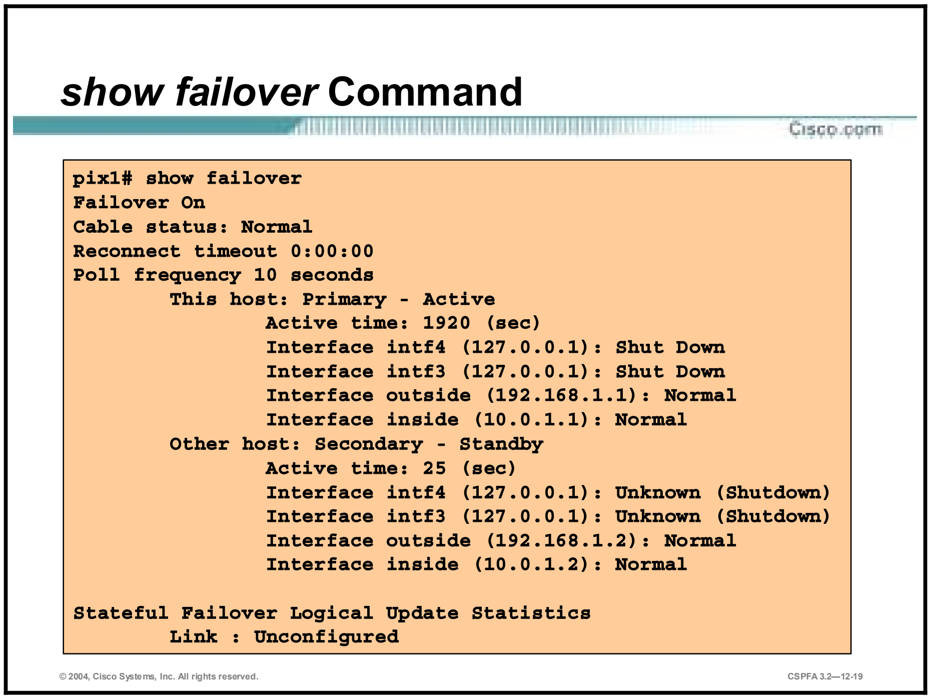


Step 4—Powering on the Secondary Firewall

- Replication of primary PIX Firewall to secondary PIX Firewall

CSPFA 3.2—12-18

Step 4 Power on the secondary firewall. When the secondary firewall completes its initial bootup, the primary firewall recognizes it and starts synchronizing and replicating the configurations. As the configurations synchronize, the messages “sync started” and “sync completed” appear.

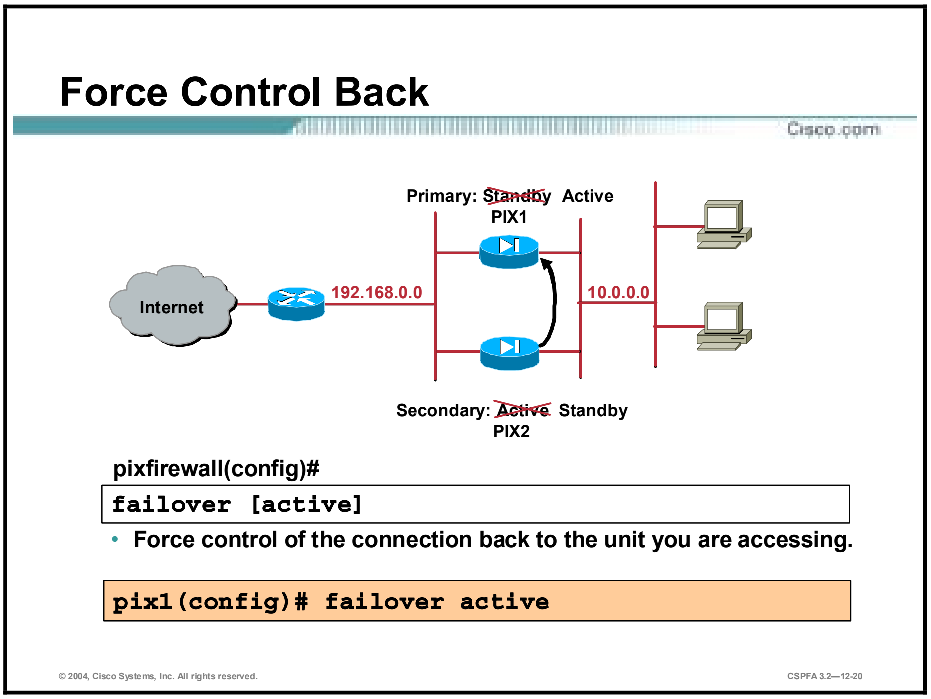


show failover Command

```
pix1# show failover
Failover On
Cable status: Normal
Reconnect timeout 0:00:00
Poll frequency 10 seconds
        This host: Primary - Active
                Active time: 1920 (sec)
                Interface intf4 (127.0.0.1): Shut Down
                Interface intf3 (127.0.0.1): Shut Down
                Interface outside (192.168.1.1): Normal
                Interface inside (10.0.1.1): Normal
        Other host: Secondary - Standby
                Active time: 25 (sec)
                Interface intf4 (127.0.0.1): Unknown (Shutdown)
                Interface intf3 (127.0.0.1): Unknown (Shutdown)
                Interface outside (192.168.1.2): Normal
                Interface inside (10.0.1.2): Normal

Stateful Failover Logical Update Statistics
        Link : Unconfigured
```

CSPFA 3.2—12-19

Use the **show failover** command to check the status of the primary and secondary units after secondary unit powerup and replication is completed. Notice that the cable status and the configured secondary interfaces are changed to normal.



Force Control Back

pixfirewall(config)#

```
failover [active]
```

- Force control of the connection back to the unit you are accessing.

```
pix1(config)# failover active
```

CSPFA 3.2—12-20

Use the **failover active** command when you need to force control of the connection back to the unit you are accessing, such as when you want to switch control back from a unit after you have fixed a problem and want to restore service to the primary unit. Either enter the **no failover active** command on the secondary unit to switch service to the primary or the **failover active** command on the primary unit.

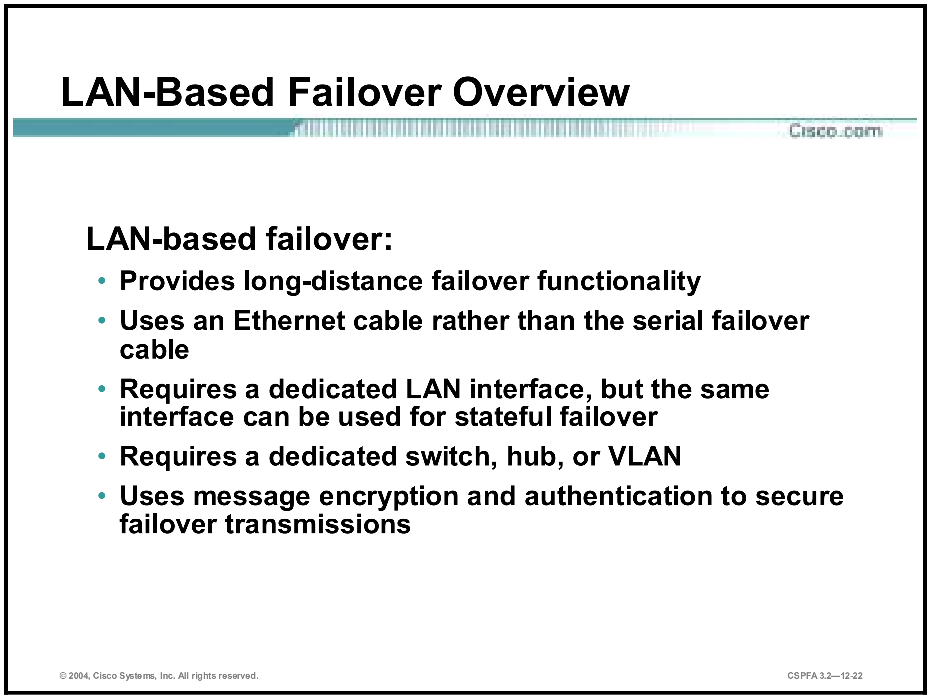
The **failover reset** command forces both firewalls back to an unfailed state. This command can be entered from either firewall, but it is best to always enter commands at the active firewall. Entering the **failover reset** command at the active firewall unfails the standby firewall.

The syntax for the **failover reset** command is as follows:

```
failover reset
```

LAN-Based Failover Configuration

This topic explains how to configure LAN-based failover.



LAN-Based Failover Overview

Cisco.com

LAN-based failover:

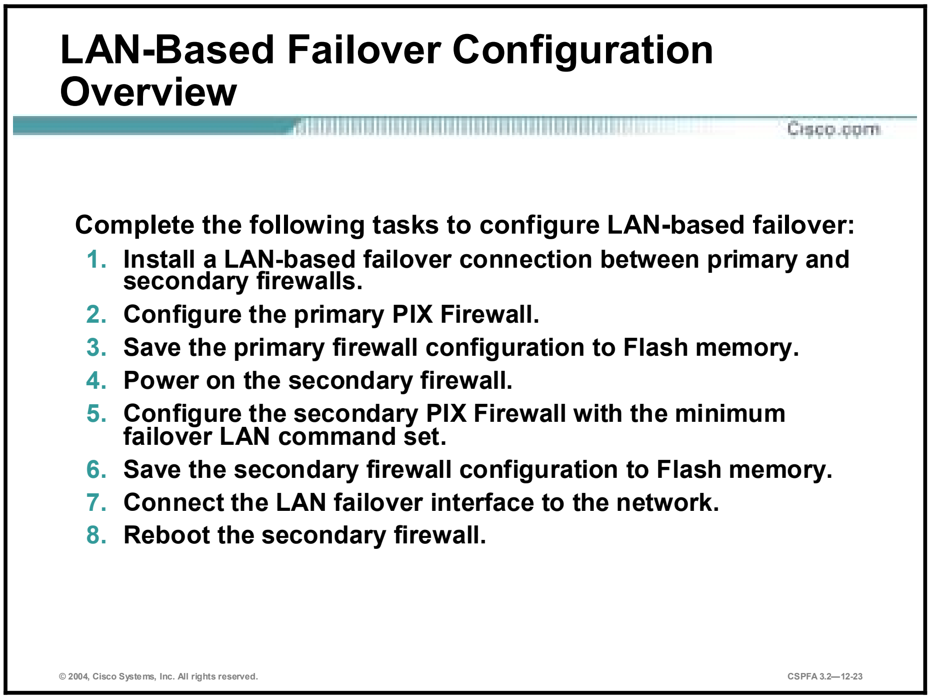
- Provides long-distance failover functionality
- Uses an Ethernet cable rather than the serial failover cable
- Requires a dedicated LAN interface, but the same interface can be used for stateful failover
- Requires a dedicated switch, hub, or VLAN
- Uses message encryption and authentication to secure failover transmissions

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—12-22

LAN-based failover overcomes the distance limitations imposed by the six-foot length of the failover cable. With LAN-based failover, an Ethernet cable can be used to replicate configuration from the primary PIX Firewall to the secondary PIX Firewall; the special failover cable is not required. Instead, LAN-based failover requires a dedicated LAN interface and a dedicated switch, hub, or VLAN. You should not use a crossover Ethernet cable to connect the two PIX Firewalls.

The same LAN interface used for LAN-based failover can also be used for stateful failover. However, the interface needs enough capacity to handle both the LAN-based failover and stateful failover traffic. If the interface does not have the necessary capacity, use two separate, dedicated interfaces.

LAN-based failover allows traffic to be transmitted over Ethernet connections that are relatively less secure than the special failover cable; therefore, to secure failover transmissions, LAN-based failover provides message encryption and authentication using a manual pre-shared key.



LAN-Based Failover Configuration Overview

Cisco.com

Complete the following tasks to configure LAN-based failover:

1. Install a LAN-based failover connection between primary and secondary firewalls.
2. Configure the primary PIX Firewall.
3. Save the primary firewall configuration to Flash memory.
4. Power on the secondary firewall.
5. Configure the secondary PIX Firewall with the minimum failover LAN command set.
6. Save the secondary firewall configuration to Flash memory.
7. Connect the LAN failover interface to the network.
8. Reboot the secondary firewall.

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—12-23

Complete the following steps to configure LAN-based failover. Detail on each step follows.

Step 1 Install a LAN-based failover connection between firewalls. Verify that any switch port that connects to a PIX Firewall interface is configured to support LAN-based failover. Disconnect the secondary PIX Firewall.

Step 2 Configure the primary PIX Firewall for failover.

Step 3 Save the primary firewall's configuration to Flash memory.

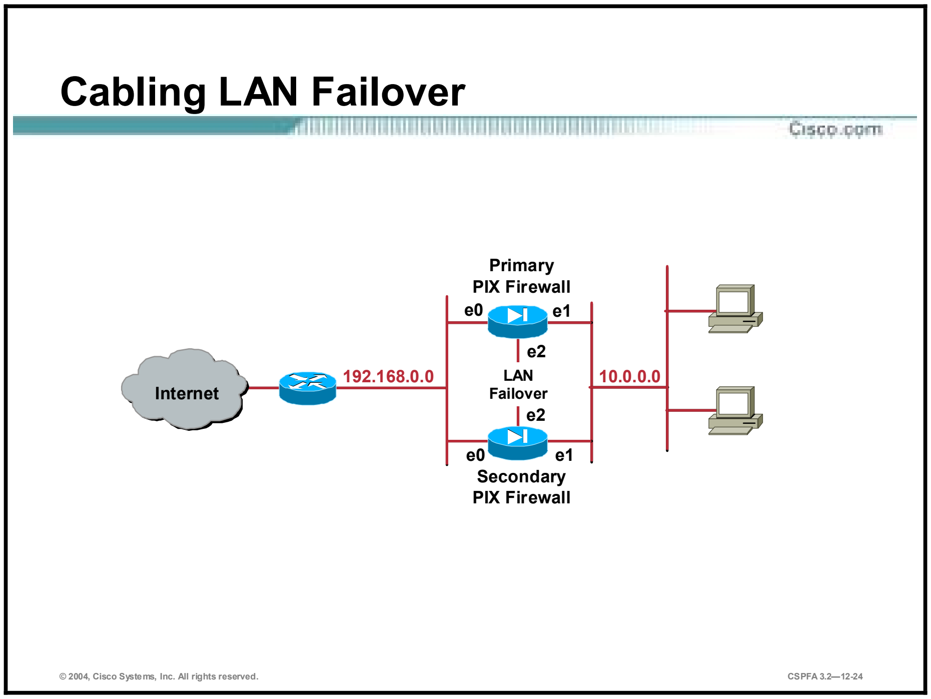
Step 4 Power on the secondary PIX Firewall.

Step 5 Configure the secondary PIX Firewall with the LAN-based failover command set.

Step 6 Save the secondary firewall's configuration to Flash memory.

Step 7 Connect the PIX Firewall LAN-based failover interface to the network.

Step 8 Reboot the secondary firewall.



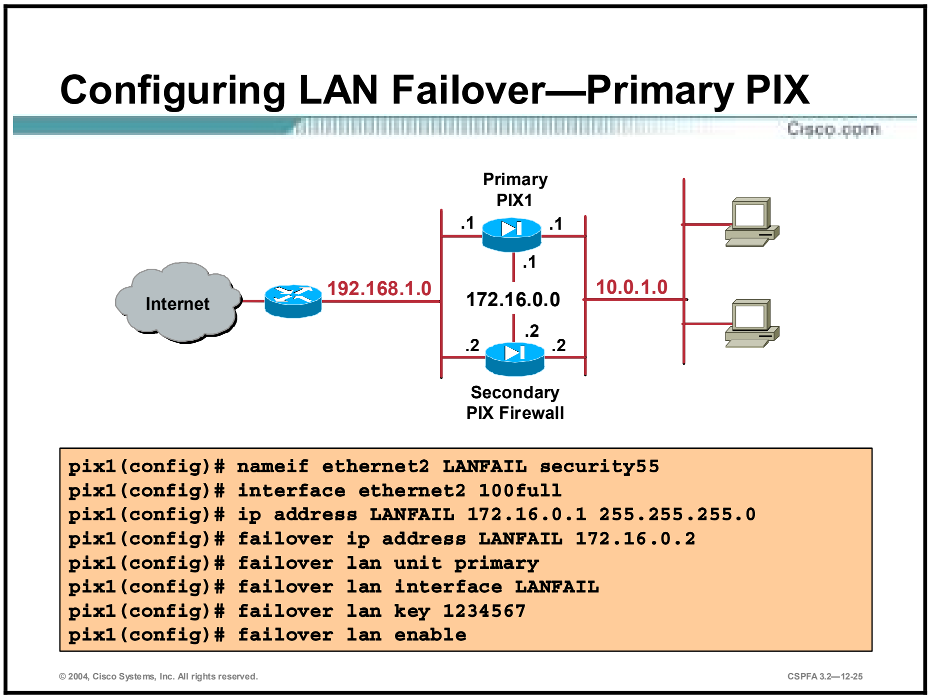
Cabling LAN Failover

CSPFA 3.2—12-24

The following steps give detail on configuring LAN-based failover:

Step 1 Install a LAN-based failover connection between firewalls, as follows:

- Perform the following on any switch that connects to the PIX Firewall:
  - — Enable portfast.
  - — Turn off trunking.
  - — Turn off channelling.
- Attach a network cable for each network interface you plan to use. The IP addresses on the secondary unit are different from the addresses on the primary unit, but they should be in the same subnet for each interface. If you plan to do stateful failover, one of the interfaces must be dedicated to this function. If the failover cable is connected to the PIX Firewall, disconnect it.



Configuring LAN Failover—Primary PIX

```
pix1(config)# nameif ethernet2 LANFAIL security55
pix1(config)# interface ethernet2 100full
pix1(config)# ip address LANFAIL 172.16.0.1 255.255.255.0
pix1(config)# failover ip address LANFAIL 172.16.0.2
pix1(config)# failover lan unit primary
pix1(config)# failover lan interface LANFAIL
pix1(config)# failover lan key 1234567
pix1(config)# failover lan enable
```

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—12-25

Step 2 Complete the following substeps to configure the primary PIX Firewall before connecting the failover LAN interface:

1. Use the **clock set** command on the active PIX Firewall to synchronize the time on the primary and secondary PIX Firewalls.

2. Ensure that you have not used the auto or the 1000auto option in any interface command in your configuration. If you have used one of these options, change it by reentering the command with the correct information. Always specify the speed for the interface, such as 10BASE-T for 10 Mbps or 100BASE-TX for 100 Mbps. Ensure that the speeds and duplexes are the same for any devices on the subnets, including switches and routers. For stateful failover, set the stateful failover dedicated interface speed to 100full or 1000sxfull. This is extremely important, and you should do this even if you are using a crossover connector to connect the PIX Firewalls directly to each other.

Note Use the **clear xlate** command after changing the **interface** command.

3. Configure a dedicated LAN-based failover interface as follows:

- Use the **nameif** command to set its name and security level.
- Use the **interface** command to enable it and set its hardware speed.
- Use the **ip address** command to assign an IP address for it.

4. Use the **failover** command to enable failover.

5. Set a **failover IP address**. A failover IP address must be set for failover to work.

6. If you are configuring stateful failover, use the **failover link** command to specify the name of the dedicated interface you are using.

7. Use the **write memory** command to save the primary PIX Firewall's configuration.

8. Connect the primary PIX Firewall's failover interface to the network.

9. Use the **no failover** command to disable failover.

10. Use the **failover lan unit** command to designate this firewall as the primary firewall.

11. Use the **failover lan interface** command to specify the name of the failover interface.

12. Use the **failover lan key** command to create a shared secret key for failover security.

13. Use the **failover lan enable** command to enable LAN-based failover.

14. Use the **failover** command to enable failover.

Step 3 Save the primary PIX Firewall's configuration to Flash memory.

The **failover lan** commands can be disabled by using their **no** forms. The syntax of the **failover lan** commands is as follows:

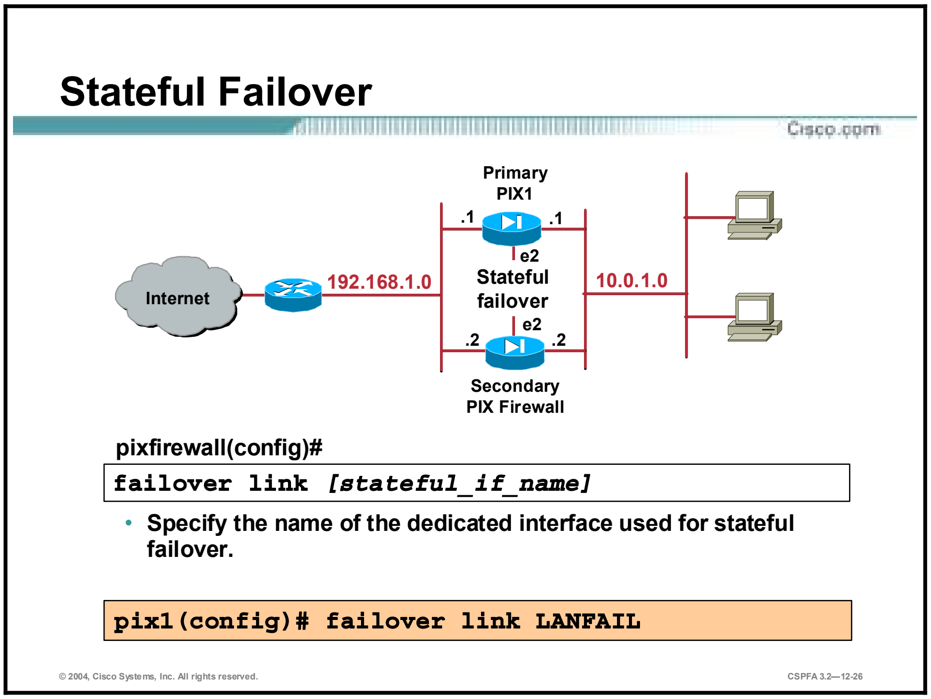
failover lan unit primary | secondary

failover lan interface ***if_name***

failover lan key ***key_secret***

failover lan enable

if_name	Specifies the interface name for LAN-based failover.
key	Enables encryption and authentication of LAN-based failover messages between PIX Firewalls.
key_secret	Specifies the shared secret key.



Stateful Failover

pixfirewall(config)#

```
failover link [stateful_if_name]
```

- Specify the name of the dedicated interface used for stateful failover.

```
pix1(config)# failover link LANFAIL
```

The stateful failover feature passes per-connection stateful information to the standby unit. After a failover occurs, the same connection information is available at the new active unit. End-user applications are not required to do a reconnect to keep the same communication session. The state information passed to the standby unit includes the global pool addresses and status, connection and translation information and status, the negotiated H.323 UDP ports, the port allocation bit map for PAT, and other details necessary to let the standby unit take over processing if the primary unit fails.

Depending on the failure, the PIX Firewall takes from 15 to 45 seconds to cause a switchover. Applications not handled by stateful failover will then require time to reconnect before the active unit becomes fully functional.

The same LAN interface used for LAN-based failover can also be used for stateful failover. However, the interface needs enough capacity to handle both the LAN-based failover and stateful failover traffic. If the interface does not have the necessary capacity, use two separate, dedicated interfaces.

Stateful failover requires an Ethernet interface with a minimum speed of 100 Mbps full duplex to be used for passing state information between the two PIX Firewall units. The stateful failover interface can be connected to either of the following:

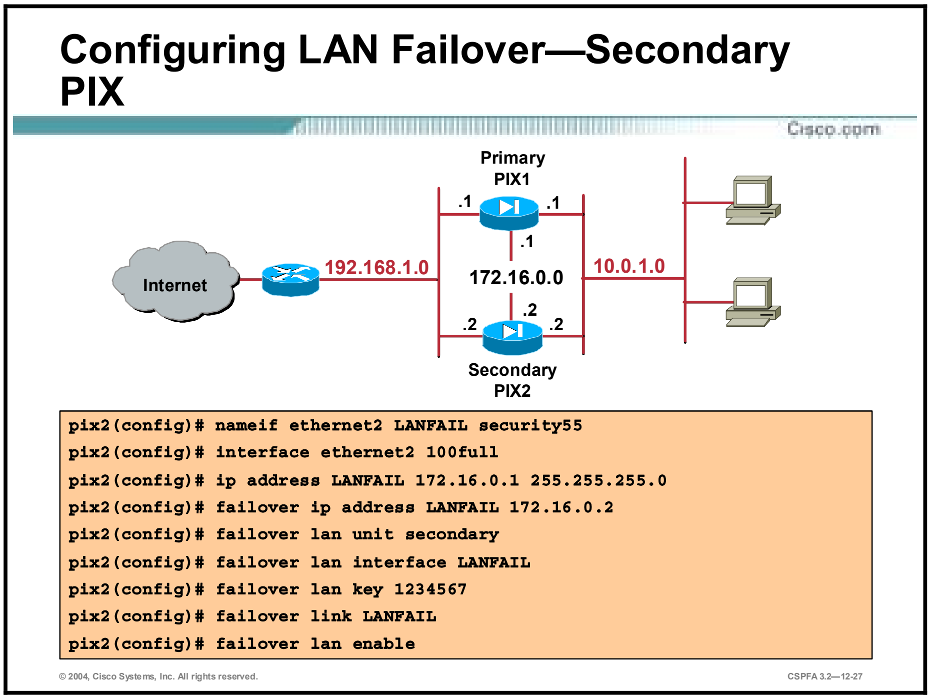
- 100Base-TX full-duplex on a dedicated switch or dedicated VLAN of a switch
- 1000Base-SX full-duplex on a dedicated switch or dedicated VLAN of a switch

Data is passed over the dedicated interface using IP Protocol 8. No hosts or routers should be on this interface.

The **failover replicate http** command enables the stateful replication of HTTP sessions in a stateful failover environment. The **no** form of this command disables HTTP replication in a stateful failover configuration. When HTTP replication is enabled, the **show failover** command displays the failover replicate http configuration.

The syntax for the **failover replicate http** command is as follows:

```
failover replicate http
```



Configuring LAN Failover—Secondary PIX

```
pix2(config)# nameif ethernet2 LANFAIL security55
pix2(config)# interface ethernet2 100full
pix2(config)# ip address LANFAIL 172.16.0.1 255.255.255.0
pix2(config)# failover ip address LANFAIL 172.16.0.2
pix2(config)# failover lan unit secondary
pix2(config)# failover lan interface LANFAIL
pix2(config)# failover lan key 1234567
pix2(config)# failover link LANFAIL
pix2(config)# failover lan enable
```

CSPFA 3.2—12-27

Step 4 Power on the secondary PIX Firewall.

Step 5 Without the LAN-based failover interface connected, complete the following substeps on the secondary firewall:

1. Use the **nameif** command to specify a name and security level for the failover interface.
2. Use the **interface** command to enable the failover interface and set its connection speed.
3. Use the **ip address** command to specify a name and security level for the failover interface.
4. Use the **failover ip address** command to assign a failover IP address to the failover interface.
5. Use the **failover lan unit** command to designate this firewall as the secondary firewall.
6. Use the **failover lan interface** command to specify the name of the failover interface.
7. Use the **failover lan key** command to enter the secret key shared with the primary firewall.
8. Use the **failover lan enable** command to enable LAN-based failover.
9. Use the **failover** command to enable failover.

Step 6 Use the **write memory** command to save your configuration to Flash memory.

In the figure, all the **failover lan** commands, except the **failover lan unit** command, are entered on the secondary PIX Firewall exactly as they would be entered on the primary. On both firewalls, Ethernet 3 is designated as the failover interface and is configured with the following parameters:

- Name—LANFAIL
- Security level—55
- Speed and duplex—100full

- IP address—172.16.0.1
- Netmask—255.255.255.0
- Failover IP address—172.16.0.2
- Failover LAN key—1234567

The output of the **show failover** command includes a section for LAN-based failover if it is enabled. An example of this additional **show failover** command output follows:

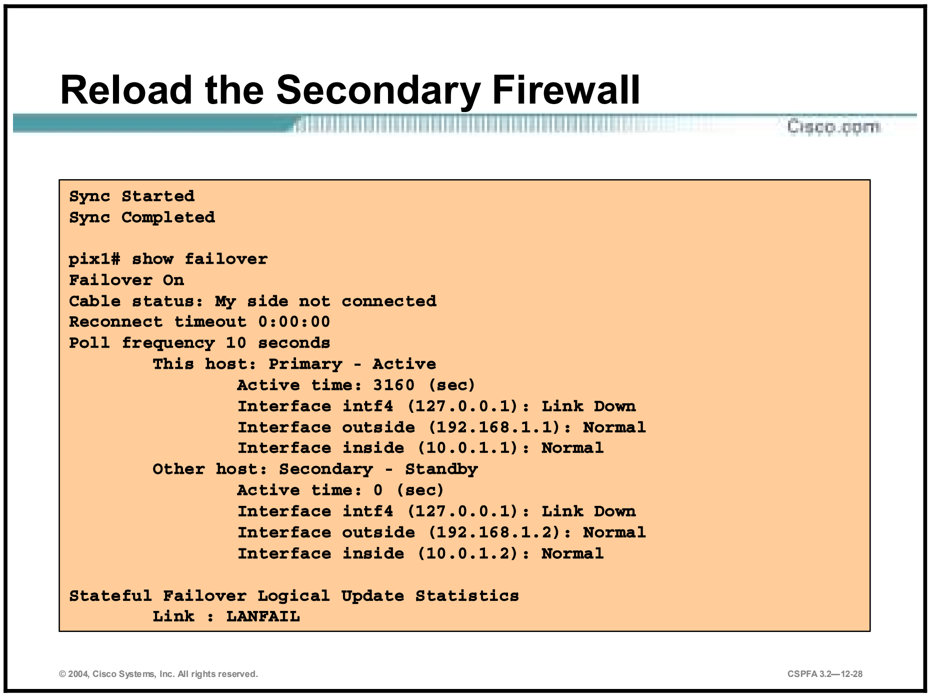
```
Lan Based Failover is Active
             interface LANFAILOVER (172.16.0.2): Normal, peer
(172.17.0.1): Normal
```

The **show failover lan** command displays only the LAN-based failover section, as follows:

```
pix# show failover lan
Lan Based Failover is Active
             interface LANFAILOVER (172.16.0.1): Normal, peer
(172.16.0.2): Normal
```

The **show failover lan detail** command is used mainly for debugging purposes and displays information similar to the following:

```
pix# show failover lan detail
Lan Failover is Active
This Pix is Primary
Command Interface is LANFAIL
Peer Command Interface IP is 172.16.0.2
My interface status is normal
Peer interface status is normal
Peer interface downtime is 0x0

Total msg send: 103093, rcvd: 103031, dropped: 0, retrans: 13,
send_err: 0
Total/Cur/Max of 51486:0:5 msgs on retransQ

msgs on retransQ if any
LAN FO cmd queue, count: 0, head: 0x0, tail: 0x0
Failover config state is 0x5c
Failover config poll cnt is 0
Failover pending tx msg cnt is 0
Failover Fmsg cnt is 0
```



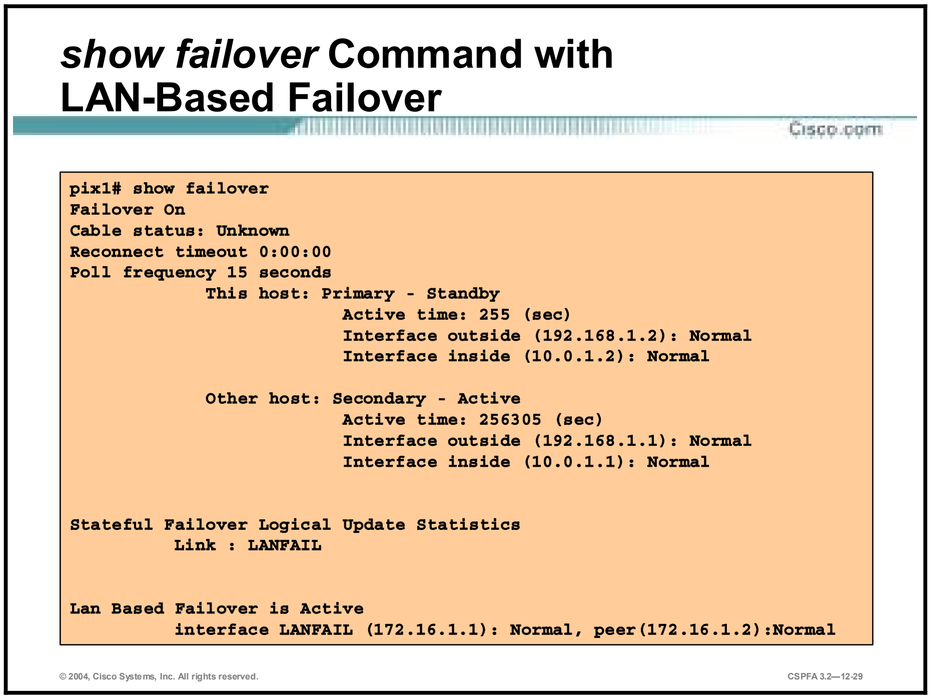
Reload the Secondary Firewall

```
Sync Started
Sync Completed

pix1# show failover
Failover On
Cable status: My side not connected
Reconnect timeout 0:00:00
Poll frequency 10 seconds
        This host: Primary - Active
                Active time: 3160 (sec)
                Interface intf4 (127.0.0.1): Link Down
                Interface outside (192.168.1.1): Normal
                Interface inside (10.0.1.1): Normal
        Other host: Secondary - Standby
                Active time: 0 (sec)
                Interface intf4 (127.0.0.1): Link Down
                Interface outside (192.168.1.2): Normal
                Interface inside (10.0.1.2): Normal

Stateful Failover Logical Update Statistics
        Link : LANFAIL
```

CSPFA 3.2—12-28

Step 7 Connect the PIX Firewall's failover interface to the network.

Step 8 Use the **reload** command to reboot the secondary PIX Firewall. The primary PIX Firewall configuration is replicated on the secondary PIX Firewall. The following messages will appear on the primary PIX Firewall: "sync started" and "sync completed." "Sync started" denotes the start of the synchronization. "Sync completed" acknowledges the completion of the replication.



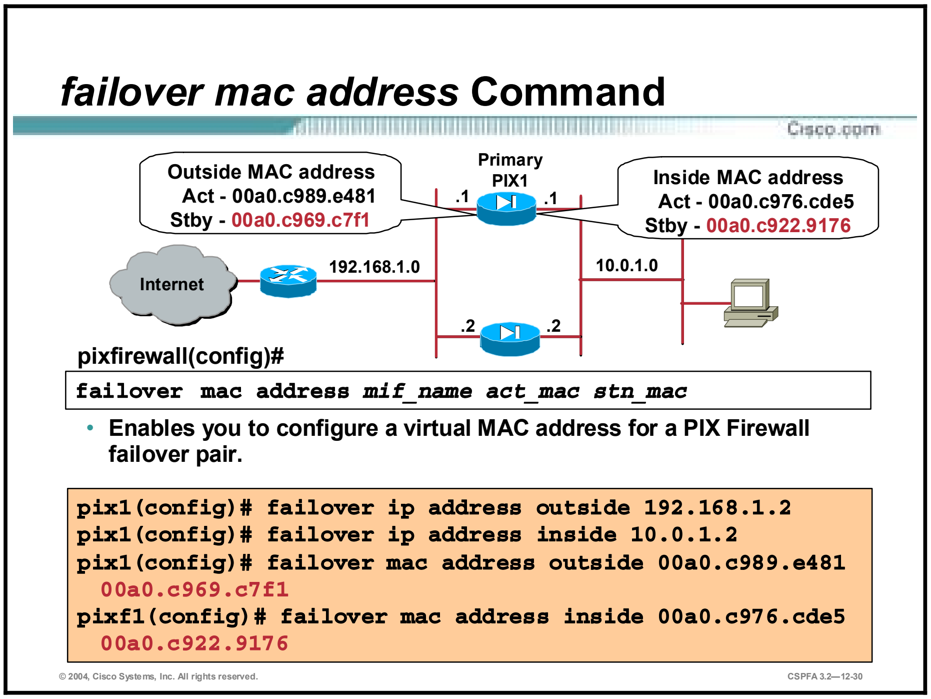
show failover Command with LAN-Based Failover

Cisco.com

```
pix1# show failover
Failover On
Cable status: Unknown
Reconnect timeout 0:00:00
Poll frequency 15 seconds
             This host: Primary - Standby
                          Active time: 255 (sec)
                          Interface outside (192.168.1.2): Normal
                          Interface inside (10.0.1.2): Normal

             Other host: Secondary - Active
                          Active time: 256305 (sec)
                          Interface outside (192.168.1.1): Normal
                          Interface inside (10.0.1.1): Normal


Stateful Failover Logical Update Statistics
          Link : LANFAIL


Lan Based Failover is Active
          interface LANFAIL (172.16.1.1): Normal, peer(172.16.1.2):Normal
```

CSPFA 3.2—12-29

The figure shows the output of the **show failover** command when LAN-based failover is configured. It includes a section for LAN-based failover, which displays the name and IP address of the interface used for LAN-based failover. Notice that this interface does not appear in the interface list that displays the status of the interfaces.



failover mac address Command

pixfirewall(config)#

```
failover mac address mif_name act_mac stn_mac
```

- Enables you to configure a virtual MAC address for a PIX Firewall failover pair.

```
pix1(config)# failover ip address outside 192.168.1.2
pix1(config)# failover ip address inside 10.0.1.2
pix1(config)# failover mac address outside 00a0.c989.e481
  00a0.c969.c7f1
pixf1(config)# failover mac address inside 00a0.c976.cde5
  00a0.c922.9176
```

CSPFA 3.2—12-30

When the primary PIX Firewall is replaced with a new primary PIX Firewall, the secondary PIX Firewall acquires the MAC address of the new primary PIX Firewall and sends out gratuitous ARPs on all interfaces to update the devices connected to the PIX Firewall. With LAN-based failover, you can prevent the MAC address change by manually setting the MAC addresses of the primary and secondary PIX Firewall using the command **failover mac address**. With LAN-based failover, the PIX Firewall can be configured to use a virtual MAC address instead of assuming the MAC address of its failover peer. If a virtual MAC address is not specified, the PIX Firewall failover pair uses the burned-in network interface card (NIC) address as the MAC address.

When adding the **failover mac address** command to your configuration, it is best to configure the virtual MAC address, save the configuration to Flash memory, and then reload the PIX Firewall pair. You must also write the complete PIX Firewall configuration, including the **failover mac address** command, into the Flash memory of the secondary PIX Firewall for the virtual MAC addressing to take effect.

The syntax of the **failover mac address** command is as follows:

failover mac address ***mif_name act_mac stn_mac***

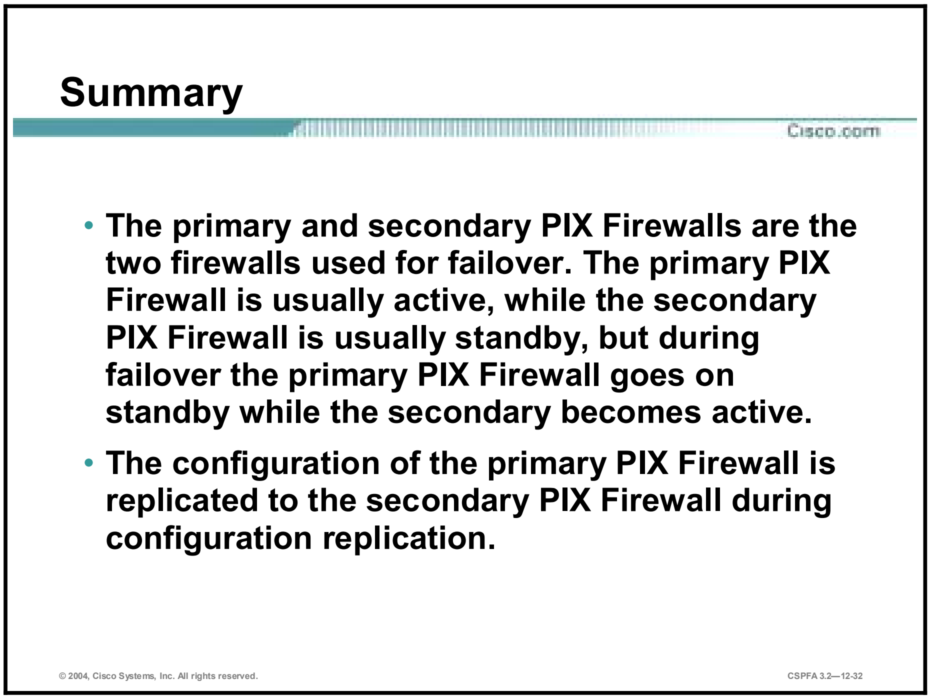
mif_name	Specifies the name of the interface to set the MAC address.
act_mac	Specifies the interface MAC address for the active PIX Firewall.
stn_mac	Specifies the interface MAC address for the standby PIX Firewall.

Note If the virtual MAC address is added when there are active connections, those connections drop.

Note The **failover mac address** command is unnecessary on an interface configured for LAN-based failover because the **failover lan interface** command does not change the IP and MAC addresses when failover occurs.

Summary

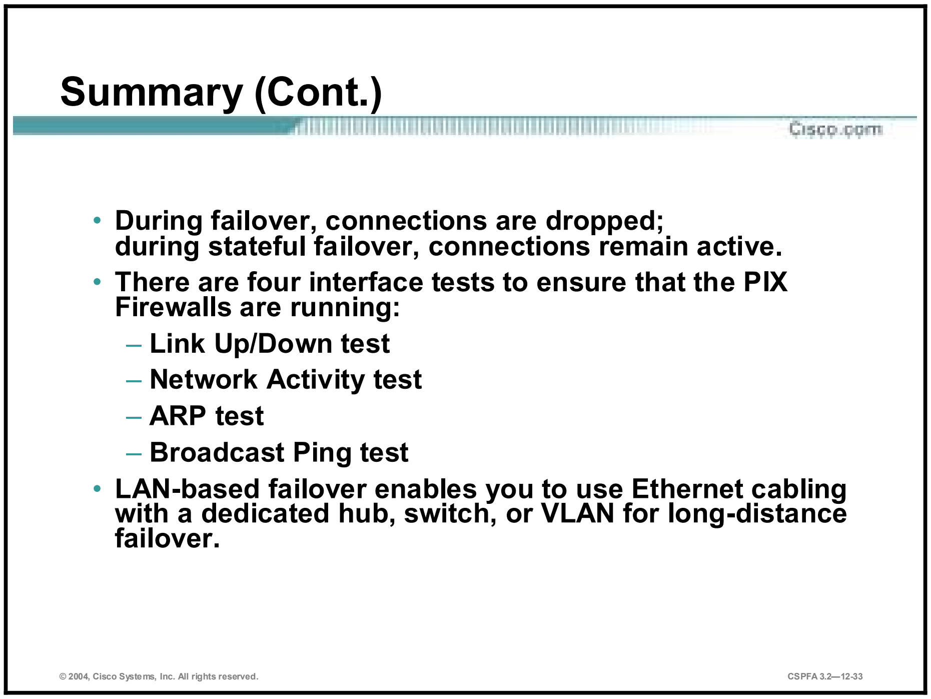
This topic summarizes what you learned in this lesson.



Summary

- The primary and secondary PIX Firewalls are the two firewalls used for failover. The primary PIX Firewall is usually active, while the secondary PIX Firewall is usually standby, but during failover the primary PIX Firewall goes on standby while the secondary becomes active.
- The configuration of the primary PIX Firewall is replicated to the secondary PIX Firewall during configuration replication.

CSPFA 3.2—12-32



Summary (Cont.)

Cisco.com

- During failover, connections are dropped; during stateful failover, connections remain active.
- There are four interface tests to ensure that the PIX Firewalls are running:
  - Link Up/Down test
  - Network Activity test
  - ARP test
  - Broadcast Ping test
- LAN-based failover enables you to use Ethernet cabling with a dedicated hub, switch, or VLAN for long-distance failover.

CSPFA 3.2—12-33

Lab Exercise—Configure LAN-Based Failover

Complete the following lab exercise to practice what you learned in this lesson.

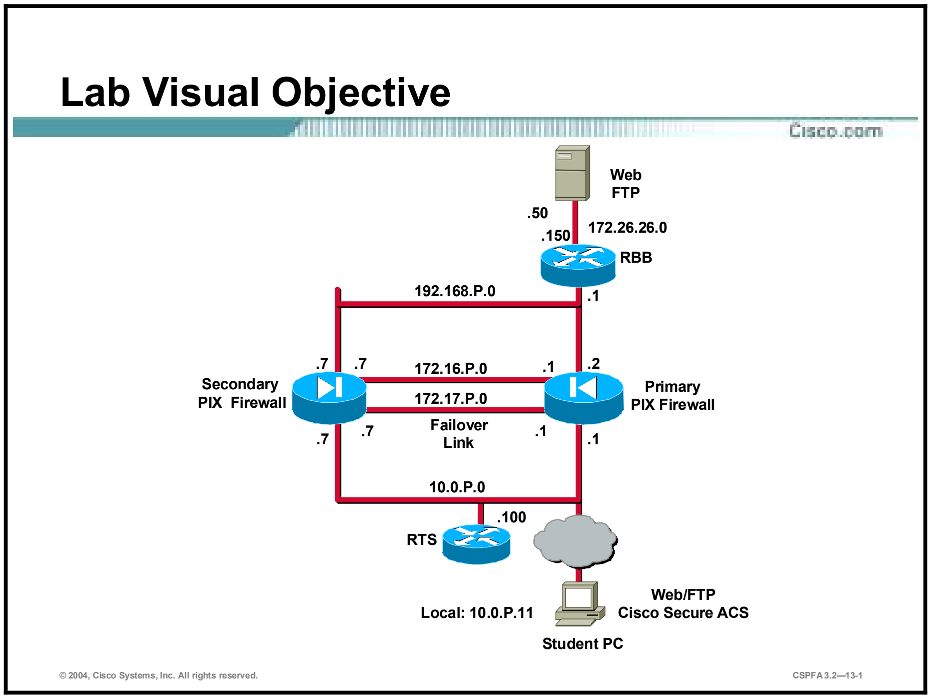
Objectives

In this lab exercise you will complete the following tasks:

- Configure the primary PIX Firewall for LAN-based stateful failover to the secondary PIX Firewall.
- Configure the secondary PIX Firewall for LAN-based failover.
- Test LAN-based stateful failover.
- Make the primary PIX Firewall active.

Visual Objectives

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—13-1

Task 1—Configure the Primary PIX Firewall for LAN-Based Stateful Failover to the Secondary PIX Firewall

Complete the following steps to configure the primary PIX Firewall for failover to the secondary PIX Firewall:

Step 1 Assign an IP address to the outside interface of the PIX Firewall:

```
pixP(config)# ip address outside 192.168.P.2 255.255.255.0
```

(where P = pod number)

Step 2 Use the **clock set** command on the active PIX Firewall to synchronize the time on both PIX Firewalls.

```
pixP(config)# clock set hh:mm:ss month day year
```

Step 3 Assign the PIX Firewall interface name (MYFAILOVER) and security level (55).

```
pixP(config)# nameif e3 MYFAILOVER security55
```

Step 4 Enable the interface for an Intel full duplex:

```
pixP(config)# interface e3 100full
```

Step 5 Assign an IP address to the interface:

```
pixP(config)# ip address MYFAILOVER 172.17.P.1 255.255.255.0
```

(where P = pod number)

Step 6 Use the **failover** command to enable failover on the primary unit:

```
pixP(config)# failover
```

Step 7 Change the failover poll time to eight seconds so the PIX Firewall triggers failover faster.

```
pixP(config)# failover poll 8
```

Step 8 Use the **show failover** command to verify that the primary PIX Firewall is active:

```
pixP(config)# show failover
Failover On
Cable status: My side not connected
Reconnect timeout 0:00:00
Poll frequency 8 seconds
This host: Secondary - Active
    Active time: 225 (sec)
    Interface intf5 (127.0.0.1): Link Down (Shutdown)
    Interface intf4 (127.0.0.1): Link Down (Shutdown)
    Interface MYFAILOVER (172.17.P.1): Link Down (Waiting)
    Interface dmz (172.16.P.1): Normal (Waiting)
    Interface outside (192.168.P.2): Normal (Waiting)
    Interface inside (10.0.P.1): Normal (Waiting)
Other host: Secondary - Standby
    Active time: 0 (sec)
    Interface intf5 (0.0.0.0): Unknown (Shutdown)
    Interface intf4 (0.0.0.0): Unknown (Shutdown)
    Interface intf3 (0.0.0.0): Unknown (Waiting)
    Interface intf2 (0.0.0.0): Unknown (Waiting)
    Interface outside (0.0.0.0): Unknown (Waiting)
    Interface inside (0.0.0.0): Unknown (Waiting)
```

Step 9 Assign a failover IP address for each interface to specify the standby unit's interface addresses:

```
pixP(config)# failover ip address inside 10.0.P.7
pixP(config)# failover ip address outside 192.168.P.7
pixP(config)# failover ip address dmz 172.16.P.7
pixP(config)# failover ip address MYFAILOVER 172.17.P.7
```

Step 10 Enter the **show failover** command to verify that the secondary unit now has IP addresses for each interface:

```
pixP(config)# show failover
Failover On
Cable status: My side not connected
Reconnect timeout 0:00:00
Poll frequency 3 seconds
This host: Secondary - Active
    Active time: 510 (sec)
    Interface intf5 (127.0.0.1): Link Down (Shutdown)
    Interface intf4 (127.0.0.1): Link Down (Shutdown)
    Interface MYFAILOVER (172.17.P.1): Link Down (Waiting)
    Interface dmz (172.16.P.1): Normal (Waiting)
    Interface outside (192.168.P.2): Normal (Waiting)
    Interface inside (10.0.P.1): Normal (Waiting)
Other host: Secondary - Standby
    Active time: 0 (sec)
    Interface intf5 (127.0.0.1): Unknown (Shutdown)
    Interface intf4 (127.0.0.1): Unknown (Shutdown)
    Interface MYFAILOVER (172.17.P.7): Unknown (Waiting)
    Interface dmz (172.16.P.7): Unknown (Waiting)
    Interface outside (192.168.P.7): Unknown (Waiting)
    Interface inside (10.0.P.7): Unknown (Waiting)
```

Step 11 Use the **failover link** command to specify the name of the dedicated interface you are using:

```
pixP(config)# failover link MYFAILOVER
```

Step 12 Save all changes to Flash memory.

```
pixP(config)# write memory
```

Step 13 Configure LAN-based failover on the primary unit. Complete the following substeps:

1. Disable failover:

```
pixP(config)# no failover
```

2. Specify the primary PIX Firewall to use for LAN-based failover:

```
pixP(config)# failover lan unit primary
```

3. Specify the interface name for LAN-based failover:

```
pixP(config)# failover lan interface MYFAILOVER
```

4. Enable encryption and authentication of LAN-based failover messages between PIX Firewalls:

```
pixP(config)# failover lan key 1234567
```

5. Enable LAN-based failover:

```
pixP(config)# failover lan enable
```

6. Enable failover:

```
pixP(config)# failover
```

Step 14 Save all changes to Flash memory:

```
pixP(config)# write memory
```

Step 15 Make sure that the primary PIX Firewall is enabled for stateful failover by using the **show failover** command:

```
pixP(config)# show failover
Failover On
```

```
 Cable status: My side not connected
 Reconnect timeout 0:00:00
 Poll frequency 8 seconds
         This host: Primary - Active
                 Active time: 510 (sec)
                 Interface intf5 (127.0.0.1): Link Down (Shutdown)
                 Interface intf4 (127.0.0.1): Link Down (Shutdown)
                 Interface dmz (172.16.P.1): Normal (Waiting)
                 Interface outside (192.168.P.2): Normal (Waiting)
                 Interface inside (10.0.P.1): Normal (Waiting)
         Other host: Secondary – Standby (Failed)
                 Active time: 0 (sec)
                 Interface intf5 (127.0.0.1): Unknown (Shutdown)
                 Interface intf4 (127.0.0.1): Unknown (Shutdown)
                 Interface dmz (172.16.P.7): Unknown (Waiting)
                 Interface outside (192.168.P.7): Unknown (Waiting)
                 Interface inside (10.0.P.7): Unknown (Waiting)
  Stateful Failover Logical Update Statistics
         Link : MYFAILOVER
         Stateful Obj    xmit       xerr       rcv        rerr
         General         0          0          0          0
         sys cmd         0          0          0          0
         up time         0          0          0          0
         xlate           0          0          0          0
         tcp conn        0          0          0          0
         udp conn        0          0          0          0
         ARP tbl         0          0          0          0
         RIP Tbl         0          0          0          0

         Logical Update Queue Information
                         Cur     Max     Total
         Recv Q:         0       0       0
         Xmit Q:         0       0       0

  Lan Based Failover is Active

         Interface MYFAILOVER (172.17.P.1): Normal, peer (172.17.P.7)
Down
```

Step 16 Wait for the failover initialization process to complete. You will see the following messages on your PIX Firewall console:

```
LAN-based Failover startup ping test failed!!
```

```
Wait for pix LAN-based failover init process to complete...
LAN-based Failover: Send hello msg and start failover monitoring
```

Step 17 Verify that you can ping the super server:

```
C:\> ping 172.26.26.50
```

Step 18 Verify that you can telnet to the backbone router:

```
C:\> telnet 192.168.P.1
```

Task 2—Configure the Secondary PIX Firewall for LAN-Based Failover

Complete the following steps to prepare the secondary PIX Firewall for failover. Your instructor will provide you with instructions for accessing the secondary firewall.

Step 1 Ask your instructor to power up your secondary PIX Firewall.

Step 2 Without the LAN-based failover interface connected, complete the following substeps on the secondary PIX Firewall:

1. When prompted to configure the PIX Firewall through interactive prompts, press <**Control Z**> to escape.
2. Enter configuration mode.
3. Assign a name and security level to the failover interface:

```
pixP(config)# nameif e3 MYFAILOVER security55
```

4. Enable the interface for an Intel full duplex:

```
pixP(config)# interface e3 100full
```

5. Assign an IP address to the interface:

```
pixP(config)# ip address MYFAILOVER 172.17.P.1
```

(where P = pod number)

6. Assign a failover IP address to the interface:

```
pixP(config)# failover ip address MYFAILOVER 172.17.P.7 255.255.255.0
```

(where P = pod number)

7. Designate this firewall as the secondary firewall:

```
pixP(config)# failover lan unit secondary
```

8. Specify the name of the interface to be used for LAN-based failover:

```
pixP(config)# failover lan interface MYFAILOVER
```

9. Enter the secret key shared with the primary PIX Firewall:

```
pixP(config)# failover lan key 1234567
```

10. Enable LAN-based failover:

```
pixP(config)# failover lan enable
```

11. Enable failover:

```
pixP(config)# failover
```

12. Save the configuration to Flash memory:

```
pixP(config)# write mem
```

Step 3 Connect the secondary PIX Firewall to the network: Ask your instructor to connect the LAN-based failover interface to the network.

Step 4 Reload the secondary PIX Firewall:

```
pixP(config)# reload
```

Task 3—Test LAN-Based Stateful Failover

Complete the following steps to test LAN-based stateful failover:

Step 1 After you see the message “Sync Complete” on your primary PIX Firewall console, telnet to the backbone router from your Windows command line:

```
C:\> telnet 192.168.P.1
```

(where P = pod number)

Step 2 When prompted for a password, enter **cisco**:

```
User Access Verification
Password: cisco
rbb>
```

Step 3 Start a continuous ping to 172.26.26.50:

```
C:\ ping 172.26.26.50 -t
```

Step 4 Reload the primary PIX Firewall:

```
pixP(config)# reload
```

Step 5 When asked to confirm the reload, press **Enter**.

Step 6 Verify that the continuous ping is still active.

Step 7 Return to your telnet connection to the backbone router. Verify that it is still active by attempting to enter privileged mode. If the router accepts the command and you are prompted for a privileged mode password, your session is still active.

```
rbb> en
```

Step 8 Enter the **show failover** command on the primary firewall and observe the output:

```
pixP(config)# show failover
Failover On
 Cable status: Normal
 Reconnect timeout 0:00:00
 Poll frequency 3 seconds
        This host: Primary - Standby
                Active time: 510 (sec)
                Interface intf5 (127.0.0.1): Link Down (Shutdown)
```

```
                 Interface intf4 (127.0.0.1): Link Down (Shutdown)
                 Interface dmz (172.16.P.7): Normal
                 Interface outside (192.168.P.7): Normal
                 Interface inside (10.0.P.7): Normal
         Other host: Secondary - Active
                 Active time: 93 (sec)
                 Interface intf5 (127.0.0.1): Link Down (Shutdown)
                 Interface intf4 (127.0.0.1): Link Down (Shutdown)
                 Interface dmz (172.16.P.1): Normal
                 Interface outside (192.168.P.2): Normal
                 Interface inside (10.0.P.1): Normal
  Stateful Failover Logical Update Statistics
         Link : MYFAILOVER
         Stateful Obj    xmit       xerr       rcv        rerr
         General         967        0        998          0
         sys cmd         675        0        897          0
         up time         4          0          2          0
         xlate           1006       0         65          0
         tcp conn        456        0        345          0
         udp conn        0          0          0          0
         ARP tbl         884        0          0          0
         RIP Tbl         0          0          0          0

Logical Update Queue Information
                         Cur     Max     Total
         Recv Q:         0       3       6740
         Xmit Q:         0       7       8665

  Lan Based Failover is Active

         Interface MYFAILOVER (172.17.P.1): Normal, peer (172.17.P.7)
Normal
```

(where P = pod number)

Task 4—Make the Primary PIX Firewall Active

Complete the following steps to make the primary PIX Firewall the active PIX Firewall:

Step 1 Make the primary PIX Firewall the active PIX Firewall by using the **failover active** command. Make sure that you are connected to the primary PIX Firewall’s console port.

```
pixP(config)# failover active
```

Step 2 Verify that the failover active command worked by using the **show failover** command. The primary PIX Firewall should show that it is in active mode and the secondary PIX Firewall should show that it is in the standby mode.

```
pixP(config)# show failover
Failover On
Cable status: Normal
Reconnect timeout 0:00:00
        This host: Primary - Active
                  Active time: 7350 (sec)
                  Interface intf5 (127.0.0.1): Link Down (Shutdown)
                  Interface intf4 (127.0.0.1): Link Down (Shutdown)
                  Interface dmz (172.16.P.1): Normal (Waiting)
                  Interface outside (192.168.P.2): Normal (Waiting)
                  Interface inside (10.0.P.1): Normal (Waiting)
        Other host: Secondary - Standby
                  Active time: 7300 (sec)
                  Interface intf5 (0.0.0.0): Link Down (Shutdown)
                  Interface intf4 (0.0.0.0): Link Down (Shutdown)
                  Interface dmz (172.16.P.7): Normal
                  Interface outside (192.168.P.7): Normal
                  Interface inside (10.0.P.7): Normal
Stateful Failover Logical Update Statistics
         Link : MYFAILOVER
         Stateful Obj    xmit       xerr      rcv        rerr
         General         999        0         1256       0
         sys cmd         775        0         987        0
         up time         8          0         6          0
         xlate           2006       0         85         0
         tcp conn        656        0         445        0
         udp conn        0          0         0          0
         ARP tbl         1884       0         0          0
         RIP Tbl         0          0         0          0

Logical Update Queue Information
                         Cur     Max     Total
         Recv Q:         0       9       7860
         Xmit Q:         0       12      9875

  Lan Based Failover is Active

         Interface MYFAILOVER (172.17.P.1): Normal, peer (172.17.P.7)
Normal
```

(where P = pod number)

Step 3 Disable failover on the primary and secondary PIX Firewalls:

```
pixP(config)# no failover
```

Step 4 Save the configuration to Flash memory:

```
pixP(config)# write mem
```

13

Switching and Routing

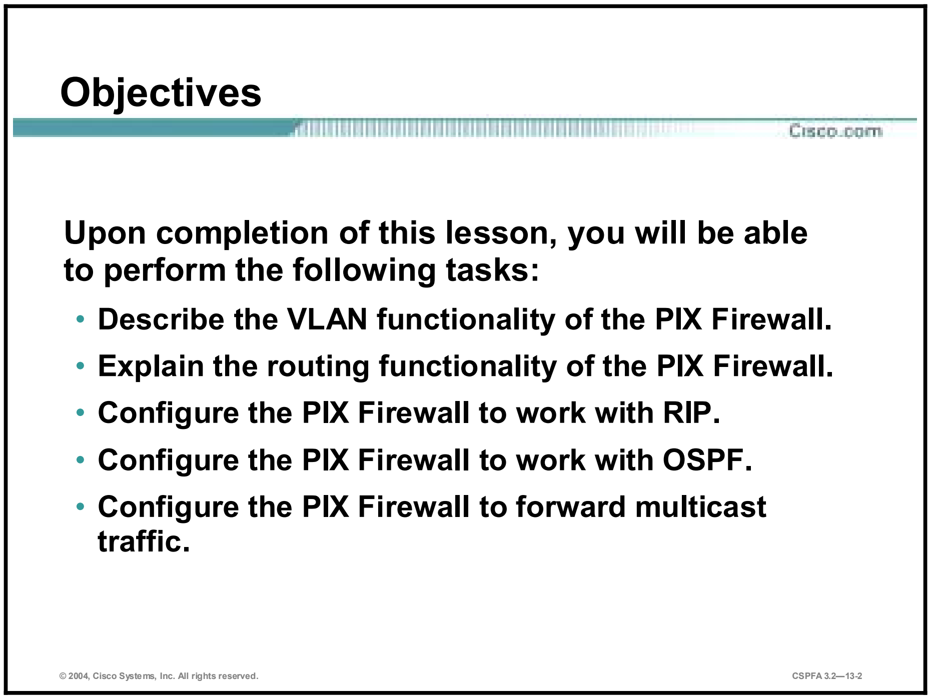
Overview

This lesson includes the following topics:

- Objectives
- Virtual LANs
- Static and dynamic routing
- OSPF
- Multicast
- Summary

Objectives

This topic lists the lesson’s objectives.



Objectives

Cisco.com

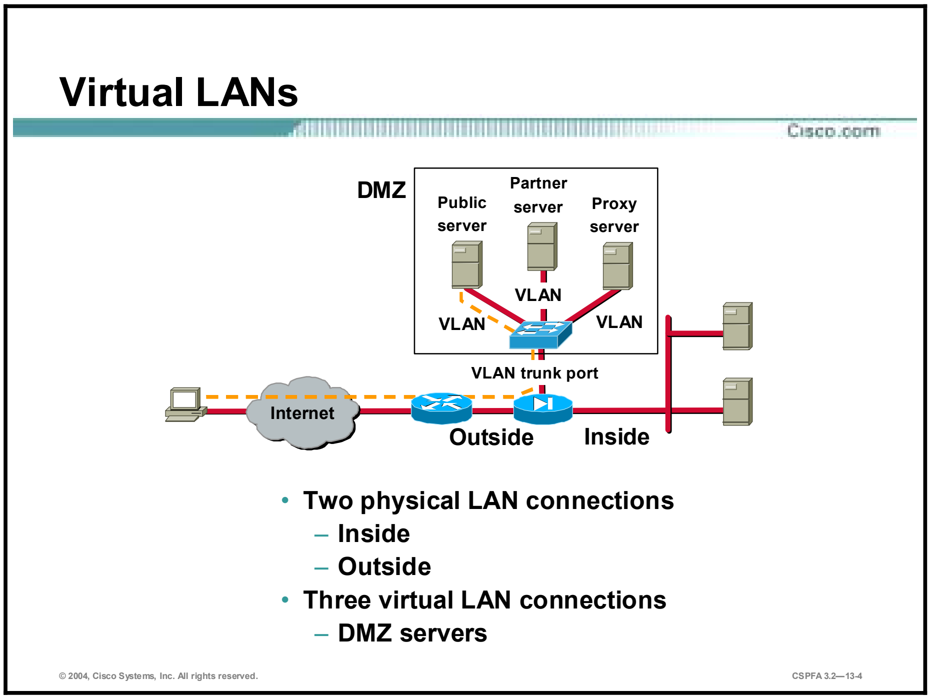
Upon completion of this lesson, you will be able to perform the following tasks:

- **Describe the VLAN functionality of the PIX Firewall.**
- **Explain the routing functionality of the PIX Firewall.**
- **Configure the PIX Firewall to work with RIP.**
- **Configure the PIX Firewall to work with OSPF.**
- **Configure the PIX Firewall to forward multicast traffic.**

CSPFA 3.2—13-2

Virtual LANS

This topic explains the Virtual LAN (VLAN) capabilities of the PIX Firewall.



Virtual LANs

- Two physical LAN connections
  - Inside
  - Outside
- Three virtual LAN connections
  - DMZ servers

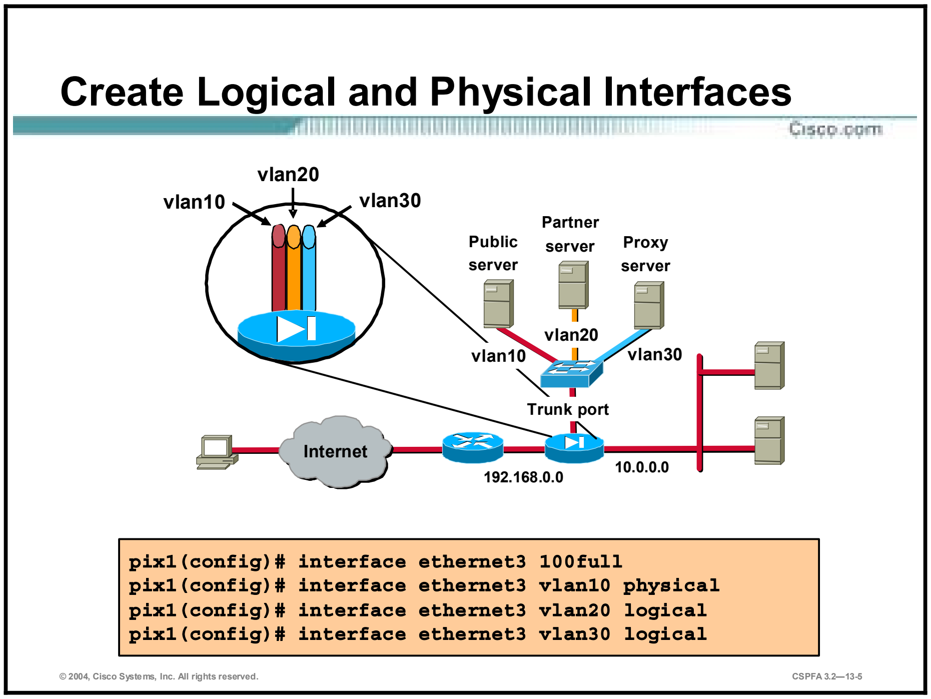
CSPFA 3.2—13-4

With PIX Firewall Software Version 6.3, the administrator can assign VLANs to physical interfaces on the PIX Firewall or configure multiple logical interfaces on a single physical interface and assign each logical interface to a specific VLAN. VLANs connect devices on one or more physical LAN segments through software so that they can act as though they are attached to the same physical LAN. VLANs make this connection based on logical (software) connections instead of physical connections. This characteristic also makes them extremely flexible and enables you to configure (or reconfigure) which segments belong to which VLAN entirely through software.

The firewall supports only 802.1Q VLANs. Specifically, the firewall supports multiple 802.1Q VLANs on a physical interface and the ability to receive and send 802.1Q-tagged packets. VLANs are not supported on the PIX Firewall 501 and PIX Firewall 506/PIX 506E.

The PIX Firewall does not currently support executable commands for LAN trunks (the physical and logical connection between two switches) because the firewall does not negotiate or participate in any bridging protocols. The firewall only displays the VLANs on the LAN trunk. The state of the LAN trunk is considered the same as the state of the physical interface by the firewall. If the link is up on the physical Ethernet, then the firewall considers the trunk as up as soon as a VLAN has been assigned or configured for it. Additionally, the VLAN is active as soon as you assign or configure a VLAN ID on the physical Ethernet interface of the firewall.

Physical interfaces are one per network interface card (NIC), in place at boot time and not removable. Logical interfaces can be many-to-one for each NIC, are created at runtime, and can be removed through software reconfiguration. A minimum of two physical interfaces is required for all PIX Firewall platforms to support VLANs.



Create Logical and Physical Interfaces

```
pix1(config)# interface ethernet3 100full
pix1(config)# interface ethernet3 vlan10 physical
pix1(config)# interface ethernet3 vlan20 logical
pix1(config)# interface ethernet3 vlan30 logical
```

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—13-5

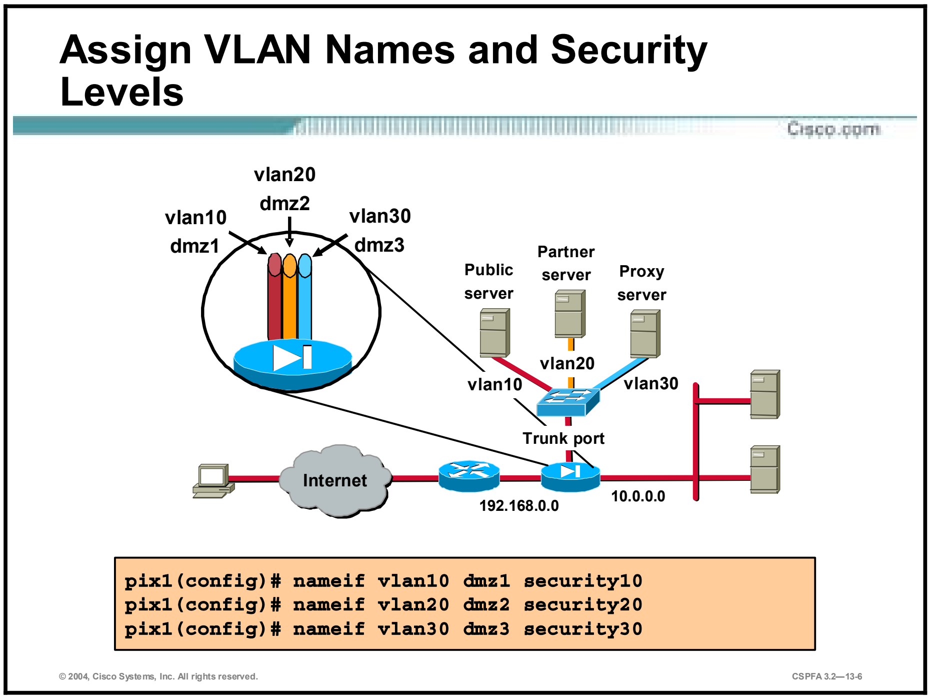
Assign VLANs to physical interfaces on the PIX Firewall with the **interface hardware_id vlan_id physical** command. With the **interface hardware_id vlan_id logical** command, configure multiple logical interfaces on a single physical interface and assign each logical interface to a specific VLAN using its VLAN ID. (It may be helpful to think of VLAN IDs as something you apply to interfaces.)

In the example in the figure, PIX Firewall Ethernet 3 interface has three VLANs—one physical VLAN, vlan10, and two logical VLANS, vlan20 and vlan30.

The syntax for the interface command is as follows:

interface *hardware_id vlan_id* [logical | physical][shutdown]

hardware_id	Identifies the network interface type.
vlan_id	The VLAN identifier. For example: vlan10, vlan20, and so on.
logical	Creates a logical interface and applies the VLAN.
physical	Applies VLAN to physical interface.
shutdown	Disables an interface.



Assign VLAN Names and Security Levels

```
pix1(config)# nameif vlan10 dmz1 security10
pix1(config)# nameif vlan20 dmz2 security20
pix1(config)# nameif vlan30 dmz3 security30
```

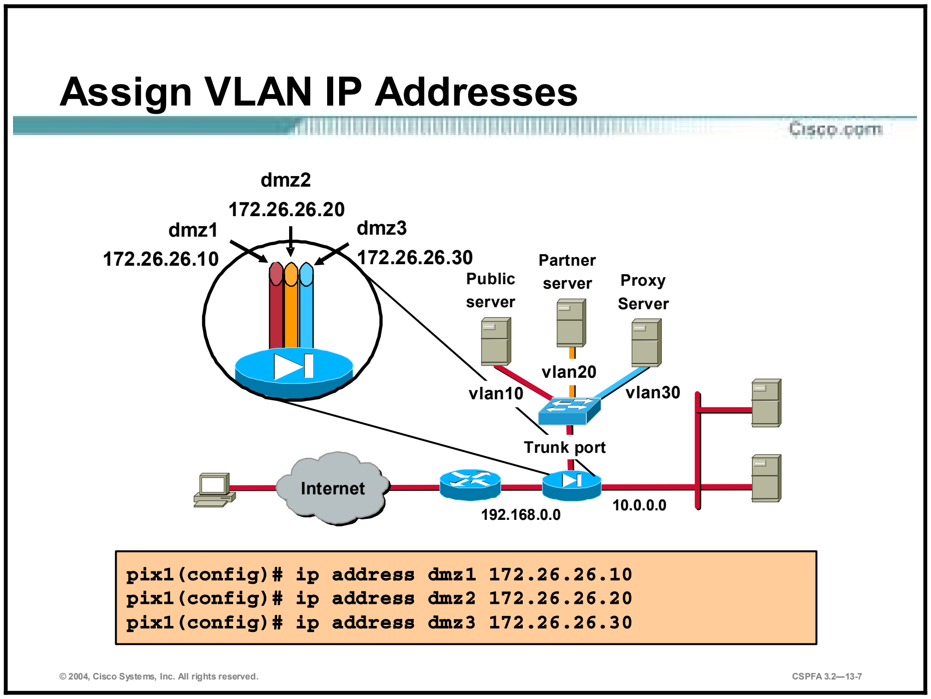
CSPFA 3.2—13-6

With the **nameif** command, the administrator defines a name and security level for each VLAN. In the example in the figure, vlan10 is named dmz1, with a security level of 10. VLAN 20 is named dmz2, with a security level of 20. Vlan30 is named dmz3, with a security level of 30.

The syntax for the **nameif** command is as follows:

nameif *{hardware_id* | *vlan_*id} *if_name security_level*

hardware_id	The hardware name for the network interface that specifies the interface's slot location on the PIX Firewall motherboard.
vlan_id	The VLAN identifier. For example: vlan10, vlan20, and so on.
if_name	A name for the internal or external network interface.
security_level	Enter 0 for the outside network or 100 for the inside network. Perimeter interfaces can use any number between 1 and 99.



Assign VLAN IP Addresses

```
pix1(config)# ip address dmz1 172.26.26.10
pix1(config)# ip address dmz2 172.26.26.20
pix1(config)# ip address dmz3 172.26.26.30
```

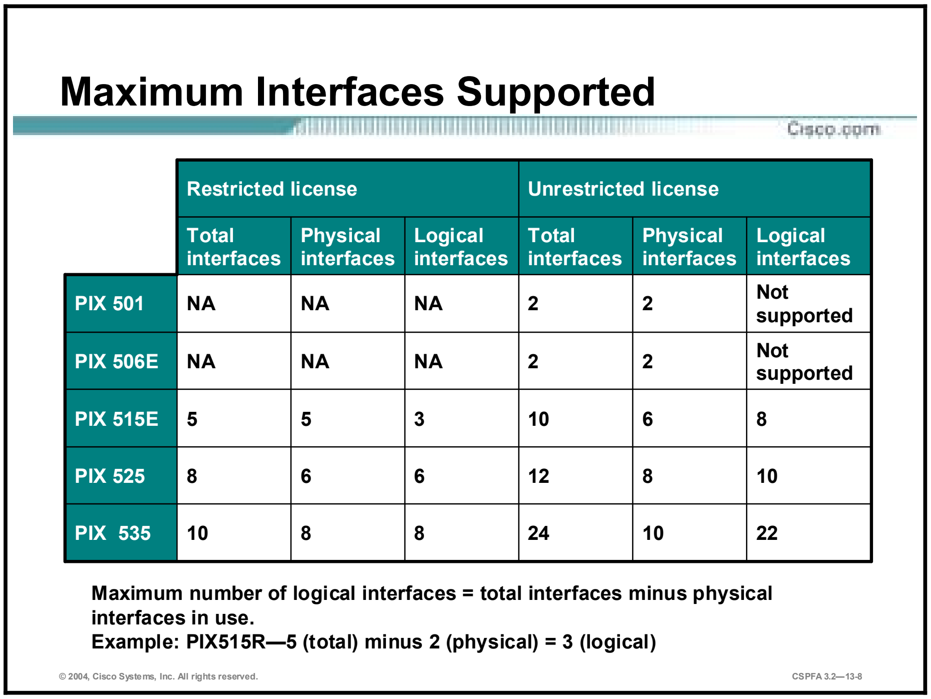
CSPFA 3.2—13-7

Use the **ip address** command to assign IP addresses to the VLANs. In the example in the figure, dmz1, dmz2, and dmz3 are assigned the following IP addresses: 172.26.26.10, 172.26.26.20, and 172.26.26.30.

The syntax for the **ip address** command is as follows:

ip address ***if_name ip_address***

if_name	A name for the internal or external network interface.
ip_address	PIX Firewall unit’s network interface IP address. Each interface IP address must be unique. Two or more interfaces must not be given the same IP address or be given IP addresses that are on the same IP network.



Maximum Interfaces Supported

	Restricted license			Unrestricted license		
	Total interfaces	Physical interfaces	Logical interfaces	Total interfaces	Physical interfaces	Logical interfaces
PIX 501	NA	NA	NA	2	2	Not supported
PIX 506E	NA	NA	NA	2	2	Not supported
PIX 515E	5	5	3	10	6	8
PIX 525	8	6	6	12	8	10
PIX 535	10	8	8	24	10	22

Maximum number of logical interfaces = total interfaces minus physical interfaces in use.
Example: PIX515R—5 (total) minus 2 (physical) = 3 (logical)

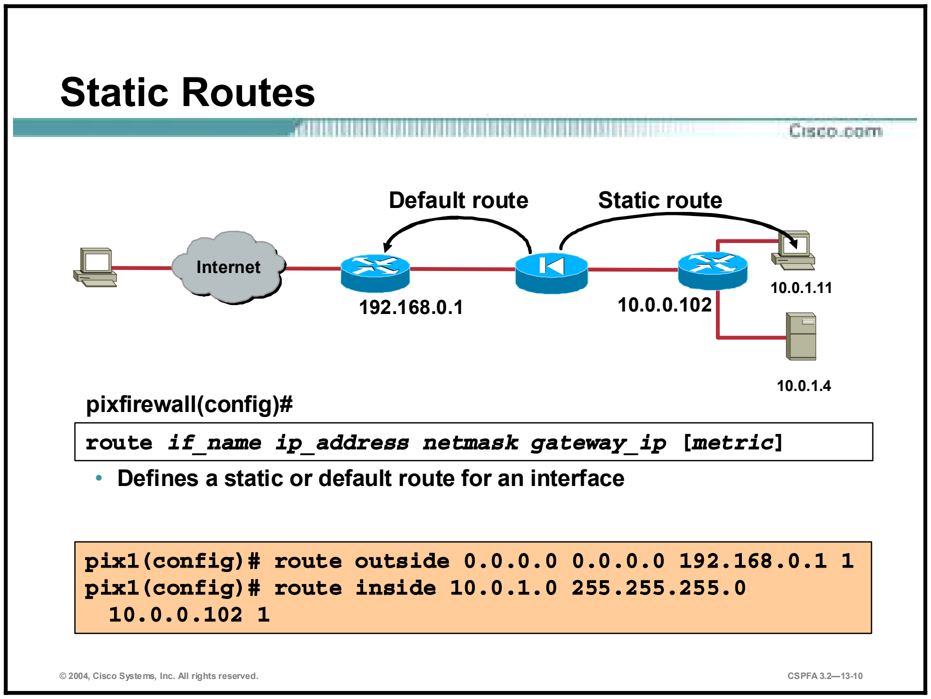
CSPFA 3.2—13-8

VLANs are not supported on the PIX Firewall 501 and PIX Firewall 506/PIX 506E. The number of logical interfaces that you can configure on the other PIX Firewall appliances varies by platform and license type. For example, a PIX Firewall 515E with a restricted license supports up to three logical interfaces. A PIX Firewall 515 with an unrestricted license supports up to eight logical interfaces.

In the example in the figure, the chart defines the maximum supported interfaces of the PIX Firewall family. The type of license is given across the top of the chart. The PIX Firewall appliances are given at the left side of the chart. The maximum number of logical interfaces configurable on a PIX Firewall is determined by taking the total number of interfaces available on a specific PIX Firewall model and license type, then subtracting the number of currently configured physical interfaces. The result is the total number of available logical interfaces. For example, if a PIX Firewall 525R has four physical interfaces, the available number of logical interfaces is eight: The total number of interfaces, 12, minus the number of physical interfaces, 4, equals available number of logical interfaces, 8.

Static and Dynamic Routing

This topic explains the routing capabilities of the PIX Firewall.



Static Routes

pixfirewall(config)#

```
route if_name ip_address netmask gateway_ip [metric]
```

- Defines a static or default route for an interface

```
pix1(config)# route outside 0.0.0.0 0.0.0.0 192.168.0.1 1
pix1(config)# route inside 10.0.1.0 255.255.255.0
  10.0.0.102 1
```

CSPFA 3.2—13-10

Although the PIX Firewall is not a router, it does have certain routing capabilities. You can use the **route** command explained in the previous topic to create static routes for accessing networks outside a router on any interface. The effect of a static route is like stating "to send a packet to the specified network, give it to this router." In the example in the figure, the PIX Firewall sends all packets destined to the 10.1.1.0 network to the router at 10.0.0.3. All traffic for which the PIX Firewall has no route is sent to 192.168.0.1, the gateway in the default route. To enter a default route, set the ip_address and netmask arguments to 0.0.0.0, or the shortened form of 0. Only one default route can be used.

All routes entered using the **route** command are stored in the configuration when it is saved. They can be displayed by using the **show route** command, and you can clear most routes by using the **clear route** command. The only routes not removed with the **clear route** command are those that show the keyword CONNECT when you issue the **show route** command. These are routes that the PIX Firewall automatically creates in its routing table when you enter an IP address for a PIX Firewall interface. A route created in this manner is a route to the network directly connected to that interface. The figure shows examples of these automatically created routes.

Although the gateway argument in the **route** command usually specifies the IP address of the gateway router, the next hop address for this route, you can also specify one of the PIX Firewall's interfaces. When a **route** command statement uses the IP address of one of the PIX Firewall's interfaces as the gateway IP address, the PIX Firewall broadcasts an Address Resolution Protocol (ARP) request for the MAC address corresponding to the destination IP address in the packet instead of broadcasting the ARP request for the MAC address corresponding to the gateway IP address.

The following steps show how the PIX Firewall handles routing in this situation:

Step 1 The PIX Firewall receives a packet from the inside interface destined to IP address X.

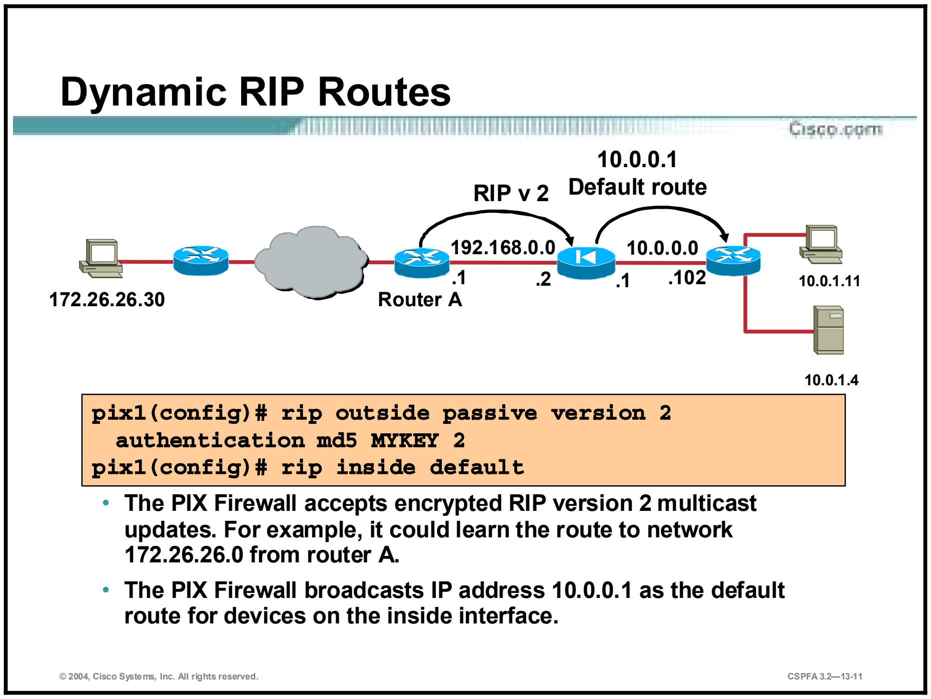
Step 2 Because a default route is set to itself, the PIX Firewall sends out an ARP for address X.

Step 3 Any Cisco router on the outside interface LAN that has a route to address X replies back to the PIX Firewall with its own MAC address as the next hop. Cisco IOS software has proxy ARP enabled by default.

Step 4 The PIX Firewall sends the packet to router.

Step 5 The PIX Firewall adds the entry to its ARP cache for IP address X with the MAC address being that of the router.

Note: Do not configure the PIX Firewall with a default route when using the setroute argument of the **ip address dhcp** command. To clear a DHCP default route, use the **clear route static** command.



Dynamic RIP Routes

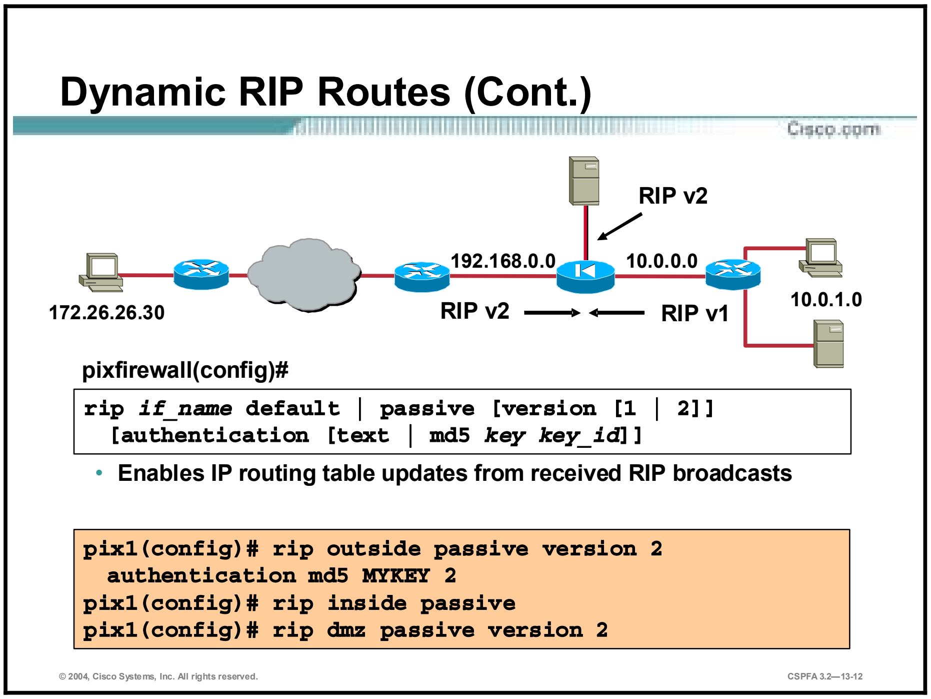
```
pix1(config)# rip outside passive version 2
  authentication md5 MYKEY 2
pix1(config)# rip inside default
```

- The PIX Firewall accepts encrypted RIP version 2 multicast updates. For example, it could learn the route to network 172.26.26.0 from router A.
- The PIX Firewall broadcasts IP address 10.0.0.1 as the default route for devices on the inside interface.

CSPFA 3.2—13-11

Another way to build the PIX Firewall’s routing table is by enabling the Routing Information Protocol (RIP) with the **rip** command. You can configure the PIX Firewall to learn routes dynamically from RIP version 1 or RIP version 2 broadcasts. Although the PIX Firewall uses the dynamically learned routes itself to forward traffic to the appropriate destinations, it does not propagate learned routes to other devices. The PIX Firewall cannot pass RIP updates between interfaces. It can, however, advertise one of its interfaces as a default route.

The figure shows the PIX Firewall learning routes from a router on its outside interface and broadcasting a default route on its inside interface. Message Digest 5 (MD5) authentication is used on the outside interface to enable the PIX Firewall to accept the encrypted RIP updates. Both the PIX Firewall and router A are configured with the encryption key MYKEY and its key_id value of 2.



Dynamic RIP Routes (Cont.)

pixfirewall(config)#

```
rip if_name default | passive [version [1 | 2]]
  [authentication [text | md5 key key_id]]
```

- Enables IP routing table updates from received RIP broadcasts

```
pix1(config)# rip outside passive version 2
  authentication md5 MYKEY 2
pix1(config)# rip inside passive
pix1(config)# rip dmz passive version 2
```

CSPFA 3.2—13-12

Use the **rip** command to configure the PIX Firewall to learn routes dynamically from RIP version 1 or RIP version 2 broadcasts. When RIP version 2 is configured in passive mode, the PIX Firewall accepts RIP version 2 multicast updates with an IP destination of 224.0.0.9. For the RIP version 2 default mode, the PIX Firewall transmits default route updates using an IP destination of 224.0.0.9. Configuring RIP version 2 registers the multicast address 224.0.0.9 on the interface specified in the command so that the PIX Firewall can accept multicast RIP version 2 updates. When the RIP version 2 commands for an interface are removed, the multicast address is unregistered from the interface card.

If you specify RIP version 2, you can also encrypt RIP updates using MD5 encryption. Ensure that the key and key_id values are the same as in use on any device in your network that makes RIP version 2 updates.

IP routing table updates are enabled by default. Use the **no rip** command to disable the PIX Firewall IP routing table updates. The **clear rip** command removes all the **rip** commands from the configuration.

Note: Static routes override dynamic routes.

The example in the figure combines RIP version 1 and 2 commands that enable the following:

- Version 2 passive RIP using MD5 authentication on the outside interface to encrypt the key used by the PIX Firewall and other RIP peers, such as routers
- Broadcasting a default route on the outside interface using MD5 authentication
- Version 1 passive RIP listening on the inside interface
- Version 2 passive RIP listening on the DMZ interface

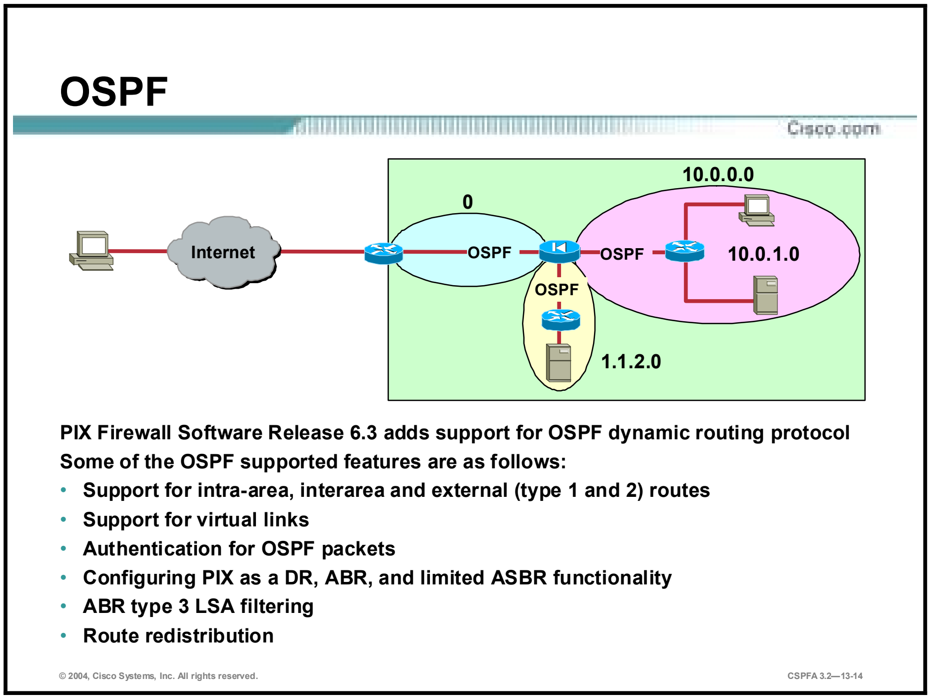
The syntax for the **rip** command is as follows:

rip *if_name* default | passive [version [1 | 2]] [authentication [text | md5 *key key_id*]]

if_name	The internal or external network interface name.
default	Broadcasts a default route on the interface.
passive	Enables passive RIP on the interface. The PIX Firewall listens for RIP routing broadcasts and uses that information to populate its routing tables.
version	The version of RIP. Use version 2 for RIP update encryption. Use version 1 to provide backward compatibility with the older version.
authentication	Enables RIP version 2 authentication.
text	Sends RIP updates as clear text.
md5	Sends RIP updates using MD5 authentication.
key	The key to encrypt RIP updates. This value must be the same on the routers and any other device that provides RIP version 2 updates. The key is a text string of up to 16 characters in length.
key_id	The key identification value. The key_id can be a number from 1 to 255. Use the same key_id in use on the routers and any other device that provides RIP version 2 updates.

OSPF

This topic provides a basic explanation of the Open Shortest Path First (OSPF) capabilities of the PIX Firewall.



OSPF

PIX Firewall Software Release 6.3 adds support for OSPF dynamic routing protocol

Some of the OSPF supported features are as follows:

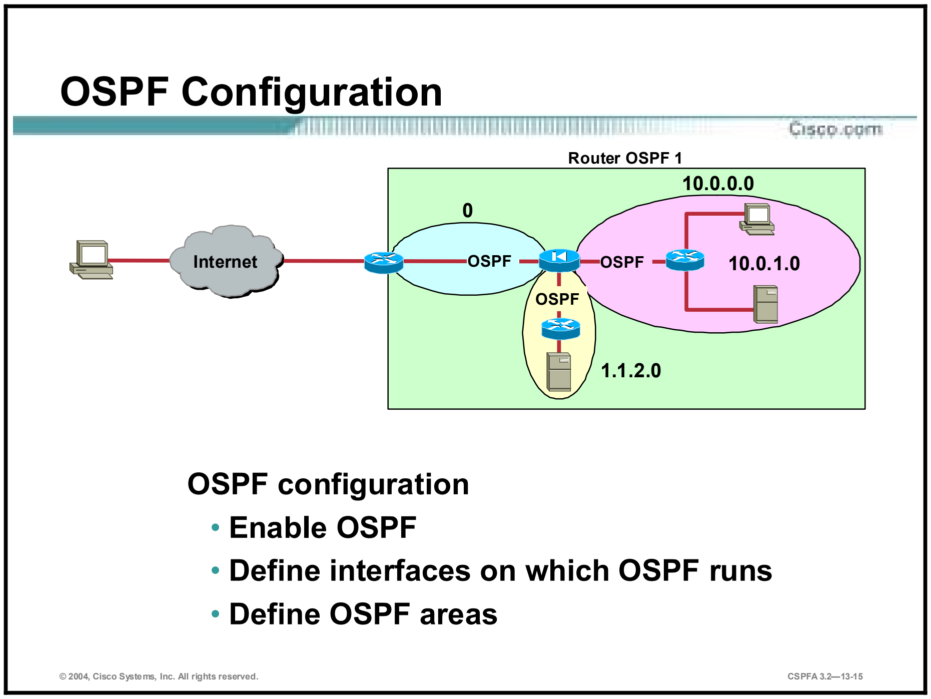
- **Support for intra-area, interarea and external (type 1 and 2) routes**
- **Support for virtual links**
- **Authentication for OSPF packets**
- **Configuring PIX as a DR, ABR, and limited ASBR functionality**
- **ABR type 3 LSA filtering**
- **Route redistribution**

CSPFA 3.2—13-14

Prior to PIX Firewall Software Version 6.3, the PIX Firewall supported static routing and limited dynamic routing through passive RIP. PIX Firewall Software Version 6.3 introduces support for dynamic routing using the OSPF routing protocol. OSPF is widely deployed in large internetworks because of its efficient use of network bandwidth and its rapid convergence after changes in topology. Some of the PIX Firewall OSPF supported features are:

- Support for intra-area, interarea and external (type 1 and 2) routes
- Support for virtual links
- Authentication for OSPF packets
- Configuring the PIX Firewall as a designated router (DR), area border router (ABR) and limited autonomous system boundary router (ASBR)
- Support for stub and not so stubby areas (NSSAs)
- ABR type 3 link-state advertisement (LSA) filtering
- Route redistribution

Note: OSPF routing is not supported on the PIX Firewall 501. OSPF and RIP cannot be enabled simultaneously on the PIX Firewall.



OSPF Configuration

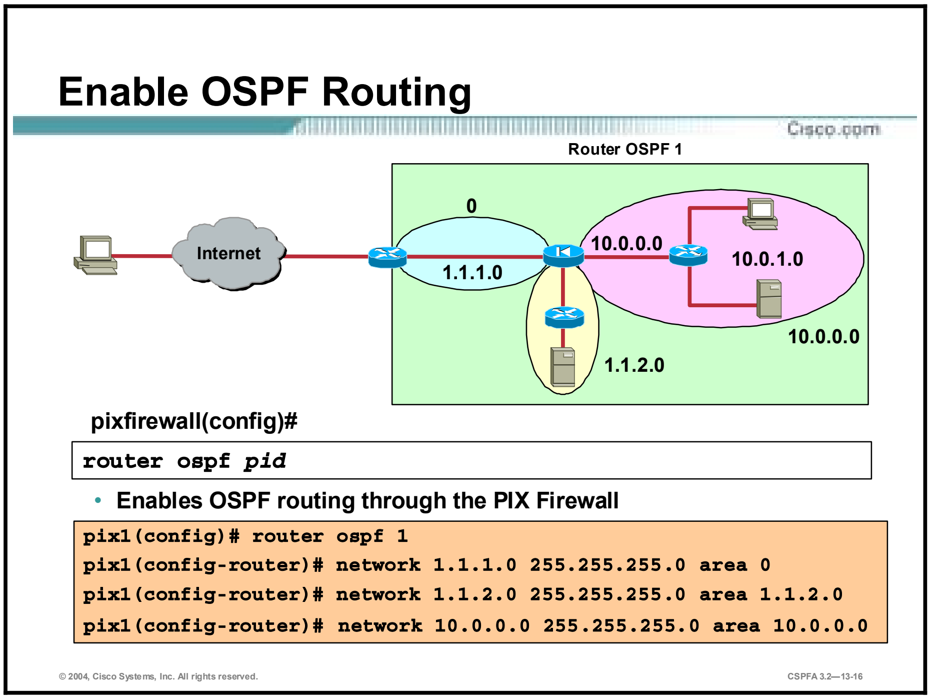
OSPF configuration

- Enable OSPF
- Define interfaces on which OSPF runs
- Define OSPF areas

CSPFA 3.2—13-15

To configure OSPF on the PIX Firewall requires the administrator to do the following:

- Enable OSPF
- Define the PIX Firewall interfaces on which OSPF runs
- Define OSPF areas



Enable OSPF Routing

pixfirewall(config)#

```
router ospf pid
```

- Enables OSPF routing through the PIX Firewall

```
pix1(config)# router ospf 1
pix1(config-router)# network 1.1.1.0 255.255.255.0 area 0
pix1(config-router)# network 1.1.2.0 255.255.255.0 area 1.1.2.0
pix1(config-router)# network 10.0.0.0 255.255.255.0 area 10.0.0.0
```

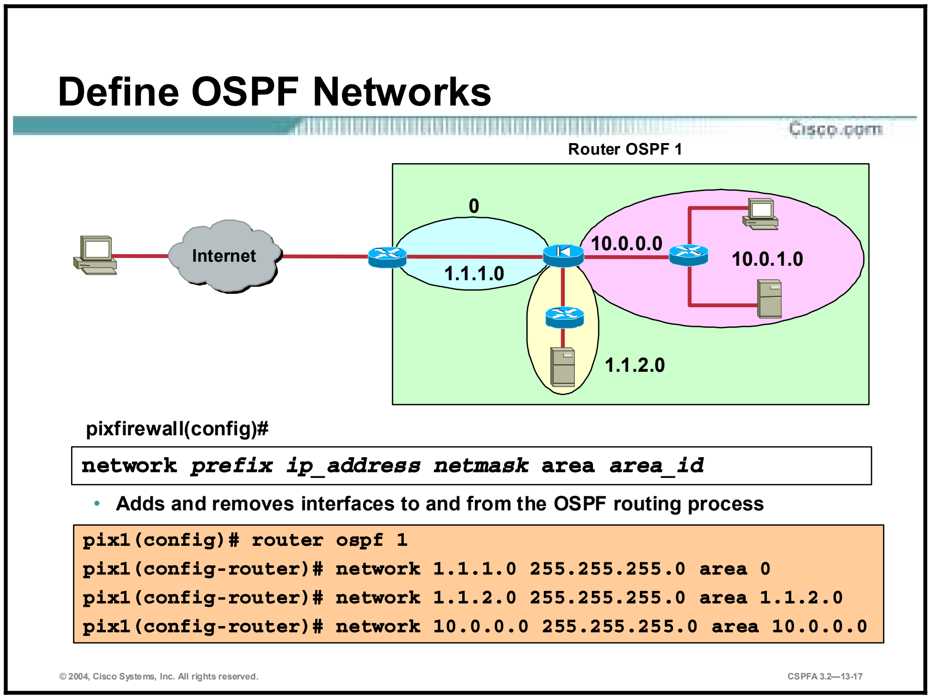
CSPFA 3.2—13-16

To enable OSPF routing, use the **router ospf** command. The syntax for the **router ospf** command is as follows:

router ospf ***pid***

pid	Internally used identification parameter for an OSPF routing process. You assign it locally on the firewall, and its value can be from 1 to 65535. A unique value must be assigned for each OSPF routing process. PIX Firewall Software Version 6.3 supports a maximum of two OSPF processes.

The PIX Firewall can be configured for one or two processes, or OSPF routing domains. If the PIX Firewall is functioning as an ABR and it is configured for one process, the PIX Firewall will pass type 3 LSA between defined OSPF areas. In the example in the figure, the PIX Firewall is configured for one OSPF process, OSPF 1. The PIX Firewall is operating as an ABR between area 0 and the other two defined areas, area 1.1.2.0 and area 10.0.0.0. The PIX Firewall passes type 3 LSA between the areas.



Define OSPF Networks

pixfirewall(config)#

```
network prefix ip_address netmask area area_id
```

- Adds and removes interfaces to and from the OSPF routing process

```
pix1(config)# router ospf 1
pix1(config-router)# network 1.1.1.0 255.255.255.0 area 0
pix1(config-router)# network 1.1.2.0 255.255.255.0 area 1.1.2.0
pix1(config-router)# network 10.0.0.0 255.255.255.0 area 10.0.0.0
```

CSPFA 3.2—13-17

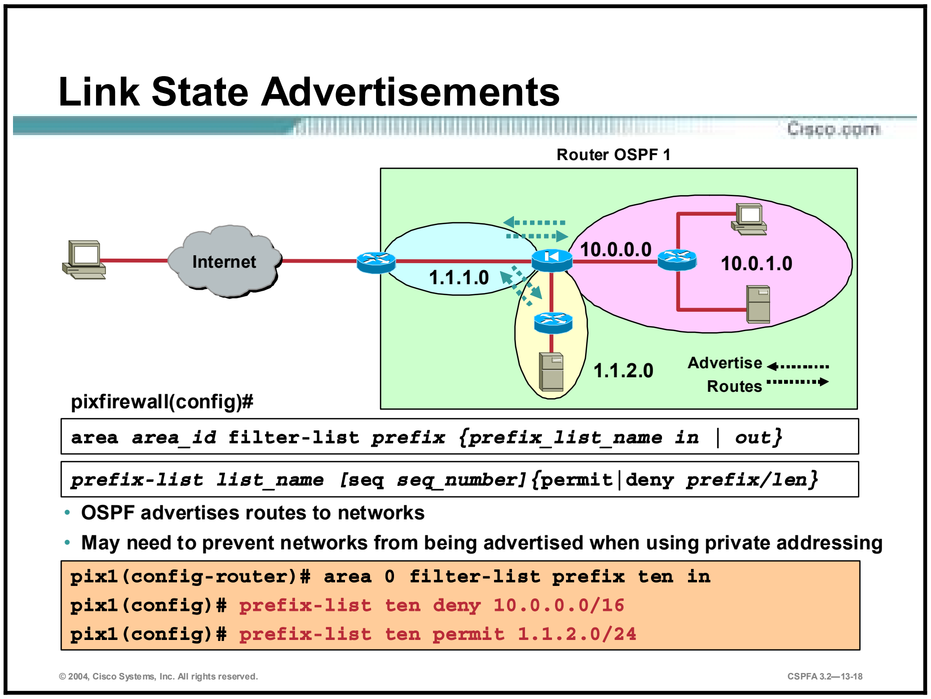
To define the interfaces on which OSPF runs and the area ID for those interfaces, use the **network prefix ip_address netmask area area_id** subcommand.

The syntax for the **network** command is as follows:

network *prefix ip_address netmask* area *area_id*

network	Adds/removes interfaces to/from the OSPF routing process.
area *area_id*	The ID of the area that is to be associated with the OSPF address range.

In the example in the figure, the three PIX Firewall interfaces are configured for OSPF. The outside interface, network 1.1.1.0, is configured as area 0. The Demilitarized Zone (DMZ) interface, network 1.1.2.0, is configured as network 1.1.2.0. The inside interface, network 10.0.0.0, is configured as area 10.0.0.0. LSA type 3 advertisements pass between the three interfaces.



Link State Advertisements

pixfirewall(config)#

```
area area_id filter-list prefix {prefix_list_name in | out}
```

```
prefix-list list_name [seq seq_number]{permit|deny prefix/len}
```

- OSPF advertises routes to networks
- May need to prevent networks from being advertised when using private addressing

```
pix1(config-router)# area 0 filter-list prefix ten in
pix1(config)# prefix-list ten deny 10.0.0.0/16
pix1(config)# prefix-list ten permit 1.1.2.0/24
```

CSPFA 3.2—13-18

When the PIX Firewall is defined with one OSPF routing process, routing advertisements are passed between areas. However, private network addresses can also be passed between areas, private to public. To prevent the PIX Firewall from passing LSA type 3 advertisements between areas, configure the **prefix-list** and **area** commands.

The **prefix-list** commands are ABR type 3 LSA filtering commands. ABR type 3 LSA filtering extends the capability of an ABR that is running OSPF to filter type 3 LSAs between different OSPF areas. This filtering is based on a prefix list defined by you, using the **prefix-list** commands. Once configured, only the specified prefixes are sent from one area to another area, and all other prefixes are restricted to their OSPF area. This type of area filtering can be applied to traffic going into or coming out of an OSPF area or to both the incoming and outgoing traffic for that area. Use the **area** command to apply the prefix list to an area.

To create an entry in a prefix list, use the **prefix-list list_name** command. The syntax for the network command is as follows:

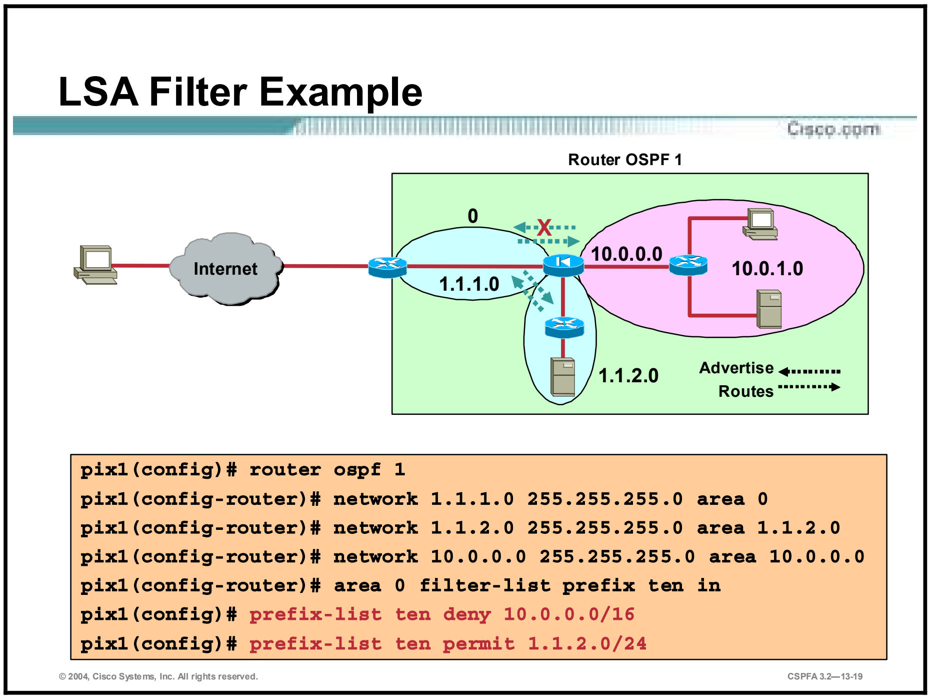
prefix-list *list_name* [seq *seq_number*]{permit | deny *prefix* / *le*n}

deny	Denies access for a matching condition.
permit	Permits access for a matching condition.
seq *seq_number*	Specifies the sequence number for the prefix list entry, from 1 to 4294967295. However, the list_name and seq_number together must be less than 64 characters combined.

Apply the prefix list filter to an area. The syntax for the **area** command is as follows:

area ***area_id*** **filter-list prefix** ***{prefix_list_name*** **in | out}**

area *area_id*	When used in the context of a prefix list, *area_id* is the identifier of the area on which filtering is configured.
prefix_list_name	Name of a prefix list.
in	Applies the configured prefix list to prefixes advertised inbound to the specified area.
out	Applies the configured prefix list to prefixes advertised outbound from the specified area.



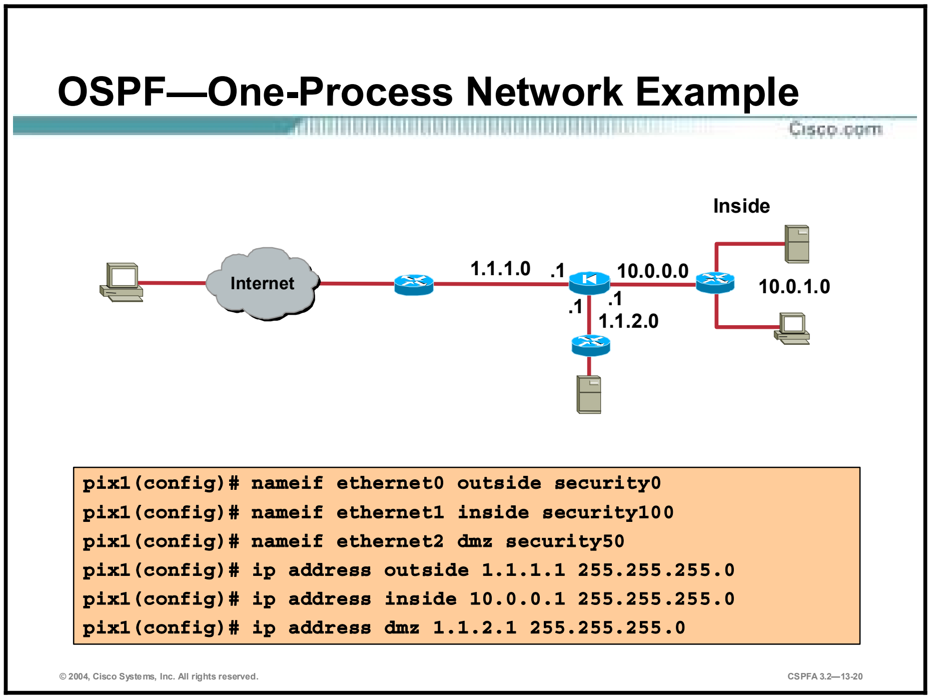
LSA Filter Example

```
pix1(config)# router ospf 1
pix1(config-router)# network 1.1.1.0 255.255.255.0 area 0
pix1(config-router)# network 1.1.2.0 255.255.255.0 area 1.1.2.0
pix1(config-router)# network 10.0.0.0 255.255.255.0 area 10.0.0.0
pix1(config-router)# area 0 filter-list prefix ten in
pix1(config)# prefix-list ten deny 10.0.0.0/16
pix1(config)# prefix-list ten permit 1.1.2.0/24
```

CSPFA 3.2—13-19

In the example in the figure, the PIX Firewall is defined with one OSPF process. The PIX Firewall in the figure by definition forwards LSA type 3 advertisements between areas. The LSA type advertisements between the outside and DMZ interfaces are permitted.

The LSA type 3 advertisements between the inside interface and the outside interface are an issue. The administrator does not want the inside private network addresses advertised outside the private network. To filter the addresses, the administrator defines two prefix list entries, prefix-list ten deny 10.0.0.0/16 and prefix-list ten permit 1.1.2.0/24. The first prefix list filters denies any LSA type advertisements matching the 10.0.0.0/16 network address. The second prefix list entry permits LSA type 3 advertisements matching 1.1.2.0/24 network address. The two prefix lists are applied inbound to area 0. Any LSA type 3 advertisements into area 0 matching 1.1.2.0/24 are permitted. Any LSA type 3 advertisements into area 0 matching 10.0.0.0/16 are denied. The DMZ network is advertised to the outside interface. The private network is not advertised to the outside interface.



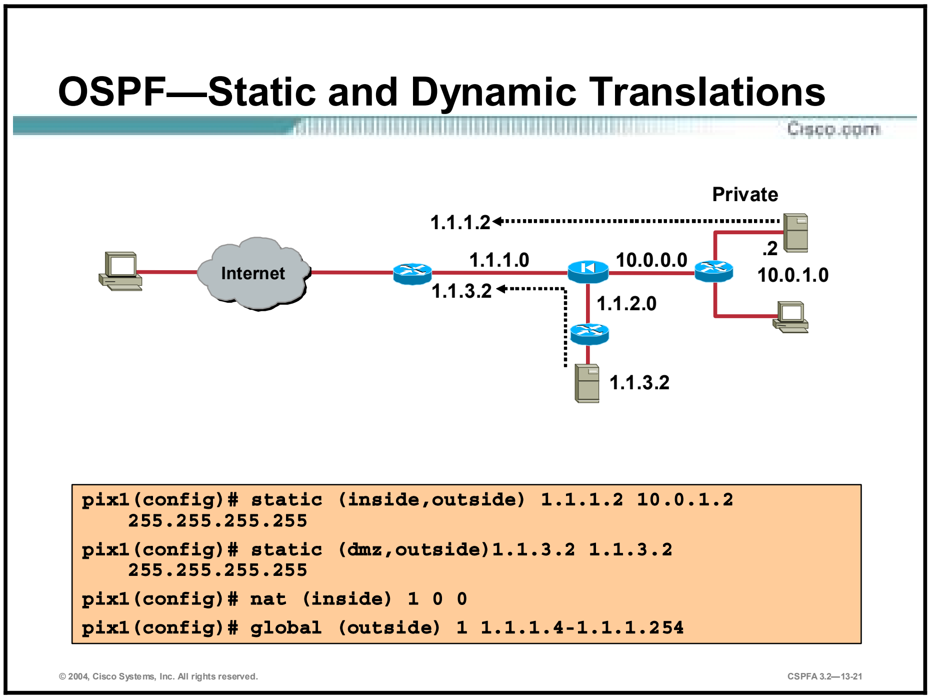
OSPF—One-Process Network Example

```
pix1(config)# nameif ethernet0 outside security0
pix1(config)# nameif ethernet1 inside security100
pix1(config)# nameif ethernet2 dmz security50
pix1(config)# ip address outside 1.1.1.1 255.255.255.0
pix1(config)# ip address inside 10.0.0.1 255.255.255.0
pix1(config)# ip address dmz 1.1.2.1 255.255.255.0
```

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—13-20

The figure shows an example of a PIX Firewall OSPF configuration. There are three interfaces, all using OSPF routing. The interfaces are the outside, dmz, and inside interfaces. In the example in the figure, the interfaces are named and IP addresses assigned.

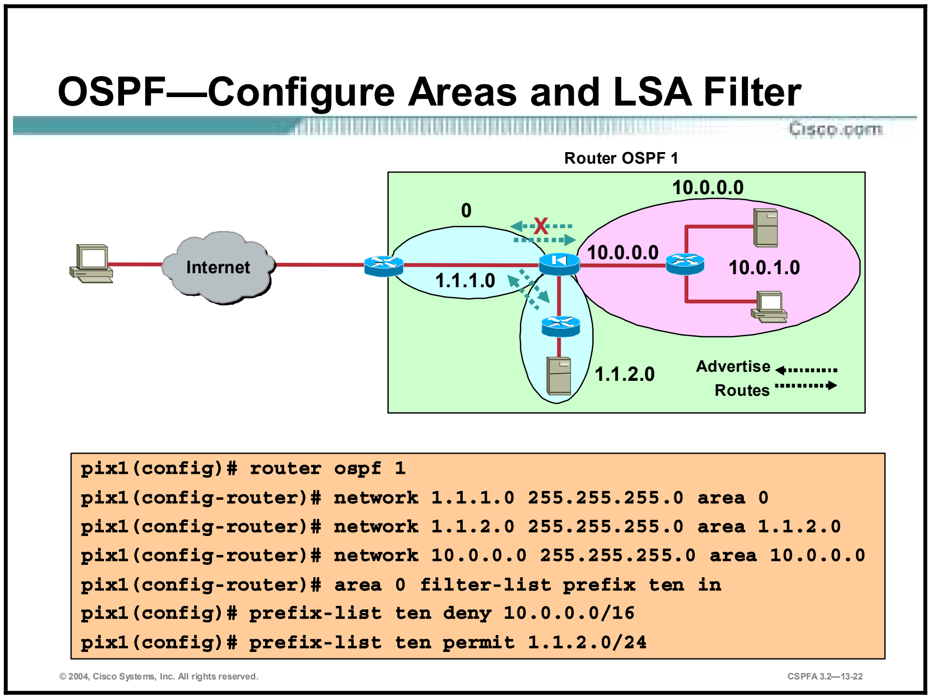


OSPF—Static and Dynamic Translations

```
pix1(config)# static (inside,outside) 1.1.1.2 10.0.1.2
    255.255.255.255
pix1(config)# static (dmz,outside)1.1.3.2 1.1.3.2
    255.255.255.255
pix1(config)# nat (inside) 1 0 0
pix1(config)# global (outside) 1 1.1.1.4-1.1.1.254
```

CSPFA 3.2—13-21

In the example in the figure, static and dynamic translations are defined. From the inside network, the inside server is statically translated to 1.1.1.2. All other inside address are dynamically translated to the IP address range of 1.1.1.4 to 1.1.1.254. The dmz server IP address, 1.1.3.2, is visible on the outside network. It has the same source address on the dmz and outside interfaces. It is not translated.

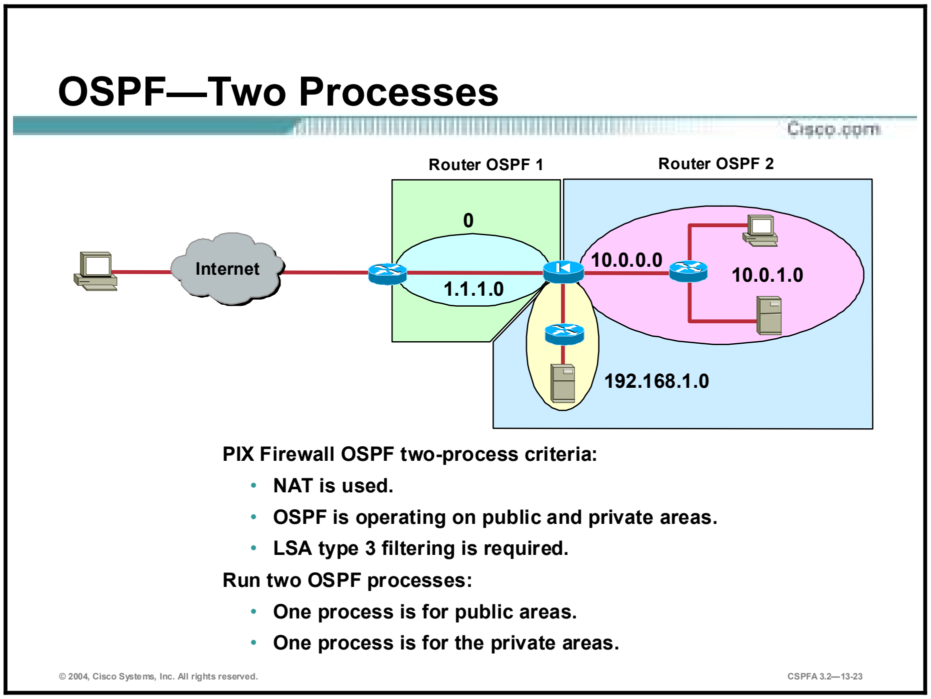


OSPF—Configure Areas and LSA Filter

```
pix1(config)# router ospf 1
pix1(config-router)# network 1.1.1.0 255.255.255.0 area 0
pix1(config-router)# network 1.1.2.0 255.255.255.0 area 1.1.2.0
pix1(config-router)# network 10.0.0.0 255.255.255.0 area 10.0.0.0
pix1(config-router)# area 0 filter-list prefix ten in
pix1(config)# prefix-list ten deny 10.0.0.0/16
pix1(config)# prefix-list ten permit 1.1.2.0/24
```

CSPFA 3.2—13-22

In the example in the figure, the PIX Firewall is defined with one OSPF process. There are three OSPF areas, area 0, area 1.1.2.0, and area 10.0.0.0. LSA type 3 advertisements should pass between the area 0 and area 1.1.2.0. Both areas have public addresses, 1.1.1.0 and 1.1.2.0. LSA type 3 advertisements should be filtered from area 10.0.0.0, private network, to area 0, public network. The administrator does not want the private network addresses advertised on the public network. The **prefix-list** command defines which matching network address LSA type 3 messages are permitted and denied. Network 10.0.0.0/16 LSA type 3 messages are denied. Network 1.1.2.0/24 LSA type 3 messages are permitted. The prefix filter list is applied to the area 0 in the inbound direction. LAS type 3 advertisements from area 10.0.0.0 to area 0 are denied by the PIX Firewall. Inbound LSA type 3 advertisements from area 1.1.2.0 to area 0 are permitted by the PIX Firewall.



OSPF—Two Processes

PIX Firewall OSPF two-process criteria:

- NAT is used.
- OSPF is operating on public and private areas.
- LSA type 3 filtering is required.

Run two OSPF processes:

- One process is for public areas.
- One process is for the private areas.

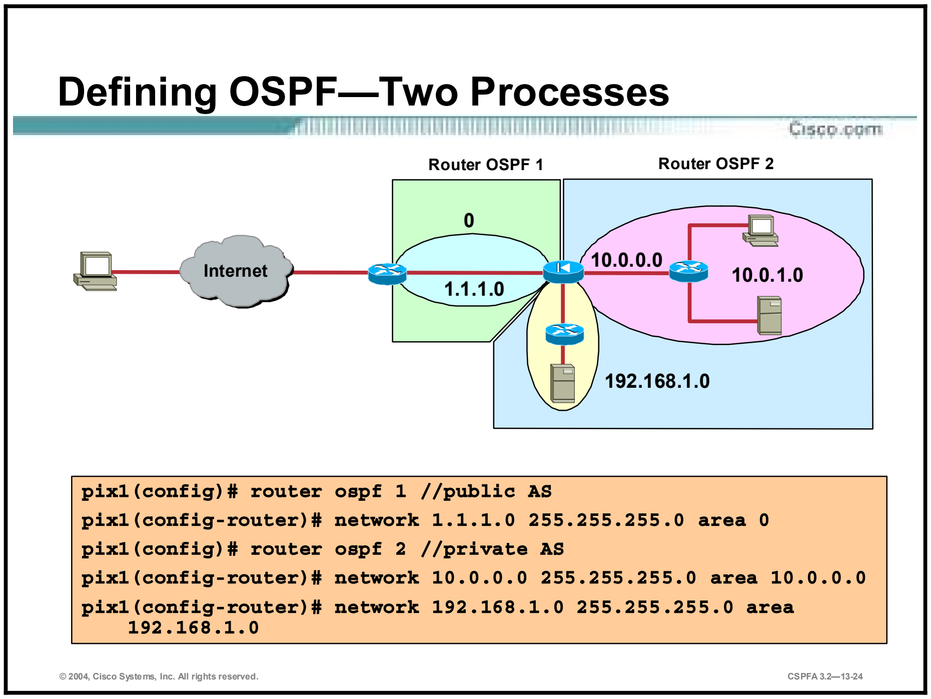
CSPFA 3.2—13-23

Defining a PIX Firewall with two OSPF processes enables the PIX Firewall to pass LSA type 3 advertisements between areas but not between processes. In the example in the figure, there are two defined process areas. OSPF process ID 1 encompasses OSPF area 0. OSPF process ID 2 encompasses areas 10.0.0.0 and 192.168.1.0. With two OSPF processes defined, LSA type 3 advertisements can pass between areas within a process; for example, 192.168.1.0 and 10.0.0.0. LSA type 3 advertisements cannot pass between areas defined by different processes. For example, 10.0.0.0 LSA type 3 advertisements cannot pass to area 0.

It might be advantageous to use two OSPF processes for the following scenario:

- NAT is used.
- OSPF is operating on the public and private interfaces.
- LSA type 3 advertisement filtering is required.

A maximum of two processes can be defined for each PIX Firewall.



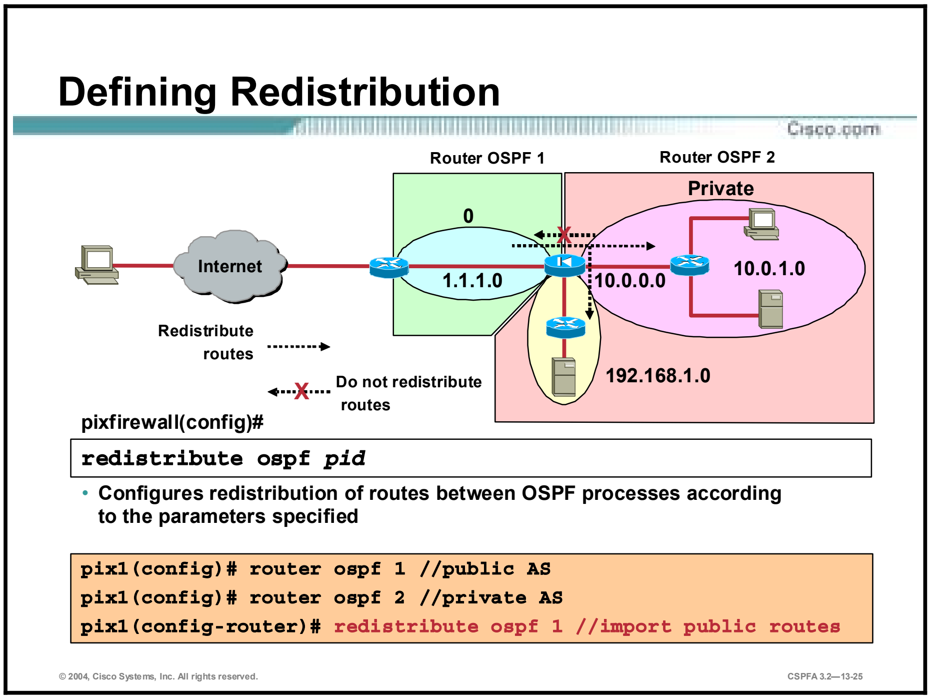
Defining OSPF—Two Processes

```
pix1(config)# router ospf 1 //public AS
pix1(config-router)# network 1.1.1.0 255.255.255.0 area 0
pix1(config)# router ospf 2 //private AS
pix1(config-router)# network 10.0.0.0 255.255.255.0 area 10.0.0.0
pix1(config-router)# network 192.168.1.0 255.255.255.0 area
    192.168.1.0
```

CSPFA 3.2—13-24

To configure two areas, define the router OSPF PID first. Next define the network and areas belonging to the OSPF process ID (PID). In the figure in the example, there are two OSPF PIDs, OSPF 1 and OSPF 2. OSPF 1 is defined first. Network 1.1.1.0/24 is associated with area 0.

OSPF 2 is configured next. Within OSPF 2, there are two networks, 10.0.0.0 and 192.168.1.0. Network 10.0.0.0/24 is associated with OSPF area 10.0.0.0. Network 192.168.1.0/24 is associated with area 192.168.1.0. LSA type 3 advertisements can pass between areas of OSPF 2. LSA type 3 advertisements cannot pass between OSPF 1 and 2.



Defining Redistribution

pixfirewall(config)#

```
redistribute ospf pid
```

- Configures redistribution of routes between OSPF processes according to the parameters specified

```
pix1(config)# router ospf 1 //public AS
pix1(config)# router ospf 2 //private AS
pix1(config-router)# redistribute ospf 1 //import public routes
```

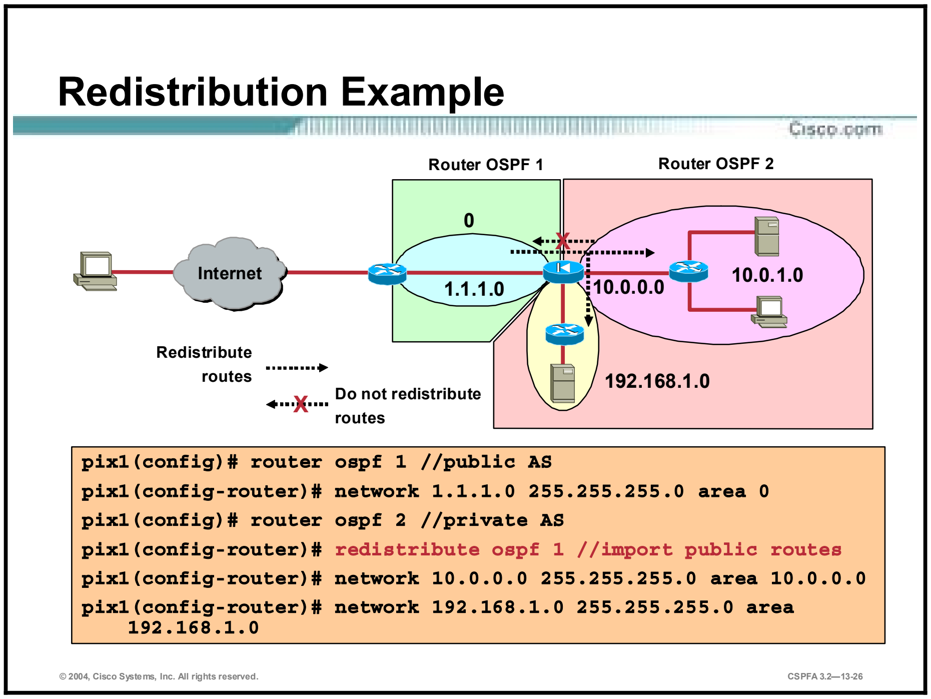
CSPFA 3.2—13-25

It may be advantageous for PID 2 to receive LSA type 3 advertisements to update its OSPF routing table. To redistribute routes from one OSPF routing domain to another, use the **redistribute** command.

The syntax for the command is as follows:

redistribute ospf ***pid***

redistribute	Configures redistribution between OSPF processes according to the parameters specified.
pid	Internally used identification parameter for an OSPF routing process. You assign it locally on the firewall, and the value can be from 1 to 65535. A unique value must be assigned for each OSPF routing process. PIX Firewall Software Version 6.3 supports a maximum of two OSPF processes.

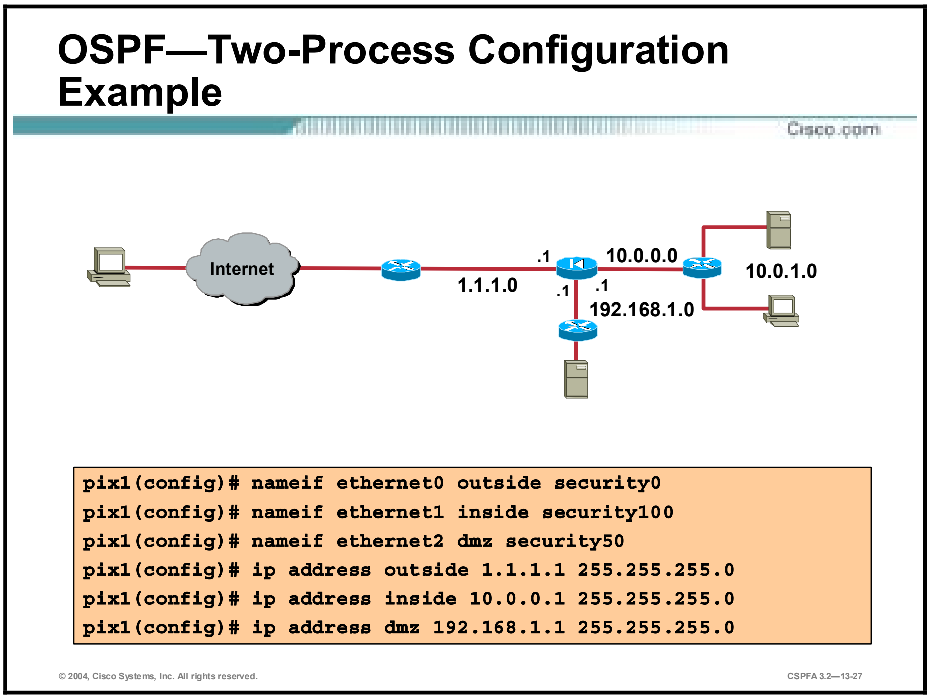


Redistribution Example

```
pix1(config)# router ospf 1 //public AS
pix1(config-router)# network 1.1.1.0 255.255.255.0 area 0
pix1(config)# router ospf 2 //private AS
pix1(config-router)# redistribute ospf 1 //import public routes
pix1(config-router)# network 10.0.0.0 255.255.255.0 area 10.0.0.0
pix1(config-router)# network 192.168.1.0 255.255.255.0 area
    192.168.1.0
```

CSPFA 3.2—13-26

In the example in the figure, there are two OSPF routing domains, OSPF 1 and OSPF 2. By default, OSPF routes are not distributed between OSPF 1 and OSPF 2. To distribute routes from OSPF 1 to OSPF 2, the administrator enters the **redistribute ospf 1** command. This command enables the PIX Firewall to redistribute routes one way from OSPF 1 to OSPF 2.

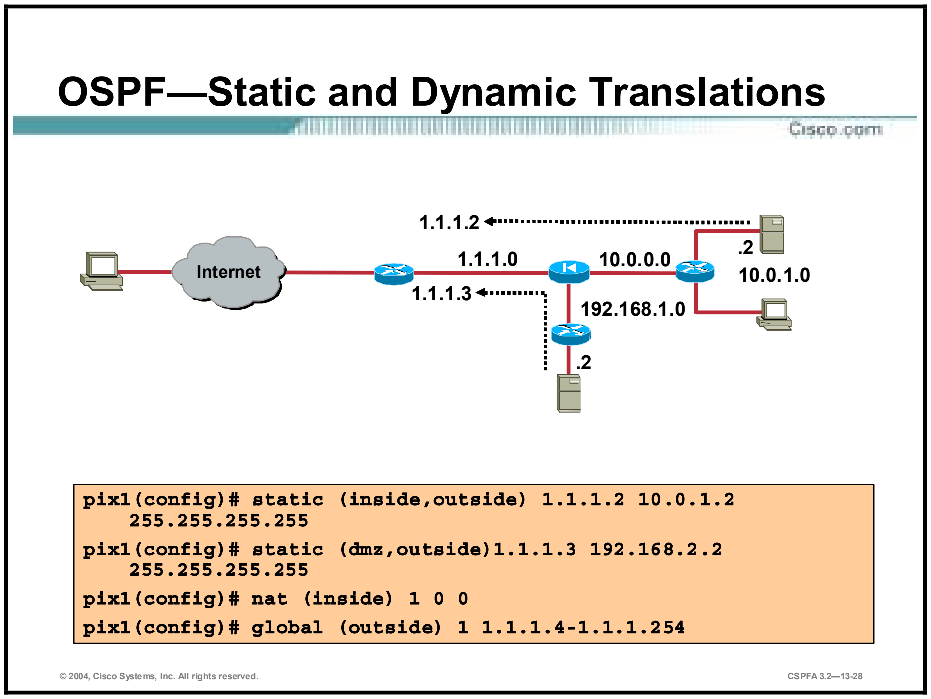


OSPF—Two-Process Configuration Example

```
pix1(config)# nameif ethernet0 outside security0
pix1(config)# nameif ethernet1 inside security100
pix1(config)# nameif ethernet2 dmz security50
pix1(config)# ip address outside 1.1.1.1 255.255.255.0
pix1(config)# ip address inside 10.0.0.1 255.255.255.0
pix1(config)# ip address dmz 192.168.1.1 255.255.255.0
```

CSPFA 3.2—13-27

The figure shows an example of a PIX Firewall configuration. There are three interfaces, all using OSPF routing. The interfaces are the outside, dmz, and inside interfaces. In the example in the figure, the interfaces are named and IP addresses assigned.

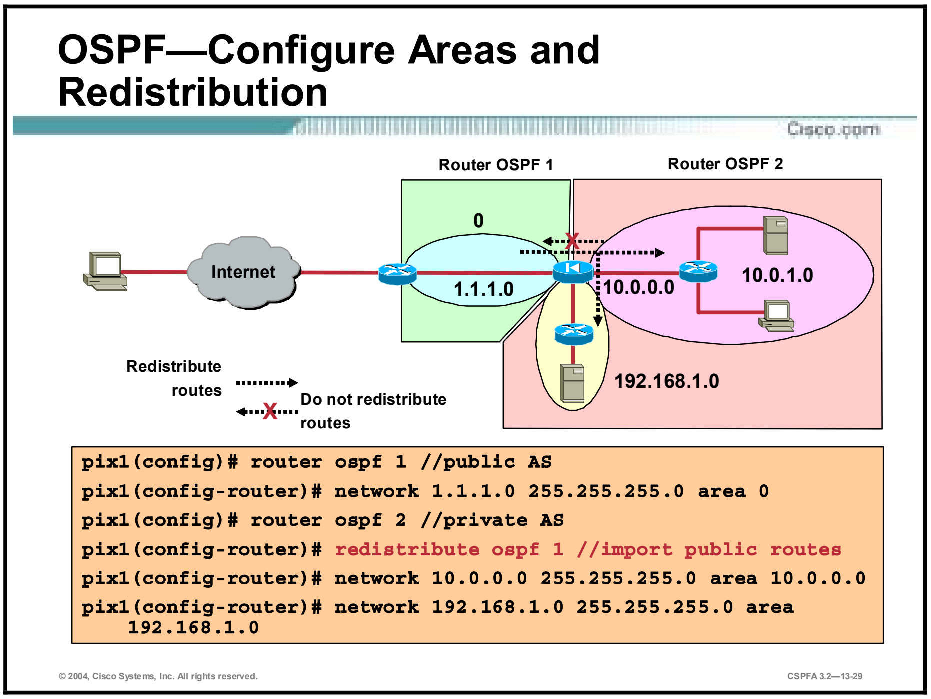


OSPF—Static and Dynamic Translations

```
pix1(config)# static (inside,outside) 1.1.1.2 10.0.1.2
    255.255.255.255
pix1(config)# static (dmz,outside)1.1.1.3 192.168.2.2
    255.255.255.255
pix1(config)# nat (inside) 1 0 0
pix1(config)# global (outside) 1 1.1.1.4-1.1.1.254
```

CSPFA 3.2—13-28

In the example in the figure, static and dynamic translations are defined. From the inside network, the inside server is statically translated to 1.1.1.2. All other inside address are dynamically translated to the IP address range of 1.1.1.4 to 1.1.1.254. The dmz server IP address is statically translated to 1.1.1.3.



OSPF—Configure Areas and Redistribution

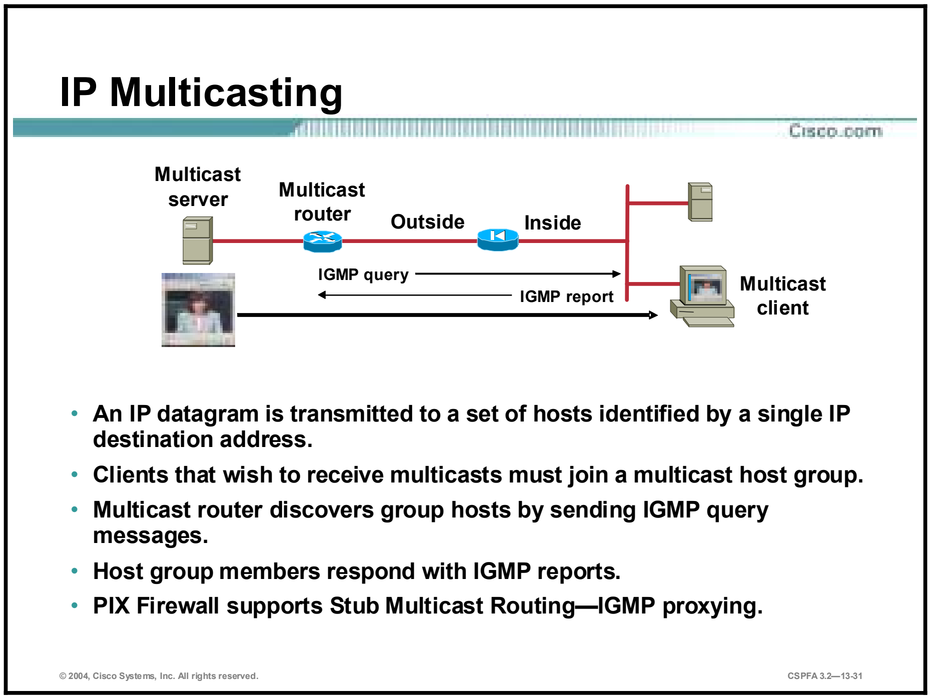
```
pix1(config)# router ospf 1 //public AS
pix1(config-router)# network 1.1.1.0 255.255.255.0 area 0
pix1(config)# router ospf 2 //private AS
pix1(config-router)# redistribute ospf 1 //import public routes
pix1(config-router)# network 10.0.0.0 255.255.255.0 area 10.0.0.0
pix1(config-router)# network 192.168.1.0 255.255.255.0 area
    192.168.1.0
```

CSPFA 3.2—13-29

In the example in the figure, the PIX Firewall is defined with two OSPF processes, or routing domains. The OSPF 1 routing domain comprises area 0, network 1.1.1.0/24. The OSPF 2 routing domain comprises two areas, area 10.0.0.0 and area 192.168.1.0. LSA type 3 advertisements can pass within routing domains, but not between the OSPF routing domains. To enable OSPF 2, the private network, to receive OSPF routing updates from OSPF 1, configure the **redistribute** command. This command enables the PIX Firewall to redistribute routes from OSPF 1 to OSPF 2. The PIX Firewall is not configured to distribute the routes in the opposite direction.

Multicast

This topic explains how to configure the PIX Firewall to allow multicast traffic.



IP Multicasting

- An IP datagram is transmitted to a set of hosts identified by a single IP destination address.
- Clients that wish to receive multicasts must join a multicast host group.
- Multicast router discovers group hosts by sending IGMP query messages.
- Host group members respond with IGMP reports.
- PIX Firewall supports Stub Multicast Routing—IGMP proxying.

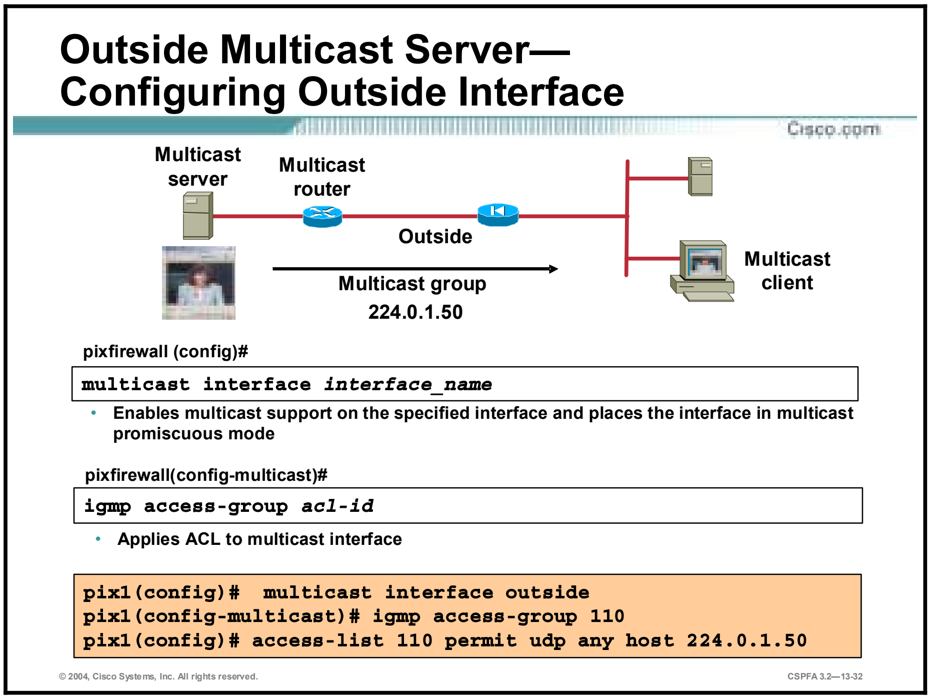
CSPFA 3.2—13-31

IP multicasting is a bandwidth-conserving technology that reduces traffic by simultaneously delivering a single stream of information to multiple recipients. Applications that take advantage of multicast include video conferencing, corporate communications, distance learning, and distribution of software, stock quotes, and news.

IP multicasting is actually the transmission of an IP datagram to a "host group," a set of hosts identified by a single IP destination address. In order for this to work, hosts that wish to receive multicasts must "tune in" to the multicast by joining a multicast host group, and routers that forward multicast datagrams must know which hosts belong to which group. Routers discover this information by sending Internet Group Management Protocol (IGMP) query messages through their attached local networks. Host members of a multicast group respond to the query by sending IGMP "reports" noting the multicast groups to which they belong. If a host is removed from a multicast group, it sends a "leave" message to the multicast router.

In software versions 6.2 and higher, the PIX Firewall supports Stub Multicast Routing (SMR), which enables it to pass multicast traffic. This feature is necessary when hosts that need to receive multicast transmissions are separated from the multicast router by a PIX Firewall. With SMR, the PIX Firewall acts as an IGMP proxy agent. It forwards IGMP messages from hosts to the upstream multicast router, which takes responsibility for forwarding multicast datagrams from one multicast group to all other networks that have members in the group. When SMR is used, it is not necessary to construct Generic Route Encapsulation (GRE) tunnels to allow multicast traffic to bypass the PIX Firewall.

Note: The GRE protocol is used for tunneling data across an IP network.



Outside Multicast Server—Configuring Outside Interface

pixfirewall (config)#

```
multicast interface interface_name
```

- Enables multicast support on the specified interface and places the interface in multicast promiscuous mode

pixfirewall(config-multicast)#

```
igmp access-group acl-id
```

- Applies ACL to multicast interface

```
pix1(config)#  multicast interface outside
pix1(config-multicast)# igmp access-group 110
pix1(config)# access-list 110 permit udp any host 224.0.1.50
```

CSPFA 3.2—13-32

When hosts that need to receive a multicast transmission are separated from the multicast router by a PIX Firewall, configure the PIX Firewall to forward IGMP reports from the downstream hosts and to forward multicast transmissions from the upstream router. Complete the following steps to allow hosts to receive multicast transmissions through the PIX Firewall:

Step 1 Use the **multicast interface** command to enable multicast forwarding on the outside interface and place the interfaces in multicast promiscuous mode. When you enter this command, the command line interface (CLI) enters multicast subcommand mode, and the prompt changes to (config-multicast)#. From this prompt, you can enter the **igmp** commands for further multicast support. The **clear multicast** command clears all multicast settings.

Step 2 (Optional.) Use the permit option of the **access-list** command to configure an access control list (ACL) that allows traffic to the desired Class D destination addresses. You can also use the deny option to deny access to transmissions from specific multicast groups. Within the ACL, the destination-addr argument is the Class D address of the multicast group to which you want to permit or deny multicast transmissions. If you use ACLs for this purpose, you must also use the **igmp access-group** command to apply the ACL to the currently selected interface.

The syntax for the **multicast interface** command is as follows:

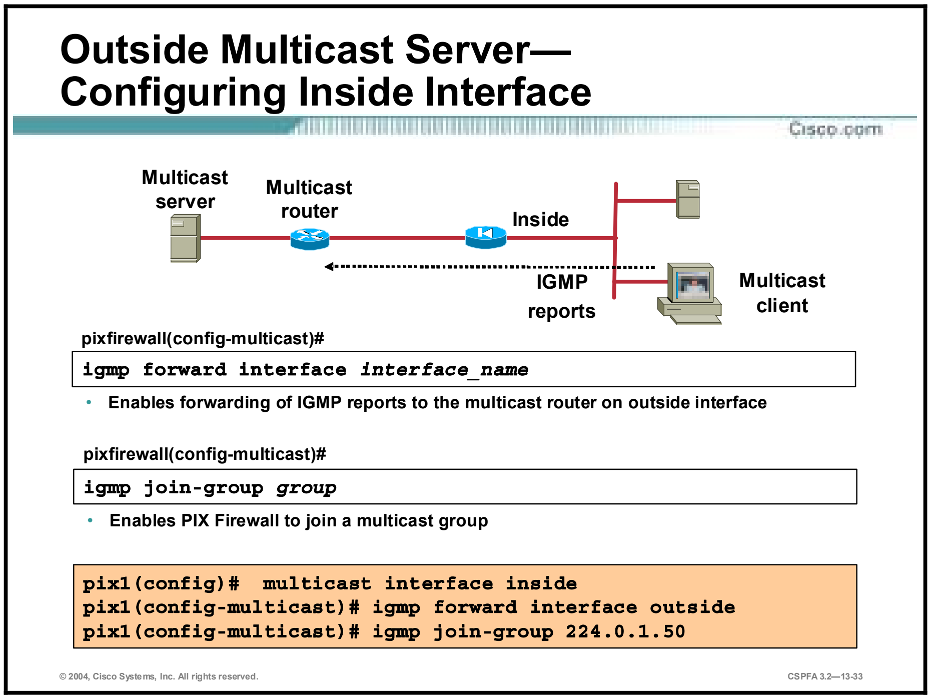
multicast interface ***interface_name***

interface_name	The name of the interface to pass multicast traffic.

The syntax for the **igmp** sub-command above is as follows:

igmp access-group *-id*

id	Access control list ID.



Outside Multicast Server—Configuring Inside Interface

pixfirewall(config-multicast)#

```
igmp forward interface interface_name
```

- Enables forwarding of IGMP reports to the multicast router on outside interface

pixfirewall(config-multicast)#

```
igmp join-group group
```

- Enables PIX Firewall to join a multicast group

```
pix1(config)#  multicast interface inside
pix1(config-multicast)# igmp forward interface outside
pix1(config-multicast)# igmp join-group 224.0.1.50
```

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—13-33

When hosts that need to receive a multicast transmission are separated from the multicast router by a PIX Firewall, configure the PIX Firewall to forward IGMP reports from the downstream hosts and to forward multicast transmissions from the upstream router. Complete the following steps to allow hosts to receive multicast transmissions through the PIX Firewall:

Step 1 Use the **multicast interface** command to enable multicast forwarding on the inside interface and place the interfaces in multicast promiscuous mode. When you enter this command, the CLI enters multicast subcommand mode, and the prompt changes to (config-multicast)#. From this prompt, you can enter the **igmp** commands for further multicast support. The **clear multicast** command clears all multicast settings.

Step 2 Use the **igmp forward** command to enable IGMP forwarding on the PIX Firewall. The **igmp forward** command enables forwarding of all IGMP host report and leave messages received by the PIX Firewall to the specified interface. The interface specified is the PIX Firewall interface connected to the multicast router. In the example in the figure, this is the outside interface.

Note: All IGMP commands (except **show igmp** and **clear igmp**) are available only as submode commands of the **multicast interface** command.

Step 3 (Optional.) Use the **igmp join-group** command to configure the PIX Firewall to join a multicast group. This command configures the interface to be a statically connected member of the specified group. It allows the PIX Firewall to act for a client that may not be able to respond via IGMP but that still requires reception. The **igmp join-group** command is applied to the downstream interface toward the receiving hosts.

A multicast group is defined by a Class D IP address. Although Internet IP multicasting uses the entire range of 224.0.0.0 to 239.255.255.255, any group address you assign must be within the range 224.0.0.2 to 239.255.255.255. Because the address 224.0.0.0 is the base address for Internet IP multicasting, it cannot be assigned to any group. The address 224.0.0.1 is assigned to the permanent group of all IP hosts (including gateways). This is used to address all multicast hosts on the directly connected network. There is no multicast address (or any other IP address) for all hosts on the total Internet.

The syntax for the **multicast interface** command is as follows:

multicast interface ***interface_name***

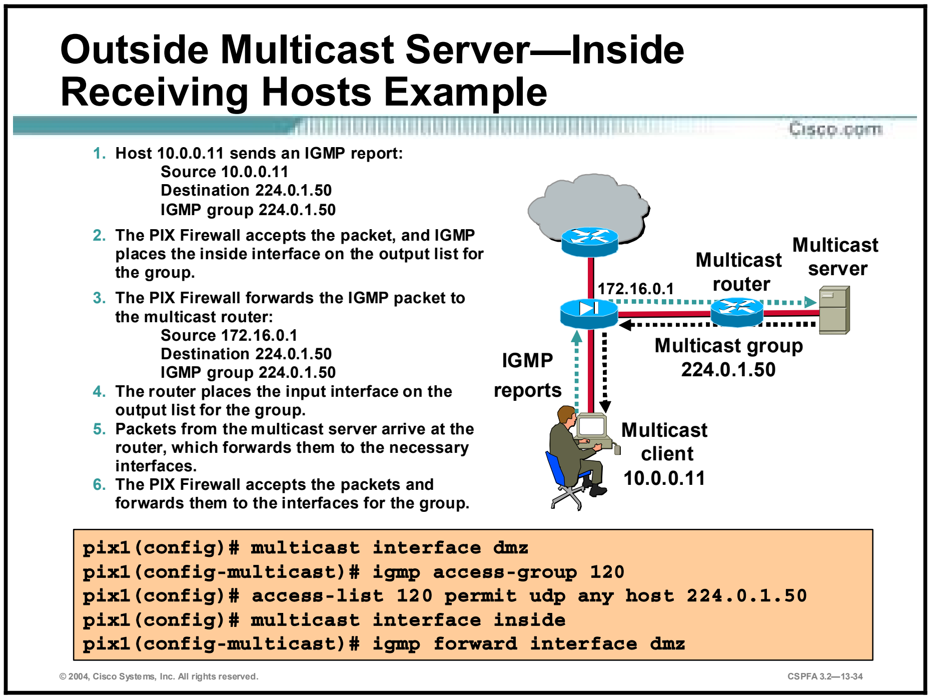
interface_name	The name of the interface to pass multicast traffic.

The syntax for the **igmp** sub-commands above is as follows:

igmp forward interface ***mc-source-if-name***

igmp join-group ***group***

mc-source-if-name	PIX Firewall interface connected to the multicast router.
group	The IP address of the multicast group being joined.



Outside Multicast Server—Inside Receiving Hosts Example

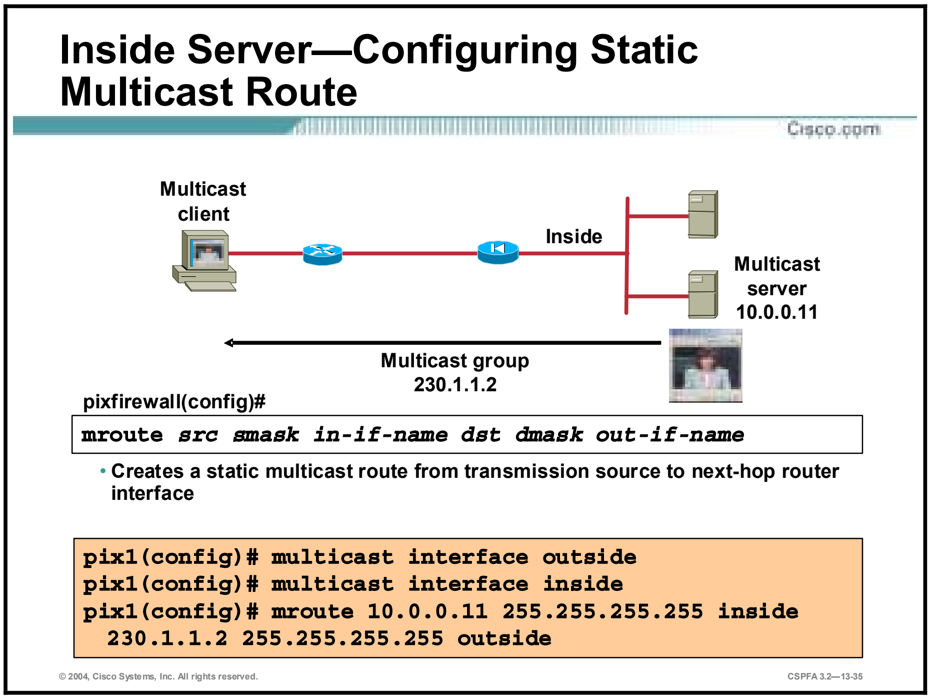
1. Host 10.0.0.11 sends an IGMP report:
   Source 10.0.0.11
   Destination 224.0.1.50
   IGMP group 224.0.1.50
2. The PIX Firewall accepts the packet, and IGMP places the inside interface on the output list for the group.
3. The PIX Firewall forwards the IGMP packet to the multicast router:
   Source 172.16.0.1
   Destination 224.0.1.50
   IGMP group 224.0.1.50
4. The router places the input interface on the output list for the group.
5. Packets from the multicast server arrive at the router, which forwards them to the necessary interfaces.
6. The PIX Firewall accepts the packets and forwards them to the interfaces for the group.

```
pix1(config)# multicast interface dmz
pix1(config-multicast)# igmp access-group 120
pix1(config)# access-list 120 permit udp any host 224.0.1.50
pix1(config)# multicast interface inside
pix1(config-multicast)# igmp forward interface dmz
```

CSPFA 3.2—13-34

The figure shows use of the **multicast** command with corresponding **igmp** subcommands. Multicast is permitted on the dmz and inside interfaces. The **igmp forward** command enables the PIX Firewall to forward IGMP reports from inside hosts to the multicast router on its dmz interface.

In the example in the figure, host 10.0.0.11 joins multicast group 224.0.1.50. The PIX Firewall enables host 10.0.0.11 to receive multicasts from the multicast server.



Inside Server—Configuring Static Multicast Route

pixfirewall(config)#

```
mroute src smask in-if-name dst dmask out-if-name
```

- Creates a static multicast route from transmission source to next-hop router interface

```
pix1(config)# multicast interface outside
pix1(config)# multicast interface inside
pix1(config)# mroute 10.0.0.11 255.255.255.255 inside
  230.1.1.2 255.255.255.255 outside
```

When a multicast transmission source is on a protected (or more secure) interface of a PIX Firewall, you must specifically configure the PIX Firewall to forward multicast transmissions from the source. Enable multicast forwarding on the PIX Firewall interfaces toward each network containing hosts that are registered to receive the multicast transmissions. Complete the following steps to configure the PIX Firewall to forward multicast transmissions from an inside source:

Step 1 Use the **multicast interface** command to enable multicast forwarding on each PIX Firewall interface.

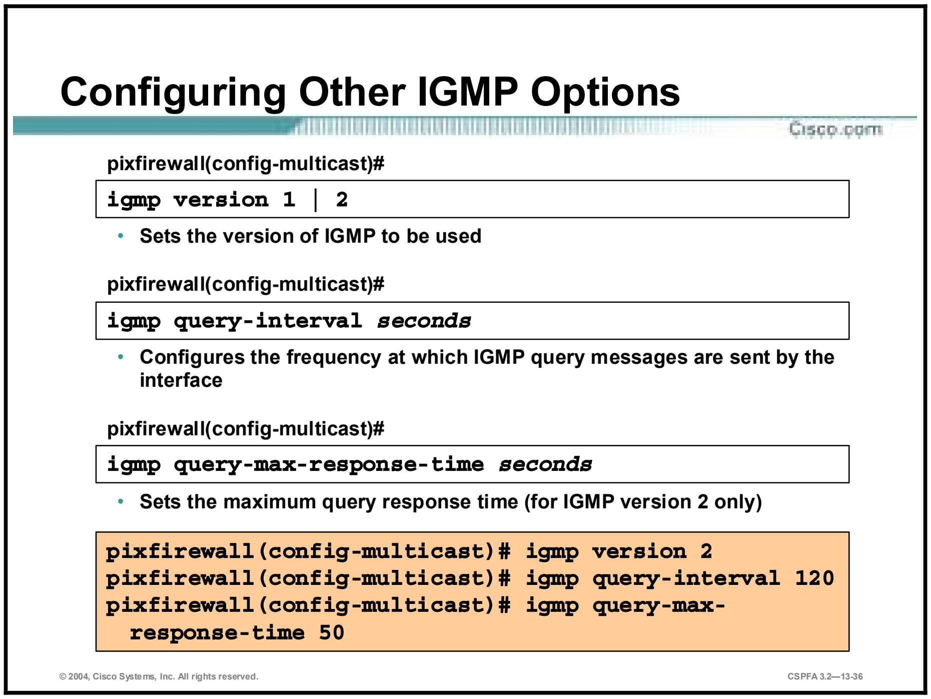
Step 2 Use the **mroute** command to create a static route from the transmission source to the next-hop router interface. Use the **clear mroute** command to clear static multicast routes.

The syntax for the **mroute** commands is as follows:

mroute ***src smask in-if-name dst dmask out-if-name***

src	The multicast source address.
smask	The multicast source mask.
in-if-name	The input interface to pass multicast traffic.
dst	The class D address of the multicast group.
dmask	The destination network address mask.
out-if-name	The output interface to pass multicast traffic.

In the figure, multicast traffic is enabled on the inside and outside interfaces. A static multicast route is configured to enable inside multicast server 10.0.0.11 to transmit multicasts to members of group 230.1.1.2 on the outside interface.



Configuring Other IGMP Options

Cisco.com

pixfirewall(config-multicast)#

```
igmp version 1 | 2
```

- Sets the version of IGMP to be used

pixfirewall(config-multicast)#

```
igmp query-interval seconds
```

- Configures the frequency at which IGMP query messages are sent by the interface

pixfirewall(config-multicast)#

```
igmp query-max-response-time seconds
```

- Sets the maximum query response time (for IGMP version 2 only)

```
pixfirewall(config-multicast)# igmp version 2
pixfirewall(config-multicast)# igmp query-interval 120
pixfirewall(config-multicast)# igmp query-max-
  response-time 50
```

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—13-36

There are some other IGMP options that can be set by an administrator. The administrator can choose an IGMP version and configure the IGMP timers with the **igmp query-interval**, and **igmp query-max-response-time** commands. To specify the version of IGMP, use with the **igmp version** command. This configures which version of IGMP is used on the subnet represented by the specified interface. The default is version 2.

Version 2 offers bandwidth-conserving features not available in version 1. The Leave Group message is one of these features. The advantage of this message is reducing the bandwidth waste between the time the last host in a subnet drops membership and the time the router times out for its queries and decides there are no more members present for that group.

For further information on the differences in versions 1 and 2, see RFC 2236.

Use the **igmp query-interval** command to configure the frequency at which IGMP query messages are sent by the interface. The default is 60 seconds. The permitted range of values is from 1 to 65535. Use the command **no igmp query-**interval to set the query interval back to the default.

The **igmp query-max-response-time** command specifies the maximum query response time and is only available with IGMP version 2. The default is 10 seconds. The permitted range of values is from 1 to 65535. Use the command **no igmp query-max-response-time** to set the query response time back to the default.

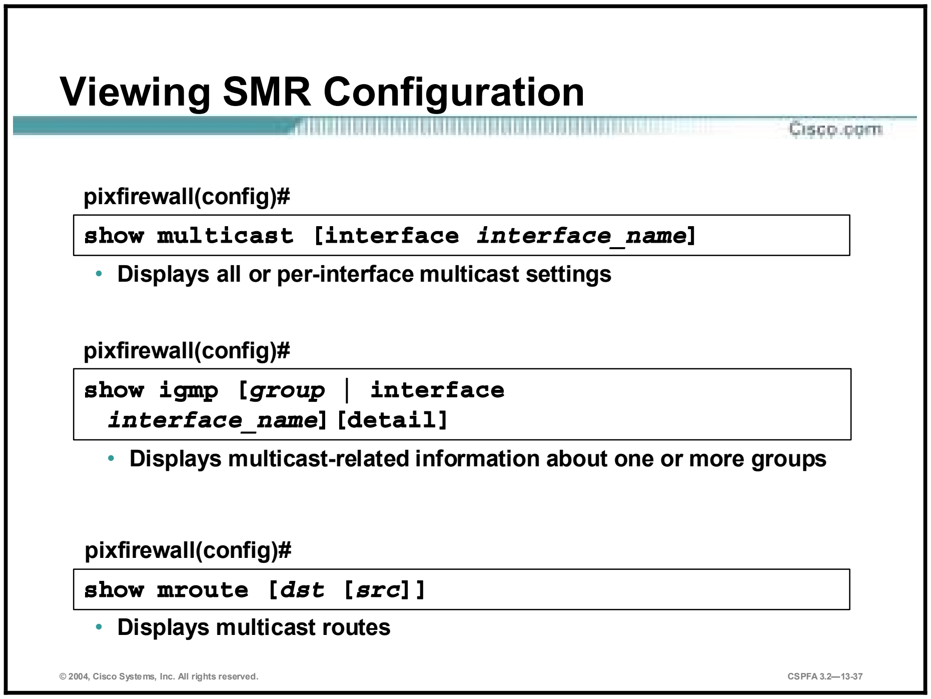
The syntax for these **igmp** commands is as follows:

igmp version 1 | 2

igmp query-interval ***seconds***

igmp query-max-response-time ***seconds***

query-interval	The query response time interval.
query-max- response-time	The maximum query response time interval.
seconds	The number of seconds to wait.



Viewing SMR Configuration

Cisco.com

pixfirewall(config)#

```
show multicast [interface interface_name]
```

- Displays all or per-interface multicast settings

pixfirewall(config)#

```
show igmp [group | interface
  interface_name] [detail]
```

- Displays multicast-related information about one or more groups

pixfirewall(config)#

```
show mroute [dst [src]]
```

- Displays multicast routes

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—13-37

The following commands can be used to view the current multicast and IGMP configuration:

- **show multicast**—Displays all or per-interface multicast settings. This command also displays IGMP configuration for the interface.
- **show igmp**—Displays multicast-related information about one or more groups.
- **show mroute**—Shows multicast routes.

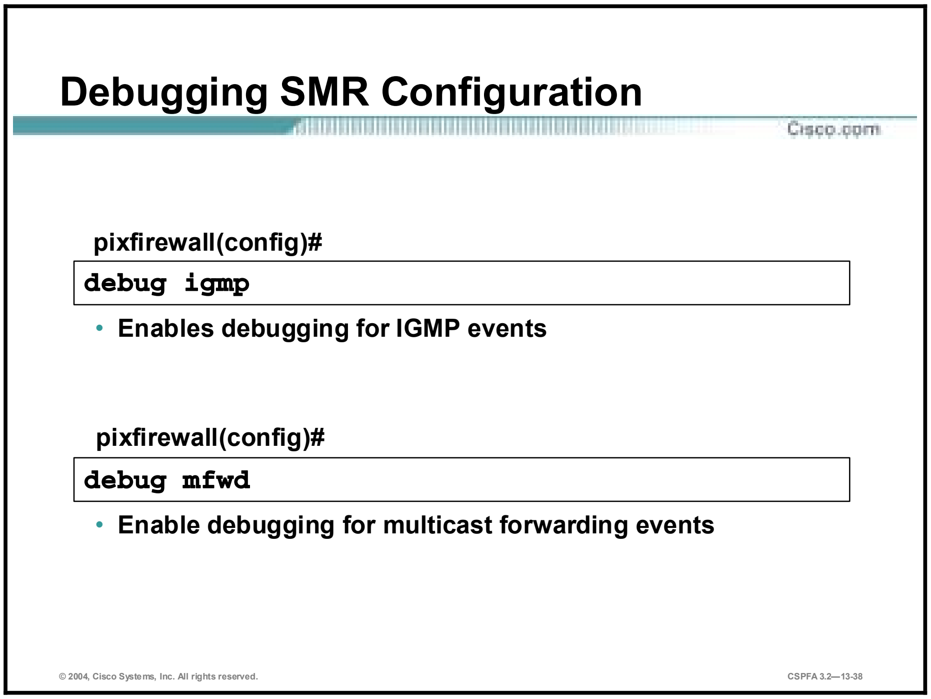
The syntax for these **show** commands is as follows:

show multicast [interface *interface_name*]

show igmp [*group* | interface *interface-name*] [detail]

show mroute [*dst* [*src*]]

interface_name	The name of the interface for which you wish to view configuration settings.
group	The address of the multicast group.
detail	Displays all information in the IGMP table.
dst	The Class D address of the multicast group.
src	The IP address of the multicast source.



Debugging SMR Configuration

pixfirewall(config)#

```
debug igmp
```

- Enables debugging for IGMP events

pixfirewall(config)#

```
debug mfwd
```

- Enable debugging for multicast forwarding events

CSPFA 3.2—13-38

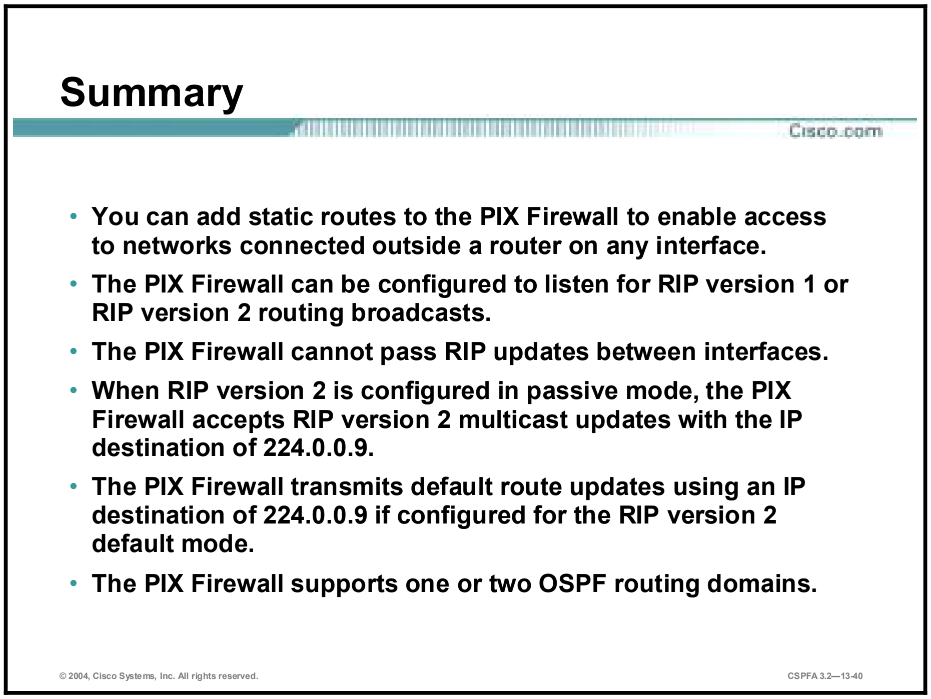
You can use the following commands for debugging your SMR configuration:

- **debug igmp**—Enables or disables debugging for IGMP events.
- **debug mfwd**—Enables or disables debugging for multicast forwarding events.

Each of these commands can be removed by using its **no** form.

Summary

This topic summarizes the information you learned in this lesson.

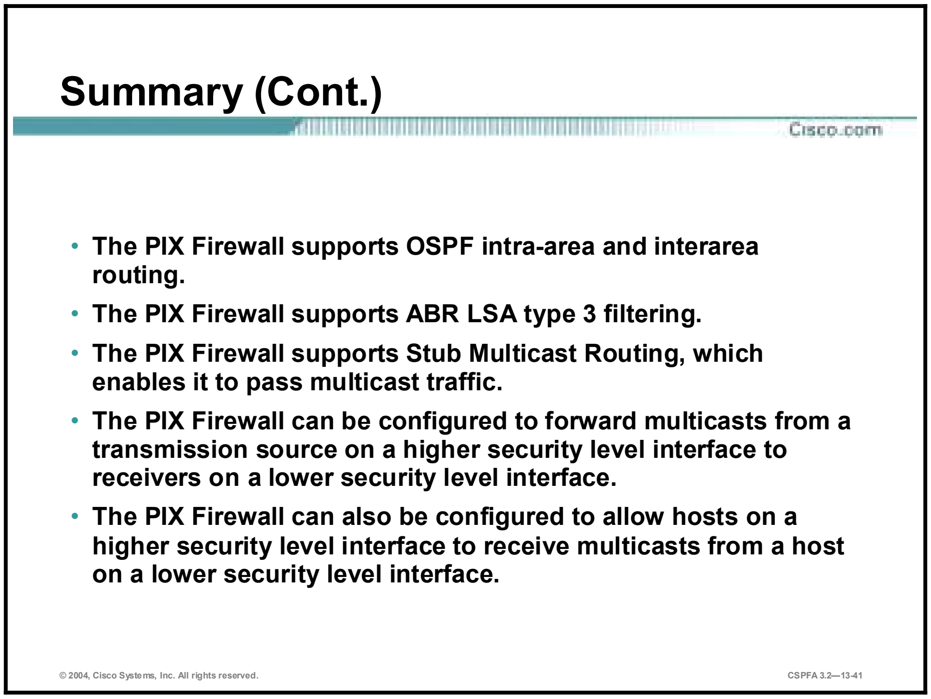


Summary

Cisco.com

- You can add static routes to the PIX Firewall to enable access to networks connected outside a router on any interface.
- The PIX Firewall can be configured to listen for RIP version 1 or RIP version 2 routing broadcasts.
- The PIX Firewall cannot pass RIP updates between interfaces.
- When RIP version 2 is configured in passive mode, the PIX Firewall accepts RIP version 2 multicast updates with the IP destination of 224.0.0.9.
- The PIX Firewall transmits default route updates using an IP destination of 224.0.0.9 if configured for the RIP version 2 default mode.
- The PIX Firewall supports one or two OSPF routing domains.

CSPFA 3.2—13-40



Summary (Cont.)

Cisco.com

- The PIX Firewall supports OSPF intra-area and interarea routing.
- The PIX Firewall supports ABR LSA type 3 filtering.
- The PIX Firewall supports Stub Multicast Routing, which enables it to pass multicast traffic.
- The PIX Firewall can be configured to forward multicasts from a transmission source on a higher security level interface to receivers on a lower security level interface.
- The PIX Firewall can also be configured to allow hosts on a higher security level interface to receive multicasts from a host on a lower security level interface.

CSPFA 3.2—13-41

14

Virtual Private Network Configuration

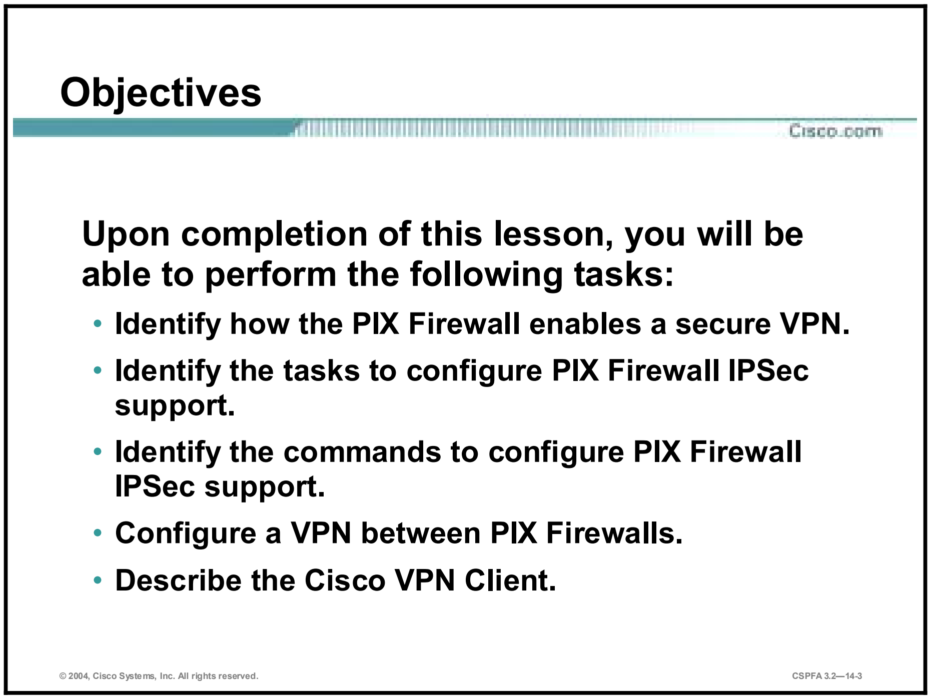
Overview

This lesson discusses how virtual private networks (VPNs) are configured. This lesson includes the following topics:

- Objectives
- The PIX Firewall enables a secure VPN
- How IPSec works
- IPSec configuration tasks
- Task 1—Prepare to configure VPN support
- Task 2 Configure IKE parameters
- Task 3 Configure IPSec parameters
- Task 4—Test and verify VPN configuration
- The Cisco VPN client
- Scale PIX Firewall VPNs
- Summary
- Lab exercise

Objectives

This topic lists the lesson’s objectives.



Objectives

Upon completion of this lesson, you will be able to perform the following tasks:

- Identify how the PIX Firewall enables a secure VPN.
- Identify the tasks to configure PIX Firewall IPSec support.
- Identify the commands to configure PIX Firewall IPSec support.
- Configure a VPN between PIX Firewalls.
- Describe the Cisco VPN Client.

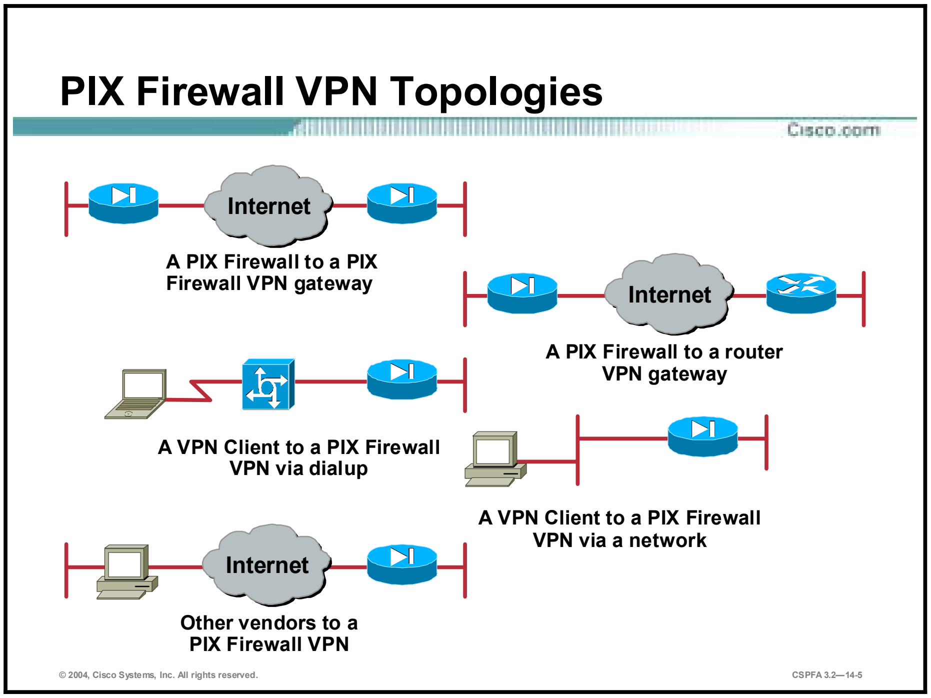
CSPFA 3.2—14-3

The PIX Firewall Enables a Secure VPN

A virtual private network (VPN) is a service offering secure, reliable connectivity over a shared, public network infrastructure such as the Internet. Because the infrastructure is shared, connectivity can be provided at lower cost than existing dedicated private networks.

The Cisco PIX Firewall is a powerful enabler of VPN services. The PIX Firewall's high performance, conformance to open standards, and ease of configuration make it a versatile VPN gateway.

The PIX Firewall VPN lesson covers the basics of IPSec and PIX Firewall VPNs with a focus on PIX Firewall gateway to PIX Firewall gateway communications.



PIX Firewall VPN Topologies

A PIX Firewall to a PIX Firewall VPN gateway

A PIX Firewall to a router VPN gateway

A VPN Client to a PIX Firewall VPN via dialup

A VPN Client to a PIX Firewall VPN via a network

Other vendors to a PIX Firewall VPN

CSPFA 3.2—14-5

The PIX Firewall enables VPNs in several topologies, as illustrated in the figure:

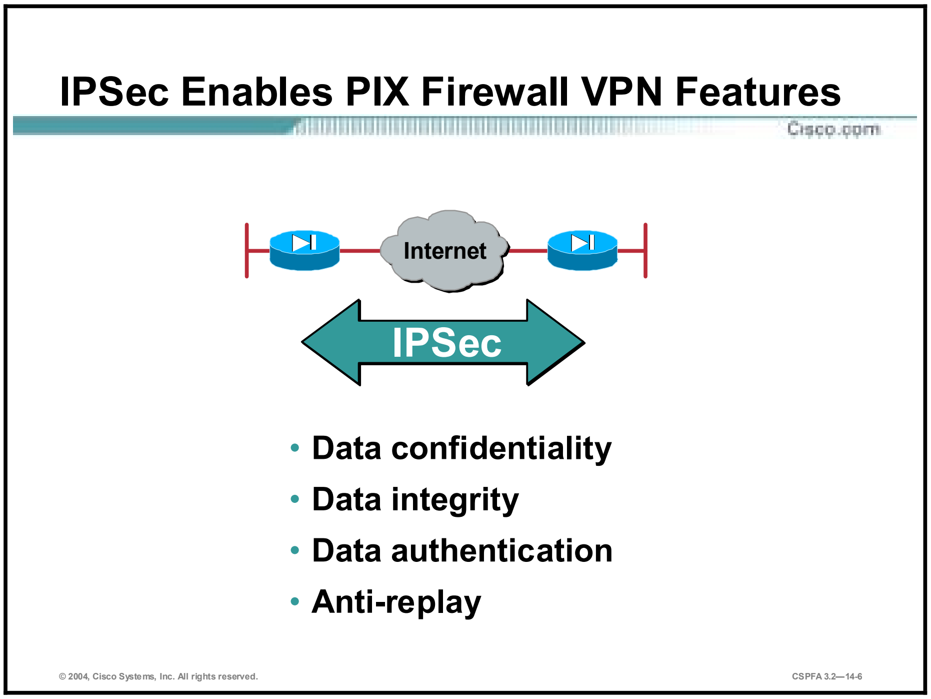
- PIX Firewall to PIX Firewall secure VPN gateway Two or more PIX Firewalls can enable a VPN, which secures traffic from devices behind the PIX Firewalls. The secure VPN gateway topology prevents the user from having to implement VPN devices or software inside the network, making the secure gateway transparent to users.
- PIX Firewall to Cisco IOS router secure VPN gateway The PIX Firewall and Cisco router, running Cisco VPN software, can interoperate to create a secure VPN gateway between networks.
- Cisco VPN Client to PIX Firewall via dialup—The PIX Firewall can become a VPN endpoint for the Cisco VPN Client over a dialup network. The dialup network can consist of ISDN, Public Switched Telephone Network (PSTN) (analog modem), or digital subscriber line (DSL) communication channels.
- Cisco VPN Client to PIX Firewall via network The PIX Firewall can become a VPN endpoint for the Cisco VPN Client over an IP network.

- Other vendor products to PIX Firewall   Products from other vendors can connect to the PIX Firewall if they conform to open VPN standards.

A VPN itself can be constructed in a number of scenarios. The most common are as follows:

- Internet VPN—A private communications channel over the public access Internet. This type of VPN can be divided into the following:
  - Connecting remote offices across the Internet
  - Connecting remote dial users to their home gateway via an Internet Service Provider (ISP) (sometimes called a Virtual Private Dial Network, or VPDN)
- Intranet VPN—A private communication channel within an enterprise or organization that may or may not involve traffic traversing a WAN.
- Extranet VPN   A private communication channel between two or more separate entities that may involve data traversing the Internet or some other WAN.

In all cases the VPN or tunnel consists of two endpoints that may be represented by PIX Firewalls, Cisco routers, individual client workstations running the Cisco VPN Client, or other vendors' VPN products that conform to open standards.



IPSec Enables PIX Firewall VPN Features

Cisco.com

- Data confidentiality
- Data integrity
- Data authentication
- Anti-replay

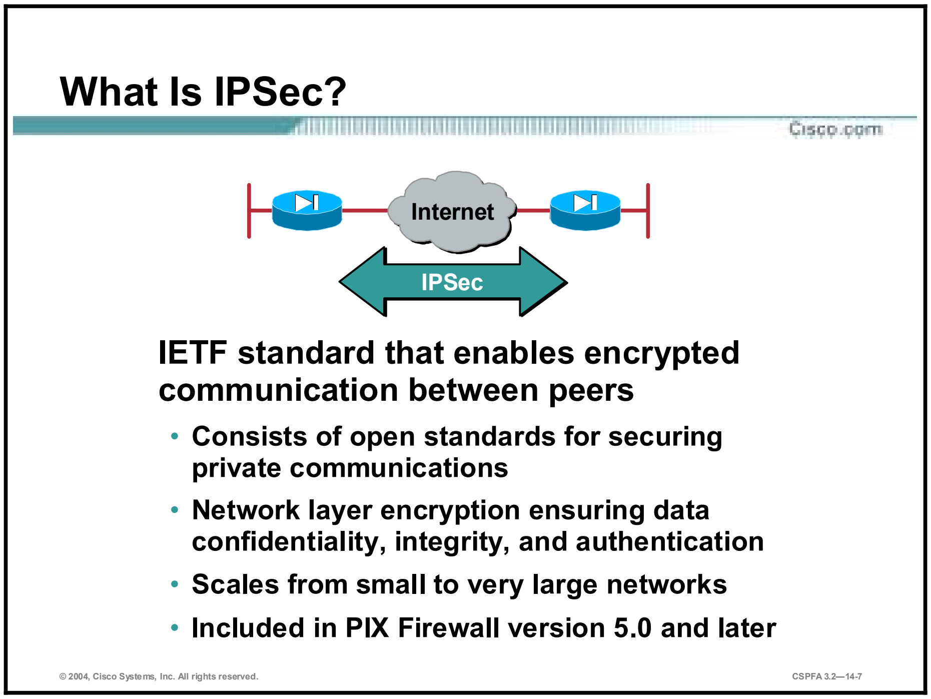
CSPFA 3.2—14-6

PIX Firewall versions 5.0 and higher use the industry-standard IP Security (IPSec) protocol suite to enable advanced VPN features. The PIX Firewall IPSec implementation is based on Cisco IOS IPSec that runs in Cisco routers.

IPSec provides a mechanism for secure data transmission over IP networks, ensuring confidentiality, integrity, and authenticity of data communications over unprotected networks such as the Internet.

IPSec enables the following PIX Firewall VPN features:

- Data confidentiality—The IPSec sender can encrypt packets before transmitting them across a network.
- Data integrity—The IPSec receiver can authenticate IPSec peers and packets sent by the IPSec sender to ensure that the data has not been altered during transmission.
- Data origin authentication   The IPSec receiver can authenticate the source of the IPSec packets sent. This service is dependent upon the data integrity service.
- Anti-replay—The IPSec receiver can detect and reject replayed packets, helping prevent spoofing and man-in-the-middle attacks.



Cisco.com
Internet
IPSec

What Is IPSec?

IETF standard that enables encrypted communication between peers

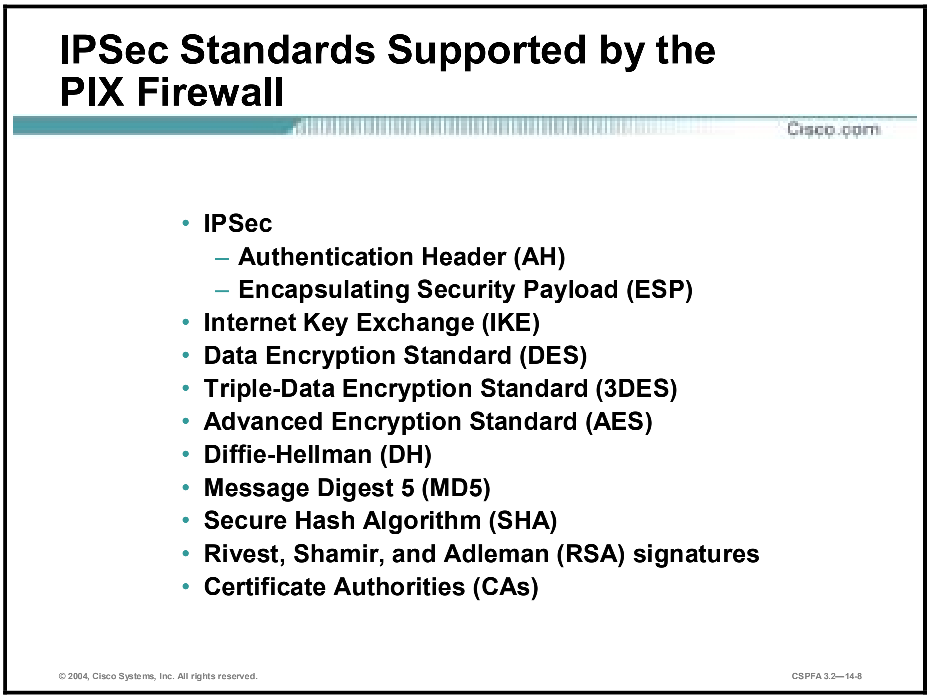
- **Consists of open standards for securing private communications**
- **Network layer encryption ensuring data confidentiality, integrity, and authentication**
- **Scales from small to very large networks**
- **Included in PIX Firewall version 5.0 and later**

CSPFA 3.2—14-7

The PIX Firewall uses the open IPSec protocol to enable secure VPNs. IPSec is a set of security protocols and algorithms used to secure data at the network layer. IPSec and related security protocols conform to open standards promulgated by the Internet Engineering Task Force (IETF) and documented RFCs and IETF-draft papers.

IPSec acts at the network layer, protecting and authenticating IP packets between a PIX Firewall and other participating IPSec devices (peers), such as PIX Firewalls, Cisco routers, the Cisco VPN Client, and other IPSec-compliant products.

IPSec can be used to scale from small to very large networks. It is included in PIX Firewall version 5.0 and later.



IPSec Standards Supported by the PIX Firewall

Cisco.com

- IPSec
  - Authentication Header (AH)
  - Encapsulating Security Payload (ESP)
- Internet Key Exchange (IKE)
- Data Encryption Standard (DES)
- Triple-Data Encryption Standard (3DES)
- Advanced Encryption Standard (AES)
- Diffie-Hellman (DH)
- Message Digest 5 (MD5)
- Secure Hash Algorithm (SHA)
- Rivest, Shamir, and Adleman (RSA) signatures
- Certificate Authorities (CAs)

CSPFA 3.2—14-8

The PIX Firewall supports the following IPSec and related standards:

- IPSec
- Internet Key Exchange (IKE)
- Data Encryption Standard (DES)
- Triple-Data Encryption Standard (3DES)
- Advanced Encryption Standard (AES)
- Diffie-Hellman (DH)
- Message Digest 5 (MD5)
- Secure Hash Algorithm-1 (SHA-1)
- Rivest, Shamir, and Adleman (RSA) signatures
- Certificate Authority (CA)

IPSec

IPSec is a framework of open standards that provides data confidentiality, data integrity, and data authentication between participating peers at the IP layer. IPSec can be used to protect one or more data flows between IPSec peers. IPSec is documented in a series of Internet RFCs, all available at http://www.ietf.org/html.charters/ipsec-charter.html. The overall IPSec implementation is guided by “Security Architecture for the Internet Protocol,” RFC 2401. IPSec consists of the following two main protocols:

- Authentication Header (AH)   A security protocol that provides authentication and optional replay-detection services. AH acts as a “digital signature” to ensure that tampering has not occurred with the data in the IP packet. AH does not provide data encryption and decryption services. AH can be used either by itself or with Encapsulating Security Payload (ESP).

- Encapsulating Security Payload (ESP) A security protocol that provides data confidentiality and protection with optional authentication and replay-detection services. The PIX Firewall uses ESP to encrypt the data payload of IP packets. ESP can be used either by itself or in conjunction with AH.

IKE

IKE is a hybrid protocol that provides utility services for IPSec: authentication of the IPSec peers, negotiation of IKE and IPSec security associations (SAs), and establishment of keys for encryption algorithms used by IPSec. IKE is synonymous with Internet Security Association Key Management Protocol (ISAKMP) in PIX Firewall configuration.

DES

DES is used to encrypt and decrypt packet data. DES is used by both IPSec and IKE. DES uses a 56-bit key, ensuring high performance encryption.

3DES

3DES is a variant of DES, which iterates three times with three separate keys, effectively doubling the strength of DES. 3DES is used by IPSec to encrypt and decrypt data traffic. 3DES uses a 168-bit key, ensuring strong encryption.

AES

The National Institute of Standards and Technology (NIST) recently adopted the new Advanced Encryption Standard to replace DES encryption in cryptographic devices. AES provides stronger security than DES and is computationally more efficient than 3DES. AES offers three different key strengths: 128-, 192-, and 256-bit keys.

DH

DH is a public-key cryptography protocol. It enables two parties to establish a shared secret key over an insecure communications channel. DH is used within IKE to establish session keys. 768-bit, 1024-bit, and 1536-bit DH groups are supported in the PIX Firewall.

MD5

MD5 is a hash algorithm used to authenticate packet data. The PIX Firewall uses the MD5 hash-based message authentication code (HMAC) variant, which provides an additional level of hashing. A hash is a one-way encryption algorithm that takes an input message of arbitrary length and produces a fixed-length output message. IKE, AH, and ESP use MD5 for authentication.

SHA-1

SHA is a hash algorithm used to authenticate packet data. The PIX Firewall uses the SHA-1 HMAC variant, which provides an additional level of hashing. IKE, AH, and ESP use SHA-1 for authentication.

RSA Signatures

RSA is a public-key cryptographic system used for authentication. IKE on the PIX Firewall uses a DH exchange to determine secret keys on each IPSec peer used by encryption algorithms. The DH exchange can be authenticated with RSA (or pre-shared keys).

CA

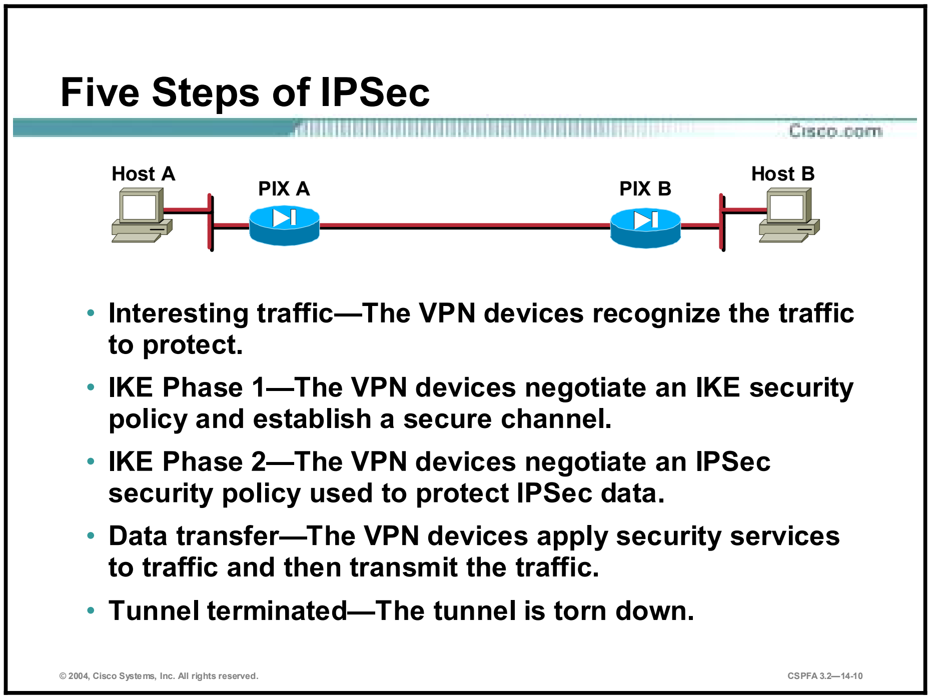
The CA support of the PIX Firewall enables the IPSec-protected network to scale by providing the equivalent of a digital identification card to each device. When two IPSec peers wish to communicate, they exchange digital certificates to prove their identities (thus removing the need to manually exchange public keys with each peer or to manually specify a shared key at each peer). The digital certificates are obtained from a CA. CA support on the PIX Firewall uses RSA signatures to authenticate the CA exchange.

Security Association (SA)

The concept of an SA is fundamental to IPSec. An SA is a connection between IPSec peers that determines the IPSec services available between the peers, similar to a TCP or UDP port. Each IPSec peer maintains an SA database in memory containing SA parameters. SAs are uniquely identified by the IPSec peer address, security protocol, and Security Parameter Index (SPI). You will need to configure SA parameters and monitor SAs on the PIX Firewall.

How IPSec Works

This topic details the individual steps of IPSec.



Five Steps of IPSec

Host A
PIX A
PIX B
Host B

- Interesting traffic—The VPN devices recognize the traffic to protect.
- IKE Phase 1—The VPN devices negotiate an IKE security policy and establish a secure channel.
- IKE Phase 2—The VPN devices negotiate an IPSec security policy used to protect IPSec data.
- Data transfer—The VPN devices apply security services to traffic and then transmit the traffic.
- Tunnel terminated—The tunnel is torn down.

CSPFA 3.2—14-10

The goal of IPSec is to protect the desired data with the needed security services. IPSec operation can be broken down into five primary steps:

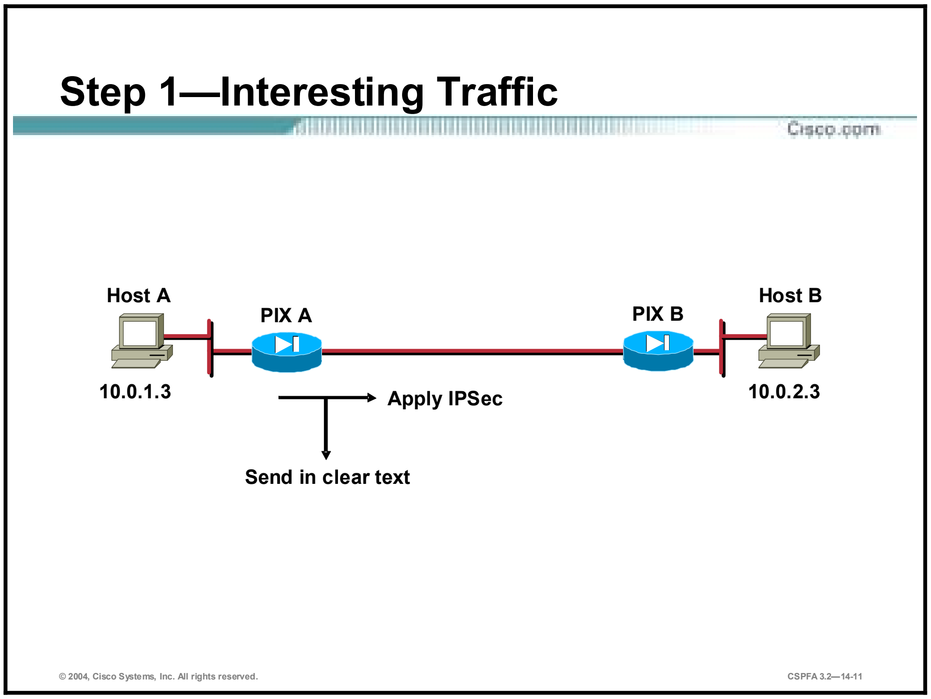
Step 1 Interesting traffic—Traffic is deemed interesting when the VPN device recognizes that the traffic you want to send needs to be protected.

Step 2 IKE Phase 1—A basic set of security services are negotiated and agreed upon between peers. These security services protect all subsequent communications between the peers. IKE Phase 1 sets up a secure communications channel between peers.

Step 3 IKE Phase 2—IKE negotiates IPSec SA parameters and sets up matching IPSec SAs in the peers. These security parameters are used to protect data and messages exchanged between endpoints.

Step 4 Data transfer—Data is transferred between IPSec peers based on the IPSec parameters and keys stored in the SA database.

Step 5 IPSec tunnel termination—IPSec SAs terminate through deletion or by timing out.



Step 1—Interesting Traffic

Host A

10.0.1.3

PIX A

PIX B

Host B

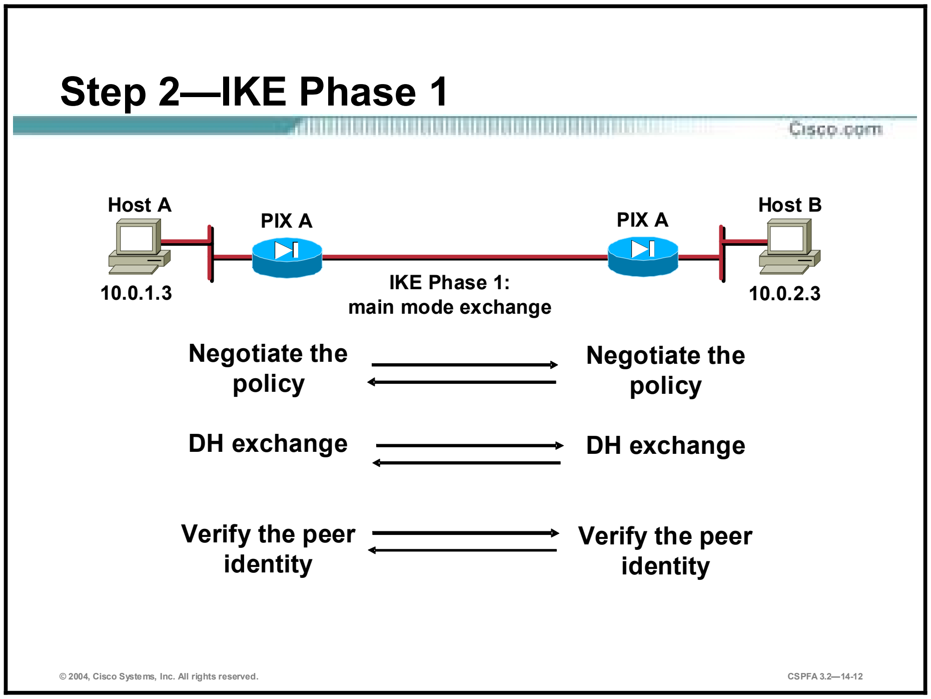
10.0.2.3

Apply IPSec

Send in clear text

CSPFA 3.2—14-11

Determining which traffic needs to be protected is done as part of formulating a security policy for use of a VPN. The policy is used to determine which traffic needs to be protected and which traffic can be sent in the clear. For every inbound and outbound datagram, there are two choices: apply IPSec or bypass IPSec and send the datagram in clear text. For every datagram protected by IPSec, the system administrator must specify the security services applied to the datagram. The security policy database specifies the IPSec protocols, modes, and algorithms applied to the traffic. The services are then applied to traffic destined to each particular IPSec peer. With the Cisco VPN Client, you use menu windows to select connections that you want secured by IPSec. When interesting traffic transits the IPSec client, the client initiates the next step in the process: negotiating an IKE Phase 1 exchange.



Step 2—IKE Phase 1

Cisco.com

Host A

10.0.1.3

PIX A

IKE Phase 1:
main mode exchange

PIX A

Host B

10.0.2.3

Negotiate the policy ⇄ Negotiate the policy

DH exchange ⇄ DH exchange

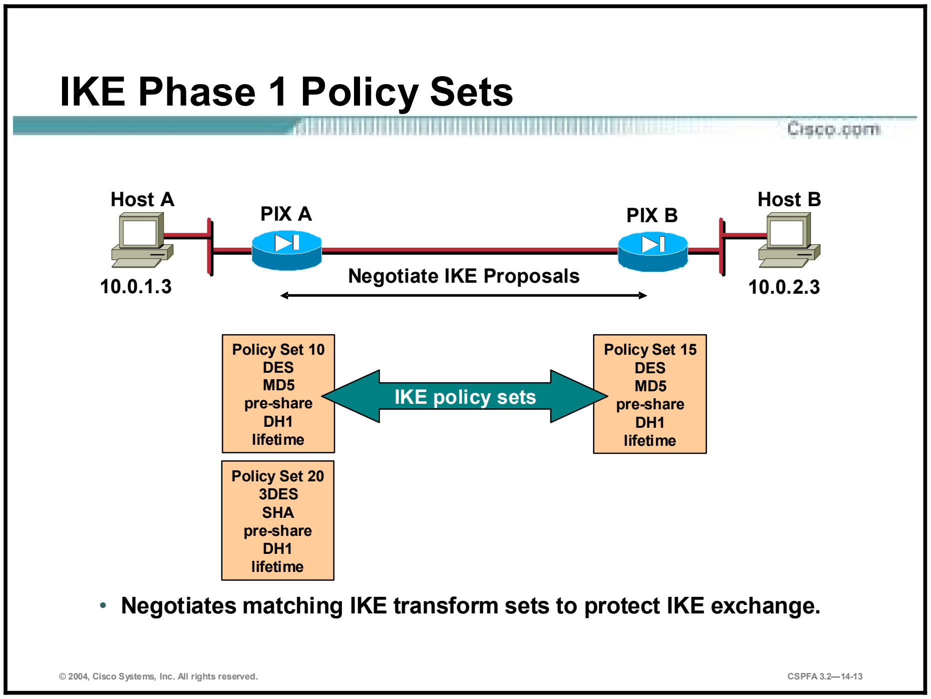
Verify the peer identity ⇄ Verify the peer identity

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—14-12

The basic purpose of IKE Phase 1 is to negotiate IKE policy sets, authenticate the peers, and set up a secure channel between the peers. IKE Phase 1 occurs in two modes: main mode and aggressive mode.

- Main mode has three two-way exchanges between the initiator and receiver:
  - First exchange—The algorithms and hashes used to secure the IKE communications are negotiated and agreed upon between peers.
  - Second exchange—This exchange uses a DH exchange to generate shared secret keys and pass nonces, which are random numbers sent to the other party, signed, and returned to prove their identity. The shared secret key is used to generate all the other encryption and authentication keys.
  - Third exchange—This exchange verifies the other side’s identity. It is used to authenticate the remote peer. The main outcome of main mode is a secure communication path for subsequent exchanges between the peers. Without proper authentication, you might establish a secure communication channel with a hacker who could be stealing all your sensitive material.
- In the aggressive mode, fewer exchanges are done and with fewer packets. The first exchange covers almost all of the steps: the IKE policy set negotiation; the DH public key generation; a nonce, which the other party signs; and an identity packet, which can be used to verify the identity of the other party via a third party. The receiver sends everything back that is needed to complete the exchange. The only thing left is for the initiator to confirm the exchange.



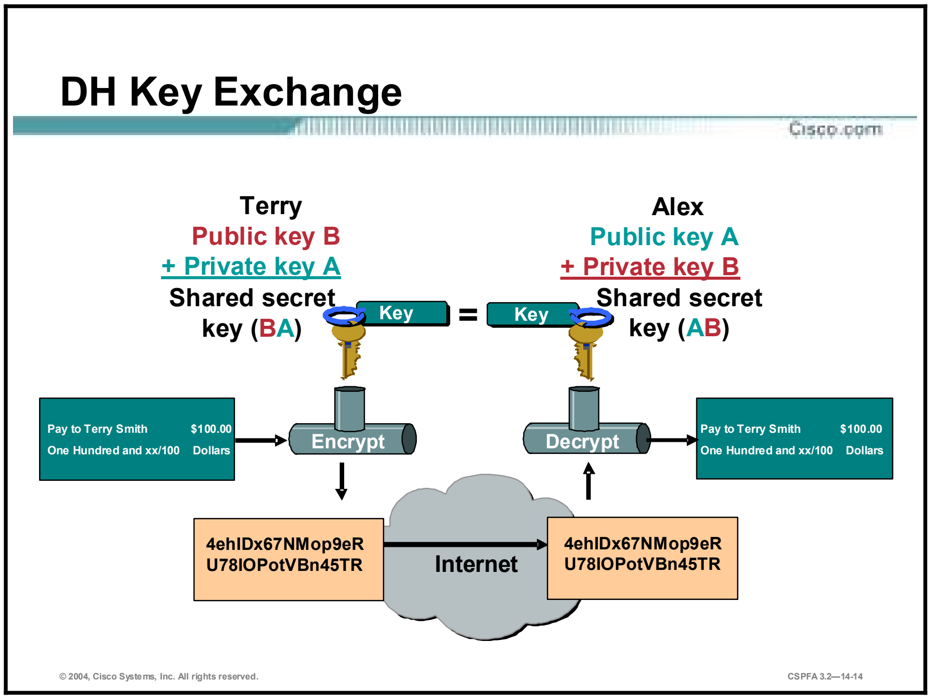
IKE Phase 1 Policy Sets

- **Negotiates matching IKE transform sets to protect IKE exchange.**

When trying to make a secure connection between hosts A and B through the Internet, IKE security proposals are exchanged between PIX Firewalls A and B. The proposals identify the IPSec protocol being negotiated (for example, ESP). Under each proposal, the originator must delineate which algorithms are employed in the proposal (for example, DES with MD5). Rather than negotiate each algorithm individually, the algorithms are grouped into sets, called IKE transform sets. A transform set delineates which encryption algorithm, authentication algorithm, mode, and key length are proposed. These IKE proposals and transform sets are exchanged during the IKE main mode first exchange phase. If a transform set match is found between peers, the main mode continues. If no match is found, the tunnel is shut down.

In the example in the figure, PIX Firewall A sends IKE transform sets 10 and 20 to PIX Firewall B. PIX Firewall B compares its set, transform set 15, with those received from PIX Firewall A. In this instance, there is a match: PIX Firewall A's transform set 10 matches PIX Firewall B's transform set 15.

In a point-to-point application, each end may only need a single IKE policy set defined. However, in a hub and spoke environment, the central site may require multiple IKE policy sets to satisfy all the remote peers.



DH Key Exchange

Terry
Public key B
+ Private key A
Shared secret key (BA)

Key = Key

Alex
Public key A
+ Private key B
Shared secret key (AB)

Pay to Terry Smith $100.00
One Hundred and xx/100 Dollars

Encrypt

4ehIDx67NMop9eR
U78IOPotVBn45TR

Internet

4ehIDx67NMop9eR
U78IOPotVBn45TR

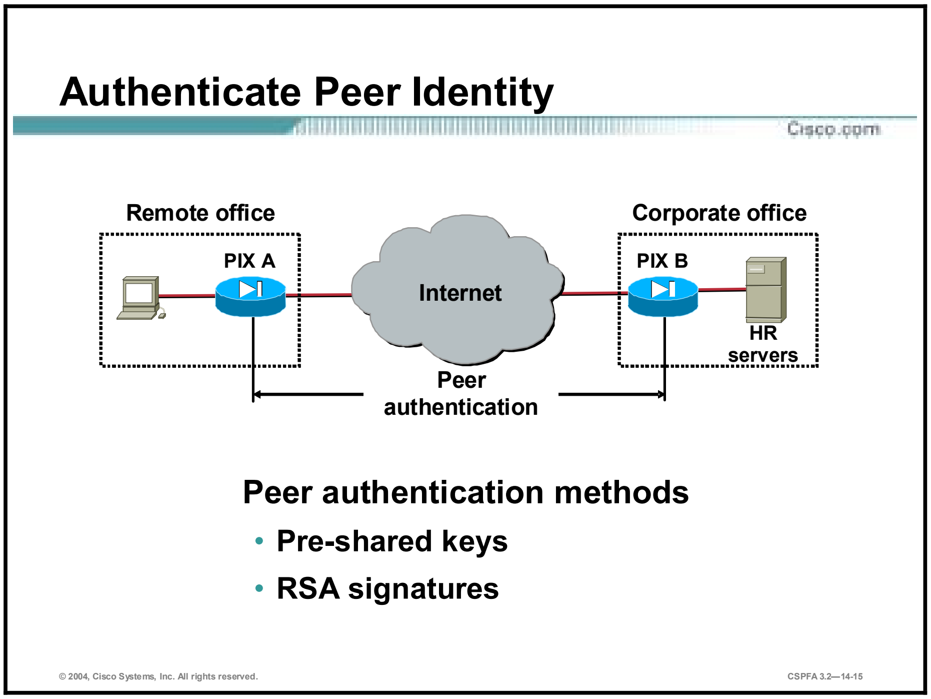
Decrypt

Pay to Terry Smith $100.00
One Hundred and xx/100 Dollars

CSPFA 3.2—14-14

DH key exchange is a public key exchange method that provides a way for two peers to establish a shared secret key over an insecure communications path. With DH, there are several different DH algorithms or groups defined, DH groups 1–7. A group number defines an algorithm and unique values. For instance, group 1 defines exponentiation over a prime modulus (MODP) algorithm with a 768-bit prime number. Group 2 defines a MODP algorithm with a 1024-bit prime number. During IKE Phase 1, the group is negotiated between peers. Between Cisco VPN devices, either group 1 or 2 is supported.

After the group negotiations are completed, the shared secret key is calculated. The shared secret key, SKEYID, is used in the derivation of three other keys: SKEYID_a, SKEYID_d, and SKEYID_e. Each key has a separate purpose. SKEYID_a is the keying material used during the authentication process. SKEYID_d is the keying material used to derive keys for non-ISAKMP SAs. SKEYID_e is the keying material used in the encryption process. All four keys are calculated during IKE Phase 1.



Authenticate Peer Identity

Peer authentication methods

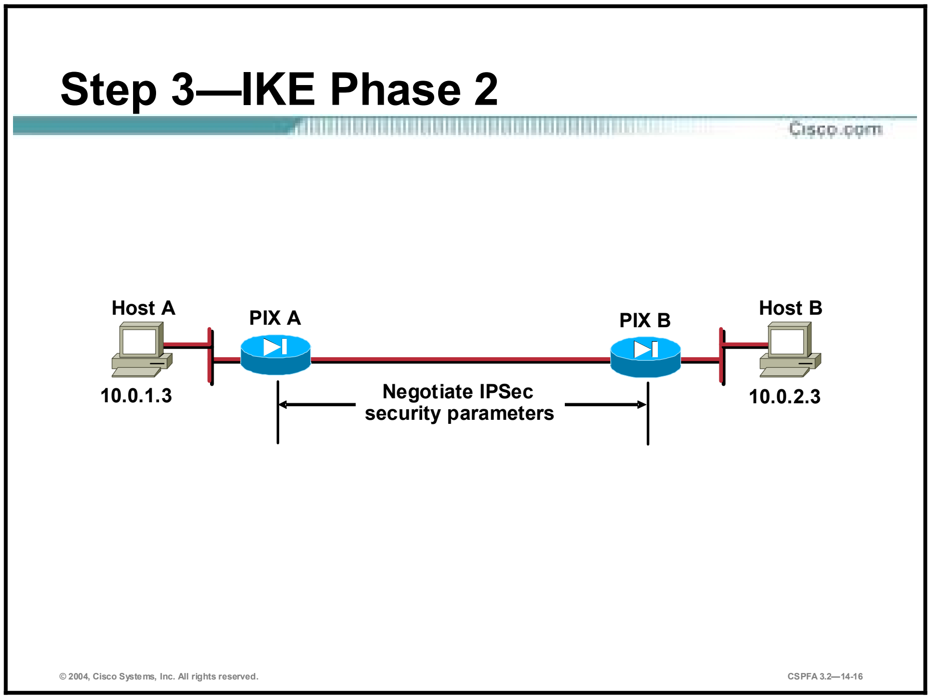
- **Pre-shared keys**
- **RSA signatures**

CSPFA 3.2—14-15

When conducting business over the Internet, you must know who is at the other end of the tunnel. The device on the other end of the VPN tunnel must be authenticated before the communications path is considered secure. The last exchange of IKE Phase 1 is used to authenticate the remote peer.

The PIX Firewall supports two data origin authentication methods:

- Pre-shared keys—A secret key value entered for each peer is manually used to authenticate the peer.
- RSA signatures—Uses the exchange of digital certificates to authenticate the peers.



Step 3—IKE Phase 2

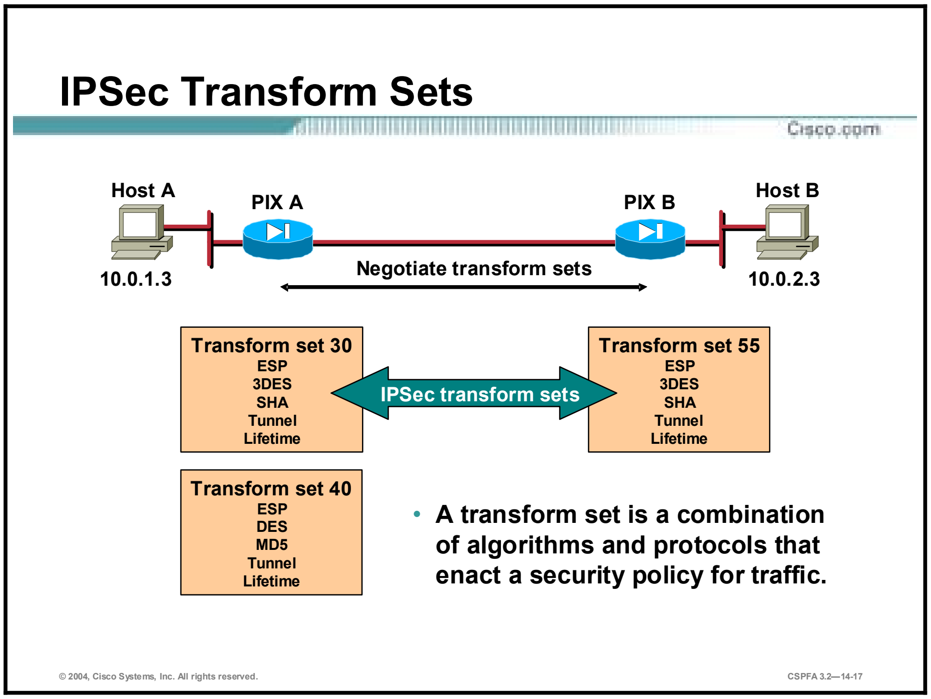
CSPFA 3.2—14-16

The purpose of IKE Phase 2 is to negotiate the IPSec security parameters used to secure the IPSec tunnel. IKE Phase 2 performs the following functions:

- Negotiates IPSec security parameters and IPSec transform sets
- Establishes IPSec SAs
- Periodically renegotiates IPSec SAs to ensure security
- Optionally performs an additional DH exchange

IKE Phase 2 has one mode, called quick mode. Quick mode occurs after IKE has established the secure tunnel in Phase 1. It negotiates a shared IPSec transform, derives shared secret keying material used for the IPSec security algorithms, and establishes IPSec SAs. Quick mode exchanges nonces that are used to generate new shared secret key material and prevent replay attacks from generating bogus SAs.

Quick mode is also used to renegotiate a new IPSec SA when the IPSec SA lifetime expires. Quick mode is used to refresh the keying material used to create the shared secret key based on the keying material derived from the DH exchange in Phase 1.



IPSec Transform Sets

Cisco.com

Host A

10.0.1.3

PIX A

Negotiate transform sets

PIX B

Host B

10.0.2.3

Transform set 30
ESP
3DES
SHA
Tunnel
Lifetime

IPSec transform sets

Transform set 55
ESP
3DES
SHA
Tunnel
Lifetime

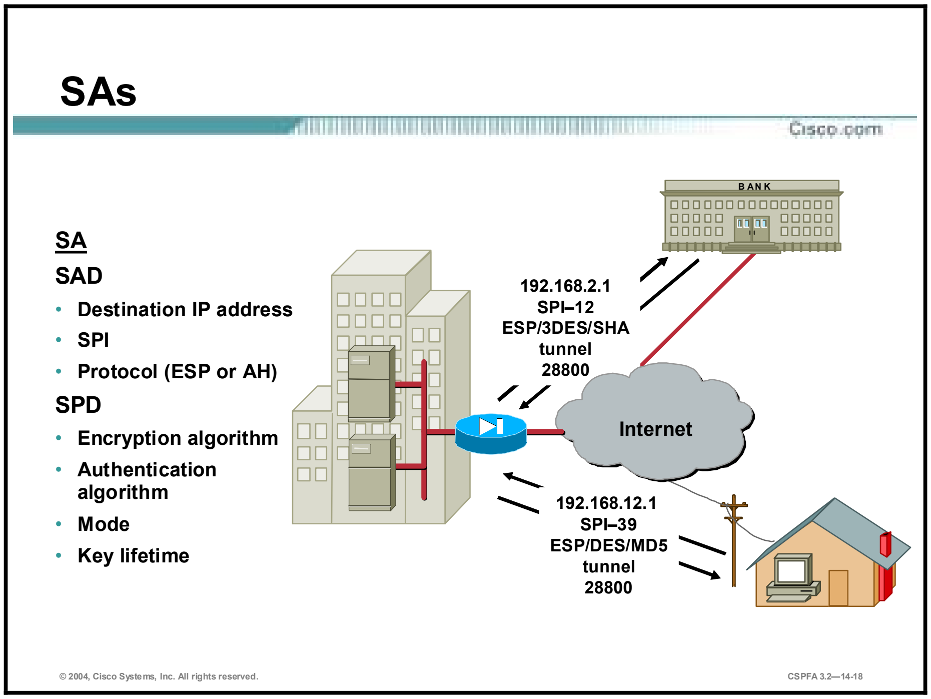
Transform set 40
ESP
DES
MD5
Tunnel
Lifetime

- **A transform set is a combination of algorithms and protocols that enact a security policy for traffic.**

CSPFA 3.2—14-17

The ultimate goal of IKE Phase 2 is to establish a secure IPSec session between endpoints. Before that can happen, each pair of endpoints negotiates the level of security required (for example, encryption and authentication algorithms for the session). Rather than negotiate each protocol individually, the protocols are grouped into sets, called IPSec transform sets. IPSec transform sets are exchanged between peers during quick mode. If a match is found between sets, IPSec session-establishment continues. If no match is found, the session is halted.

In the example in the figure, PIX Firewall A sends IPSec transform set 30 and 40 to PIX Firewall B. PIX Firewall B compares its set, transform set 55, with those received from PIX Firewall A. In this instance, there is a match. PIX Firewall A's transform set 30 matches PIX Firewall B's transform set 55. These encryption and authentication algorithms form an SA.



SAs

SA

SAD

- **Destination IP address**
- **SPI**
- **Protocol (ESP or AH)**

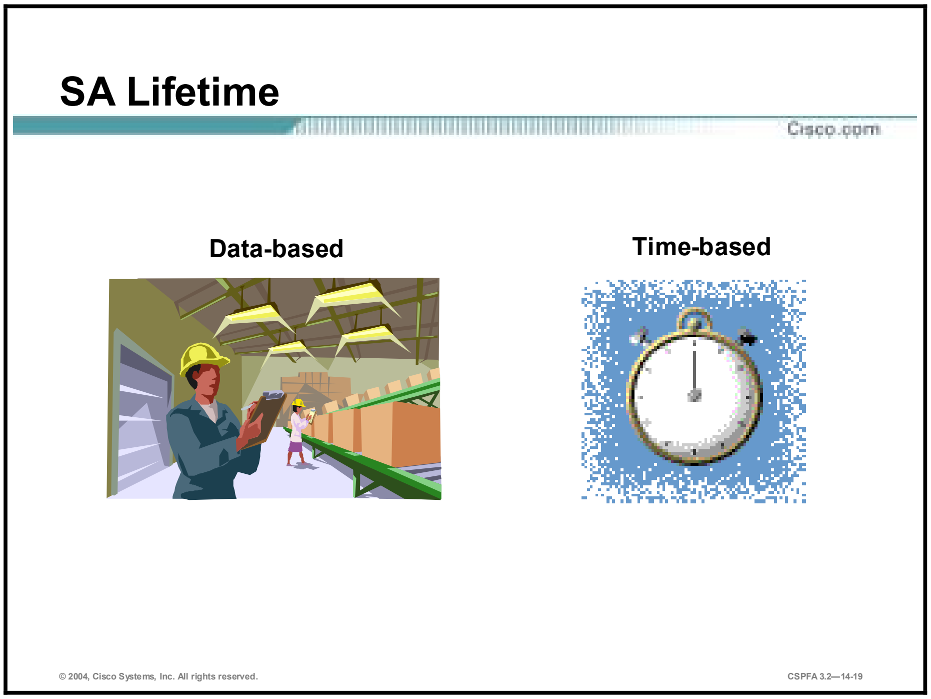
SPD

- **Encryption algorithm**
- **Authentication algorithm**
- **Mode**
- **Key lifetime**

CSPFA 3.2—14-18

When the security services are agreed upon between peers, each VPN peer device enters the information in a security policy database (SPD). The information includes the encryption and authentication algorithm, destination IP address, transport mode, key lifetime, and so on. This information is referred to as the SA. An SA is a one-way logical connection that provides security to all traffic traversing the connection. Because most traffic is bidirectional, two SAs are required: one for inbound and one for outbound traffic. The VPN device indexes the SA with a number, a Security Parameter Index (SPI). Rather than send the individual parameters of the SA across the tunnel, the source gateway, or host, inserts the SPI into the ESP header. When the IPSec peer receives the packet, it looks up the destination IP address, IPSec protocol, and SPI in its SA database (SAD), and then processes the packet according to the algorithms listed under the SPD.

The IPSec SA is a compilation of the SAD and SPD. SAD is used to identify the SA destination IP address, IPSec protocol, and SPI number. The SPD defines the security services applied to the SA, encryption and authentication algorithms, and mode and key lifetime. For example, in the corporate-to-bank connection, the security policy provides a very secure tunnel using 3DES, SHA, tunnel mode, and a key lifetime of 28800. The SAD value is 192.168.1.1, ESP, and SPI-12. For the remote user accessing e-mail, a less secure policy is negotiated using DES, MD5, tunnel mode, and a key lifetime of 28800. The SAD values are a destination IP address of 192.168.6.1, ESP, and an SPI-39.



SA Lifetime

Cisco.com

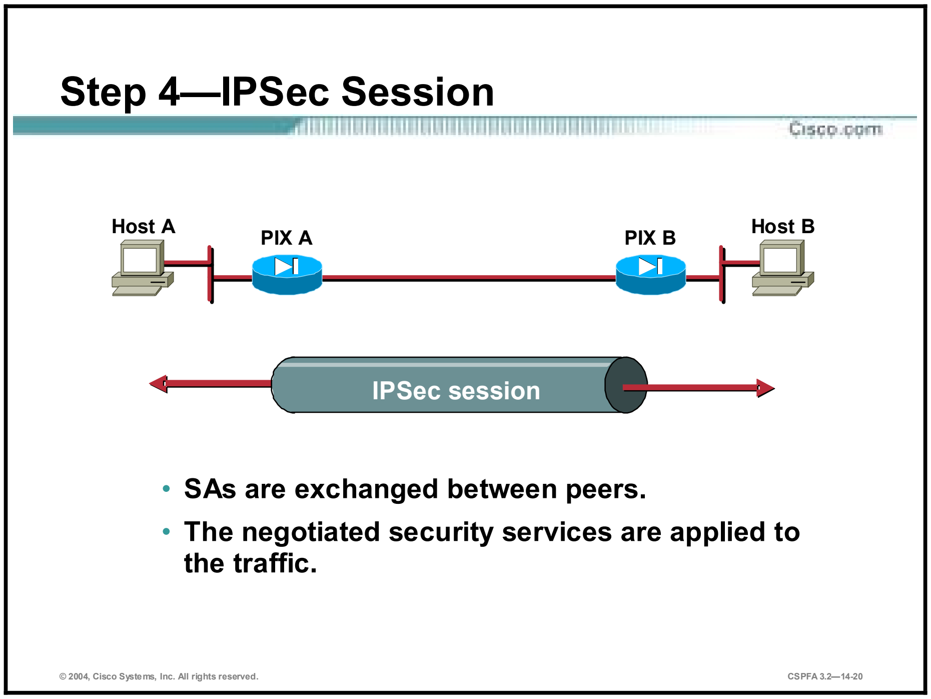
Data-based

Time-based

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—14-19

The longer you keep a password on your company PC, the more vulnerable it becomes. The same is true of keys and Security Associations (SAs). For good security, the SA and keys should be changed periodically. There are two parameters to consider: lifetime type and duration. The first parameter, lifetime type, defines how the lifetime is measured, by the number of bytes transmitted or the amount of time transpired? The second parameter, the duration, is expressed in either kilobytes of data or seconds of time. For example, you might specify a lifetime based on 10,000 kilobytes of data transmitted or 28,800 seconds of time expired. The keys and SAs remain active until their lifetime expires or until some external event—the client drops the tunnel—causes them to be deleted.



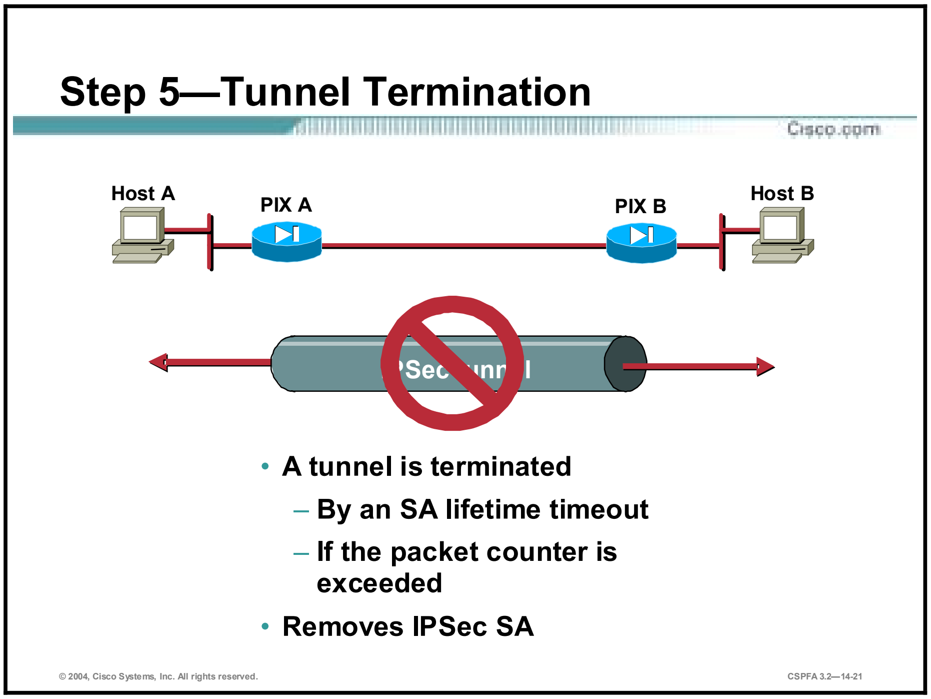
Step 4—IPSec Session

Cisco.com

- SAs are exchanged between peers.
- The negotiated security services are applied to the traffic.

CSPFA 3.2—14-20

After IKE Phase 2 is complete and quick mode has established IPSec SAs, traffic is exchanged between hosts A and B via a secure tunnel. Interesting traffic is encrypted and decrypted according to the security services specified in the IPSec SA.



Step 5—Tunnel Termination

Host A
PIX A
PIX B
Host B
IPSec Tunnel

- A tunnel is terminated
  - By an SA lifetime timeout
  - If the packet counter is exceeded
- Removes IPSec SA

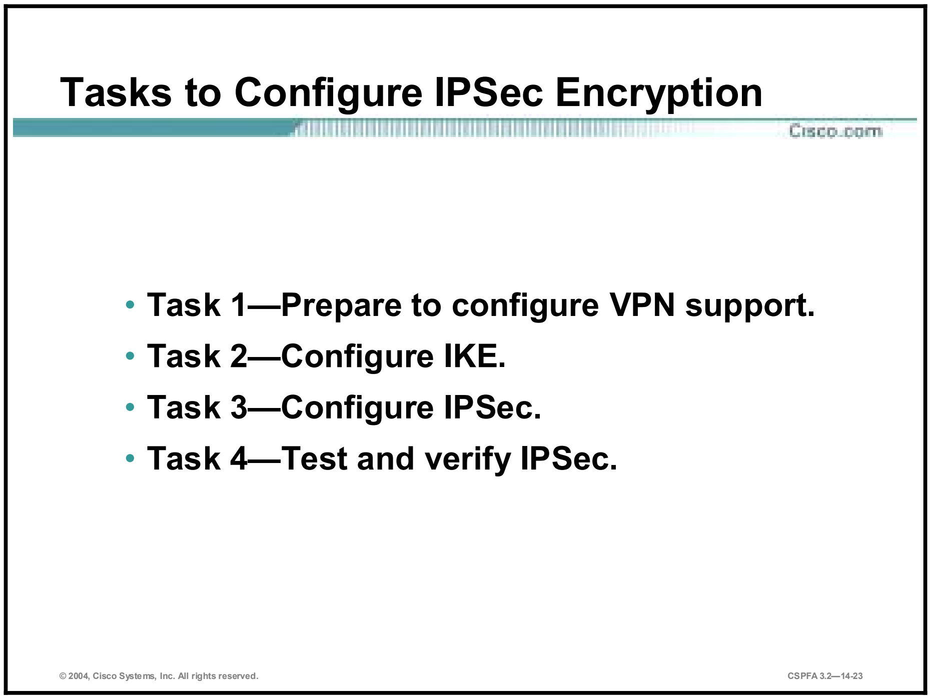
© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—14-21

IPSec SAs terminate through deletion or by timing out. An SA can time out when a specified number of seconds has elapsed or when a specified number of bytes has passed through the tunnel. When the SAs terminate, the keys are also discarded. When subsequent IPSec SAs are needed for a flow, IKE performs a new Phase 2, and, if necessary, a new Phase 1 negotiation. A successful negotiation results in new SAs and new keys. New SAs are usually established before the existing SAs expire, so that a given flow can continue uninterrupted.

IPSec Configuration Tasks

This topic describes the tasks you perform when configuring an IPSec-based VPN.



Tasks to Configure IPSec Encryption

- Task 1—Prepare to configure VPN support.
- Task 2—Configure IKE.
- Task 3—Configure IPSec.
- Task 4—Test and verify IPSec.

© 2004, Cisco Systems, Inc. All rights reserved.

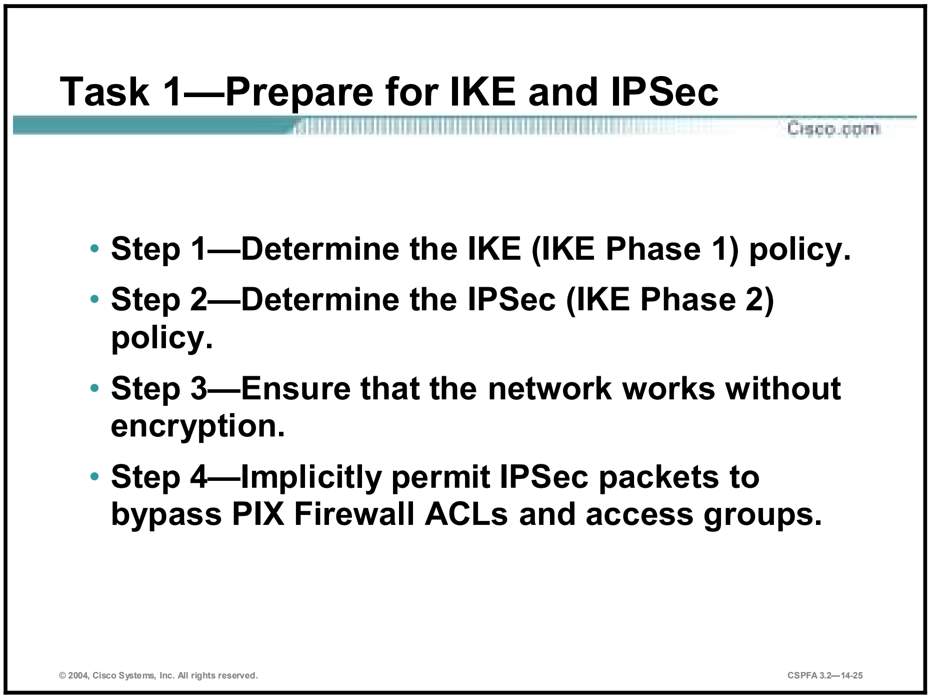
CSPFA 3.2—14-23

The rest of this lesson demonstrates how to configure an IPSec-based VPN between two PIX Firewalls operating as secure gateways, using pre-shared keys for authentication. The four overall tasks used to configure IPSec encryption on the PIX Firewall are summarized below. Subsequent topics of this lesson discuss each configuration task in greater detail. The following are the four tasks:

- Task 1 Prepare to configure VPN support. This task consists of several steps that determine IPSec policies, ensure that the network works, and ensure that the PIX Firewall can support IPSec.
- Task 2—Configure IKE parameters. This task consists of several configuration steps that ensure that IKE can set up secure channels to desired IPSec peers. IKE can set up IPSec SAs, enabling IPSec sessions. IKE negotiates IKE parameters and sets up IKE SAs during an IKE Phase 1 exchange called main mode.
- Task 3 Configure IPSec parameters. This task consists of several configuration steps that specify IPSec SA parameters between peers, and set global IPSec values. IKE negotiates SA parameters and sets up IPSec SAs during an IKE Phase 2 exchange called quick mode.
- Task 4 Test and verify VPN configuration. After you configure IPSec, you will need to verify that you have configured it correctly, and ensure that it works.

Task 1—Prepare to Configure VPN Support

Successful implementation of an IPSec network requires advance preparation before beginning the configuration of individual PIX Firewalls. This topic outlines how to determine network design details.



Task 1—Prepare for IKE and IPSec

- Step 1—Determine the IKE (IKE Phase 1) policy.
- Step 2—Determine the IPSec (IKE Phase 2) policy.
- Step 3—Ensure that the network works without encryption.
- Step 4—Implicitly permit IPSec packets to bypass PIX Firewall ACLs and access groups.

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—14-25

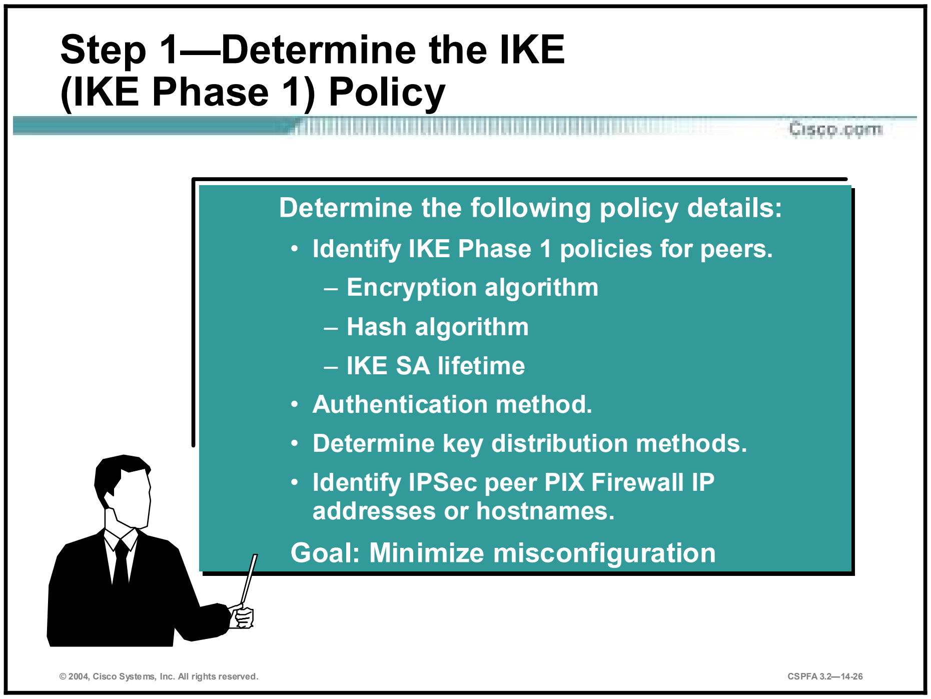
Configuring IPSec encryption can be complicated. You must plan in advance if you want to configure IPSec encryption correctly the first time and minimize misconfiguration. You should begin this task by defining the overall security needs and strategy based on the overall company security policy. Some planning steps include the following:

Step 1 Determine the IKE (IKE Phase 1) policy—Determine the IKE policies between peers based on the number and location of IPSec peers.

Step 2 Determine the IPSec (IKE Phase 2) policy Identify IPSec peer details such as IP addresses and IPSec modes. Determine the IPSec policies applied to the encrypted data passing between peers.

Step 3 Ensure that the network works without encryption—Ensure that basic connectivity has been achieved between IPSec peers using the desired IP services before configuring PIX Firewall IPSec.

Step 4 Implicitly permit IPSec packets to bypass PIX Firewall access control lists (ACLs), access groups, and conduits In this step you enter the **sysopt connection permit-ipsec** command.



Step 1—Determine the IKE (IKE Phase 1) Policy

Determine the following policy details:

- Identify IKE Phase 1 policies for peers.
  - Encryption algorithm
  - Hash algorithm
  - IKE SA lifetime
- Authentication method.
- Determine key distribution methods.
- Identify IPSec peer PIX Firewall IP addresses or hostnames.

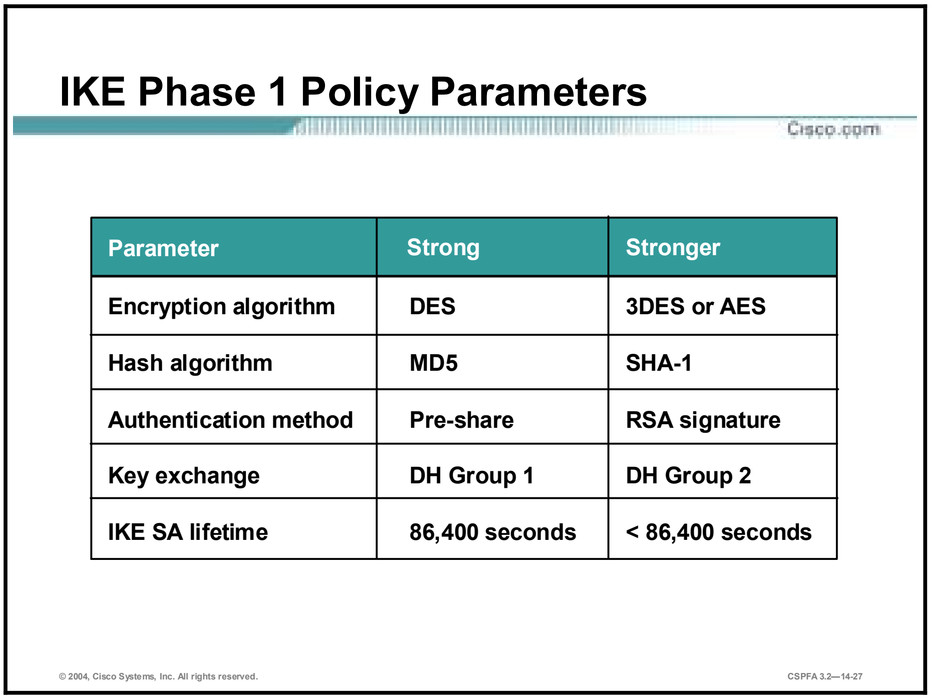
Goal: Minimize misconfiguration

CSPFA 3.2—14-26

An IKE policy defines a combination of security parameters to be used during the IKE negotiation. You should determine the IKE policy, and then configure it. Some planning steps include the following:

- Identify IKE Phase 1 (ISAKMP) policies for peers. An IKE policy defines a combination of security parameters to be used during the IKE negotiation. Each IKE negotiation begins by each peer agreeing on a common (shared) IKE policy. The IKE policy suites must be determined in advance of configuration.
- Choose the authentication method based on the key distribution method. Cisco PIX Firewall software supports either pre-shared keys or RSA Signatures to authenticate IPSec peers. This lesson focuses on using pre-shared keys.
- Determine key distribution methods based on the numbers and locations of IPSec peers. You may wish to use a CA server to support scalability of IPSec peers. You must then configure IKE to support the selected key distribution method.
- Identify IPSec peer PIX Firewall IP addresses or hostnames. You need to determine the details of all of the IPSec peers that will use IKE for establishing SAs.

The goal of advance planning is to minimize misconfiguration.



IKE Phase 1 Policy Parameters

Parameter	Strong	Stronger
Encryption algorithm	DES	3DES or AES
Hash algorithm	MD5	SHA-1
Authentication method	Pre-share	RSA signature
Key exchange	DH Group 1	DH Group 2
IKE SA lifetime	86,400 seconds	< 86,400 seconds

CSPFA 3.2—14-27

An IKE policy defines a combination of security parameters to be used during the IKE negotiation. A group of policies makes up a protection suite of multiple policies that enable IPSec peers to establish IKE sessions and SAs with a minimum of configuration.

Create IKE Policies for a Purpose

IKE negotiations must be protected, so each IKE negotiation begins by each peer agreeing on a common (shared) IKE policy. This policy states which security parameters will be used to protect subsequent IKE negotiations.

After the two peers agree upon a policy, an SA established at each peer identifies the security parameters of the policy. These SAs apply to all subsequent IKE traffic during the negotiation.

You can create multiple, prioritized policies at each peer to ensure that at least one policy will match a remote peer's policy.

Define IKE Policy Parameters

You can select specific values for each IKE parameter, per the IKE standard. You choose one value over another based on the security level you desire and the type of IPSec peer to which you will connect.

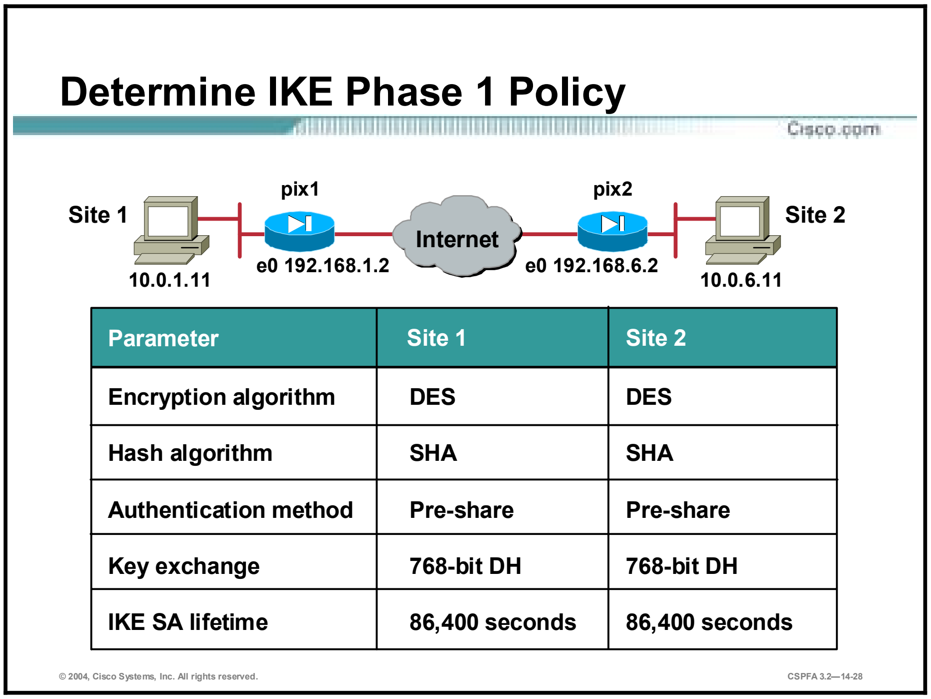
There are five parameters to define in each IKE policy, as outlined in the previous figure and in the following table. The figure shows the relative strength of each parameter, and the table shows the default values.

IKE Policy Parameters

Parameter	Accepted Values	Keyword	Default
Message encryption algorithm	56-bit DES 168-bit 3DES AES 128-bit key AES 192-bit key AES 256-bit key	**des** **3des** **aes** **aes-192** **aes-256**	DES
Message integrity (hash) algorithm	SHA-1 (HMAC variant) MD5 (HMAC variant)	**sha** **md5**	SHA-1
Peer authentication method	Pre-shared keys RSA signatures	**pre-share** **rsa-sig**	RSA signatures
Key exchange parameters (Diffie-Hellman group identifier)	768-bit Diffie-Hellman 1024-bit Diffie-Hellman 1536-bit Diffie-Hellman	**1** **2** **5**	768-bit Diffie-Hellman
ISAKMP-established security association's lifetime	Can specify any number of seconds	—	86,400 seconds (1 day)

Note 3DES provides stronger encryption than DES. AES provides stronger security than DES and is computationally more efficient than 3DES. 3DES may be restricted for export or import into some countries.

Note RSA signatures are used with CA support and require enrollment to a CA server.



Determine IKE Phase 1 Policy

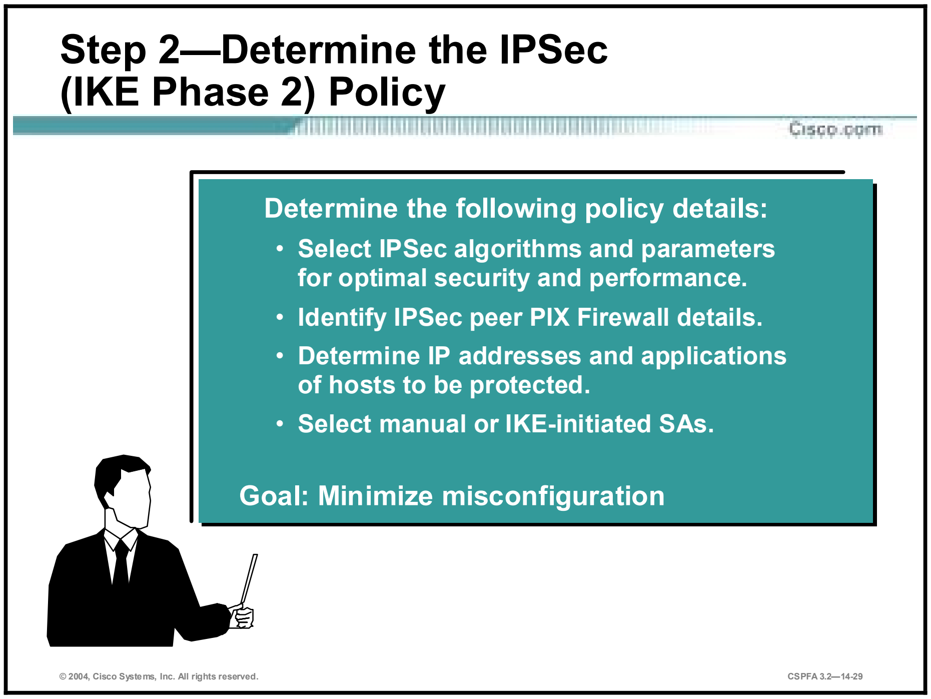
Cisco.com

Parameter	Site 1	Site 2
Encryption algorithm	DES	DES
Hash algorithm	SHA	SHA
Authentication method	Pre-share	Pre-share
Key exchange	768-bit DH	768-bit DH
IKE SA lifetime	86,400 seconds	86,400 seconds

CSPFA 3.2—14-28

An IKE policy defines a combination of security parameters to be used during the IKE negotiation. A group of policies makes up a protection suite of multiple policies that enable IPSec peers to establish IKE sessions and SAs with a minimum of configuration.

You should determine IKE policy details for each IPSec peer before configuring IKE. The figure shows a summary of some IKE policy details that will be configured in the examples in this lesson.



Step 2—Determine the IPSec (IKE Phase 2) Policy

Determine the following policy details:

- Select IPSec algorithms and parameters for optimal security and performance.
- Identify IPSec peer PIX Firewall details.
- Determine IP addresses and applications of hosts to be protected.
- Select manual or IKE-initiated SAs.

Goal: Minimize misconfiguration

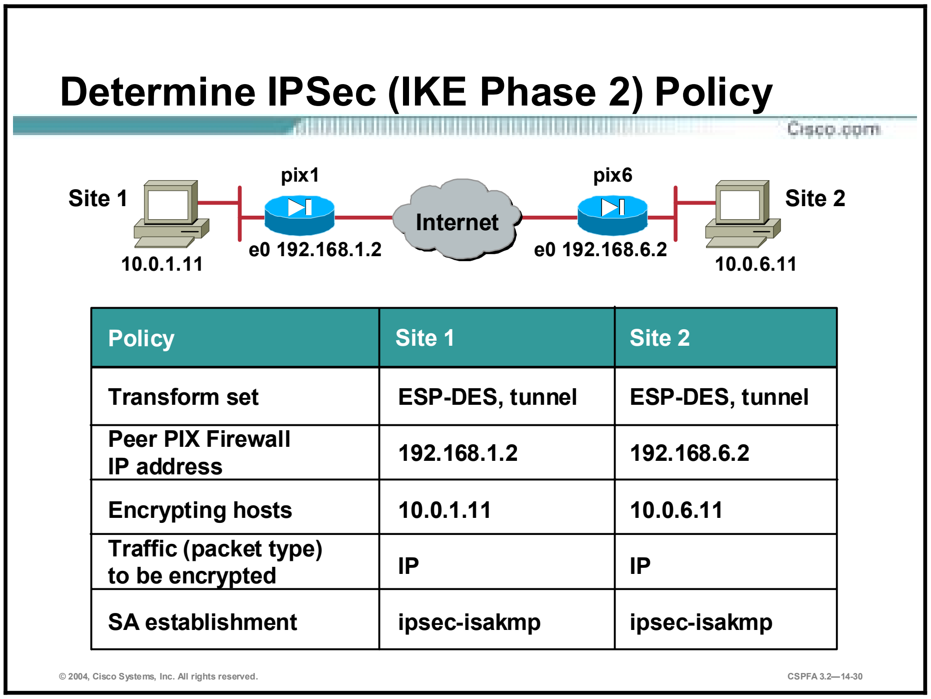
CSPFA 3.2—14-29

Planning for IPSec (IKE Phase 2) is another important step you should complete before actually configuring the PIX Firewall. Items to determine at this stage include the following:

- Select IPSec algorithms and parameters for optimal security and performance. You should determine what type of IPSec security will be used to secure interesting traffic. Some IPSec parameters require you to make tradeoffs between high performance and stronger security.
- Identify IPSec peer details. You must identify the IP addresses and hostnames of all IPSec peers that you will connect to.
- Determine IP addresses and applications of hosts to be protected at the local peer and remote peer.
- Decide whether SAs are manually established or are established via IKE.

Note IPSec SAs can be configured manually, but this practice is not recommended because IKE is easier to configure.

The goal of this planning step is to gather the precise data you will need in later steps to minimize misconfiguration.



Determine IPSec (IKE Phase 2) Policy

Policy	Site 1	Site 2
Transform set	ESP-DES, tunnel	ESP-DES, tunnel
Peer PIX Firewall IP address	192.168.1.2	192.168.6.2
Encrypting hosts	10.0.1.11	10.0.6.11
Traffic (packet type) to be encrypted	IP	IP
SA establishment	ipsec-isakmp	ipsec-isakmp

CSPFA 3.2—14-30

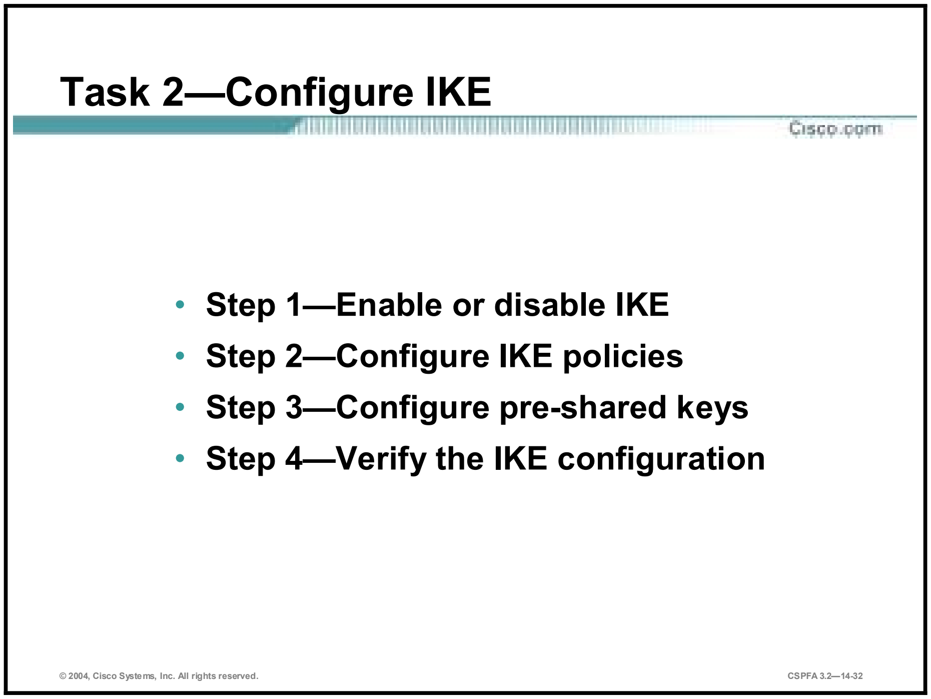
Determining network design details includes defining a more detailed security policy for protecting traffic. You can then use the detailed policy to help select IPSec transform sets and modes of operation. Your security policy should answer the following questions:

- What protections are required or are acceptable for the protected traffic?
- What traffic should or should not be protected?
- Which PIX Firewall interfaces are involved in protecting internal networks, external networks, or both?
- What are the peer IPSec endpoints for the traffic?
- How should SAs be established?

The figure above shows a summary of IPSec encryption policy details that will be configured in the examples later in this lesson.

Task 2—Configure IKE Parameters

The next major task in configuring PIX Firewall IPSec is to configure IKE parameters gathered in the previous task. This topic presents the steps used to configure IKE policies.



Task 2—Configure IKE

- Step 1—Enable or disable IKE
- Step 2—Configure IKE policies
- Step 3—Configure pre-shared keys
- Step 4—Verify the IKE configuration

CSPFA 3.2—14-32

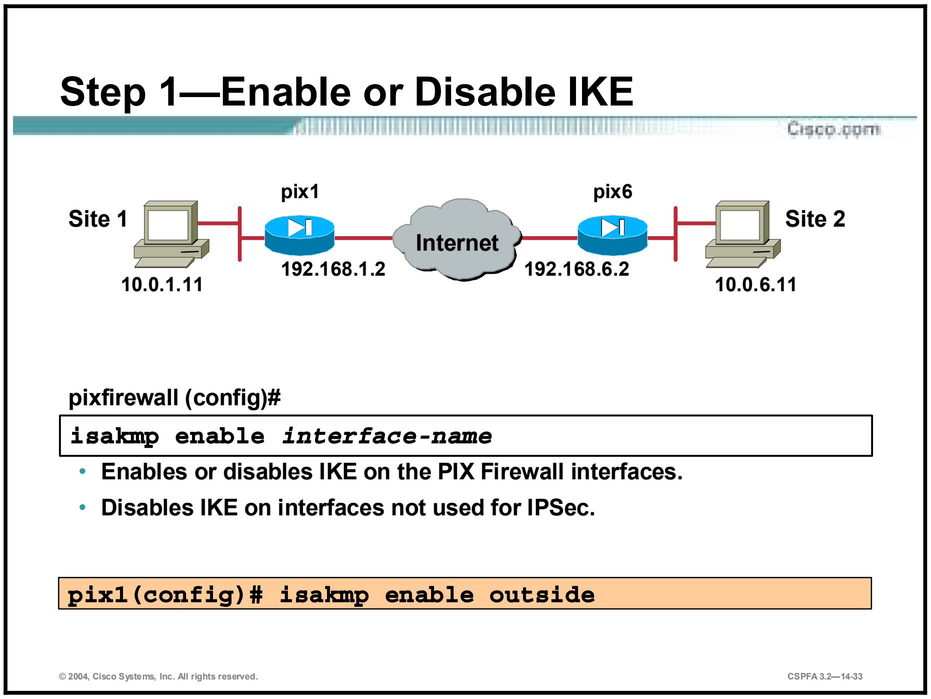
Configuring IKE consists of the following essential steps:

Step 1 Enable or disable IKE.

Step 2 Configure an IKE Phase 1 policy.

Step 3 Configure the IKE pre-shared keys.

Step 4 Verify IKE Phase 1 configuration.



Step 1—Enable or Disable IKE

pixfirewall (config)#

```
isakmp enable interface-name
```

- Enables or disables IKE on the PIX Firewall interfaces.
- Disables IKE on interfaces not used for IPSec.

```
pix1(config)# isakmp enable outside
```

CSPFA 3.2—14-33

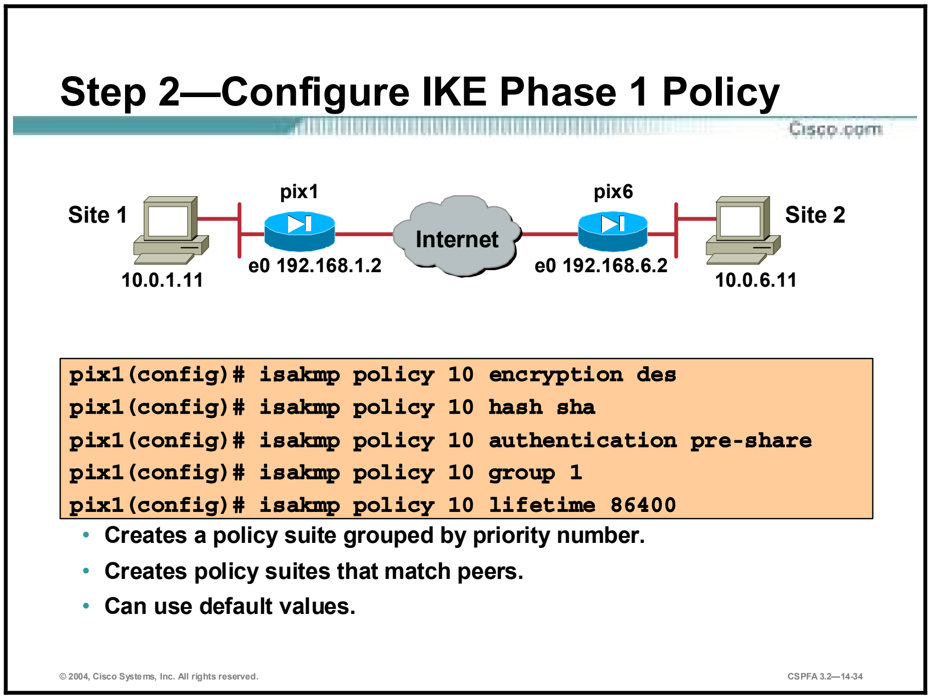
The following gives detail on the steps for configuring IKE:

Step 1 Enable or disable IKE (ISAKMP) negotiation:

```
pixfirewall(config)# isakmp enable interface-name
```

Interface-name	The name of the interface on which to enable ISAKMP negotiation.

Specify the PIX Firewall interface on which the IPSec peer will communicate. IKE is enabled by default and for individual PIX Firewall interfaces. Use the **no isakmp enable** *interface-name* command to disable IKE.



Step 2—Configure IKE Phase 1 Policy

```
pix1(config)# isakmp policy 10 encryption des
pix1(config)# isakmp policy 10 hash sha
pix1(config)# isakmp policy 10 authentication pre-share
pix1(config)# isakmp policy 10 group 1
pix1(config)# isakmp policy 10 lifetime 86400
```

- Creates a policy suite grouped by priority number.
- Creates policy suites that match peers.
- Can use default values.

CSPFA 3.2—14-34

Step 2 Configure an IKE Phase 1 policy with the **isakmp policy** command to match expected IPSec peers by completing the following substeps:

1. Identify the policy with a unique priority designation, according to the table.

```
pixfirewall(config)# isakmp policy priority
```

aes	Specifies that encrypted IKE messages protected by this suite are encrypted using AES with a 128-bit key.
aes-192	Specifies that encrypted IKE messages protected by this suite are encrypted using AES with a 192-bit key.
aes-256	Specifies that encrypted IKE messages protected by this suite are encrypted using AES with a 256-bit key.
des	Specifies 56-bit DES-cipher block chaining (CBC) as the encryption algorithm to be used in the IKE policy.
3des	Specifies that the Triple DES encryption algorithm is to be used in the IKE policy.
group 1	Specifies that the 768-bit Diffie-Hellman group is to be used in the IKE policy. This is the default value.
group 2	Specifies that the 1024-bit Diffie-Hellman group 2 be used in the IKE policy.
group 5	Specifies that the 1536-bit Diffie-Hellman group 5 be used in the IKE policy.
lifetime seconds	Specifies how many seconds each security association should exist before expiring. To propose a finite lifetime, use an integer from 120 to 86,400 seconds (one day). Specify 0 seconds for infinite lifetime.
md5	Specifies MD5 (HMAC variant) as the hash algorithm to be used in the IKE policy.

pre-share	Specifies pre-shared keys as the authentication method.
priority	Uniquely identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 65,534, with 1 being the highest priority and 65,534 the lowest.
rsa-sig	Specify RSA Signatures as the authentication method. RSA Signatures provide non-repudiation for the IKE negotiation. This basically means you can prove to a third party whether you had an IKE negotiation with the peer.
sha	Specify SHA-1 (HMAC variant) as the hash algorithm to be used in the IKE policy. This is the default hash algorithm.

2. Specify the encryption algorithm (the default is des):

```
pixfirewall(config)# isakmp policy priority encryption {des | 3des |
aes |aes-192 | aes-256}
```

3. Specify the hash algorithm (the default is **sha**):

```
pixfirewall(config)# isakmp policy priority hash {md5 | sha}
```

4. Specify the authentication method:

```
pixfirewall(config)# isakmp policy priority authentication {pre-share
| rsa-sig}
```

Note If you specify the authentication method of pre-shared keys, you must manually configure these keys, which is outlined in Step 3.

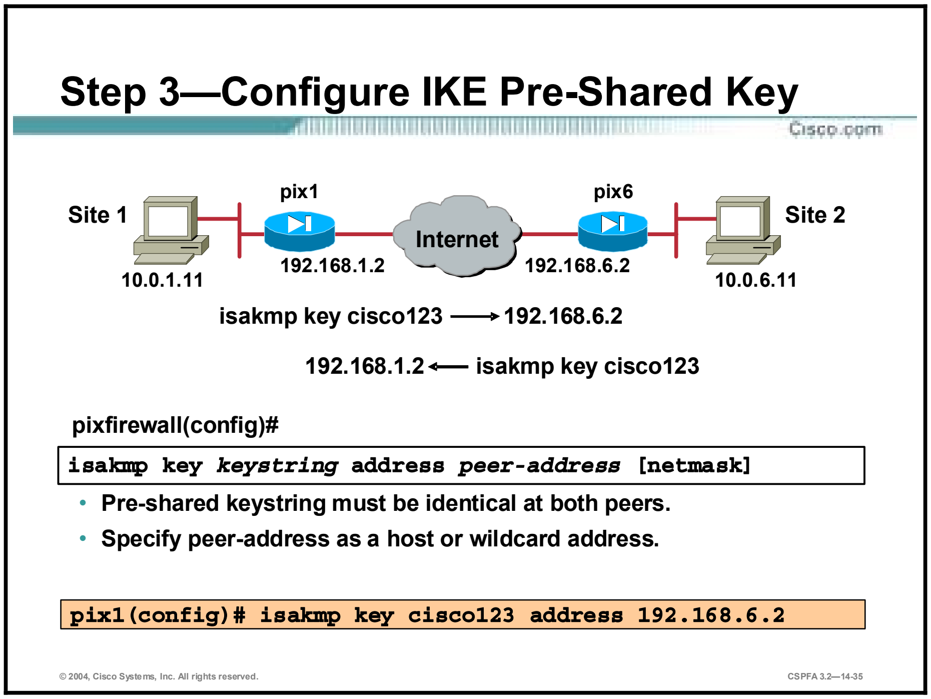
5. Specify the DH group identifier (the default is group 1):

```
pixfirewall(config)# isakmp policy priority group 1 | 2 | 5
```

6. Specify the IKE SA's lifetime (the default is 86,400):

```
pixfirewall(config)# isakmp policy priority lifetime seconds
```

Note PIX Firewall software has preset default values. If you enter a default value for a given policy parameter, it will not be written in the configuration. If you do not specify a value for a given policy parameter, the default value is assigned. You can observe configured and default values with the **show isakmp policy** command.



Step 3—Configure IKE Pre-Shared Key

Cisco.com
Site 1
pix1
pix6
Site 2
Internet
10.0.1.11
192.168.1.2
192.168.6.2
10.0.6.11
isakmp key cisco123 → 192.168.6.2
192.168.1.2 ← isakmp key cisco123

pixfirewall(config)#

```
isakmp key keystring address peer-address [netmask]
```

- Pre-shared keystring must be identical at both peers.
- Specify peer-address as a host or wildcard address.

```
pix1(config)# isakmp key cisco123 address 192.168.6.2
```

CSPFA 3.2—14-35

Step 3 Configure the IKE pre-shared key:

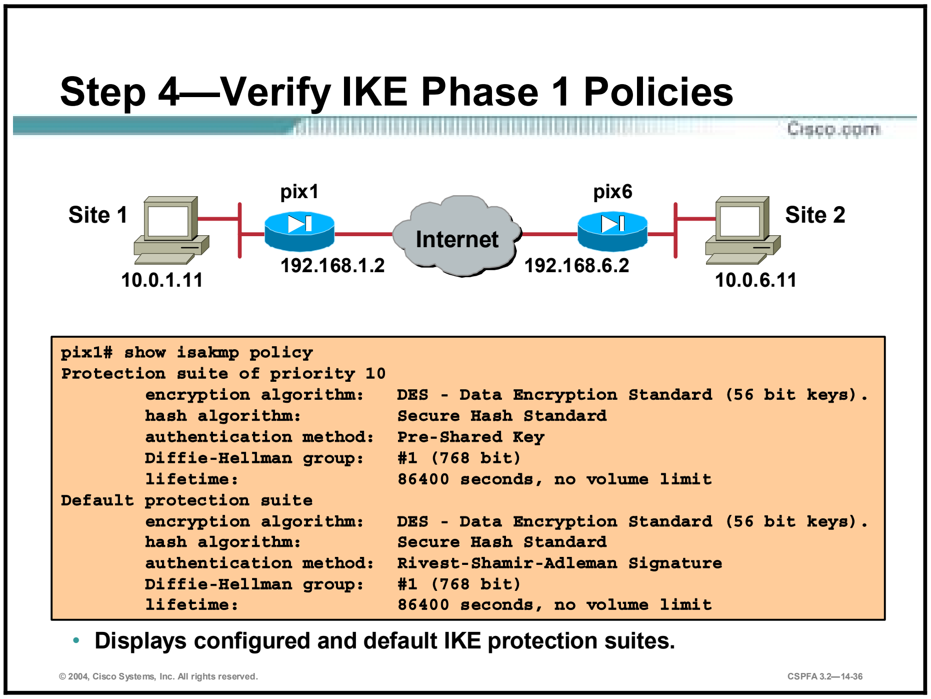
```
pixfirewall(config)# isakmp key keystring address peer-address
[netmask mask]
```

key	Specifies the authentication pre-shared key. Use any combination of alphanumeric characters up to 128 bytes. This pre-shared key must be identical at both peers.
peer-address	Specifies the IPSec peer's IP address for the pre-shared key.

The keystring is any combination of alphanumeric characters up to 128 bytes. This pre-shared key must be identical at both peers.

The peer-address and netmask should point to the IP address of the IPSec peer. A wildcard peer address and netmask of 0.0.0.0 0.0.0.0 may be configured to share the pre-shared key among many peers. However, it is strongly recommended that you use a unique key for each peer.

You can also use the peer's hostname for the pre-shared key.



Step 4—Verify IKE Phase 1 Policies

Cisco.com
Site 1
pix1
pix6
Site 2
Internet
10.0.1.11
192.168.1.2
192.168.6.2
10.0.6.11

```
pix1# show isakmp policy
Protection suite of priority 10
        encryption algorithm:   DES - Data Encryption Standard (56 bit keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #1 (768 bit)
        lifetime:               86400 seconds, no volume limit
Default protection suite
        encryption algorithm:   DES - Data Encryption Standard (56 bit keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Rivest-Shamir-Adleman Signature
        Diffie-Hellman group:   #1 (768 bit)
        lifetime:               86400 seconds, no volume limit
```

- **Displays configured and default IKE protection suites.**

CSPFA 3.2—14-36

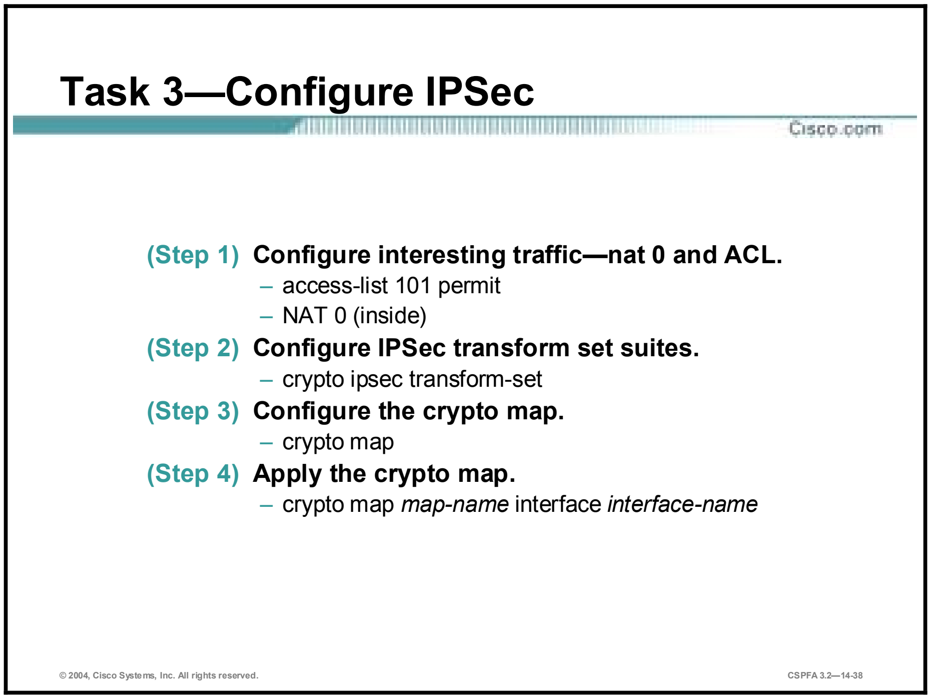
Step 4 Verify IKE Phase 1 policies.

The **show isakmp policy** command displays configured and default policies, as shown in the figure. The **show isakmp** command displays configured policies much as they would appear with the **write terminal** command, as follows:

```
pix1# show isakmp
isakmp enable outside
isakmp key ******** address 192.168.6.2 netmask 255.255.255.255
isakmp policy 10 authentication pre-share
isakmp policy 10 encryption des
isakmp policy 10 hash sha
isakmp policy 10 group 1
isakmp policy 10 lifetime 86400
```

Task 3—Configure IPSec Parameters

The next major task in configuring PIX Firewall IPSec is to configure the previously gathered IPSec parameters. This topic presents the steps used to configure IPSec parameters for IKE pre-shared keys.



Task 3—Configure IPSec

(Step 1) **Configure interesting traffic—nat 0 and ACL.**
- access-list 101 permit
- NAT 0 (inside)

(Step 2) **Configure IPSec transform set suites.**
- crypto ipsec transform-set

(Step 3) **Configure the crypto map.**
- crypto map

(Step 4) **Apply the crypto map.**
- crypto map *map-name* interface *interface-name*

CSPFA 3.2—14-38

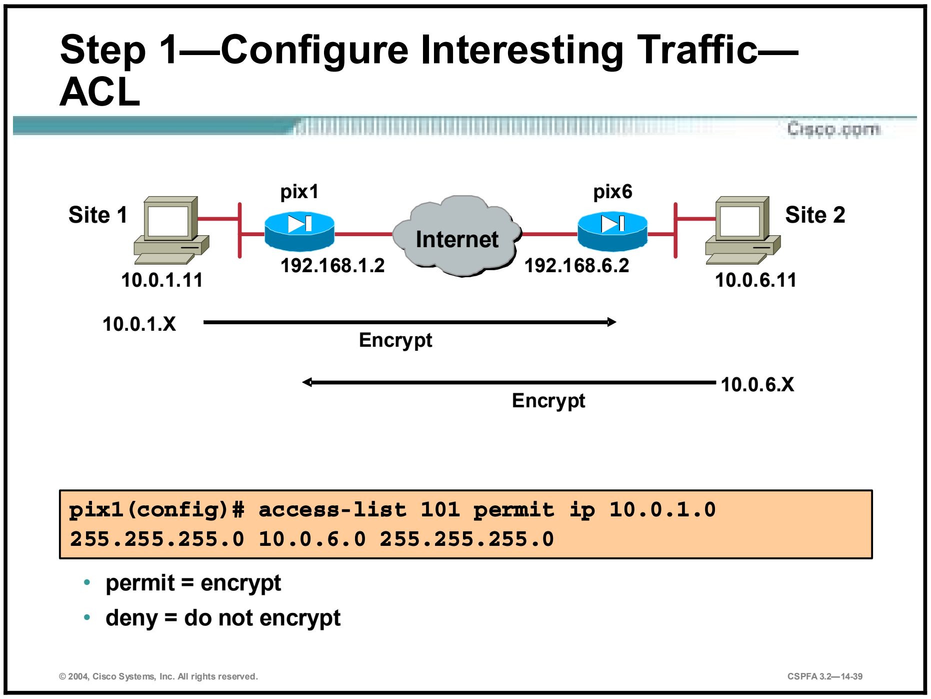
The tasks and commands used to configure IPSec encryption on a PIX Firewall are summarized as follows:

Step 1 Configure interesting traffic.

Step 2 Configure IPSec transform set suites.

Step 3 Configure the crypto map.

Step 4 Apply the crypto map to the interface.



Step 1—Configure Interesting Traffic—ACL

```
pix1(config)# access-list 101 permit ip 10.0.1.0
255.255.255.0 10.0.6.0 255.255.255.0
```

- permit = encrypt
- deny = do not encrypt

CSPFA 3.2—14-39

The following discusses each configuration step in greater detail.

Step 1 Configure the ACL as follows:

```
pixfirewall(config)# access-list acl_ID {deny | permit} protocol
source_addr source_mask destination_addr destination_mask
```

acl_ID	Name of an ACL. You can use either a name or number.
deny	Instructs the PIX Firewall to route traffic in the clear.
permit	Causes all IP traffic that matches the specified conditions to be protected by crypto, using the policy described by the corresponding crypto map entry.
protocol	Indicates which IP packet types to encrypt.
source_addr	Networks, subnets, or hosts.
source_mask	Netmask bits (mask) to be applied to source_addr.
destination_addr	Networks, subnets, or hosts.
destination_mask	Netmask bits (mask) to be applied to destination_addr.

Note PIX Firewall version 5.0 supports only the IP protocol. PIX Firewall versions 5.1 and higher support greater protocol and port granularity.

Crypto ACLs are traffic selection ACLs. They are used to define which IP traffic is interesting and will be protected by IPSec, and which traffic will not be protected by IPSec. Crypto ACLs perform the following functions:

- Indicate the data flow to be protected by IPSec
- Select outbound traffic to be protected by IPSec

- Process inbound traffic in order to filter out and discard traffic that should be protected by IPSec
- Determine whether or not to accept requests for IPSec SAs for the requested data flows when processing IKE negotiations

---

Note Although the ACL syntax is unchanged from the ACL applied to PIX Firewall interfaces, the meanings are slightly different for crypto ACLs—**permit** specifies that matching packets must be encrypted while **deny** specifies that matching packets need not be encrypted.

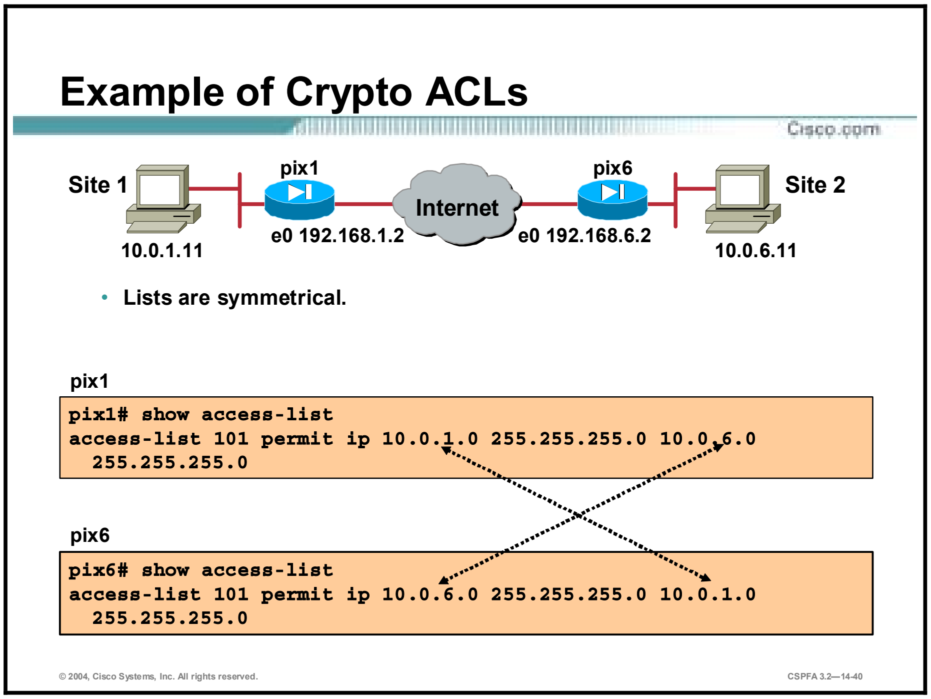
Any unprotected inbound traffic that matches a permit entry in the ACL for a crypto map entry, flagged as IPSec, will be dropped because this traffic was expected to be protected by IPSec.

If you want certain traffic to receive one combination of IPSec protection (for example, authentication only) and other traffic to receive a different combination of IPSec protection (for example, both authentication and encryption), you must create two different crypto ACLs to define the two different types of traffic. These different ACLs are then used in different crypto map entries, which specify different IPSec policies.

In a later configuration step, you will associate the crypto ACLs to particular interfaces when you configure and apply crypto map sets to the interfaces.

Caution It is recommended that you avoid using the ***any*** keyword to specify source or destination addresses. The ***permit any any*** statement is strongly discouraged, because this will cause all outbound traffic to be protected (and all protected traffic sent to the peer, specified in the corresponding crypto map entry), and will require protection for all inbound traffic. All inbound packets that lack IPSec protection will then be silently dropped, including packets for routing protocols, Network Time Protocol (NTP), echo, echo response, and so on.

Try to be as restrictive as possible when defining which packets to protect in a crypto ACL. If you must use the any keyword in a permit statement, you must preface that statement with a series of deny statements to filter out any traffic (that would otherwise fall within that permit statement) that you do not want protected.



Example of Crypto ACLs

- Lists are symmetrical.

pix1

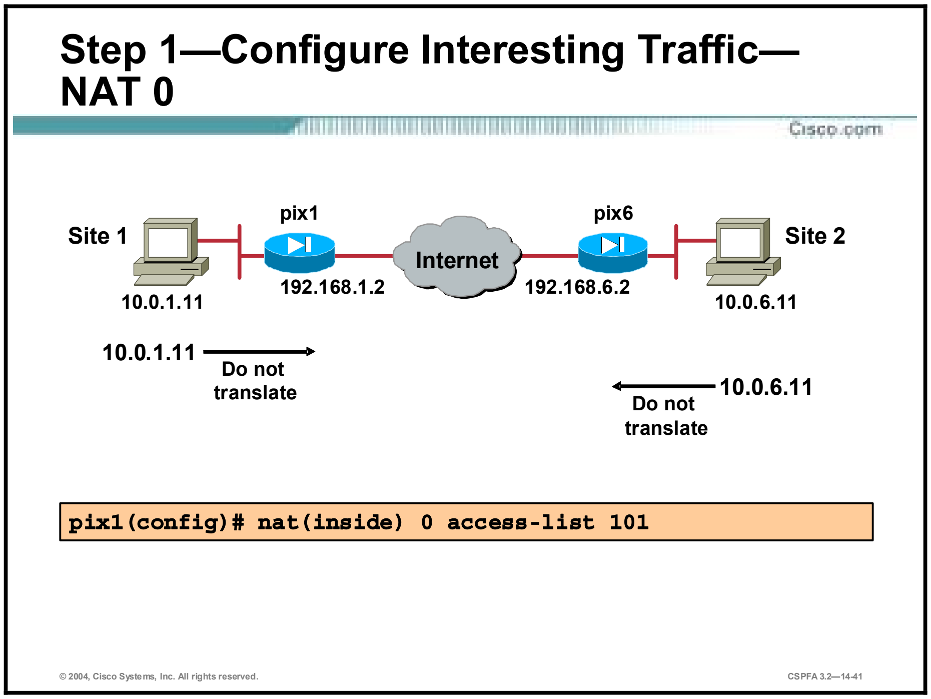
```
pix1# show access-list
access-list 101 permit ip 10.0.1.0 255.255.255.0 10.0.6.0
  255.255.255.0
```

pix6

```
pix6# show access-list
access-list 101 permit ip 10.0.6.0 255.255.255.0 10.0.1.0
  255.255.255.0
```

CSPFA 3.2—14-40

Use the **show access-list** command to display currently configured ACLs. The figure contains an example ACL for each of the peer PIX Firewalls. In the pix1 ACL, the source network is 10.0.1.0 and the destination network is 10.0.6.0. In the pix6 ACL, the source network is 10.0.6.0 and the destination address is 10.0.1.0. The ACLs are symmetrical.



Step 1—Configure Interesting Traffic—NAT 0

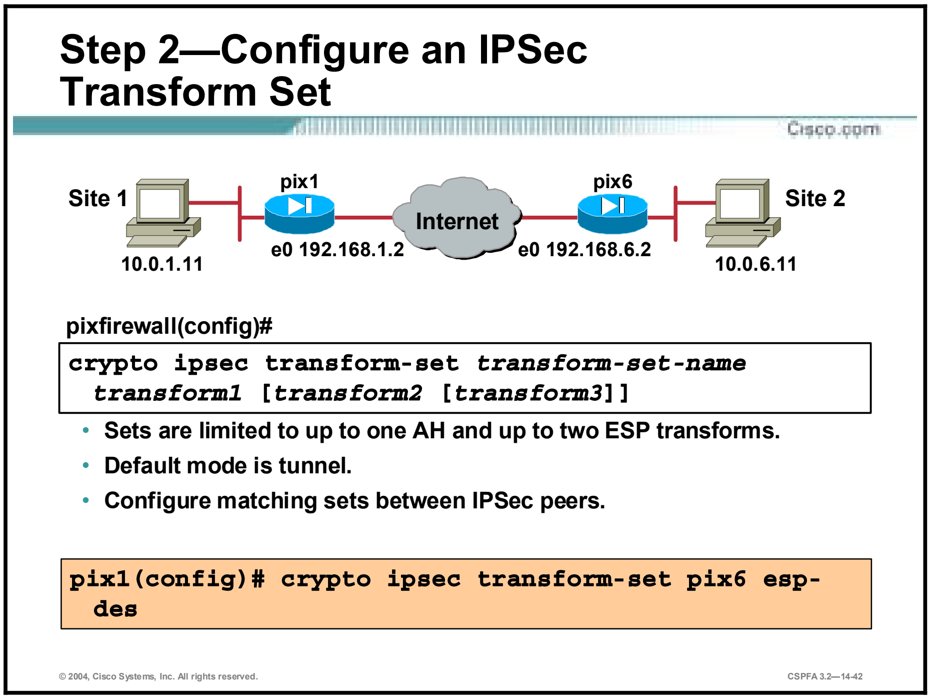
```
pix1(config)# nat(inside) 0 access-list 101
```

CSPFA 3.2—14-41

The **nat 0** command instructs the PIX Firewall not to use Network Address Translation (NAT) for any traffic deemed interesting traffic for IPSec. All traffic that matches the **access-list** command statement will be exempt from NAT services. In the figure, traffic matching access-list 101, traffic from 10.0.1.0/24 to 10.0.6.0/24, is exempt from NAT.

nat [(*if_name*)] **0** [**access-list** *acl_id*]

access-list	Associates **access-list** command statements with the **nat 0** command and exempts traffic that matches the ACL from NAT processing.
acl_id	The ACL name.
if_name	The name of the network interface, as specified by the **nameif** command, through which the hosts or network designated by *local_ip* are accessed.



Step 2—Configure an IPSec Transform Set

pixfirewall(config)#

```
crypto ipsec transform-set transform-set-name
  transform1 [transform2 [transform3]]
```

- Sets are limited to up to one AH and up to two ESP transforms.
- Default mode is tunnel.
- Configure matching sets between IPSec peers.

```
pix1(config)# crypto ipsec transform-set pix6 esp-
  des
```

CSPFA 3.2—14-42

Step 2 Configure an IPSec transform set:

```
pixfirewall(config)# crypto ipsec transform-set transform-set-name
transform1 [transform2 [transform3]]
```

transform-set-name	The name of the transform set to create or modify.
transform1 ***transform2*** ***transform3***	Specify up to three transforms.

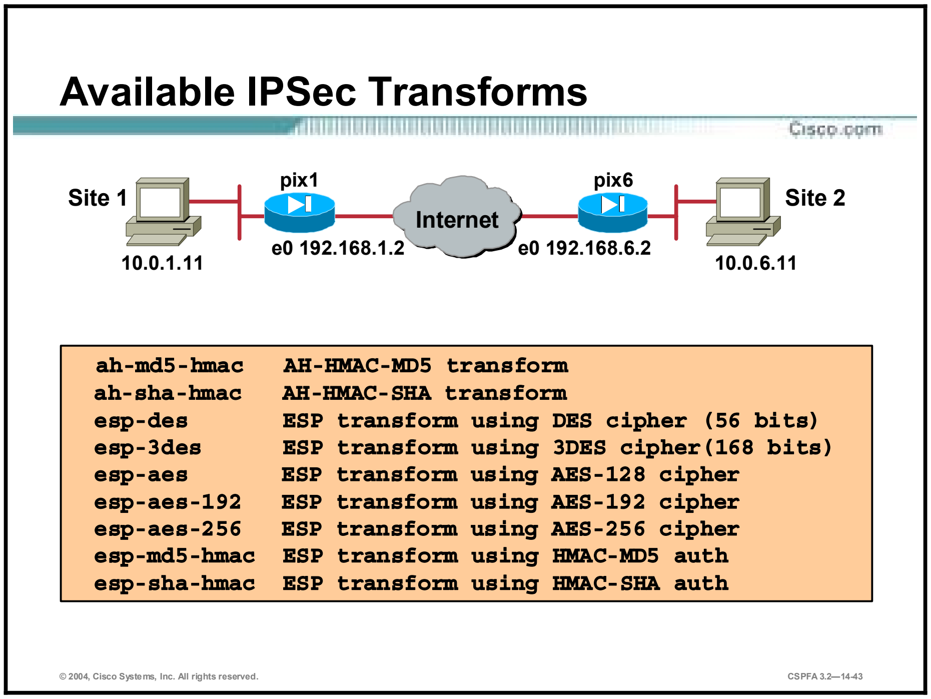
Transforms define the IPSec security protocols and algorithms. Each transform represents an IPSec security protocol (ESP, AH, or both) plus the algorithm you want to use.

You can specify multiple transform sets, and then specify one or more of these transform sets in a crypto map entry. The transform set defined in the crypto map entry will be used in the IPSec SA negotiation to protect the data flows specified by the ACL of that crypto map entry.

During the IPSec SA negotiation, the peers agree to use a particular transform set for protecting a particular data flow.

A transform set equals an AH transform and an ESP transform plus the mode (transport or tunnel). Transform sets are limited to one AH and two ESP transforms. The default mode is tunnel. Be sure to configure matching transform sets between IPSec peers.

Note In PIX Firewall versions 6.0 and higher, Layer 2 Tunneling Protocol (L2TP) is the only protocol that can use the IPSec transport mode. The PIX Firewall discards all other types of packets using IPSec transport mode.



Available IPSec Transforms

Cisco.com
Site 1
10.0.1.11
pix1
e0 192.168.1.2
Internet
pix6
e0 192.168.6.2
Site 2
10.0.6.11

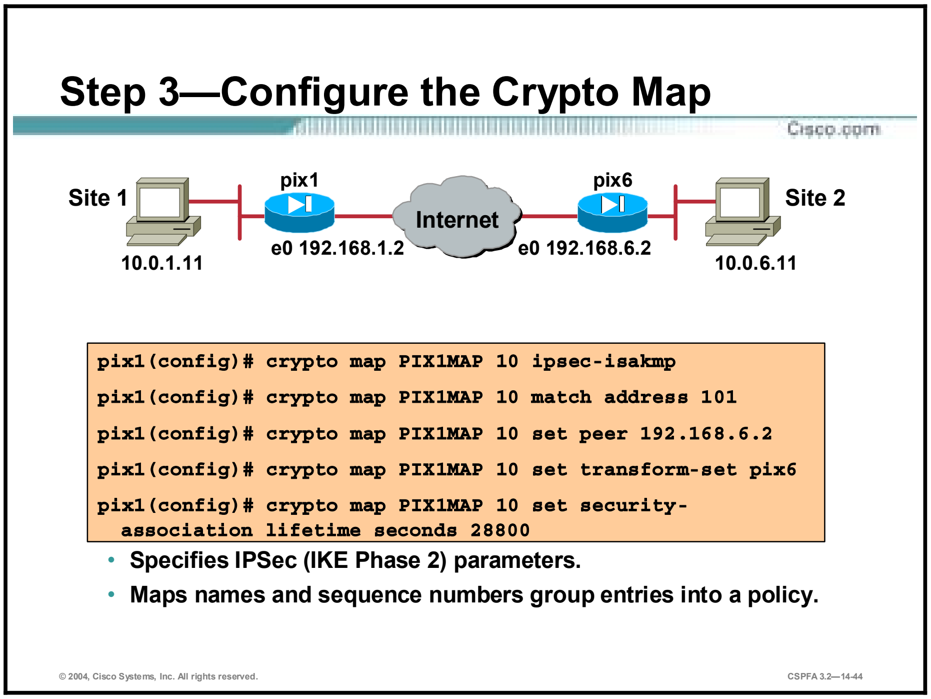
ah-md5-hmac	AH-HMAC-MD5 transform
ah-sha-hmac	AH-HMAC-SHA transform
esp-des	ESP transform using DES cipher (56 bits)
esp-3des	ESP transform using 3DES cipher(168 bits)
esp-aes	ESP transform using AES-128 cipher
esp-aes-192	ESP transform using AES-192 cipher
esp-aes-256	ESP transform using AES-256 cipher
esp-md5-hmac	ESP transform using HMAC-MD5 auth
esp-sha-hmac	ESP transform using HMAC-SHA auth

CSPFA 3.2—14-43

The PIX Firewall supports the transforms listed in the figure.

Choosing IPSec transforms combinations can be complex. The following tips may help you select transforms that are appropriate for your situation:

- If you want to provide data confidentiality, include an ESP encryption transform.
- Also consider including an ESP authentication transform or an AH transform to provide authentication services for the transform set:
  - To ensure data authentication for the outer IP header as well as the data, include an AH transform.
  - To ensure data authentication, use either ESP or AH. You can choose from the MD5 or SHA (HMAC keyed hash variants) authentication algorithms.
- The SHA algorithm is generally considered stronger than MD5, but it is slower.
- Examples of acceptable transform combinations are as follows:
  - esp-des for high performance encryption
  - ah-md5-hmac for authenticating packet contents with no encryption
  - esp-3des and esp-md5-hmac for strong encryption and authentication
  - ah-sha-hmac and esp-3des and esp-sha-hmac for strong encryption and authentication



Step 3—Configure the Crypto Map

```
pix1(config)# crypto map PIX1MAP 10 ipsec-isakmp
pix1(config)# crypto map PIX1MAP 10 match address 101
pix1(config)# crypto map PIX1MAP 10 set peer 192.168.6.2
pix1(config)# crypto map PIX1MAP 10 set transform-set pix6
pix1(config)# crypto map PIX1MAP 10 set security-
  association lifetime seconds 28800
```

- Specifies IPSec (IKE Phase 2) parameters.
- Maps names and sequence numbers group entries into a policy.

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—14-44

Step 3 Configure the crypto map with the **crypto map** command by completing the following substeps:

1. Create a crypto map entry in IPSec ISAKMP mode:

```
pixfirewall(config)# crypto map map-name seq-num {ipsec-isakmp |
ipsec-manual | [dynamic dynamic-map-name]}
```

Ipsec-isakmp	Indicate that IKE will be used to establish the IPSec security associations for protecting the traffic specified by this crypto map entry.
Ipsec-manual	Indicate that IKE will not be used to establish the IPSec security associations for protecting the traffic specified by this crypto map entry. Note Manual configuration of SAs is not supported on the PIX 501.
map *map-name*	The name of the crypto map set.

This identifies the crypto map with a unique crypto map name and sequence number.

2. Assign an ACL to the crypto map entry:

```
pixfirewall(config)# crypto map map-name seq-num match address access-
list-name
```

match address	Specify an access list for a crypto map entry.

3. Specify the peer to which the IPSec protected traffic can be forwarded:

```
pixfirewall(config)# crypto map map-name seq-num set peer {hostname
|ip-address}
```

hostname	Specify a peer by its IP address, or by its hostname as defined by the PIX Firewall **name** command.
seq-num	The number you assign to the crypto map entry.
set peer	Specify an IPSec peer in a crypto map entry.
Ip_address	Specify a peer by its IP address.

This specifies the peer hostname or IP address. You can specify multiple peers by repeating this command.

4. Specify which transform sets are allowed for this crypto map entry.

```
pixfirewall(config)# crypto map map-name seq-num set transform-set
[transform-set-name1 [transform-set-name2, transform-set-name6]]
```

seq-num	The number you assign to the crypto map entry.
set *transform-set*	Specify which transform sets can be used with the crypto map entry.
transform-set-name	The name of the transform set. For an ipsec-manual crypto map entry, you can specify only one transform set. For an ipsec-isakmp or dynamic crypto map entry, you can specify up to six transform sets.

If you use multiple transform sets, list them in order of priority (highest priority first). You can specify up to six transform sets.

5. (Optional.) Specify whether IPSec should ask for Perfect Forward Secrecy (PFS) when requesting new SAs for this crypto map entry, or should require PFS in requests received from the peer:

```
pixfirewall(config)# crypto map map-name seq-num set pfs [group1 |
group2]
```

group 1	Specify that IPSec should use the 768-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.
group 2	Specifies the IPSec peer's IP address for the pre-shared key. Specify that IPSec should use the 1024-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.
set pfs	Specify that IPSec should ask for PFS. With PFS, every time a new security association is negotiated, a new Diffie-Hellman exchange occurs. (This exchange requires additional processing time.)

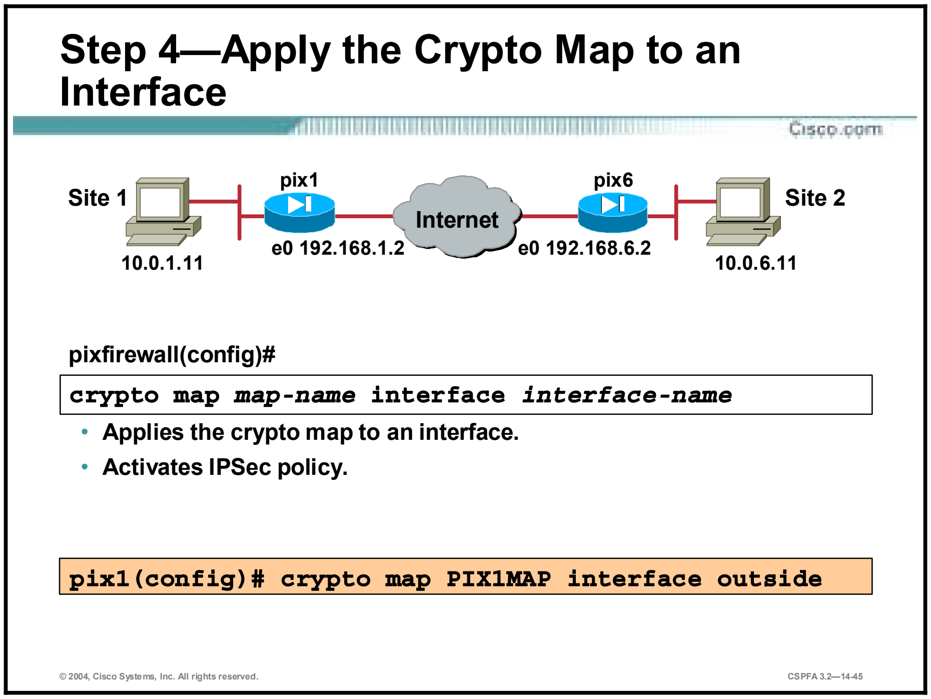
Note PFS provides additional security for DH key exchanges at a cost of additional processing.

6. (Optional.) Specify the SA lifetime for the crypto map entry if you want the SAs for this entry to be negotiated using different IPSec SA lifetimes other than the global lifetimes:

```
pixfirewall(config)# crypto map map-name seq-num set
security-association lifetime {seconds seconds | kilobytes kilobytes}
```

set security-association lifetime	Set the lifetime a security association will last in either seconds or kilobytes. For use with either **seconds** or **kilobyte** keywords.

7. (Optional.) Specify dynamic crypto maps with the **crypto dynamic-map** ***dynamic-map-name dynamic-seq-num*** command. A dynamic crypto map entry is essentially a crypto map entry without all the parameters configured. It acts as a policy template where the missing parameters are later dynamically configured (as the result of an IPSec negotiation) to match a peer's requirements. This allows peers to exchange IPSec traffic with the PIX Firewall even if the PIX Firewall does not have a crypto map entry specifically configured to meet all the peer's requirements.



Step 4—Apply the Crypto Map to an Interface

pixfirewall(config)#

```
crypto map map-name interface interface-name
```

- Applies the crypto map to an interface.
- Activates IPSec policy.

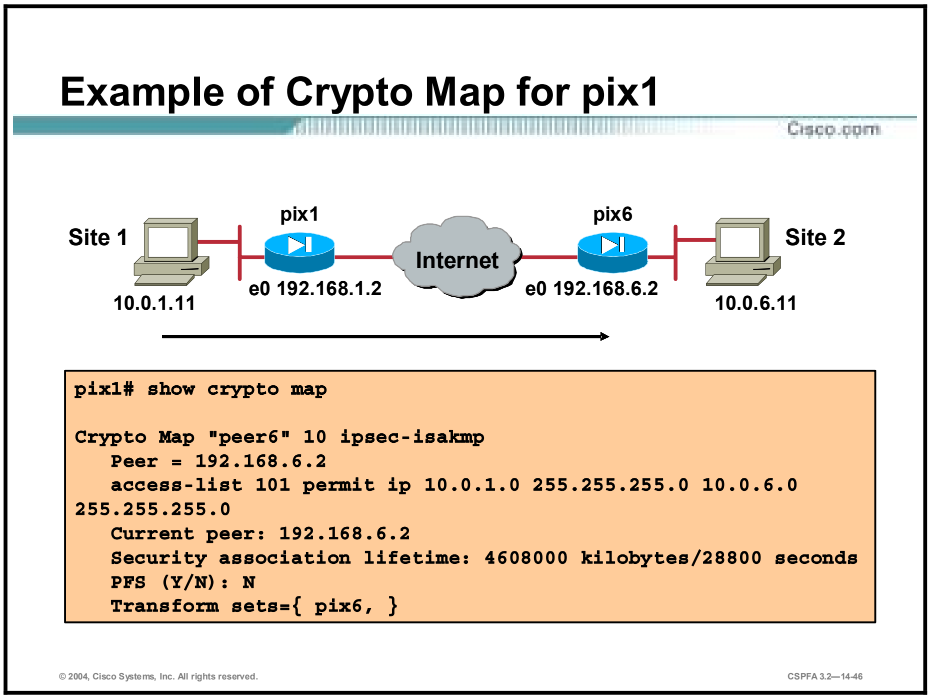
```
pix1(config)# crypto map PIX1MAP interface outside
```

Step 4 Apply the crypto map to an interface:

```
pixfirewall(config)# crypto map map-name interface interface-name
```

map ***map-name***	The name of the crypto map set.
interface ***interface-name***	Specifies the identifying interface to be used by the PIX Firewall to identify itself to peers. If IKE is enabled, and you are using a CA to obtain certificates, this should be the interface with the address specified in the CA certificates.

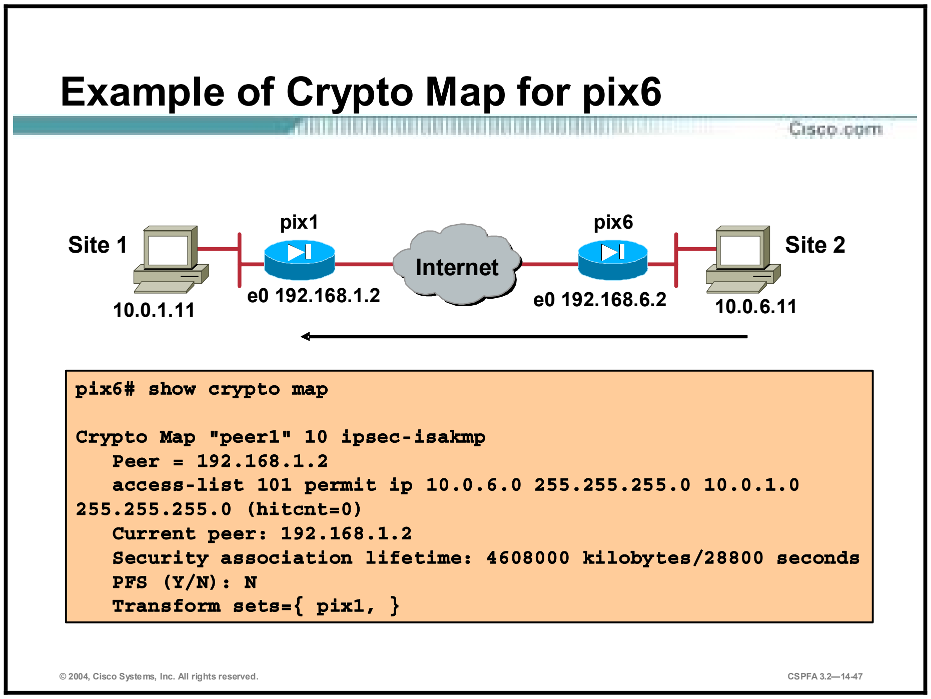
This command applies the crypto map to an interface and the command activates the IPSec policy.



Example of Crypto Map for pix1

```
pix1# show crypto map

Crypto Map "peer6" 10 ipsec-isakmp
   Peer = 192.168.6.2
   access-list 101 permit ip 10.0.1.0 255.255.255.0 10.0.6.0
255.255.255.0
   Current peer: 192.168.6.2
   Security association lifetime: 4608000 kilobytes/28800 seconds
   PFS (Y/N): N
   Transform sets={ pix6, }
```

Use the **show crypto map** command to verify the crypto map configuration. Consider the example of a crypto map for PIX Firewall 1 in the figure.



Example of Crypto Map for pix6

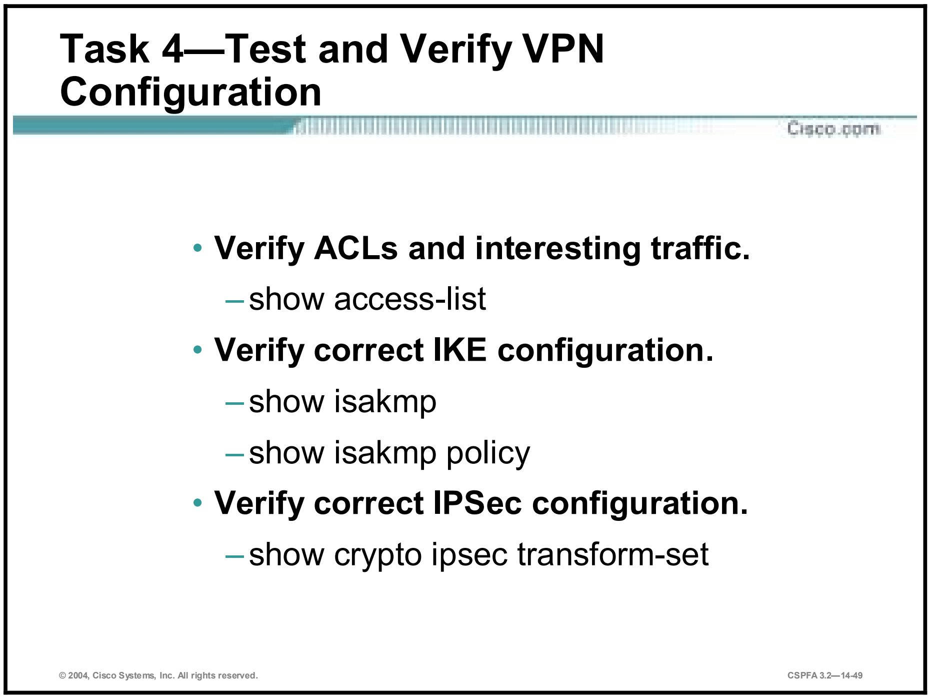
```
pix6# show crypto map

Crypto Map "peer1" 10 ipsec-isakmp
   Peer = 192.168.1.2
   access-list 101 permit ip 10.0.6.0 255.255.255.0 10.0.1.0
255.255.255.0 (hitcnt=0)
   Current peer: 192.168.1.2
   Security association lifetime: 4608000 kilobytes/28800 seconds
   PFS (Y/N): N
   Transform sets={ pix1, }
```

CSPFA 3.2—14-47

Consider the example of a crypto map for PIX Firewall 6 in the figure.

Task 4—Test and Verify VPN Configuration

The last major task in configuring PIX Firewall IPSec is to test and verify the IKE and IPSec configurations accomplished in the previous tasks. This topic presents the methods and commands used to test and verify VPN configuration.



Task 4—Test and Verify VPN Configuration

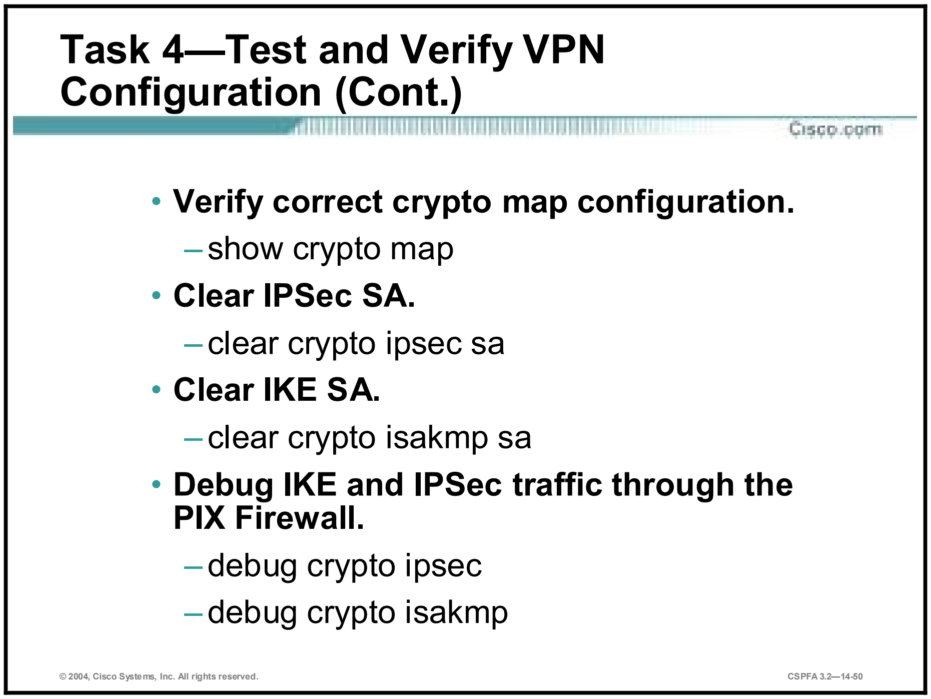
- **Verify ACLs and interesting traffic.**
  - show access-list
- **Verify correct IKE configuration.**
  - show isakmp
  - show isakmp policy
- **Verify correct IPSec configuration.**
  - show crypto ipsec transform-set

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—14-49

You can perform the following actions to test and verify that you have correctly configured the VPN on the PIX Firewall:

- Verify ACLs and select interesting traffic with the **show access-list** command.
- Verify correct IKE configuration with the **show isakmp** and **show isakmp policy** commands.
- Verify correct IPSec configuration of transform sets with the **show crypto ipsec transform-set** command.



Task 4—Test and Verify VPN Configuration (Cont.)

- **Verify correct crypto map configuration.**
  - show crypto map
- **Clear IPSec SA.**
  - clear crypto ipsec sa
- **Clear IKE SA.**
  - clear crypto isakmp sa
- **Debug IKE and IPSec traffic through the PIX Firewall.**
  - debug crypto ipsec
  - debug crypto isakmp

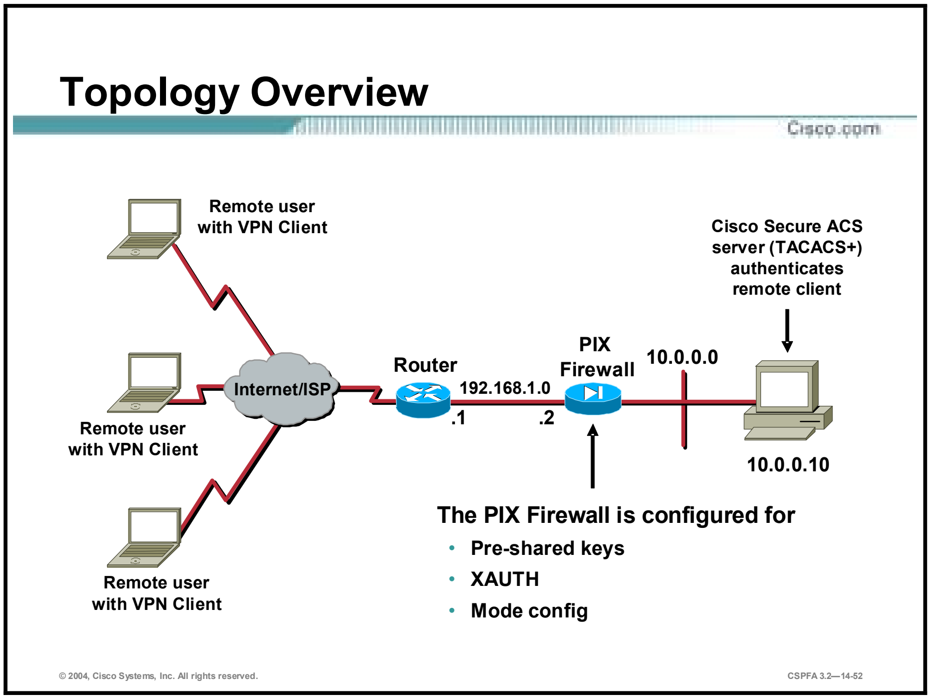
CSPFA 3.2—14-50

You can perform the following actions to test and verify that you have correctly configured the VPN on the PIX Firewall:

- Verify the correct crypto map configuration with the **show crypto map** command.
- Clear IPSec SAs for testing of SA establishment with the **clear crypto ipsec sa** command.
- Clear IKE SAs for testing of IKE SA establishment with the **clear crypto isakmp sa** command.
- Debug IKE and IPSec traffic through the PIX Firewall with the **debug crypto ipsec** and **debug crypto isakmp** commands.

The Cisco VPN Client

This topic introduces the Cisco VPN Client and explains how to use it to create a secure VPN with the PIX Firewall.



Topology Overview

Cisco.com
Remote user with VPN Client
Remote user with VPN Client
Remote user with VPN Client
Internet/ISP
Router
192.168.1.0
.1
.2
PIX Firewall
10.0.0.0
Cisco Secure ACS server (TACACS+) authenticates remote client
10.0.0.10

The PIX Firewall is configured for

- Pre-shared keys
- XAUTH
- Mode config

CSPFA 3.2—14-52

The PIX Firewall supports the Cisco VPN Client, a software program that runs on all current Windows operating systems including Windows 95, 98, Millennium Edition (ME), NT 4.0, 2000, and XP. The Cisco VPN Client enables you to establish secure, end-to-end encrypted tunnels to Cisco remote access VPN devices supporting the Unified Client framework.

The Cisco VPN Client is intended for use by low or high-speed remote users who need to securely access their corporate networks across the Internet. Users can load it on their PCs and launch tunnels as needed to establish connections to a VPN device that supports the Cisco VPN Client. The software Cisco VPN Client supports only the device on which it is loaded.

As a remote user, you probably first connect to the Internet before establishing your VPN tunnel. The Cisco VPN Client enables you to use plain old telephone service (POTS), ISDN, DSL, or cable modem connection technologies for this connection. It is compatible with Point-to-Point Protocol (PPP) over Ethernet-based DSL and has been tested with Network Telesystems Ethernet, Wind River Poet, and Point-to-Point Protocol over Ethernet (PPPoE) for Windows 95, 98, ME, NT, and 2000.

The Cisco VPN Client can also be preconfigured for mass deployments, and initial logins require very little user intervention. VPN access policies and configurations are downloaded from the central gateway and pushed to the Cisco VPN Client when a connection is established, allowing simple deployment and management as well as high scalability. Items pushed to the Cisco VPN Client from the central site Cisco VPN 3000 Concentrator include the following:

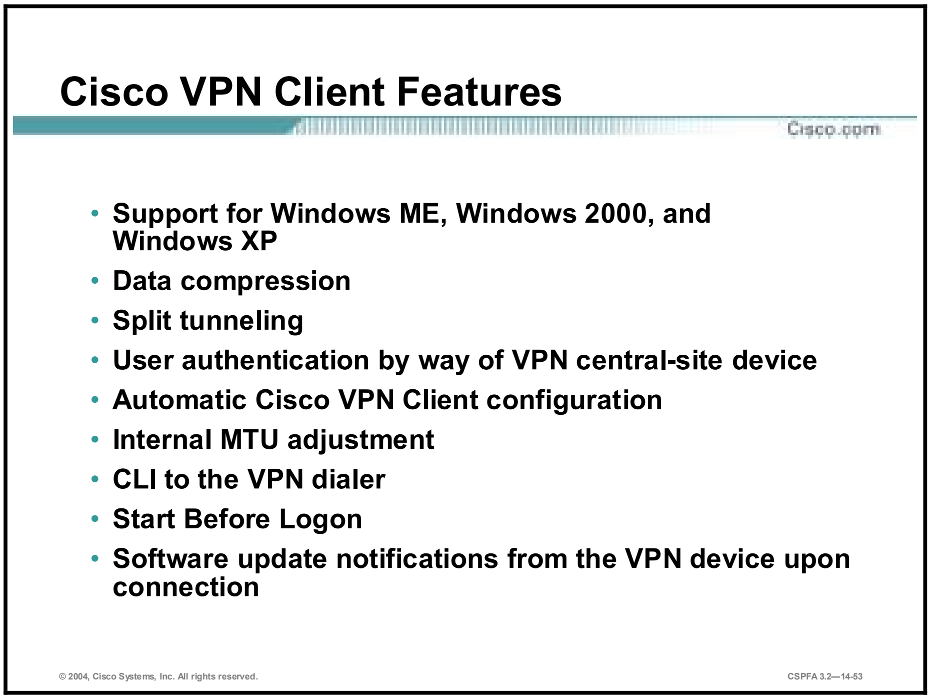
- Domain Name System (DNS)
- Windows Internet Name Service (WINS)

- Split tunneling networks
- Default domain name
- IP address
- Ability to save a password for the VPN connection

Note The Cisco VPN Client works with any Cisco VPN-enabled products that support the Unified Client framework. The Cisco Unified Client framework is an initiative that enables consistent Cisco VPN Client operation between Service Providers and Enterprises and compatibility across various Cisco VPN platforms. The Unified Client framework currently includes the Cisco VPN 3000 Concentrator Series and the Cisco PIX 500 Firewall series.

The network topology in the figure shows that the remote user with the Cisco VPN Client installed will set up an IPSec tunnel to the PIX Firewall via remote access. The PIX Firewall is configured for pre-shared keys, XAUTH, and IKE mode configuration, and has VPN groups configured with the **vpngroup** command to enable the PIX Firewall to push IPSec policy to the Cisco VPN Client.

Pre-shared keys are used for authentication, although the PIX Firewall and Cisco VPN Client also support digital certificates. Cisco Secure Access Control Server (ACS) TACACS+ is used for user authentication via XAUTH.



Cisco VPN Client Features

- Support for Windows ME, Windows 2000, and Windows XP
- Data compression
- Split tunneling
- User authentication by way of VPN central-site device
- Automatic Cisco VPN Client configuration
- Internal MTU adjustment
- CLI to the VPN dialer
- Start Before Logon
- Software update notifications from the VPN device upon connection

CSPFA 3.2—14-53

The Cisco VPN Client features the following:

- Support for Windows ME, Windows 2000, and Windows XP.
- STAC Lempel-Ziv (LZS) data compression, which also benefits modem users.
- Split tunneling, which provides the ability to simultaneously direct packets over the Internet in clear text and encrypted through an IPSec tunnel.
- User authentication by way of one of the following VPN central-site devices:
  - Internally through the VPN device's database
  - RADIUS (Remote Authentication Dial-In User Service)
  - NT Domain (Windows NT)
  - RSA (formerly SDI) SecurID or SoftID
- Automatic Cisco VPN Client configuration option, which provides the ability to import a configuration file.
- Setting the maximum transmission unit (MTU) size. The Cisco VPN Client can automatically set a size that is optimal for your environment. However, you can set the MTU size manually as well.
- Use the command-line interface (CLI) to access the VPN Dialer.
- Start Before Logon feature, which provides the ability to establish a VPN connection before logging on to a Windows NT platform. This includes Windows NT 4.0, Windows 2000, and Windows XP systems.
- Software update notifications from the VPN device upon connection.
- IPSec tunneling protocol.
- IKE key management protocol.

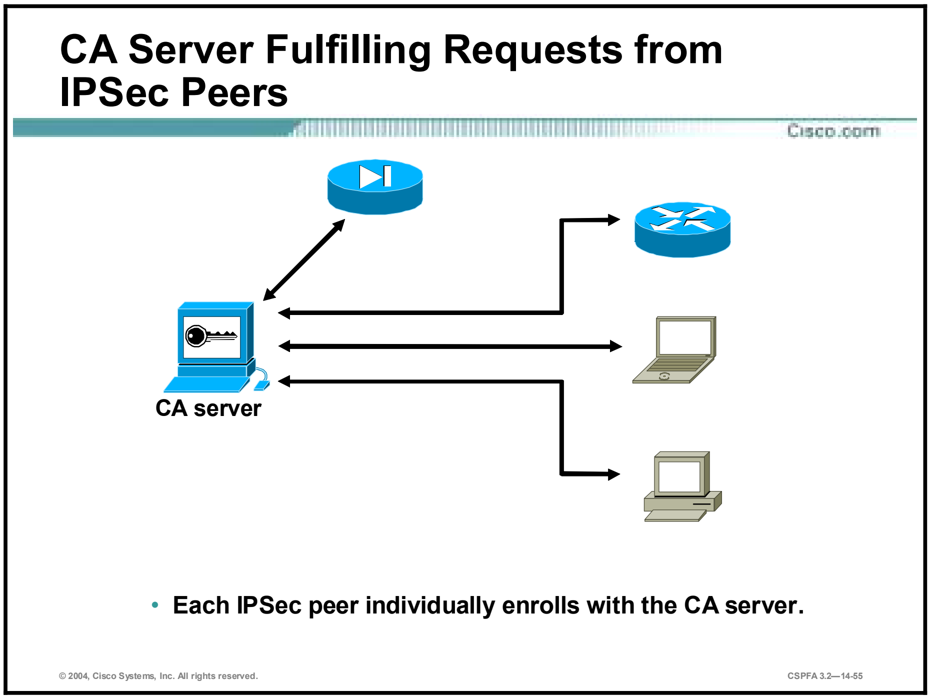
- IKE keepalives, which monitor the continued presence of a peer and report the Cisco VPN Client's continued presence to the peer. This enables the Cisco VPN Client to notify you when the peer is no longer present. Another type of keepalive keeps NAT ports alive.
- Data compression for modem users, which speeds transmission.
- Local LAN access, which is the ability to access resources on a local LAN while connected through a secure gateway to a central-site VPN device (if the central site grants permission).
- Automatic connection by way of Microsoft Dial-Up Networking or any other third-party remote access dialer.
- Log Viewer, which is an application that collects events for viewing and analysis.
- Certificate Manager, which is an application that enables you to manage your identity certificates.
- Complete browser-based, context-sensitive HTML help.
- Support for PIX Firewall platforms that run Release 6.0 and above.
- The ability to disable automatic disconnect when logging off of a Windows NT platform. This allows for roaming profile synchronization.
- Application Launcher provides the ability to launch an application or a third-party dialer from the Cisco VPN Client.
- Ability to use Entrust Entelligence certificates.

The Cisco VPN Client also supports the following IPSec attributes:

- Main mode, for negotiating phase one of establishing ISAKMP SAs
- Aggressive mode, for negotiating phase one of establishing ISAKMP SAs
- Authentication algorithm, which has the HMAC with MD5 hash function
- HMAC with SHA-1 (Secure Hash Algorithm) hash function
- Authentication mode with pre-shared keys
- X.509 digital certificates
- DH groups 1, 2, and 5
- Encryption algorithms
- 56-bit DES
- 168-bit 3DES
- AES 128-bit and AES 256-bit
- XAUTH
- Mode configuration (also known as ISAKMP configuration method)
- Tunnel encapsulation mode
- IP compression (IPComp) using LZS

Scale PIX Firewall VPNs

This topic explains how to scale PIX Firewall VPNs using Certificate Authorities (CAs).

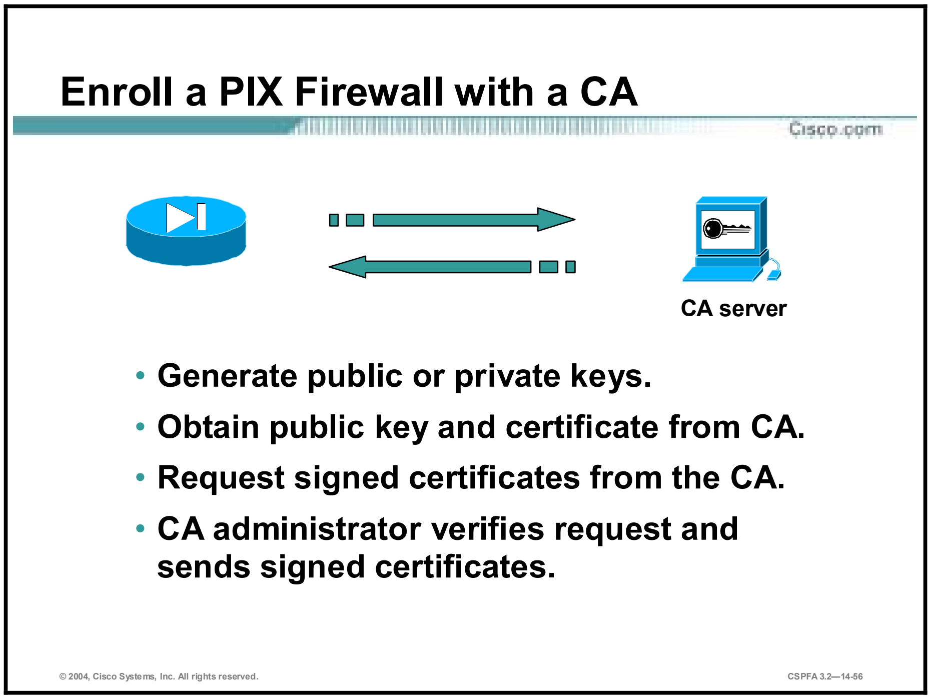


CA Server Fulfilling Requests from IPSec Peers

- Each IPSec peer individually enrolls with the CA server.

CSPFA 3.2—14-55

The use of pre-shared keys for IKE authentication works only when you have a few IPSec peers. CAs enable scaling to a large number of IPSec peers. Although there are a number of methods, using a CA server is the most scalable solution. Other IKE authentication methods require manual intervention to generate and distribute the keys on a per-peer basis. The CA server enrollment process can be largely automated so that it scales well to large deployments. Each IPSec peer individually enrolls with the CA server and obtains public and private encryption keys compatible with other peers enrolled with the server.



Enroll a PIX Firewall with a CA

Cisco.com

- Generate public or private keys.
- Obtain public key and certificate from CA.
- Request signed certificates from the CA.
- CA administrator verifies request and sends signed certificates.

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—14-56

Peers enroll with a CA server in a series of steps in which specific keys are generated and then exchanged by the PIX Firewall and the CA server to ultimately form a signed certificate. The enrollment steps can be summarized as follows:

Step 1 The PIX Firewall generates an RSA public and private key pair.

Step 2 The PIX Firewall obtains a public key and its certificate from the CA server.

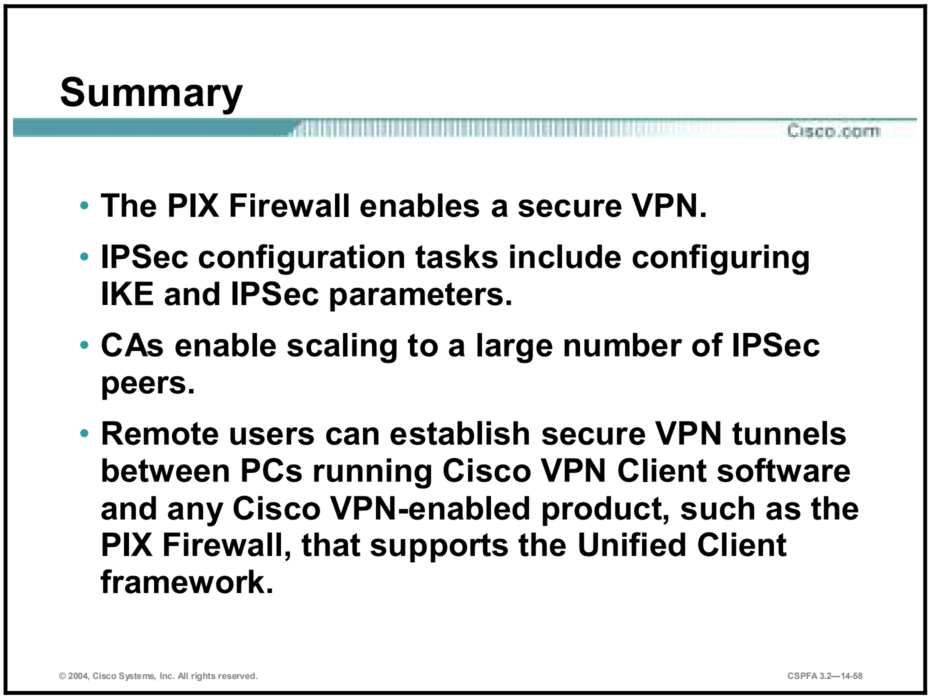
Step 3 The PIX Firewall requests a signed certificate from the CA using the generated RSA keys and the public key certificate from the CA server.

Step 4 The CA administrator verifies the request and sends a signed certificate.

Note See the “About CA” and “Configuring CA” sections in the “Configuring IPSec” chapter of the *Configuration Guide for the Cisco PIX Firewall* for more details on how CA servers work and how to configure the PIX Firewall for CA support.

Summary

This topic summarizes what you learned in this lesson.



Summary

- The PIX Firewall enables a secure VPN.
- IPSec configuration tasks include configuring IKE and IPSec parameters.
- CAs enable scaling to a large number of IPSec peers.
- Remote users can establish secure VPN tunnels between PCs running Cisco VPN Client software and any Cisco VPN-enabled product, such as the PIX Firewall, that supports the Unified Client framework.

CSPFA 3.2—14-58

Lab Exercise—Configure PIX Firewall VPNs

Complete the following lab exercise to practice what you learned in this lesson.

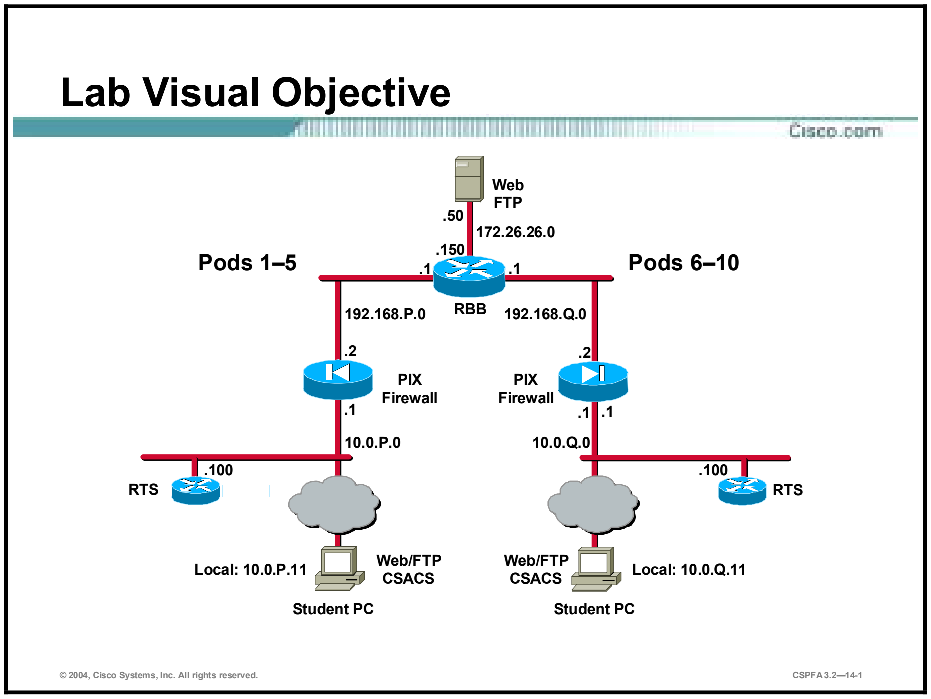
Objectives

In this lab exercise you will complete the following tasks:

- Prepare to configure VPN support.
- Configure IKE parameters.
- Configure IPSec parameters.
- Test and verify the IPSec configuration.

Visual Objective

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—14-1

Scenario

The XYZ Company has purchased PIX Firewalls to create a secure VPN over the Internet between sites. The company wants you to configure a secure VPN gateway using IPSec between two PIX Firewalls.

Task 1—Prepare to Configure VPN Support

Complete the following steps to prepare for the IKE and IPSec configuration. For this task, you will use default values except when you are directed to enter a specific value. Your IKE policy will use pre-shared keys. Your IPSec policy will use ESP mode with DES encryption.

Step 1 Verify that a static translation is configured from a global IP address on the outside interface to the internal host:

```
pixP(config)# show static

static (dmz,outside) 192.168.P.11 bastionhost netmask 255.255.255.255
0 0

static (inside,outside) 192.168.P.10 insidehost netmask
255.255.255.255 0 0
```

(where P = pod number)

Step 2 Verify that an ACL permitting Web access to your inside host has been configured:

```
pixP(config)# show access-list

access-list ACLIN; 10 elements
```

```
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 object-
group FTPSERVERS
object-group MYSERVICES
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq www(hitcnt=4)
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq ftp(hitcnt=2)
access-list ACLIN line 2 permit tcp any object-group ALLSERVERS eq www
access-list ACLIN line 2 permit tcp any host 192.168.P.11 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.10 eq www
(hitcnt=2)
access-list ACLIN line 2 permit tcp any host 192.168.P.6 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.7 eq www
(hitcnt=0)
access-list ACLIN line 3 permit icmp any any object-group PING
access-list ACLIN line 3 permit icmp any any echo (hitcnt=12)
access-list ACLIN line 3 permit icmp any any echo-reply (hitcnt=4)
access-list ACLIN line 3 permit icmp any any unreachable (hitcnt=0)
access-list ACLIN line 3 deny ip any any (hitcnt=3)
access-list ACLDMZ; 3 elements
access-list ACLDMZ line 1 permit icmp any any object-group PING
access-list ACLDMZ line 1 permit icmp any any echo (hitcnt=0)
access-list ACLDMZ line 1 permit icmp any any echo-reply (hitcnt=8)
access-list ACLDMZ line 1 permit icmp any any unreachable (hitcnt=0)
```

(where P = pod number, and Q = peer pod number)

Step 3 Verify that ACLIN is bound to the outside interface:

```
pixP(config)# show access-group
access-group ACLIN in interface outside
access-group ACLDMZ in interface dmz
```

Step 4 Ensure you can establish a web connection between pods from the student PCs using the static and ACL.

Step 5 Enable the PIX Firewall to implicitly permit any packet from an IPSec tunnel, and bypass checking with an associated **access-group** command for IPSec connections:

```
pixP(config)# sysopt connection permit-ipsec
```

Task 2—Configure IKE Parameters

Complete the following steps to configure IKE on your PIX Firewall:

Step 1 Ensure IKE is enabled on the outside interface:

```
pixP(config)# isakmp enable outside
```

Step 2 Configure a basic IKE policy using pre-shared keys for authentication:

```
pixP(config)# isakmp policy 10 authentication pre-share
```

Step 3 Set the IKE identity:

```
pixP(config)# isakmp identity address
```

Step 4 Configure the ISAKMP pre-shared key to point to the outside IP address of the peer PIX Firewall:

```
pixP(config)# isakmp key cisco123 address 192.168.Q.2 netmask
255.255.255.255
```

(where Q = peer pod number)

Task 3—Configure IPSec Parameters

Complete the following steps to configure IPSec (IKE Phase 2) parameters:

Step 1 Create an ACL to select traffic to protect. The ACL should protect IP traffic between student PC networks:

```
pixP(config)# access-list 101 permit ip host 192.168.P.10 host
192.168.Q.10
```

(where P = pod number, and Q = peer pod number)

Step 2 View your ACL:

```
pix1(config)# show access-list
access-list ACLIN; 10 elements
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 object-
group FTPSERVERS object-group MYSERVICES
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq www(hitcnt=4)
access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq ftp(hitcnt=2)
access-list ACLIN line 2 permit tcp any object-group ALLSERVERS eq www
access-list ACLIN line 2 permit tcp any host 192.168.P.11 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.10 eq www
(hitcnt=2)
access-list ACLIN line 2 permit tcp any host 192.168.P.6 eq www
(hitcnt=0)
access-list ACLIN line 2 permit tcp any host 192.168.P.7 eq www
(hitcnt=0)
access-list ACLIN line 3 permit icmp any any object-group PING
access-list ACLIN line 3 permit icmp any any echo (hitcnt=12)
access-list ACLIN line 3 permit icmp any any echo-reply (hitcnt=4)
access-list ACLIN line 3 permit icmp any any unreachable (hitcnt=0)
access-list ACLIN line 3 deny ip any any (hitcnt=3)
access-list ACLDMZ; 3 elements
access-list ACLDMZ line 1 permit icmp any any object-group PING
access-list ACLDMZ line 1 permit icmp any any echo (hitcnt=0)
access-list ACLDMZ line 1 permit icmp any any echo-reply (hitcnt=8)
access-list ACLDMZ line 1 permit icmp any any unreachable (hitcnt=0)
```

```
access-list 101; 1 elements
access-list 101 line 1 permit ip host 192.168.P.10 host 192.168.Q.10
(hitcnt=0)
```

(where P = pod number, and Q = peer pod number)

Step 3 Configure an IPSec transform set (IKE Phase 2 parameters) to use ESP and DES. Use a transform-set-name of pixQ.

```
pixP(config)# crypto ipsec transform-set pixQ esp-des
```

(where Q = peer pod number)

Step 4 Create a crypto map by completing the following substeps:

1. Create a crypto map entry. Use a map-name of peer Q.

```
pixP(config)# crypto map peerQ 10 ipsec-isakmp
```

(where Q = peer pod number)

2. Look at the crypto map and observe the defaults:

```
pixP(config)# show crypto map
Crypto Map "peerQ" 10 ipsec-isakmp
        No matching address list set.
        Current peer: 0.0.0.0
        Security association lifetime: 4608000 kilobytes/28800 seconds
        PFS (Y/N): N
        Transform sets={ }
```

3. Assign the ACL to the crypto map:

```
pixP(config)# crypto map peerQ 10 match address 101
```

(where Q = peer pod number)

4. Define the peer. The peer IP address should be set to the peer's outside interface IP address:

```
pixP(config)# crypto map peerQ 10 set peer 192.168.Q.2
```

(where Q = peer pod number)

5. Specify the transform set used to reach the peer. Use the transform set name you configured in substep 2.

```
pixP(config)# crypto map peerQ 10 set transform-set pixQ
```

(where Q = peer pod number)

6. Apply the crypto map set to the outside interface:

```
pixP(config)# crypto map peerQ interface outside
```

(where Q = peer pod number)

Task 4—Test and Verify the IPSec Configuration

Complete the following steps to test and verify the IPSec configuration:

Step 1 Verify the IKE policy you just created. Note the default values.

```
pixP(config)# show isakmp
isakmp enable outside
isakmp key ******** address 192.168.Q.2 netmask 255.255.255.255
isakmp identity address
isakmp policy 10 authentication pre-share
isakmp policy 10 encryption des
isakmp policy 10 hash sha
isakmp policy 10 group 1
isakmp policy 10 lifetime 86400
```

(where Q = peer pod number)

Step 2 Examine the IKE policies in your PIX Firewall:

```
pixP(config)# show isakmp policy
Protection suite of priority 10
        encryption algorithm:   DES - Data Encryption Standard (56 bit
                                keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #1 (768 bit)
        lifetime:               86400 seconds, no volume limit
Default protection suite
        encryption algorithm:   DES - Data Encryption Standard (56 bit
                                keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Rivest-Shamir-Adleman Signature
        Diffie-Hellman group:   #1 (768 bit)
        lifetime:               86400 seconds, no volume limit
```

Step 3 Verify the crypto ACL:

```
pixP(config)# show access-list 101
access-list 101; 1 elements
access-list 101 line 1 permit ip host 192.168.P.10 host 192.168.Q.10
(hitcnt=0)
```

(where P = pod number, and Q = peer pod number)

Step 4 Verify that the IPSec parameters (IKE Phase 2) are correct:

```
pixP(config)# show crypto ipsec transform-set
Transform set pixQ: { esp-des  }
   will negotiate = { Tunnel,  },
```

(where Q = peer pod number)

Step 5 Verify that the crypto map configuration is correct:

```
pix1(config)# show crypto map
Crypto Map: "peerQ" interfaces: { outside }
Crypto Map "peerQ" 10 ipsec-isakmp
  Peer = 192.168.Q.2
  access-list 101; 1 elements
  access-list 101 line 1 permit ip host 192.168.P.10 host 192.168.Q.10
  (hitcnt=0)
  Current peer: 192.168.Q.2
  Security association lifetime: 4608000 kilobytes/28800 seconds
  PFS (Y/N): N
  Transform sets={ pixQ, }
```

(where P = pod number, and Q = peer pod number)

Step 6 Turn on debugging for IPSec and ISAKMP:

```
pixP(config)# debug crypto ipsec
pixP(config)# debug crypto isakmp
```

Step 7 Clear the IPSec SA by using the following command:

```
pixP(config)# clear crypto ipsec sa
```

Step 8 Initiate a web session from your student PC to your peer pod's student PC, 192.168.Q.10. Observe the debug output and verify that the web session was established. The debug should state the following status indicating that IPSec was successful:

```
return status is IKMP_NO_ERROR
```

Step 9 Ensure that traffic between peers is being encrypted by completing the following substeps:

1. Examine the ISAKMP SA. Note the source and destination IP addresses as well as the state.

```
pixP(config)# show crypto isakmp sa
Total     : 1
Embryonic : 0
        dst               src        state     pending     created
     192.168.Q.2      192.168.P.2    QM_IDLE         0           1
```

2. Examine the IPSec SAs. Note the number of packets encrypted and decrypted.

```
pixP(config)# show crypto ipsec sa
interface: outside
    Crypto map tag: peerQ, local addr. 192.168.P.2

   local  ident (addr/mask/prot/port):
(192.168.P.10/255.255.255.255/0/0)
   remote ident (addr/mask/prot/port):
(192.168.Q.10/255.255.255.255/0/0)
```

```
   current_peer: 192.168.Q.2
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 39, #kas encrypt: 39, #kas digest 0
    #pkts decaps: 41, #pkts decrypt: 41, #pkts verify 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0, #pkts decompress
failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 192.168.P.2, remote crypto endpt.:
192.168.Q.2
     path mtu 1500, ipsec overhead 44, media mtu 1500
     current outbound spi: f63e125

     inbound esp sas:
      spi: 0xa464ac7f(2758061183)
        transform: esp-des ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 1, crypto map: peerQ
        sa timing: remaining key lifetime (k/sec): (4607977/28481)
        IV size: 8 bytes
        replay detection support: N

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0xf63e125(258203941)
        transform: esp-des ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 2, crypto map: peerQ
        sa timing: remaining key lifetime (k/sec): (4607978/28472)
        IV size: 8 bytes
        replay detection support: N

     outbound ah sas:

     outbound pcp sas:
```

(where P = pod number, and Q = peer pod number)

3. Generate additional traffic by clicking the **Reload** button of your web browser.

4. Examine the IPSec SAs again. Note that the packet counters have increased incrementally.

```
pixP(config)# show crypto ipsec sa
interface: outside
    Crypto map tag: peerQ, local addr. 192.168.P.2


   local  ident (addr/mask/prot/port):
(192.168.P.10/255.255.255.255/0/0)
   remote ident (addr/mask/prot/port):
(192.168.Q.10/255.255.255.255/0/0)
   current_peer: 192.168.Q.2
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 51, #pkts encrypt: 51, #pkts digest 0
    #pkts decaps: 47, #pkts decrypt: 47, #pkts verify 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0, #pkts decompress
failed: 0
    #send errors 0, #recv errors 0


     local crypto endpt.: 192.168.P.2, remote crypto endpt.:
192.168.Q.2
     path mtu 1500, ipsec overhead 44, media mtu 1500
     current outbound spi: f63e125


     inbound esp sas:
      spi: 0xa464ac7f(2758061183)
        transform: esp-des ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 1, crypto map: peerQ
        sa timing: remaining key lifetime (k/sec): (4607976/28301)
        IV size: 8 bytes
        replay detection support: N


     inbound ah sas:


     inbound pcp sas:


     outbound esp sas:
      spi: 0xf63e125(258203941)
        transform: esp-des ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 2, crypto map: peerQ
        sa timing: remaining key lifetime (k/sec): (4607976/28301)
        IV size: 8 bytes
```

```
            replay detection support: N

         outbound ah sas:

         outbound pcp sas:
```

(where P = pod number, and Q = peer pod number)

Step 10 Clear your IPSec SAs with the **clear crypto sa** command:

```
pixP(config)# clear crypto sa
```

Step 11 Remove all **isakmp** command statements from your configuration with the **clear isakmp** command:

```
pixP(config)# clear isakmp
```

Step 12 Remove all parameters entered through the **crypto map** command with the **clear crypto map** command:

```
pixP(config)# clear crypto map
```

Step 13 Remove the **sysopt** command statements from your configuration with the **clear sysopt** command:

```
pixP(config)# clear sysopt
```

Step 14 Remove ACL 101 from your configuration:

```
pixP(config)# clear access-list 101
```

15

Configuring PIX Firewall Remote Access Using Cisco Easy VPN

Overview

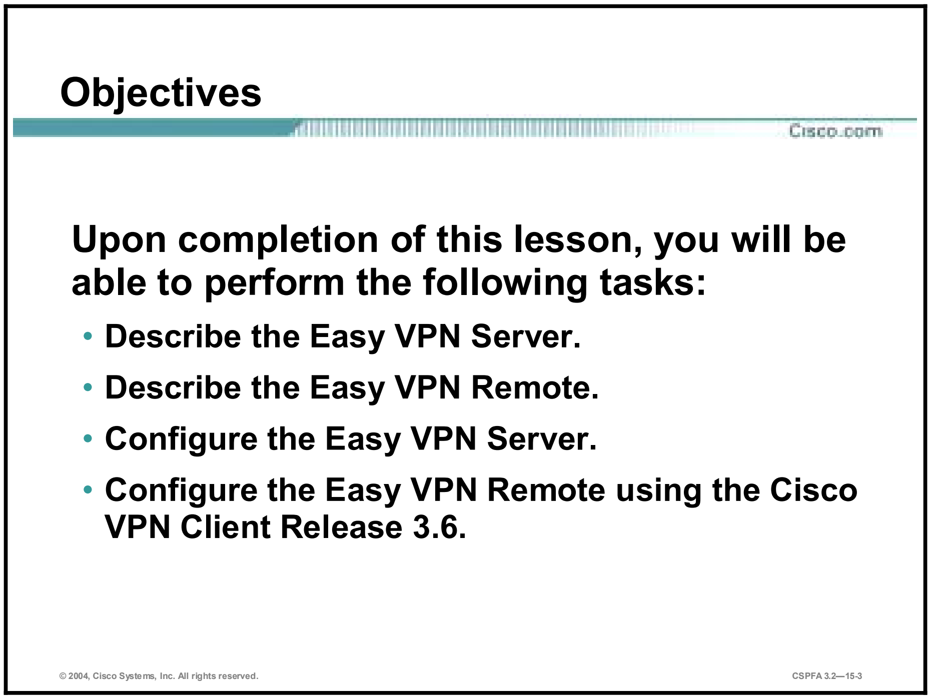
This lesson covers the configuration of Cisco PIX Firewall remote access using the Cisco Easy Virtual Private Network (VPN) and the Cisco VPN Client.

It includes the following topics:

- Objectives
- Introduction to the Cisco Easy VPN
- Overview of the Easy VPN Server
- Overview of the Easy VPN Remote feature
- Overview of the Cisco VPN 3.6 Client
- How the Cisco Easy VPN works
- Configuring the Easy VPN Server for extended authentication
- Cisco VPN Client 3.6 manual configuration tasks
- Working with the Cisco VPN 3.6 Client
- Summary
- Lab exercise

Objectives

This topic lists the lesson's objectives.



Objectives

Cisco.com

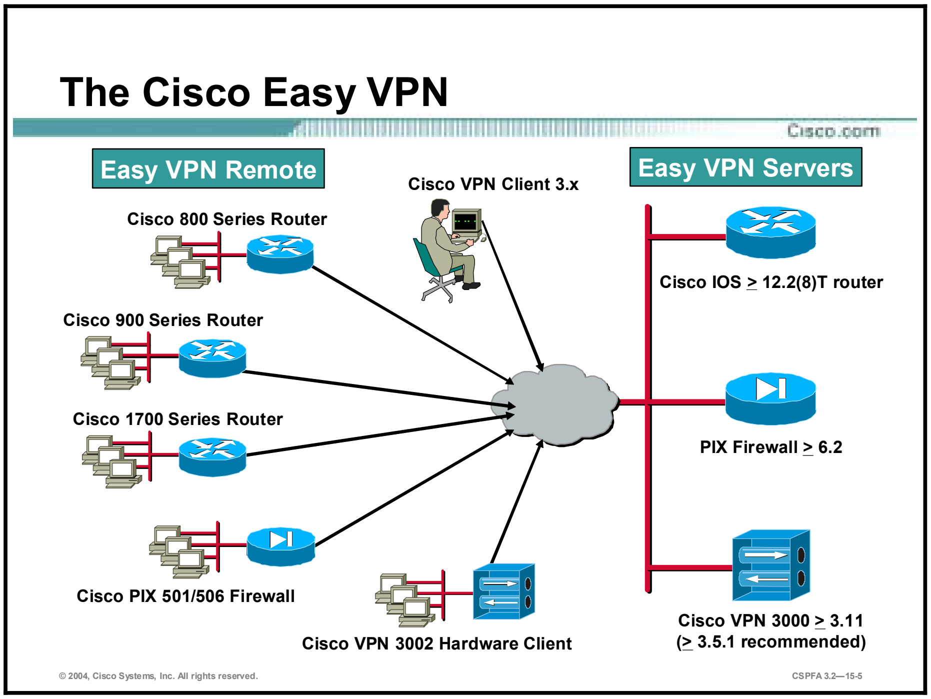
Upon completion of this lesson, you will be able to perform the following tasks:

- Describe the Easy VPN Server.
- Describe the Easy VPN Remote.
- Configure the Easy VPN Server.
- Configure the Easy VPN Remote using the Cisco VPN Client Release 3.6.

CSPFA 3.2—15-3

Introduction to the Cisco Easy VPN

This topic discusses the Cisco Easy VPN and its two components.



The Cisco Easy VPN

Cisco.com

Easy VPN Remote

Cisco 800 Series Router

Cisco 900 Series Router

Cisco 1700 Series Router

Cisco PIX 501/506 Firewall

Cisco VPN Client 3.x

Cisco VPN 3002 Hardware Client

Easy VPN Servers

Cisco IOS ≥ 12.2(8)T router

PIX Firewall ≥ 6.2

Cisco VPN 3000 ≥ 3.11
(≥ 3.5.1 recommended)

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—15-5

The Cisco Easy VPN, a software enhancement for existing Cisco PIX Firewalls and security appliances, greatly simplifies VPN deployment for remote offices and teleworkers. Based on the Cisco Unified Client framework, the Cisco Easy VPN centralizes VPN management across all Cisco VPN devices, greatly reducing the complexity of VPN deployments. The Cisco Easy VPN enables an integration of Easy VPN Remotes—Cisco routers, Cisco PIX Firewalls, the Cisco VPN 3002 Hardware Client or Software Clients—within a single deployment with a consistent policy and key management method that greatly simplifies remote side administration.

The Cisco Easy VPN consists of two components: the Cisco Easy VPN Server and the Cisco Easy VPN Remote feature.

The Easy VPN Server

The Easy VPN Server enables Cisco IOS routers, PIX Firewalls, and Cisco VPN 3000 Series Concentrators to act as VPN head-end devices in site-to-site or remote-access VPNs, where the remote office devices are using the Easy VPN Remote feature. Using this feature, security policies defined at the headend are pushed to the remote VPN device, insuring that those connections have up-to-date policies in place before the connection is established.

In addition, an Easy VPN Server-enabled device can terminate IPSec tunnels initiated by mobile remote workers running VPN Client software on PCs. This flexibility makes it possible for mobile and remote workers, such as sales people on the road or teleworkers, to access their headquarters intranet where critical data and applications exist.

The Easy VPN Remote Feature

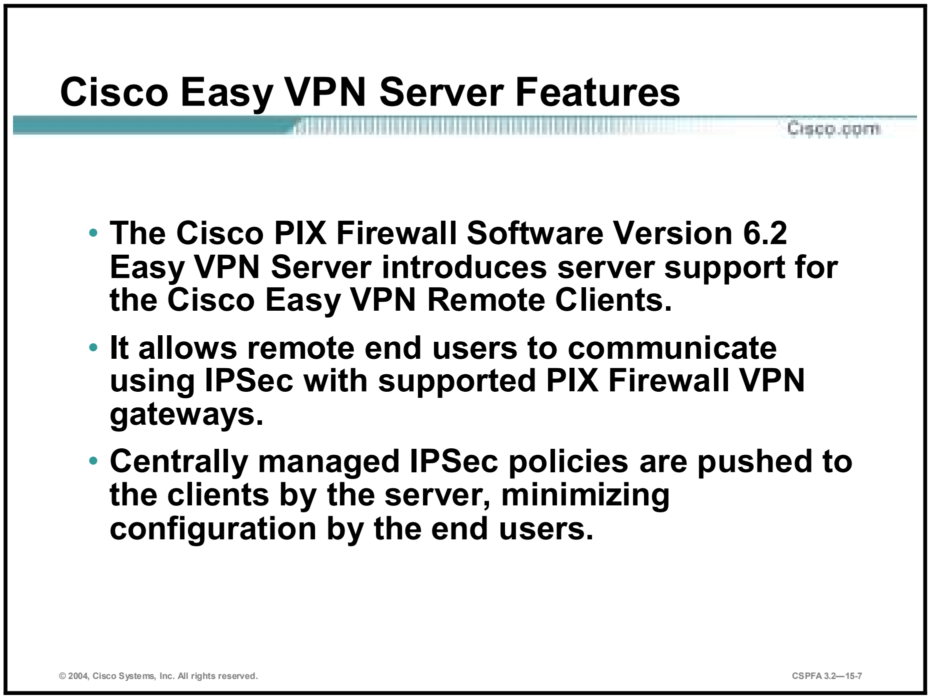
The Easy VPN Remote feature enables Cisco IOS routers, PIX Firewalls, or Cisco VPN 3002 Hardware Clients or Software Clients to act as remote VPN Clients. As such, these devices can receive security policies from an Easy VPN Server, minimizing VPN configuration requirements at the remote location. This cost-effective solution is ideal for remote offices with little IT support, or large customer premises equipment (CPE) deployments where it is impractical to individually configure multiple remote devices. This feature makes VPN configuration as easy as entering a password, which increases productivity and lowers costs as the need for local IT support is minimized.

In the example in the figure, the VPN gateway is a PIX Firewall running the Easy VPN Server feature. Remote IOS routers and VPN software clients connect to the IOS router Easy VPN Server for access to the corporate intranet.

See Cisco.com for a complete list of Cisco routers that support the Cisco Easy VPN.

Overview of the Easy VPN Server

This topic provides an overview of the Easy VPN Server features of Cisco PIX Firewall Software Version 6.2.

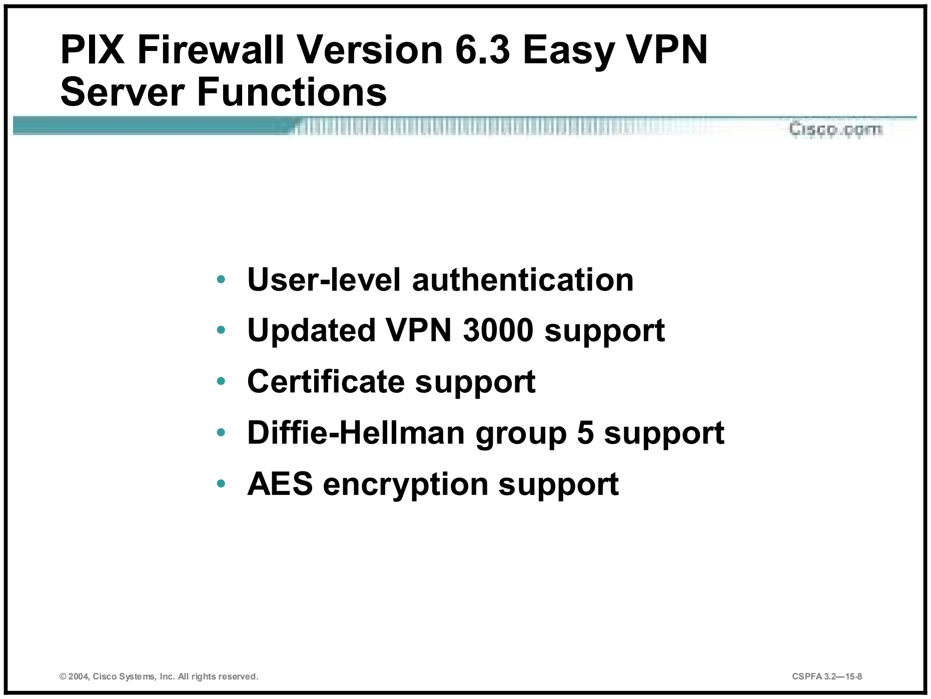


Cisco Easy VPN Server Features

- The Cisco PIX Firewall Software Version 6.2 Easy VPN Server introduces server support for the Cisco Easy VPN Remote Clients.
- It allows remote end users to communicate using IPSec with supported PIX Firewall VPN gateways.
- Centrally managed IPSec policies are pushed to the clients by the server, minimizing configuration by the end users.

CSPFA 3.2—15-7

The Cisco PIX Firewall version 6.2 Cisco Easy VPN Server introduces server support for the Cisco Easy VPN Remote Clients. It allows remote end users to communicate using IP Security (IPSec) with supported Cisco VPN gateways. Centrally managed IPSec policies are pushed to the clients by the server, minimizing configuration by the end users.



PIX Firewall Version 6.3 Easy VPN Server Functions

- **User-level authentication**
- **Updated VPN 3000 support**
- **Certificate support**
- **Diffie-Hellman group 5 support**
- **AES encryption support**

CSPFA 3.2—15-8

The PIX Firewall version 6.3 Easy VPN Server supports the following functions:

- Mode configuration version 6 support—Mode configuration version 6 is now supported for more attributes (as described in an Internet Engineering Task Force [IETF] draft submission).
- Extended Authentication (XAUTH) version 6 support—The PIX Firewall has been enhanced to support version 6 of XAUTH. XAUTH for user authentication is based on an IETF draft submission.
- Internet Key Exchange (IKE) dead peer detection (DPD)—The VPN Client implements a new keepalive scheme—IKE DPD.

  DPD enables two IPSec peers to determine if the other is still “alive” during the lifetime of a VPN connection. DPD is useful because a host may reboot or the dialup link of a remote user may disconnect without notifying the peer that the VPN connection has gone away. When an IPSec host determines that a VPN connection no longer exists, it can notify a user, attempt to switch to another IPSec host, or clean up valuable resources that were allocated for the peer that no longer exists.

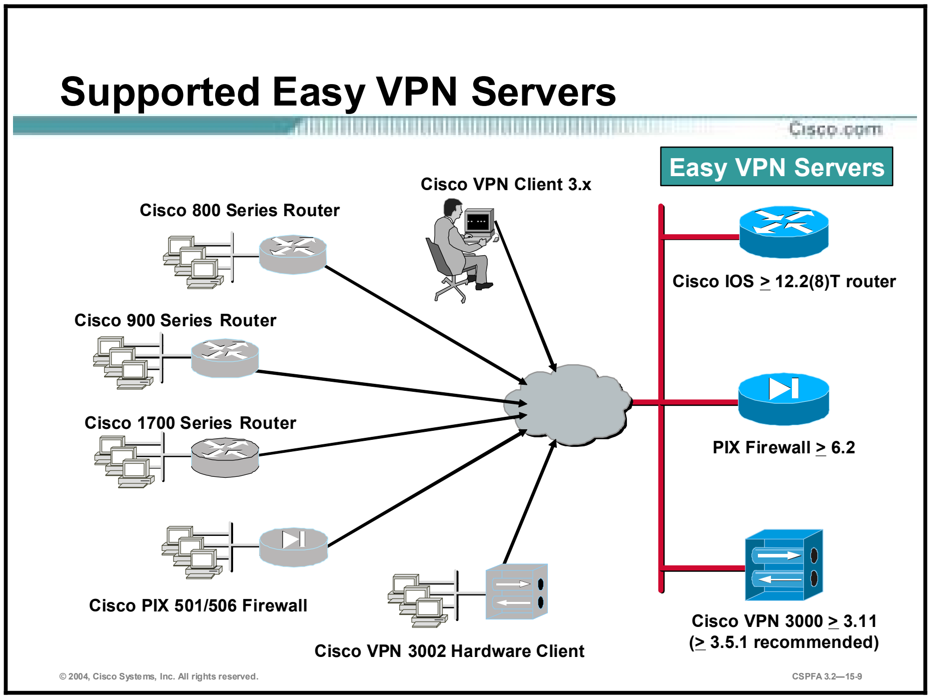
  A Cisco VPN device can be configured to send and reply to DPD messages. DPD messages are sent if no other traffic is being passed through the IPSec tunnel. If a configured amount of time has lapsed since the last inbound data was received, DPD will send a message (“DPD R-U-THERE”). DPD messages are unidirectional and are automatically sent by Cisco VPN Clients. DPD must be configured on the router only if the router wishes to send DPD messages to the Cisco VPN Client to determine the health of the Cisco VPN Client.
- Split tunneling control—Remote Cisco VPN Clients can support split tunneling, which is the ability for a Cisco VPN Client to have intranet and Internet access at the same time. If split tunneling is not configured, the Cisco VPN Client will direct all traffic through the tunnel, even traffic destined for the Internet.

- Initial contact—If a Cisco VPN Client is suddenly disconnected, the gateway may not be notified. Consequently, removal of connection information (IKE and IPSec Security Associations [SAs]) for that Cisco VPN Client will not immediately occur. Thus, if the Cisco VPN Client attempts to reconnect to the gateway again, the gateway will refuse the connection because the previous connection information is still valid. To avoid this scenario, a new capability called initial contact has been introduced; it is supported by all Cisco VPN products. If a Cisco VPN Client or router is connecting to another Cisco gateway for the first time, an initial contact message is sent that tells the receiver to ignore and delete any old connection information that has been maintained for that newly connecting peer. Initial contact ensures that connection attempts are not refused because of SA synchronization problems, which are often identified by invalid Security Parameter Index (SPI) messages and which require devices to have their connections cleared.
- Group-Based Policy Control—Policy attributes such as IP addresses, DNS, and split tunnel access can be provided on a per-group or per-user basis.

The Cisco Easy VPN supports the following IPSec options and attributes:

- Authentication algorithms
  - Hash-based message authentication code (HMAC)-Message Digest 5 (MD5)
  - HMAC- Secure Hash Algorithm-1 (SHA-1)
- Authentication types
  - Pre-shared keys
  - Rivest, Shamir, and Adleman (RSA) digital signatures (not supported by Cisco Easy VPN Remote phase II)
- Diffie-Hellman (DH) groups 1, 2, and 5
- IKE encryption algorithms
  - Data Encryption Standard (DES)
  - Triple-Data Encryption Standard (3DES)
  - Advanced Encryption Standard (AES)
- IPSec encryption algorithms
  - DES
  - 3DES
  - Advanced Encryption Standard (AES)
  - NULL
- IPSec protocol identifiers
  - Encapsulating Security Payload (ESP)
  - IP Payload Compression Protocol (IPComp)-STAC-Lempel-Ziv Compression (LZS)
- IPSec protocol mode—Tunnel mode

For information on these options and attributes, refer to the chapter “Managing VPN Remote Access” in the *Cisco PIX Firewall and VPN Configuration Guide*, Release 6.3.



Supported Easy VPN Servers

Cisco.com

Easy VPN Servers

© 2004, Cisco Systems, Inc. All rights reserved.

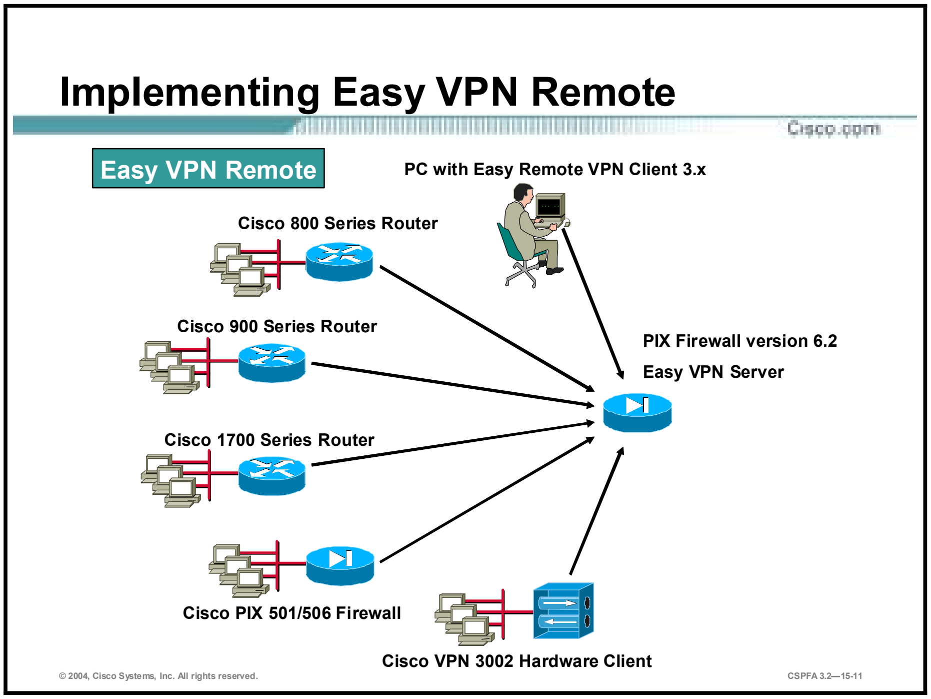
CSPFA 3.2—15-9

The Easy VPN Remote feature requires that the destination peer be a VPN gateway or Concentrator that supports the Easy VPN Server. At the time of publication, this includes the following platforms when running the indicated software releases:

- Cisco 806, Cisco 826, Cisco 827, and Cisco 828 routers—Cisco IOS Software Release 12.2(8)T or later release
- Cisco 1700 Series—Cisco IOS Software Release 12.2(8)T or later release
- Cisco 2600 Series—Cisco IOS Software Release 12.2(8)T or later release
- Cisco 3620—Cisco IOS Software Release 12.2(8)T or later release
- Cisco 3640—Cisco IOS Software Release 12.2(8)T or later release
- Cisco 3660—Cisco IOS Software Release 12.2(8)T or later release
- Cisco 7100 Series VPN routers—Cisco IOS Software Release 12.2(8)T or later release
- Cisco 7200 Series routers—Cisco IOS Software Release 12.2(8)T or later release
- Cisco 7500 Series routers—Cisco IOS Software Release 12.2(8)T or later release
- Cisco uBR905 and Cisco uBR925 cable access routers—Cisco IOS Software Release 12.2(8)T or later release
- Cisco VPN 3000 Series—Software Release 3.11 or later release
- Cisco PIX 500 Series—Software Release 6.2 or later release

Overview of the Easy VPN Remote Feature

The Cisco Easy VPN Remote feature allows certain Cisco IOS routers, Cisco PIX Firewalls, Cisco VPN 3002 Hardware or Software Clients to act as remote Cisco VPN Clients.

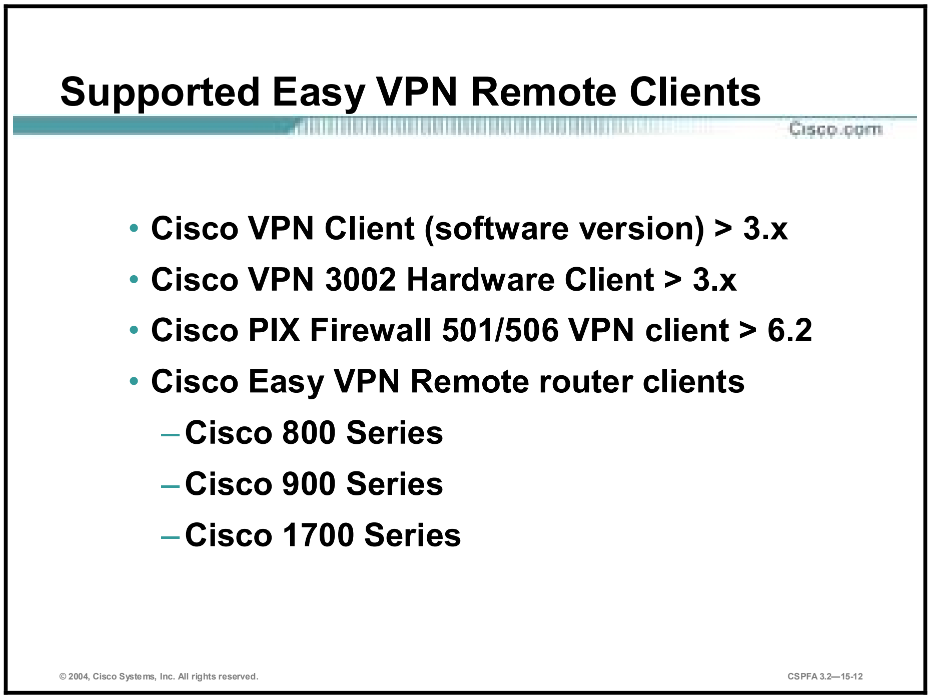


Implementing Easy VPN Remote

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—15-11

In the example in the figure, remote offices or individual PCs may access corporate computing resources using various Cisco VPN Clients terminating at a Cisco PIX Firewall.



Supported Easy VPN Remote Clients

- Cisco VPN Client (software version) > 3.x
- Cisco VPN 3002 Hardware Client > 3.x
- Cisco PIX Firewall 501/506 VPN client > 6.2
- Cisco Easy VPN Remote router clients
  - Cisco 800 Series
  - Cisco 900 Series
  - Cisco 1700 Series

CSPFA 3.2—15-12

The following list details the Cisco VPN Clients that support the Easy VPN Remote feature:

- Cisco VPN Client (software version) 3.x or later
- Cisco VPN 3002 Hardware Client 3.x or later
- Cisco PIX Firewall 501/506/506E VPN client 6.2 or later
- Cisco VPN Easy VPN Remote routers
  - Cisco 800 series
  - Cisco 900 series
  - Cisco 1700 series

See Cisco.com for the latest listing of Cisco Easy VPN Remote devices and software clients.

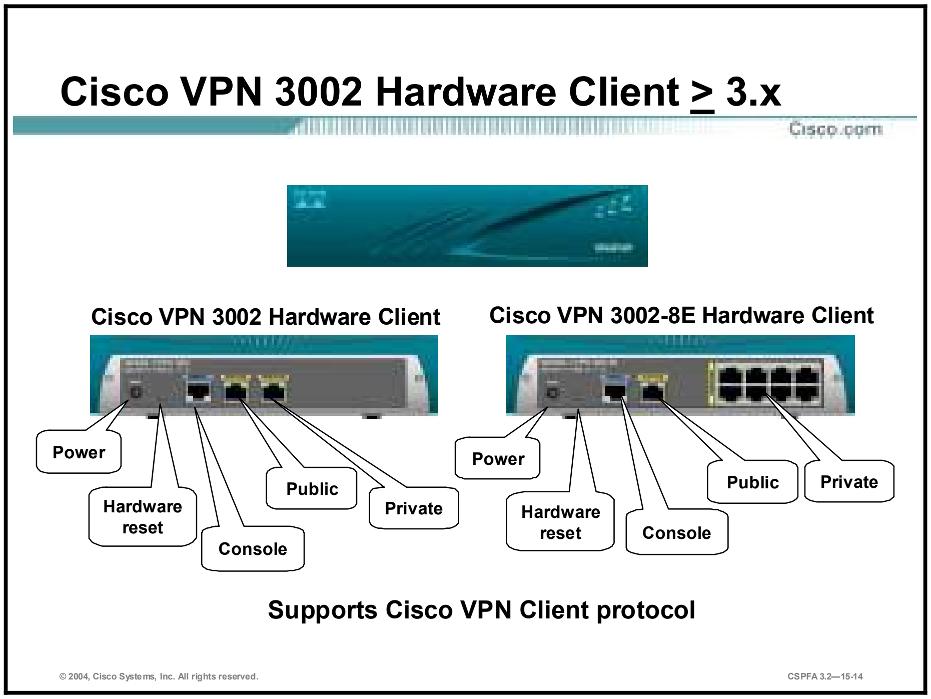
Cisco VPN Client Software Version ≥ 3.x

- Software-based Cisco VPN Client
- Supports several operating systems
- Comes standard with the Cisco VPN 3000 Series Concentrator
- Available for download from Cisco.com
- Supports Cisco VPN Client protocol

Simple to deploy and operate, the Cisco VPN Client (software version) enables customers to establish secure, end-to-end encrypted tunnels to any Easy VPN Server. This thin design, IPSec implementation is available via Cisco.com for use with any Cisco central site remote access VPN product and is included free of charge with the Cisco VPN 3000 Series Concentrator. The client can be preconfigured for mass deployments and initial logins require very little user intervention. VPN access policies and configurations are downloaded from the Easy VPN Server and pushed to the client when a connection is established, allowing simple deployment and management, as well as high scalability.

The Cisco VPN Client provides support for the following computer operating systems:

- Microsoft Windows 95 (OSR2+), 98, ME, NT 4.0, 2000, XP
- Linux (Intel)
- Solaris (UltraSPARC 32- and 64-bit)
- MAC OS X 10.1



Cisco VPN 3002 Hardware Client ≥ 3.x

Cisco.com

Supports Cisco VPN Client protocol

CSPFA 3.2—15-14

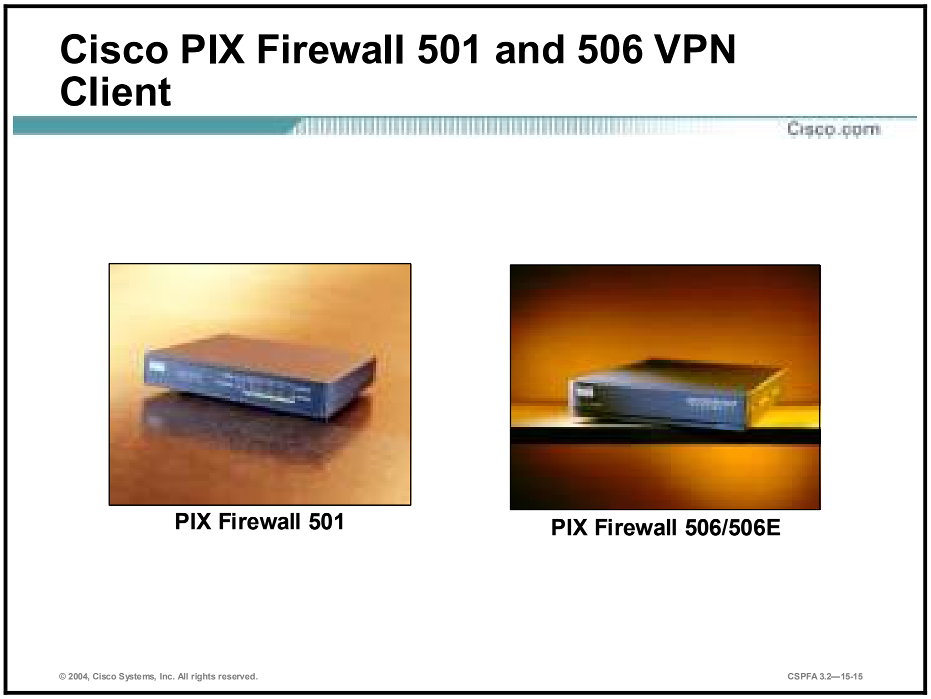
The Cisco VPN 3002 Hardware Client has the Cisco VPN Client software built into it, enabling the Hardware Client to emulate the Cisco VPN 3000 Series Concentrator Software Client. With the Hardware Client, you can plug remote site PCs into the Hardware Client, instead of loading the Cisco VPN Client software, or additional applications on all remote site PCs.

There are two versions of the Hardware Client:

- Hardware Client—One private and one public interface
- Hardware Client—8E
  - One public interface, and the private interface is a built-in eight-port 10/100BASE-T Ethernet switch (100 full duplex is locked in, not configurable)
  - Auto Medium Dependent Interface Crossover (MDIX), a technology which eliminates crossover cables

There are two modes of operation for the Hardware Client: client mode and network extension. These modes are configurable via the command line interface (CLI) or the GUI. They can be remotely managed via an IPSec tunnel or a Secure Shell (SSH).

The Hardware Client is powered by an external power supply. It auto-senses the voltage, either 110V or 220V.



Cisco PIX Firewall 501 and 506 VPN Client

Cisco.com

PIX Firewall 501

PIX Firewall 506/506E

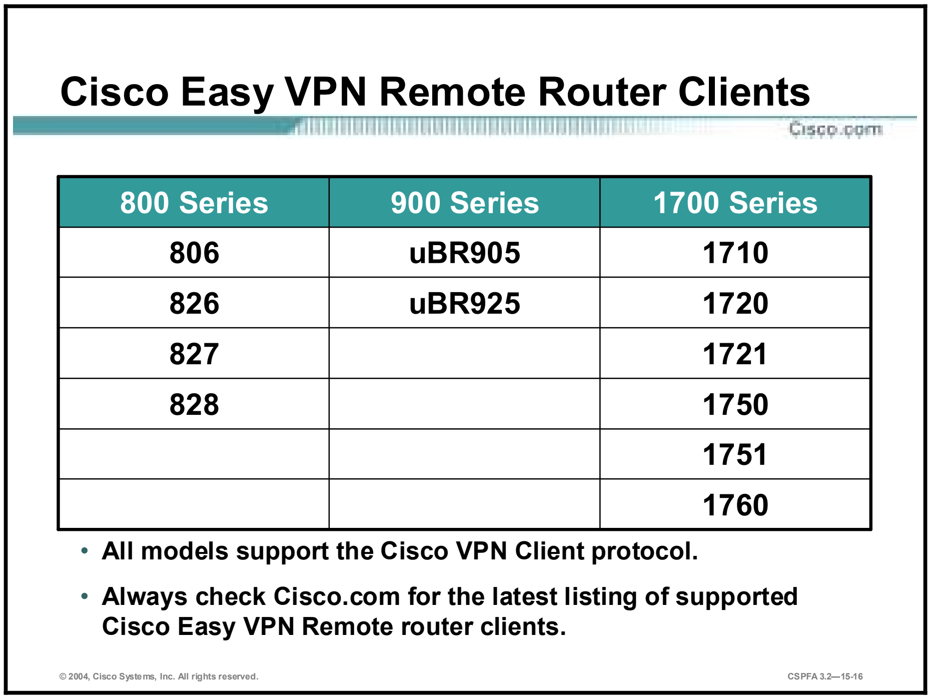
CSPFA 3.2—15-15

The PIX Firewall 501 delivers enterprise-class security for small offices and teleworkers. Ideal for securing high-speed, “always on” broadband environments, the PIX Firewall 501 provides small office networking features and powerful remote management capabilities in a compact, all-in-one solution.

The PIX Firewall 501 provides a convenient way for multiple computers to share a single broadband connection. In addition to its RS-232 (RJ-45) 9600 baud console port and its integrated 10/100BASE-T port (100BASE-T option available in release 6.3) for the outside interface, it features an integrated auto-sensing, auto-MDIX 4-port 10/100 switch for the inside interface. Auto-MDIX support eliminates the need to use crossover cables with devices connected to the switch.

The PIX Firewall 506/506E is designed for companies that are leveraging the cost advantages of the Internet and allowing employees to work remotely. It delivers full firewall protection, as well as IPSec VPN capabilities. The PIX Firewall 506/506E can connect with up to 25 VPN peers simultaneously, and provides users with a complete implementation of IPSec standards. It comes with two integrated 10/100BASE-T ports (100BASE-T option available in release 6.3 for 506E only), is compact in size (8 x 12 x 1.7 inches), and uses Trivial File Transfer Protocol (TFTP) for image download and upgrade.

Note Prior to PIX Firewall Software Release 6.3, the PIX Firewall 501 outside interface and 506E outside and inside interfaces operated at 10BASE-T. With the upgrade to software release 6.3, the PIX Firewall 501 outside interface and 506E outside and inside interfaces can operate at 10/100BASE-T. To enable the speed change on the interface requires a software upgrade only.



Cisco Easy VPN Remote Router Clients

800 Series	900 Series	1700 Series
806	uBR905	1710
826	uBR925	1720
827		1721
828		1750
		1751
		1760

- **All models support the Cisco VPN Client protocol.**
- **Always check Cisco.com for the latest listing of supported Cisco Easy VPN Remote router clients.**

CSPFA 3.2—15-16

Cable modems, xDSL routers, and other forms of broadband access provide high-performance connections to the Internet, but many applications also require the security of VPN connections that perform a high level of authentication and that encrypt the data between two particular endpoints. However, establishing a VPN connection between two VPN devices can be complicated, and typically requires tedious coordination between network administrators to configure the VPN parameters of the two devices.

The Easy VPN Remote feature eliminates much of this tedious work by implementing the Cisco VPN Client protocol, which allows most VPN parameters to be defined at a VPN remote access server. This server can be a dedicated VPN device such as a Concentrator, a Cisco PIX Firewall, or a Cisco IOS router that supports the Cisco VPN Client protocol.

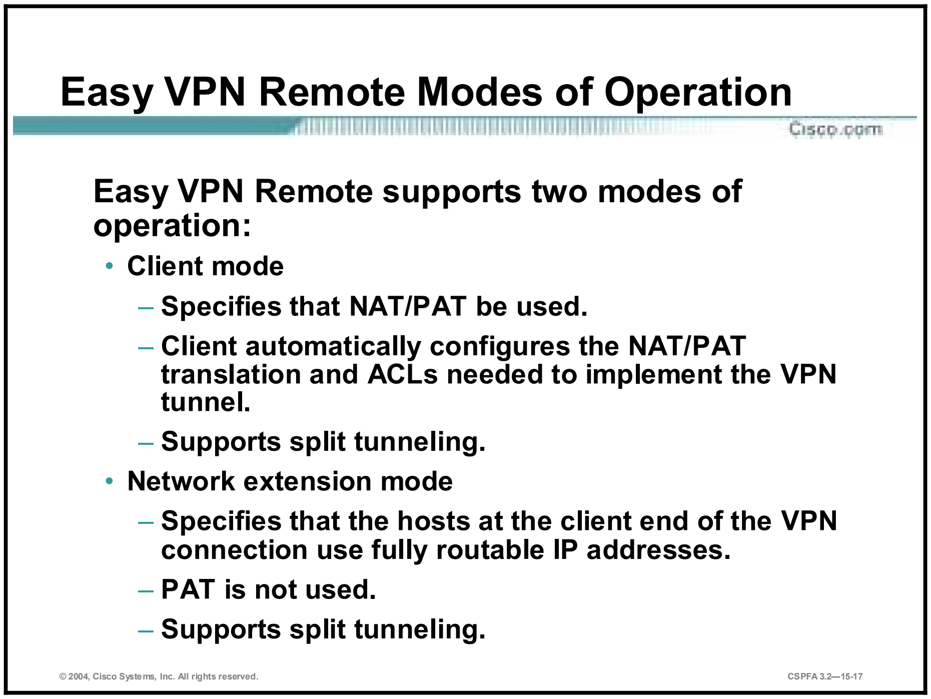
After the VPN remote access server has been configured, a VPN connection can be created with minimal configuration on an IPSec client, such as a Cisco uBR905 or Cisco uBR925 cable access router, as well as on the Cisco 806/826/827/828 and Cisco 1700 series routers. When the IPSec client initiates the VPN connection, the VPN remote access server pushes the IPSec policies to the IPSec client and creates the corresponding IPSec tunnel.

The Cisco Easy VPN Remote feature provides for automatic management of the following details:

- Negotiating tunnel parameters—Addresses, algorithms, lifetime, and so on
- Establishing tunnels according to the parameters
- Automatically creating the Network Address Translation (NAT)/Port Address Translation (PAT) and associated access lists that are needed, if any
- Authenticating users—Making sure that users are who they say they are by way of usernames, group names, and passwords
- Managing security keys for encryption and decryption
- Authenticating, encrypting, and decrypting data through the tunnel

The Cisco Router Easy VPN Remote feature was released in two phases, phase I and phase II. Phase I was introduced as part of IOS version 12.2(4)YA, and phase II as part of IOS version 12.2(8)YJ. The following table summarizes the major differences between the Cisco Easy VPN Remote phase I and phase II feature sets.

Item	Phase I–12.2(4)YA	Phase II–12.2(8)YJ
IPSec tunnel	Tunnel is automatically connected only when a Cisco Easy VPN Remote feature is configured on an interface (automatic tunnel mode).	Establishes and terminates a tunnel on demand (manual tunnel control) or automatically (automatic tunnel mode).
Inside interface	Supports only the default inside interface.	Supports multiple inside interfaces.
Outside interface	Supports only one tunnel for a single outside interface.	Supports multiple tunnels, one tunnel for each outside interface.
Default inside interface	The default inside interface is the FastEthernet0 interface for the Cisco 1700 series. The Ethernet0 interface is the default inside interface for the Cisco 800 and 900 series routers.	■ Cisco 1700 series routers no longer have a default inside interface. Inside interfaces must be configured manually. ■ On Cisco 1700 series routers, the last inside interface of a tunnel can be unconfigured only after unconfiguring the outside interface. ■ Default inside interfaces on the Cisco 800 and 900 series routers can be manually disabled if one other inside interface has been configured on the router.
Cable Dynamic Host Configuration Protocol (DHCP) proxy enhancement	Not supported.	The **cable-modem dhcp-proxy interface** configuration command is enhanced to support the loopback interface for Cisco uBR905 and uBR925 cable access routers, so that a public IP address is automatically assigned to the loopback interface.
NAT interoperability support	NAT configurations are not supported.	A manually assigned NAT configuration on an interface works with the Easy VPN Remote phase II configuration.
IOS Firewall support	IOS Firewall configurations are not supported.	Easy VPN Remote phase II feature works in conjunction with Cisco IOS Firewall configurations.
show crypto ipsec client ezvpn command	No inside or outside interfaces are shown.	Output shows the list of inside and outside interfaces for each channel.



Easy VPN Remote Modes of Operation

Easy VPN Remote supports two modes of operation:

- **Client mode**
  - **Specifies that NAT/PAT be used.**
  - **Client automatically configures the NAT/PAT translation and ACLs needed to implement the VPN tunnel.**
  - **Supports split tunneling.**
- **Network extension mode**
  - **Specifies that the hosts at the client end of the VPN connection use fully routable IP addresses.**
  - **PAT is not used.**
  - **Supports split tunneling.**

CSPFA 3.2—15-17

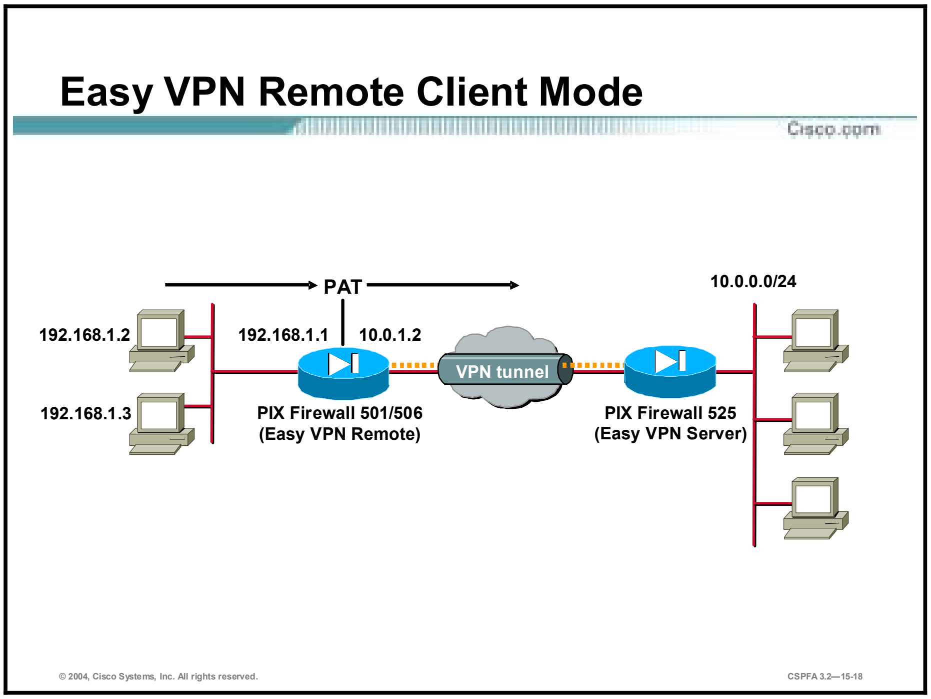
The Easy VPN Remote feature supports two modes of operation:

- Client mode—Specifies that NAT/PAT be configured to allow PCs and hosts on the client side of the VPN connection to form a private network that does not use any IP addresses in the address space of the destination server. In client mode, the Easy VPN Remote feature automatically configures the NAT/PAT translation and access control lists (ACLs) that are needed to implement the VPN connection. These configurations are automatically created when the VPN connection is initiated. When the tunnel is torn down, the NAT/PAT and ACL configurations are automatically deleted.

Note The NAT/PAT translation and ACL configurations that are created by the Easy VPN Remote feature are not written to either the startup-configuration or running-configuration files. These configurations, however, can be displayed in Cisco routers using the **show ip nat statistics** and **show access-list** commands, or in the Cisco PIX Firewalls using the **show vpnclient detail** command.

- Network extension mode—Specifies that the PCs and other hosts at the client end of the IPSec tunnel should be given IP addresses that are fully routable and reachable by the destination network over the tunneled network, so that they form one logical network. PAT is not used, which allows the client PCs and hosts to have direct access to the PCs and hosts on the destination network.

Both modes of operation also optionally support split tunneling, which allows secure access to corporate resources through the IPSec tunnel while also allowing Internet access through a connection to an Internet Service Provider (ISP) or other service—thereby eliminating the corporate network from the path for web access.

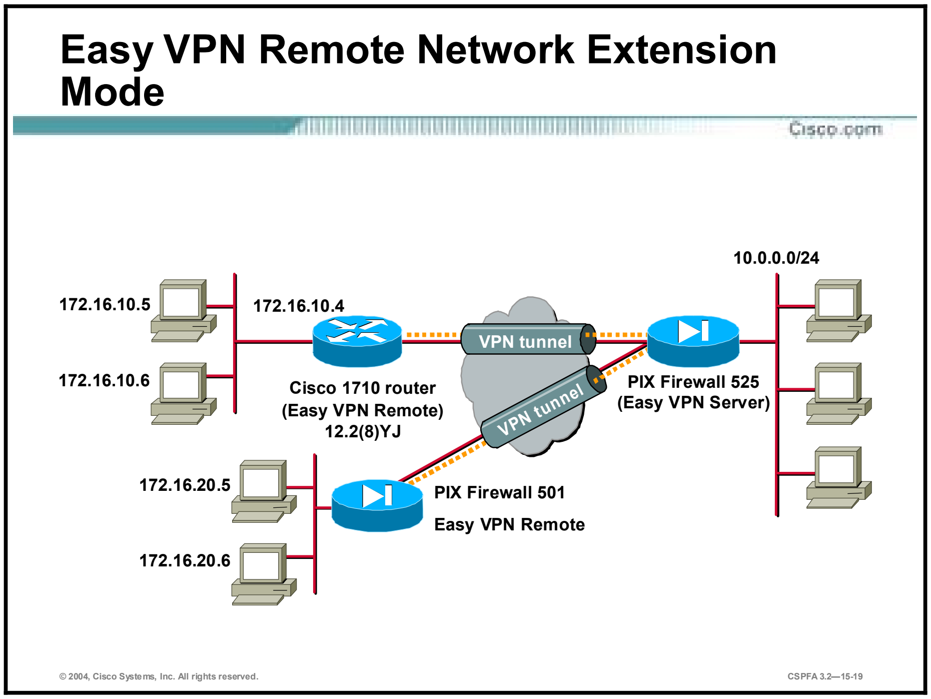


Easy VPN Remote Client Mode

CSPFA 3.2—15-18

Client mode is for those who want to deploy a VPN quickly and easily in a small office/home office (SOHO). If there is no need to see the devices behind the Hardware Client, and ease of use and installation is the key, then client mode should be implemented. In client mode, the Hardware Client uses PAT to isolate its private network from the public network. SOHO PCs behind the Hardware Client are invisible to the outside network. PAT causes all traffic from the SOHO PCs to appear on the private network as a single-source IP address. This figure illustrates the Easy VPN Remote client mode of operation. In this example, the PIX Firewall 501 provides access to two PCs, which have IP addresses in the 192.168.1.0 private network space. These PCs connect to the Ethernet interface on the PIX Firewall 501. In this example, the two PC IP addresses are translated to the PIX Firewall 501 outside IP address 10.0.1.2. The PIX Firewall 501 performs NAT/PAT translation over the IPSec tunnel so that the PCs can access the destination network.

This figure could also represent a split tunneling connection, in which the client PCs can access public resources in the Internet without including the corporate network in the path for the public resources.



Easy VPN Remote Network Extension Mode

Cisco.com

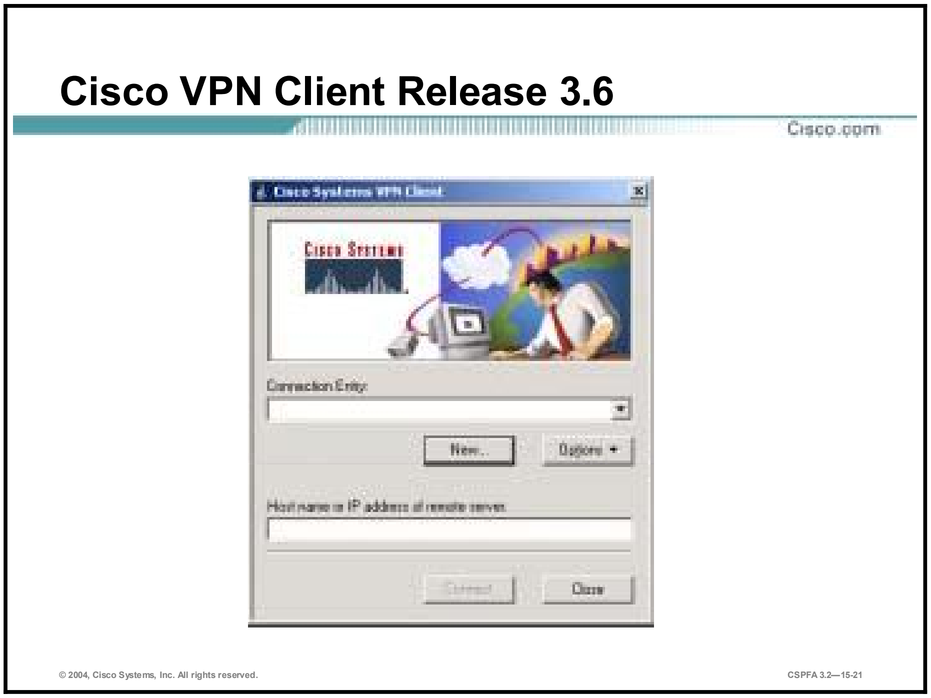
CSPFA 3.2—15-19

In network extension mode, all SOHO PCs on the Hardware Client are uniquely addressable via the tunnel. This allows direct connection to devices behind the Hardware Client. It enables central site Management Information System (MIS) personnel at the central site to directly address devices behind the Hardware Client over the IPSec tunnel. This figure illustrates the network extension mode of operation. In this example, the PIX Firewall 501 and Cisco 1700 series router both act as Cisco Easy VPN Remote clients, connecting to a PIX Firewall 525 Easy VPN Server.

The client hosts are given IP addresses that are fully routable by the destination network over the tunnel. These IP addresses could be either in the same subnet space as the destination network, or they could be in separate subnets, as long as the destination routers are configured to properly route those IP addresses over the tunnel. This provides a seamless extension of the remote network.

Overview of the Cisco VPN 3.6 Client

This topic introduces you to the Cisco VPN Client release 3.6. The Cisco VPN Client is software that enables customers to establish secure, end-to-end encrypted tunnels to any Easy VPN Server. This thin design, which is an IPSec-compliant implementation, is available via Cisco.com for customers with SMARTnet support, and is included free of charge with the Concentrator.



Cisco VPN Client Release 3.6

Cisco.com

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—15-21

This figure displays the Cisco VPN Client splash window. Users can preconfigure the connection entry (name of connection) and hostname or IP address of remote Easy VPN Servers. Clicking **Connect** initiates IKE Phase 1.

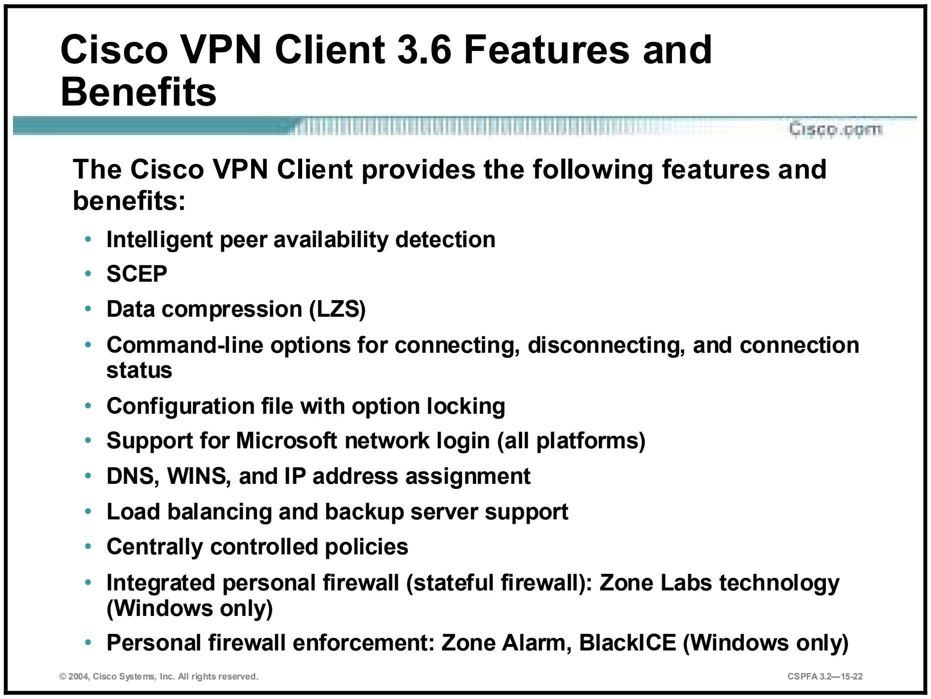
The Cisco VPN Client can be preconfigured for mass deployments, and initial logins require very little user intervention. VPN access policies and configurations are downloaded from the Easy VPN Server and pushed to the Cisco VPN Client when a connection is established, allowing simple deployment and management.

The Cisco VPN Client provides support for the following operating systems:

- Windows 95, 98, ME, NT 4.0, 2000, and XP
- Linux (Intel)
- Solaris (UltraSPARC-32 and -64 bit)
- MAC OS X 10.1.

The Cisco VPN Client is compatible with the following Cisco products (Easy VPN Servers):

- Cisco IOS software-based platforms version 12.2(8)T and later
- Cisco VPN 3000 Series Concentrator version 3.0 and later
- Cisco PIX Firewall version 6.0 and later



Cisco VPN Client 3.6 Features and Benefits

Cisco.com

The Cisco VPN Client provides the following features and benefits:

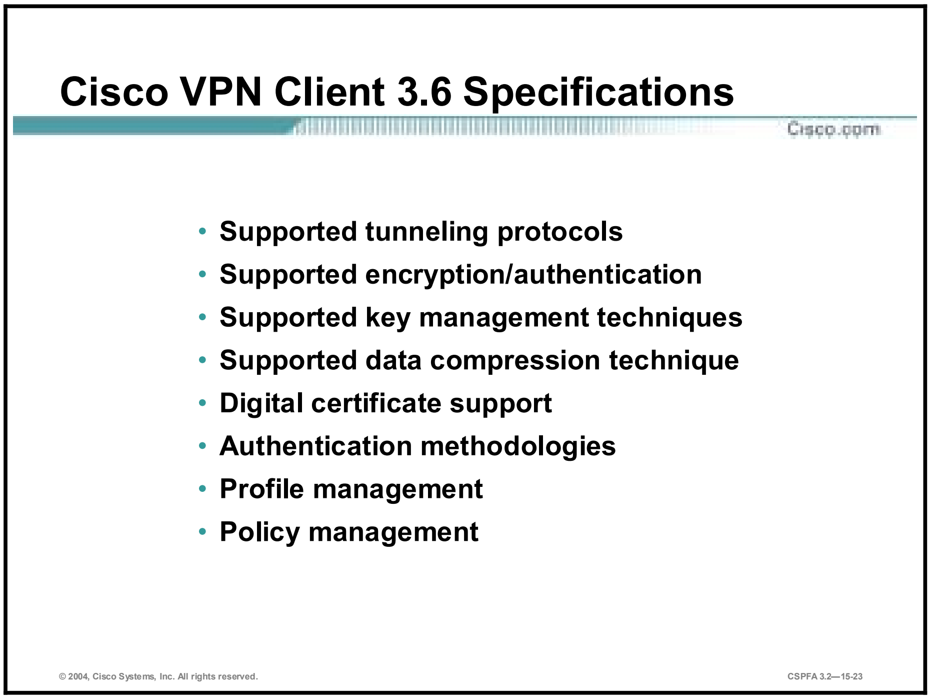
- Intelligent peer availability detection
- SCEP
- Data compression (LZS)
- Command-line options for connecting, disconnecting, and connection status
- Configuration file with option locking
- Support for Microsoft network login (all platforms)
- DNS, WINS, and IP address assignment
- Load balancing and backup server support
- Centrally controlled policies
- Integrated personal firewall (stateful firewall): Zone Labs technology (Windows only)
- Personal firewall enforcement: Zone Alarm, BlackICE (Windows only)

CSPFA 3.2—15-22

The Cisco VPN Client provides the following features and benefits:

- Intelligent peer availability detection
- Simple certificate enrollment protocol (SCEP)
- Data compression (LZS)
- Command-line options for connecting, disconnecting, and connection status
- Configuration file with option locking
- Support for Microsoft network login (all platforms)
- Domain Name System (DNS), Windows Internet Name Service (WINS), and IP address assignment
- Load balancing and backup server support
- Centrally controlled policies
- Integrated personal firewall (stateful firewall): Zone Labs technology (Windows only)
- Personal firewall enforcement: Zone Alarm, BlackICE (Windows only)

Please note that the Cisco VPN Client supports more features than the Easy VPN Server can accommodate. Always compare the Cisco VPN Client specifications against the Easy VPN Server supported and unsupported feature lists. For example, although the Cisco VPN Client supports Zone Labs and BlackICE personal firewall features, the Easy VPN Server does not.



Cisco VPN Client 3.6 Specifications

Cisco.com

- Supported tunneling protocols
- Supported encryption/authentication
- Supported key management techniques
- Supported data compression technique
- Digital certificate support
- Authentication methodologies
- Profile management
- Policy management

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—15-23

The specifications for the Cisco VPN Client release 3.6 product are as follows:

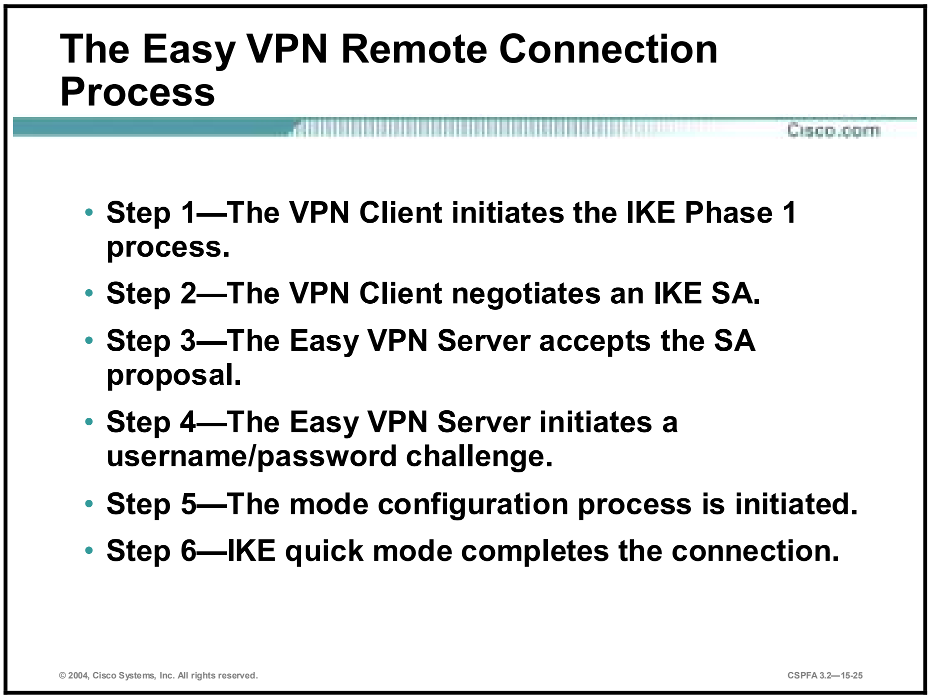
- Supported tunneling protocols:
  - IPSec Encapsulating Security Payload (ESP)
  - Transparent tunneling:
    - IPSec over TCP (NAT or PAT)
    - IPSec over UDP (NAT, PAT, or a firewall)
- Supported encryption and authentication—IPSec (ESP) using DES, 3DES (56/168-bit), and AES (128/256-bit) with MD5 or SHA
- Supported key management techniques
  - IKE—Aggressive and main mode (digital certificates)
  - DH groups 1, 2, 5, and 7
- Supported data compression technique—LZS compression
- Digital certificate support includes the following:
  - Two supported enrollment mechanisms
    - SCEP
    - Certificates enrolled with Microsoft Internet Explorer
  - The supported Certificate Authorities (CAs)
    - Entrust
    - GTE Cybertrust
    - Netscape
    - Baltimore

        - RSA Keon
        - VeriSign
        - Microsoft
    - Support is provided for the Entrust Entelligence Client
    - Smartcards—Supported via MS crypto API (CRYPT_NOHASHOID), including the following:
        - ActivCard (Schlumberger cards)
        - eAladdin
        - Gemplus
        - Datakey
- Authentication methodologies include the following:
    - XAUTH
    - Remote Authentication Dial-In User Service (RADIUS) with support for:
        - State—Or reply-message attributes (token cards)
        - Security Dynamics (RSA SecurID ready)
        - Microsoft NT Domain authentication
        - Microsoft Challenge Handshake Authentication Protocol version 2 (MSCHAPv2)—NT password expiration
        - X.509 version 3 digital certificates
- Profile management—The Cisco VPN Client can be easily distributed with preconfigured Profile Configuration Files (PCFs)
- Policy management—Internet Security Association Key Management Protocol (ISAKMP)
    - Keeps track of centrally controlled policies such as the following:
        - DNS information
        - WINS information
        - IP address
        - Default domain name
    - Contains the ability to save connection attributes.

How the Cisco Easy VPN Works

When an Easy VPN Remote client initiates a connection with a Easy VPN Server gateway, the "conversation" that occurs between the peers generally consists of the following major steps:

- Device authentication via IKE
- User authentication using IKE XAUTH
- VPN policy push (using mode configuration)
- IPSec SA creation



The Easy VPN Remote Connection Process

- Step 1—The VPN Client initiates the IKE Phase 1 process.
- Step 2—The VPN Client negotiates an IKE SA.
- Step 3—The Easy VPN Server accepts the SA proposal.
- Step 4—The Easy VPN Server initiates a username/password challenge.
- Step 5—The mode configuration process is initiated.
- Step 6—IKE quick mode completes the connection.

The following is a detailed description of the Easy VPN Remote connection process:

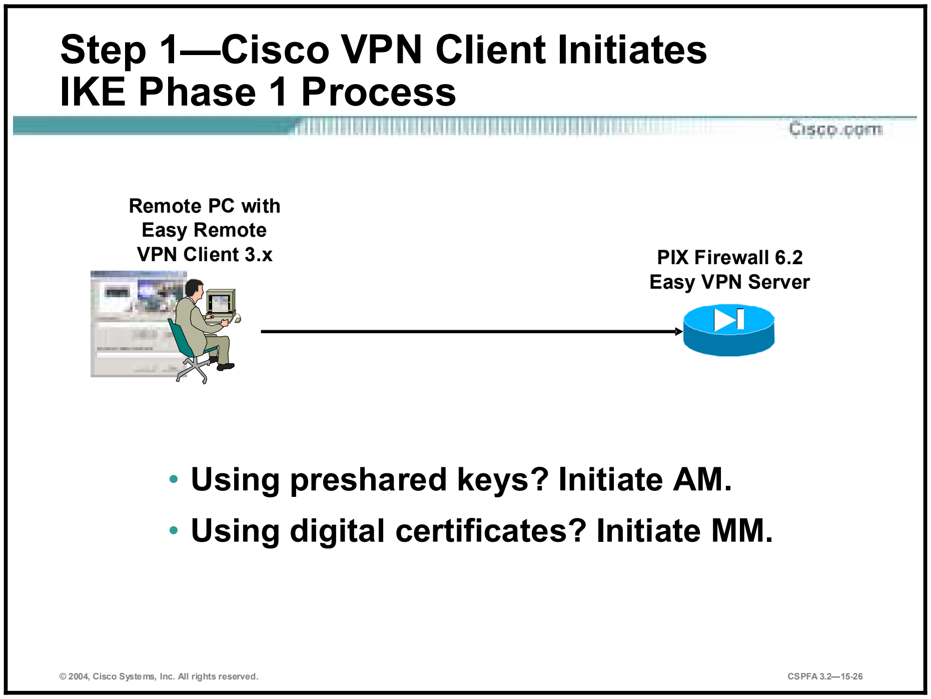
Step 1 The Cisco VPN Client initiates the IKE Phase 1 process. The Cisco VPN Client negotiates an IKE SA.

Step 3 The Easy VPN Server accepts the SA proposal.

Step 4 The Easy VPN Server initiates a username/password challenge.

Step 5 The mode configuration process is initiated.

Step 6 IKE quick mode completes the connection.



Step 1—Cisco VPN Client Initiates IKE Phase 1 Process

Remote PC with
Easy Remote
VPN Client 3.x

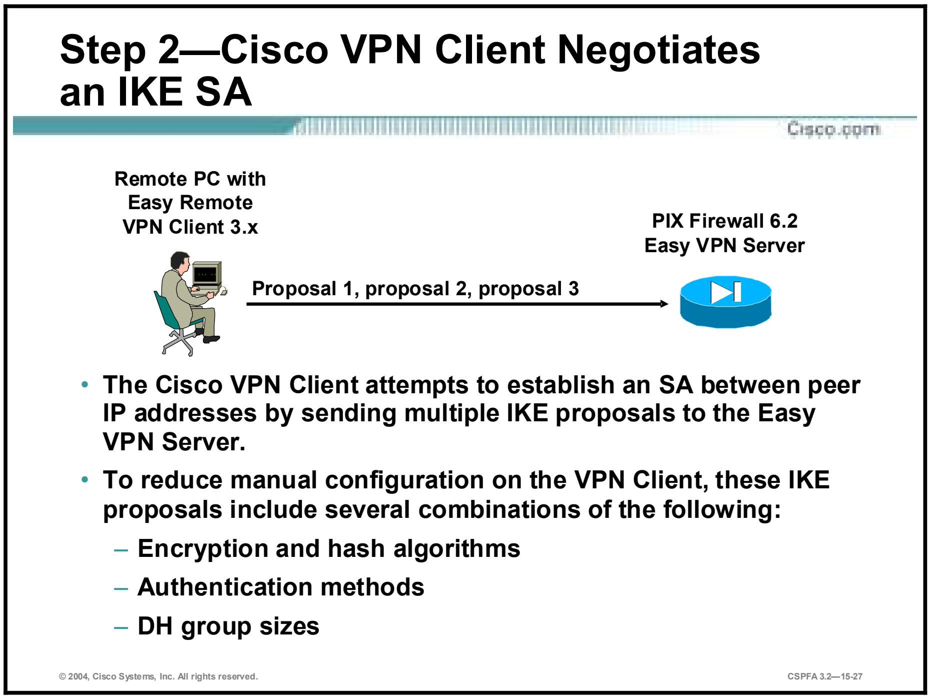
PIX Firewall 6.2
Easy VPN Server

- Using preshared keys? Initiate AM.
- Using digital certificates? Initiate MM.

CSPFA 3.2—15-26

Because there are two ways to perform authentication, the Cisco VPN Client must consider the following when initiating this phase:

- If a pre-shared key is to be used for authentication, the Cisco VPN Client initiates aggressive mode (AM). When pre-shared keys are used, the accompanying group name entered in the configuration GUI (ID_KEY_ID) is used to identify the group profile associated with this Cisco VPN Client.
- If digital certificates are to be used for authentication, the Cisco VPN Client initiates main mode (MM). When digital certificates are used, the Organizational Unit (OU) field of a distinguished name (DN) is used to identify the group profile.



Step 2—Cisco VPN Client Negotiates an IKE SA

Remote PC with
Easy Remote
VPN Client 3.x

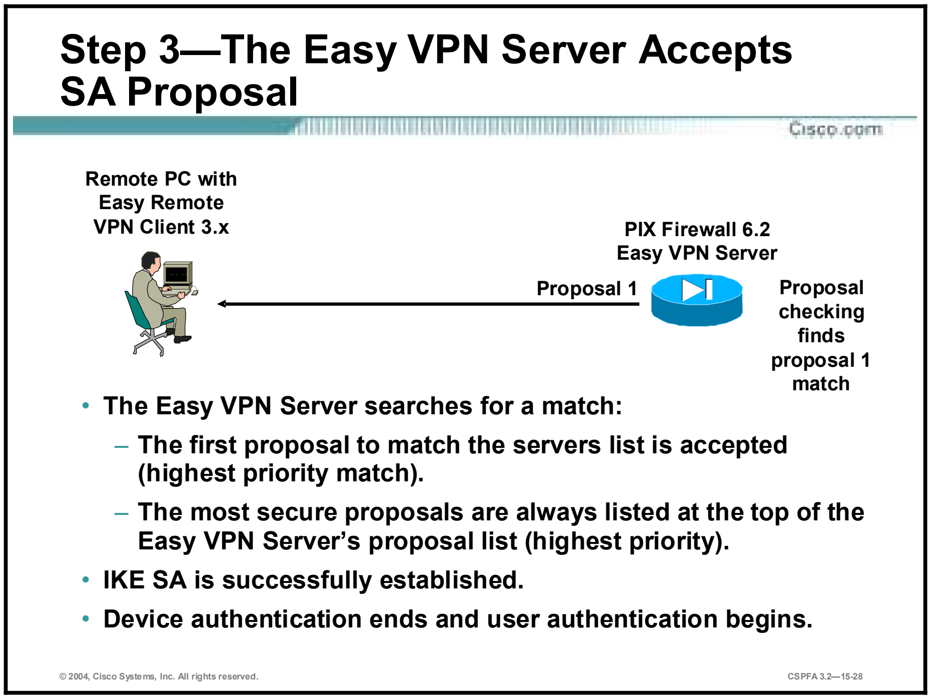
Proposal 1, proposal 2, proposal 3

PIX Firewall 6.2
Easy VPN Server

- The Cisco VPN Client attempts to establish an SA between peer IP addresses by sending multiple IKE proposals to the Easy VPN Server.
- To reduce manual configuration on the VPN Client, these IKE proposals include several combinations of the following:
  - Encryption and hash algorithms
  - Authentication methods
  - DH group sizes

CSPFA 3.2—15-27

To reduce the amount of manual configuration on the Cisco VPN Client, a fixed combination of encryption, hash algorithms, authentication methods, and DH group sizes is proposed.



Step 3—The Easy VPN Server Accepts SA Proposal

Cisco.com

Remote PC with Easy Remote VPN Client 3.x

PIX Firewall 6.2 Easy VPN Server

Proposal 1

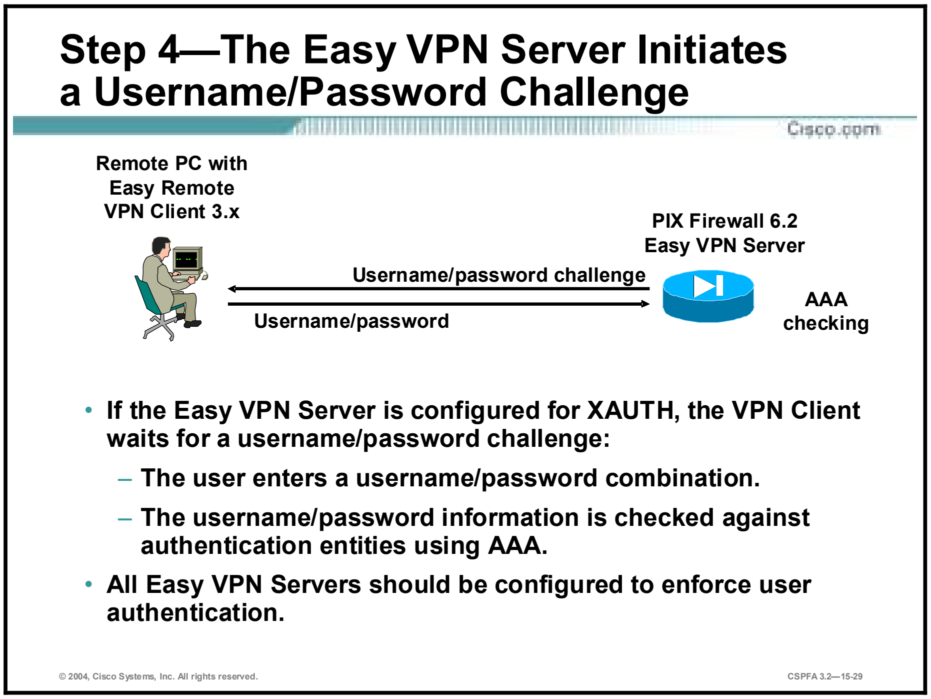
Proposal checking finds proposal 1 match

- The Easy VPN Server searches for a match:
  - The first proposal to match the servers list is accepted (highest priority match).
  - The most secure proposals are always listed at the top of the Easy VPN Server’s proposal list (highest priority).
- IKE SA is successfully established.
- Device authentication ends and user authentication begins.

CSPFA 3.2—15-28

IKE policy is global for the Easy VPN Server and can consist of several proposals. In the case of multiple proposals, the Easy VPN Server will use the first match (so you should always have your most secure policies listed first).

Device authentication ends and user authentication begins at this point.



Step 4—The Easy VPN Server Initiates a Username/Password Challenge

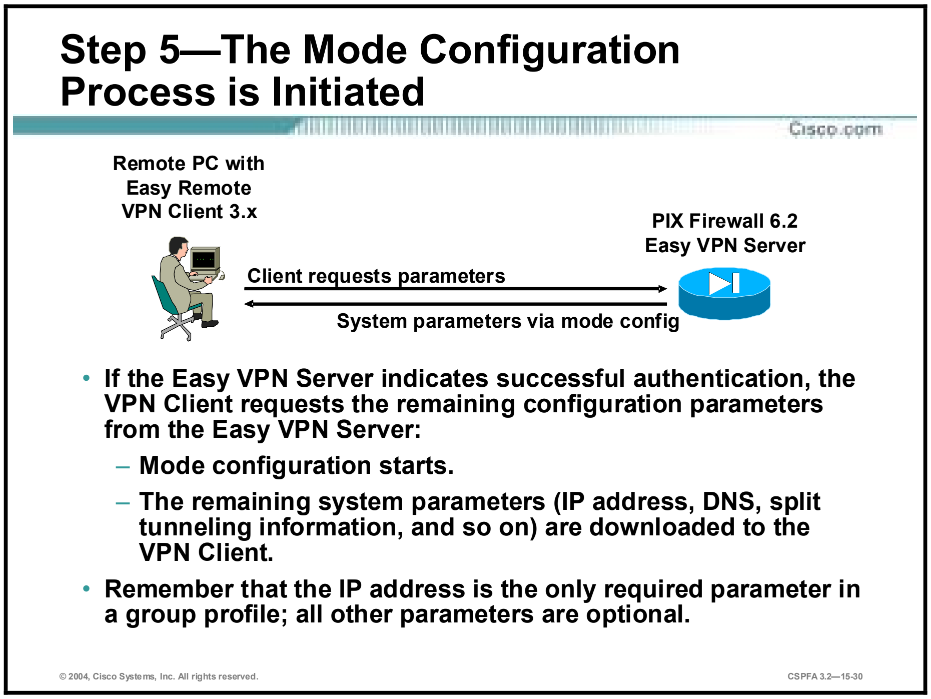
Cisco.com

- If the Easy VPN Server is configured for XAUTH, the VPN Client waits for a username/password challenge:
  - The user enters a username/password combination.
  - The username/password information is checked against authentication entities using AAA.
- All Easy VPN Servers should be configured to enforce user authentication.

CSPFA 3.2—15-29

The information that is entered is checked against authentication entities using authentication, authorization, and accounting (AAA) protocols such as RADIUS and Terminal Access Controller Access Control System Plus (TACACS+). Token cards may also be used via AAA proxy.

VPN devices that are configured to handle remote Cisco VPN Clients should always be configured to enforce user authentication.



Step 5—The Mode Configuration Process is Initiated

Remote PC with Easy Remote VPN Client 3.x

PIX Firewall 6.2 Easy VPN Server

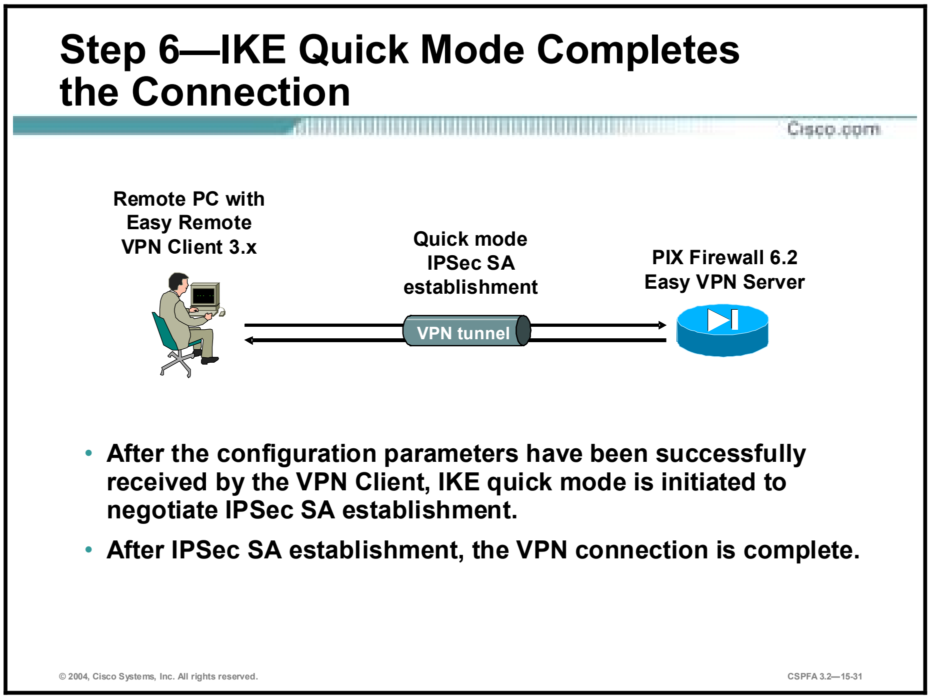
Client requests parameters

System parameters via mode config

- If the Easy VPN Server indicates successful authentication, the VPN Client requests the remaining configuration parameters from the Easy VPN Server:
  - Mode configuration starts.
  - The remaining system parameters (IP address, DNS, split tunneling information, and so on) are downloaded to the VPN Client.
- Remember that the IP address is the only required parameter in a group profile; all other parameters are optional.

CSPFA 3.2—15-30

The remaining system parameters (IP address, DNS, split tunnel attributes, and so on) are pushed to the Cisco VPN Client at this time using mode configuration. The IP address is the only required parameter in a group profile; all other parameters are optional.



Step 6—IKE Quick Mode Completes the Connection

Cisco.com

Remote PC with Easy Remote VPN Client 3.x

Quick mode IPSec SA establishment

VPN tunnel

PIX Firewall 6.2 Easy VPN Server

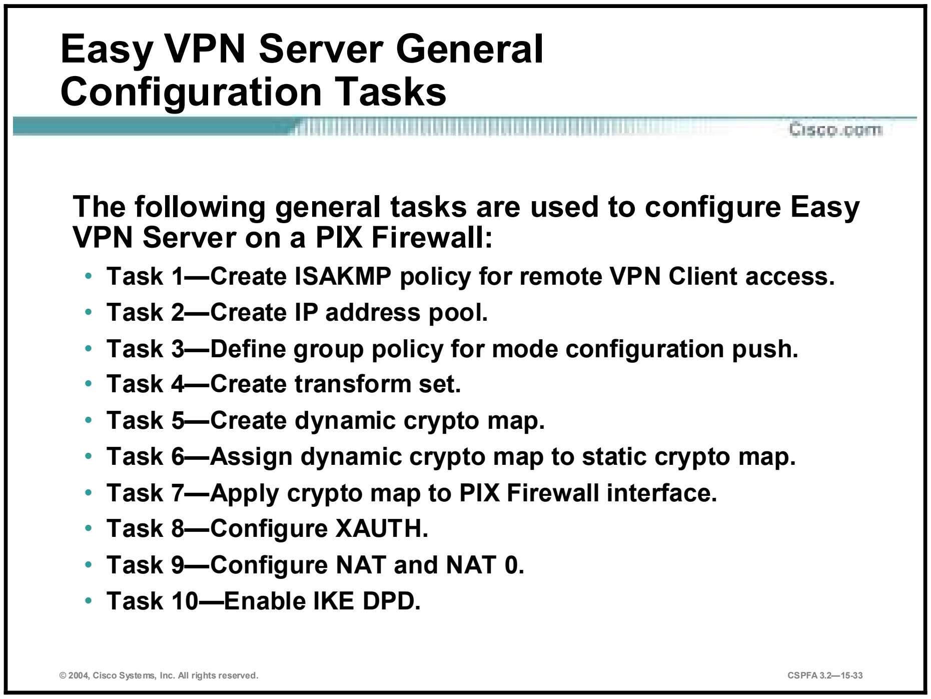
- After the configuration parameters have been successfully received by the VPN Client, IKE quick mode is initiated to negotiate IPSec SA establishment.
- After IPSec SA establishment, the VPN connection is complete.

CSPFA 3.2—15-31

After IPSec SAs are created, the connection is complete.

Configuring the Easy VPN Server for Extended Authentication

This topic examines the general tasks used to configure an Easy VPN Server to support XAUTH for Easy VPN Remote Cisco VPN Client access.



Easy VPN Server General Configuration Tasks

Cisco.com

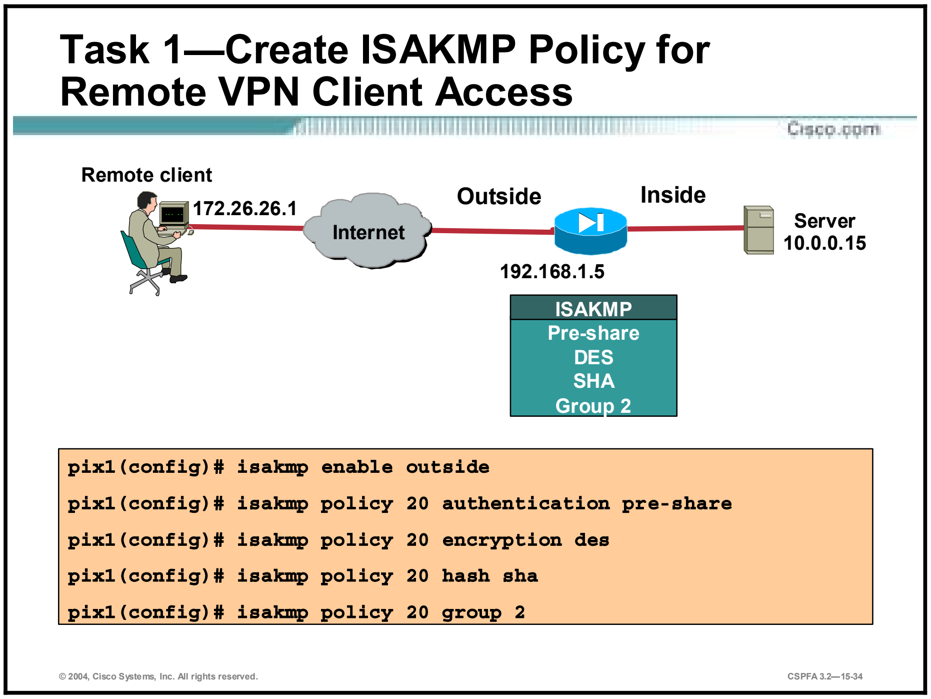
The following general tasks are used to configure Easy VPN Server on a PIX Firewall:

- **Task 1—Create ISAKMP policy for remote VPN Client access.**
- **Task 2—Create IP address pool.**
- **Task 3—Define group policy for mode configuration push.**
- **Task 4—Create transform set.**
- **Task 5—Create dynamic crypto map.**
- **Task 6—Assign dynamic crypto map to static crypto map.**
- **Task 7—Apply crypto map to PIX Firewall interface.**
- **Task 8—Configure XAUTH.**
- **Task 9—Configure NAT and NAT 0.**
- **Task 10—Enable IKE DPD.**

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—15-33

Complete the following tasks to configure an Easy VPN Server for XAUTH with Easy VPN Remote clients.

- Task 1—Create an ISAKMP policy for remote Cisco VPN Client access.
- Task 2—Create an IP address pool.
- Task 3—Define a group policy for a mode configuration push.
- Task 4—Create a transform set.
- Task 5—Create a dynamic crypto map.
- Task 6—Assign a dynamic crypto map to a static crypto map.
- Task 7—Apply a dynamic crypto map to the PIX Firewall interface.
- Task 8—Configure XAUTH.
- Task 9—Configure NAT and NAT 0.
- Task 10—Enable IKE dead peer detection (DPD).



Task 1—Create ISAKMP Policy for Remote VPN Client Access

```
pix1(config)# isakmp enable outside
pix1(config)# isakmp policy 20 authentication pre-share
pix1(config)# isakmp policy 20 encryption des
pix1(config)# isakmp policy 20 hash sha
pix1(config)# isakmp policy 20 group 2
```

CSPFA 3.2—15-34

Complete this task to configure the ISAKMP policy for Easy VPN Server. Use the standard ISAKMP configuration commands to accomplish this task. Here is a general example of how to configure the ISAKMP policy starting in global configuration mode:

```
pix1(config)# isakmp enable outside
pix1(config)# isakmp policy 20 authen pre-share
pix1(config)# isakmp policy 20 encryption 3des
pix1(config)# isakmp policy 20 hash sha
pix1(config)# isakmp policy 20 group 2
```

The syntax for these commands is as follows:

isakmp policy *priority* **authentication** {*pre-share* | *rsa-sig*}

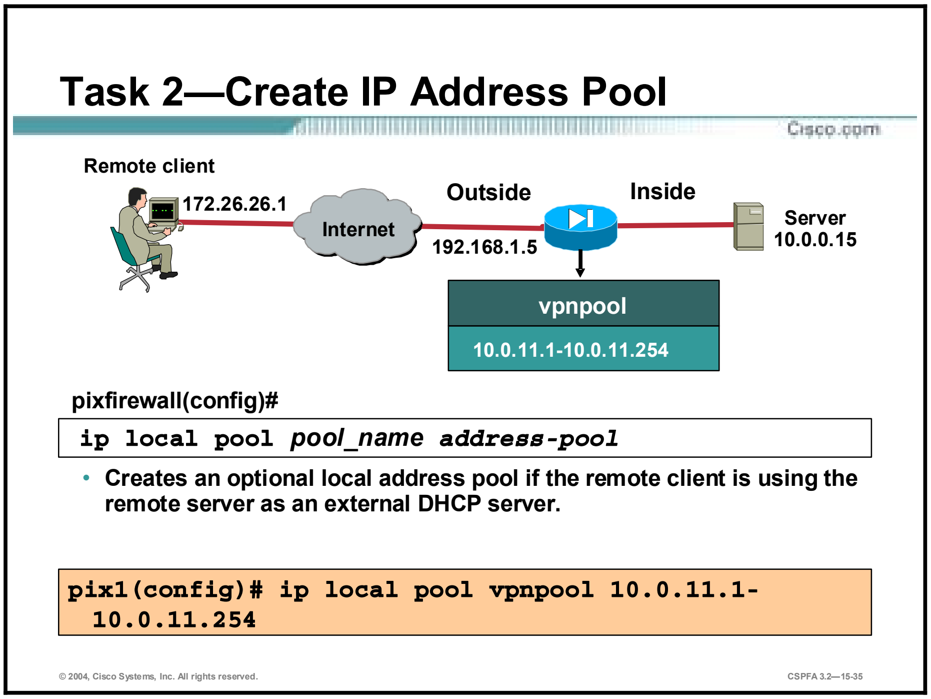
isakmp policy *priority* **encryption** {**aes** | **aes-192**| **aes-256** | **des** | **3des**}

isakmp policy *priority* **group** {**1** | **2** | **5**}

isakmp policy *priority* **hash** *md5* | *sha* **isakmp policy** *priority* **lifetime** *seconds*

aes	Selecting this option means that encrypted IKE messages protected by this suite are encrypted using AES with a 128-bit key.
aes-192	Selecting this option means that encrypted IKE messages protected by this suite are encrypted using AES with a 192-bit key.
aes-256	Selecting this option means that encrypted IKE messages protected by this suite are encrypted using AES with a 256-bit key.
des	Specifies 56-bit DES-CBC as the encryption algorithm to be used in the IKE policy.

3des	Specifies that the Triple DES encryption algorithm is to be used in the IKE policy.
group 1	Specify that the 768-bit Diffie-Hellman group is to be used in the IKE policy. This is the default value.
group 2	Specifies that the 1024-bit Diffie-Hellman group 2 be used in the IKE policy.
group 5	Specify that the 768-bit Diffie-Hellman group is to be used in the IKE policy. This is the default value.
hash	Specifies that the 1536-bit Diffie-Hellman group 5 be used in the IKE policy.
lifetime	Specify how many seconds each security association should exist before expiring. To propose a finite lifetime, use an integer from 120 to 86,400 seconds (one day). Specify 0 seconds for infinite lifetime.
md5	Specify MD5 (HMAC variant) as the hash algorithm to be used in the IKE policy.
pre-share	Specify pre-shared keys as the authentication method.
priority	Uniquely identifies the Internet Key Exchange (IKE) policy and assigns a priority to the policy. Use an integer from 1 to 65,534, with 1 being the highest priority and 65,534 the lowest.
rsa-sig	Specify RSA signatures as the authentication method. RSA signatures provide non-repudiation for the IKE negotiation. This basically means you can prove to a third party whether you had an IKE negotiation with the peer.
sha	Specify SHA-1 (HMAC variant) as the hash algorithm to be used in the IKE policy. This is the default hash algorithm.



Task 2—Create IP Address Pool

Cisco.com
Remote client
172.26.26.1
Internet
Outside
192.168.1.5
Inside
Server
10.0.0.15
vpnpool
10.0.11.1-10.0.11.254

pixfirewall(config)#

```
ip local pool pool_name address-pool
```

- **Creates an optional local address pool if the remote client is using the remote server as an external DHCP server.**

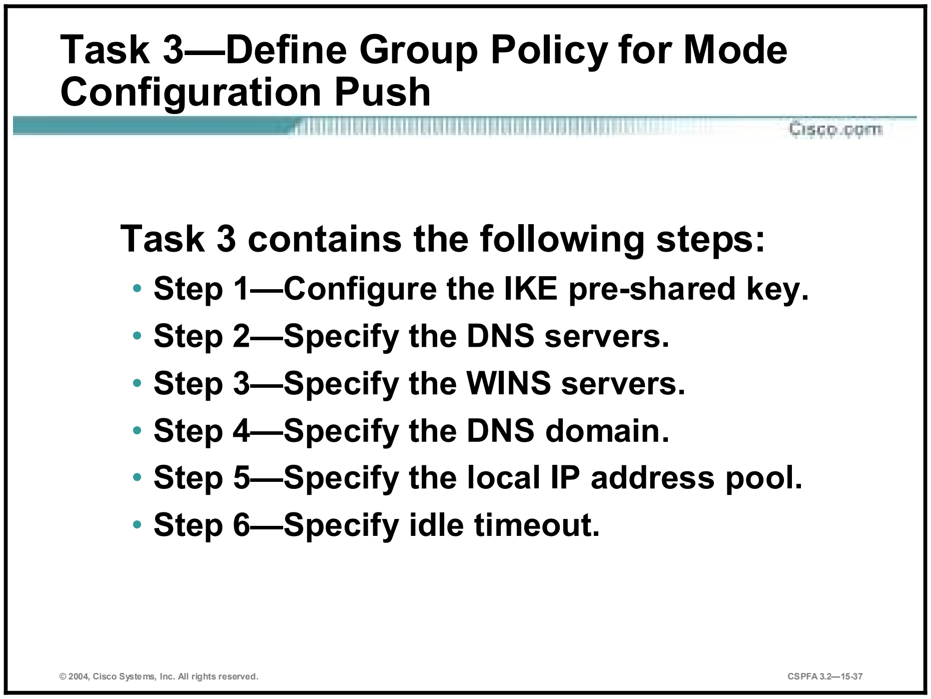
```
pix1(config)# ip local pool vpnpool 10.0.11.1-
  10.0.11.254
```

CSPFA 3.2—15-35

If you are using a local IP address pool, you will also need to configure that pool using the **ip local pool** command. The syntax for this command is as follows:

ip local pool { *pool-name low-ip-address* [*high-ip-address*]}

pool_name	Local pool name.
low-ip-address ***high-ip-address***	Local pool IP address range.



Task 3—Define Group Policy for Mode Configuration Push

Cisco.com

Task 3 contains the following steps:

- **Step 1—Configure the IKE pre-shared key.**
- **Step 2—Specify the DNS servers.**
- **Step 3—Specify the WINS servers.**
- **Step 4—Specify the DNS domain.**
- **Step 5—Specify the local IP address pool.**
- **Step 6—Specify idle timeout.**

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—15-37

Complete this task to define a group policy to be pushed during mode configuration. Although users can belong to only one group per connection, they may belong to specific groups with different policy requirements.

Use the following steps beginning in global configuration mode to define the policy attributes that are pushed to the Cisco VPN Client via mode configuration:

Step 1 Configure the IKE pre-shared key.

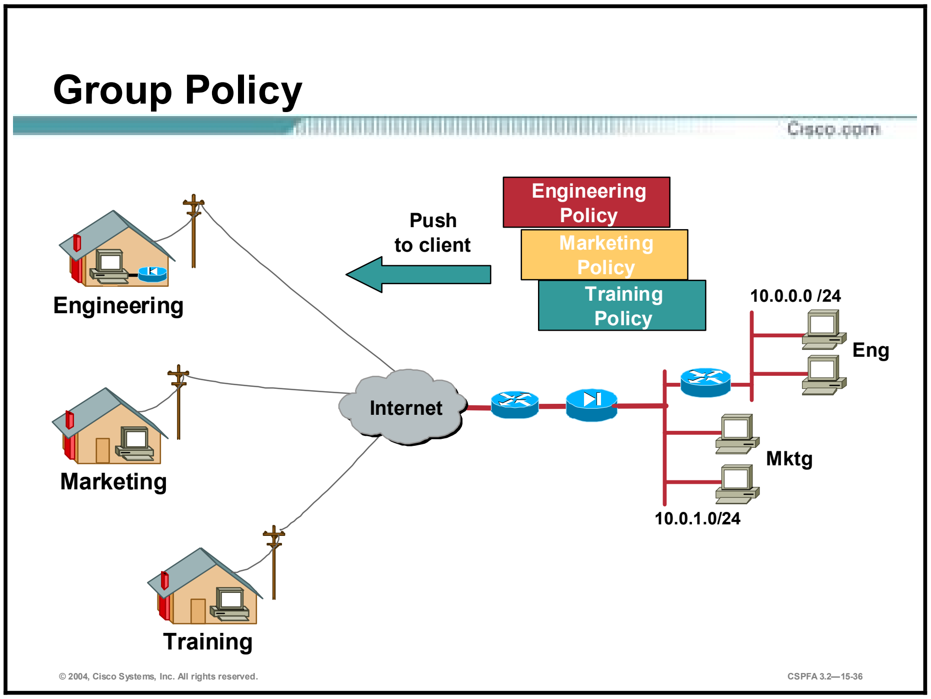
Step 2 Specify the DNS servers.

Step 3 Specify the WINS servers.

Step 4 Specify the DNS domain.

Step 5 Specify the local IP address pool.

Step 6 Specify the idle timeout.



Group Policy

Cisco.com

Push to client

Engineering Policy

Marketing Policy

Training Policy

Engineering

Marketing

Training

Internet

10.0.0.0 /24

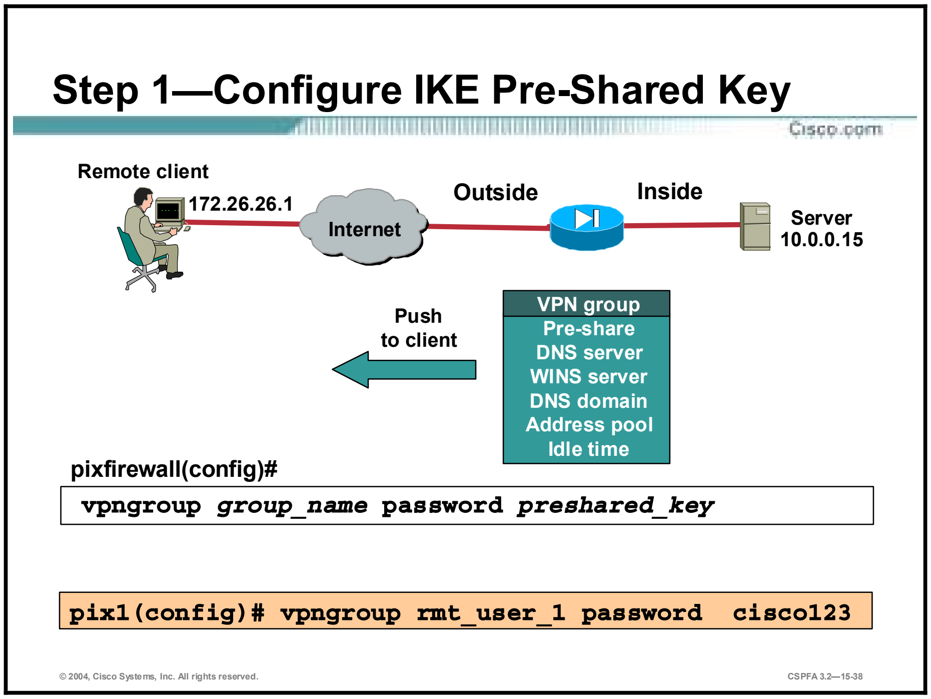
Eng

Mktg

10.0.1.0/24

CSPFA 3.2—15-36

Each remote VPN user belongs to a specific VPN group. As users establish VPN tunnels to the Easy VPN Server, they identify which group they belong to. The Easy VPN Server responds by pushing the appropriate VPN group policy to the remote user. In the figure, there are three VPN group policies configured, the engineering, marketing, and training VPN group policies. Each VPN client belongs to one group. As they establish VPN tunnels, they identify which VPN group they belong to. The central site PIX Firewall pushes a specific policy to each remote user. The VPN group policies are configured centrally on the Easy VPN server using the **vpngroup** command.



Step 1—Configure IKE Pre-Shared Key

Remote client

172.26.26.1

Internet

Outside

Inside

Server
10.0.0.15

Push
to client

VPN group
Pre-share
DNS server
WINS server
DNS domain
Address pool
Idle time

pixfirewall(config)#

```
vpngroup group_name password preshared_key
```

```
pix1(config)# vpngroup rmt_user_1 password  cisco123
```

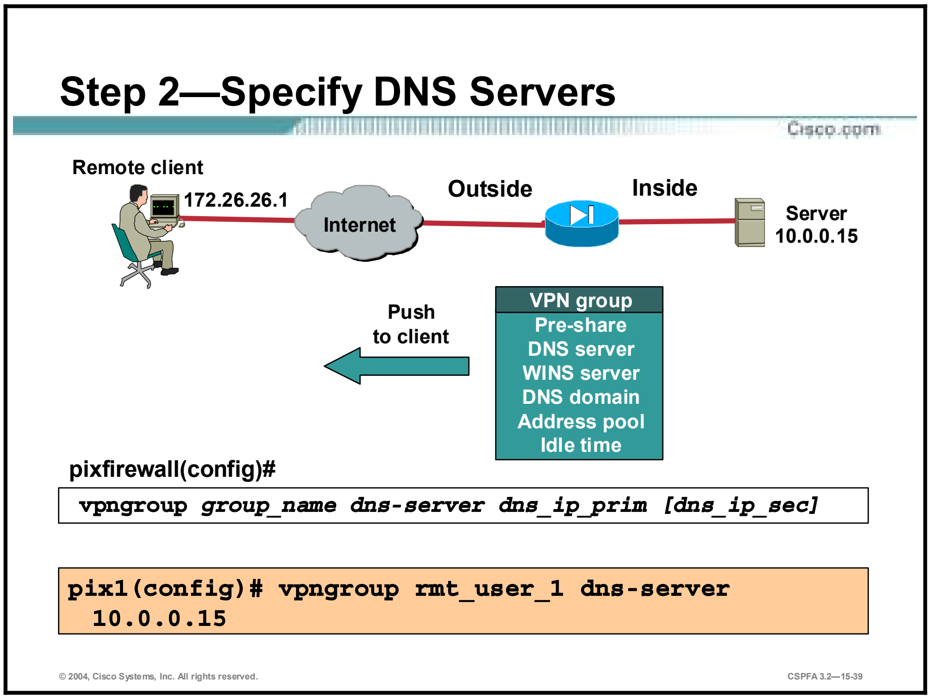
CSPFA 3.2—15-38

Step 1 Use the **vpngroup** command to specify the IKE pre-shared key when defining group policy information for the mode configuration push. You must use this command if the Cisco VPN Client identifies itself to the router with a pre-shared key.

The syntax for the **vpngroup** command is as follows:

vpnclient vpngroup *group_name* **password** *preshared_key*

group_name	The name of the VPN group configured on the VPN headend. The maximum length is 63 characters and no spaces are permitted.
password	Specifies to set the password.
preshared_key	The IKE pre-shared key used for authentication by the Easy VPN Server. The maximum length is 127 characters.



Step 2—Specify DNS Servers

pixfirewall(config)#

```
vpngroup group_name dns-server dns_ip_prim [dns_ip_sec]
```

```
pix1(config)# vpngroup rmt_user_1 dns-server
 10.0.0.15
```

CSPFA 3.2—15-39

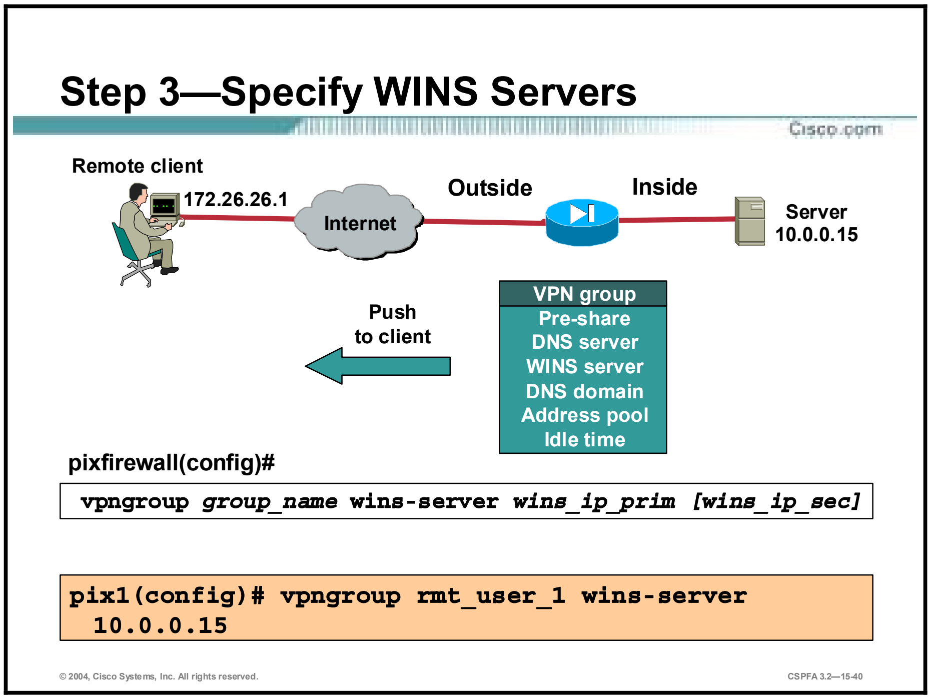
Step 2 (Optional.) Specify the primary and secondary DNS servers using the **dns** command in ISAKMP group configuration mode.

Note You must enable the **crypto isakmp configuration group** command, which specifies group policy information that needs to be defined or changed, before using the **dns** command.

The syntax for the **dns** command is as follows:

dns ***primary-server secondary-server***

primary-server	Name or IP address of the primary DNS server.
secondary-server	Name or IP address of the secondary DNS server.



Step 3—Specify WINS Servers

Cisco.com

Remote client

172.26.26.1

Internet

Outside

Inside

Server
10.0.0.15

Push
to client

VPN group
Pre-share
DNS server
WINS server
DNS domain
Address pool
Idle time

pixfirewall(config)#

```
vpngroup group_name wins-server wins_ip_prim [wins_ip_sec]
```

```
pix1(config)# vpngroup rmt_user_1 wins-server
  10.0.0.15
```

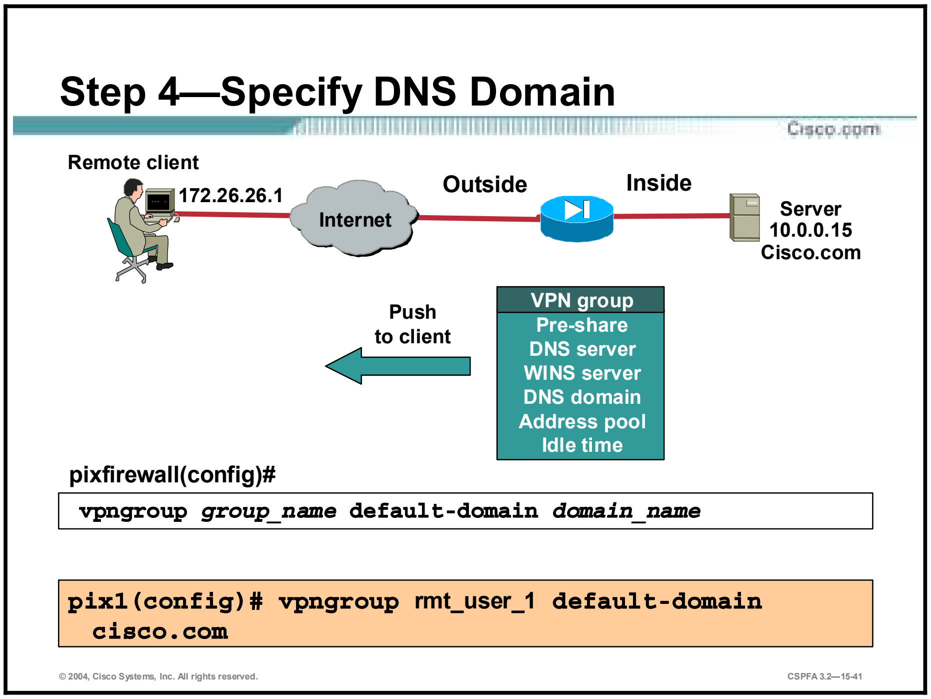
CSPFA 3.2—15-40

Step 3 (Optional.) Specify the primary and secondary WINS servers using the **wins** command in ISAKMP group configuration mode.

The syntax for the **wins** command is as follows:

wins *primary-server secondary-server*

primary-server	Name or IP address of the primary WINS server.
secondary-server	Name or IP address of the secondary WINS server.



Step 4—Specify DNS Domain

VPN group
Pre-share
DNS server
WINS server
DNS domain
Address pool
Idle time

pixfirewall(config)#

```
vpngroup group_name default-domain domain_name
```

```
pix1(config)# vpngroup rmt_user_1 default-domain
  cisco.com
```

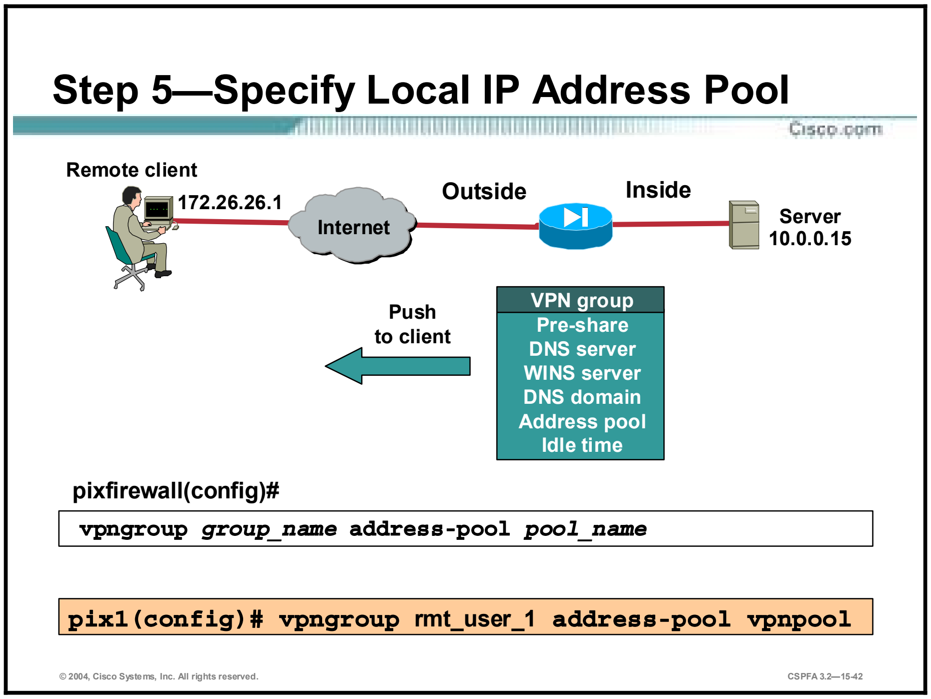
CSPFA 3.2—15-41

Step 4 (Optional.) Specify the DNS domain to which a group belongs by using the **domain** command in ISAKMP group configuration mode.

The syntax for the **domain** command is as follows:

domain ***name***

name	Name of the DNS domain.



Step 5—Specify Local IP Address Pool

Cisco.com

Remote client

172.26.26.1

Internet

Outside

Inside

Server
10.0.0.15

Push
to client

VPN group
Pre-share
DNS server
WINS server
DNS domain
Address pool
Idle time

pixfirewall(config)#

```
vpngroup group_name address-pool pool_name
```

```
pix1(config)# vpngroup rmt_user_1 address-pool vpnpool
```

CSPFA 3.2—15-42

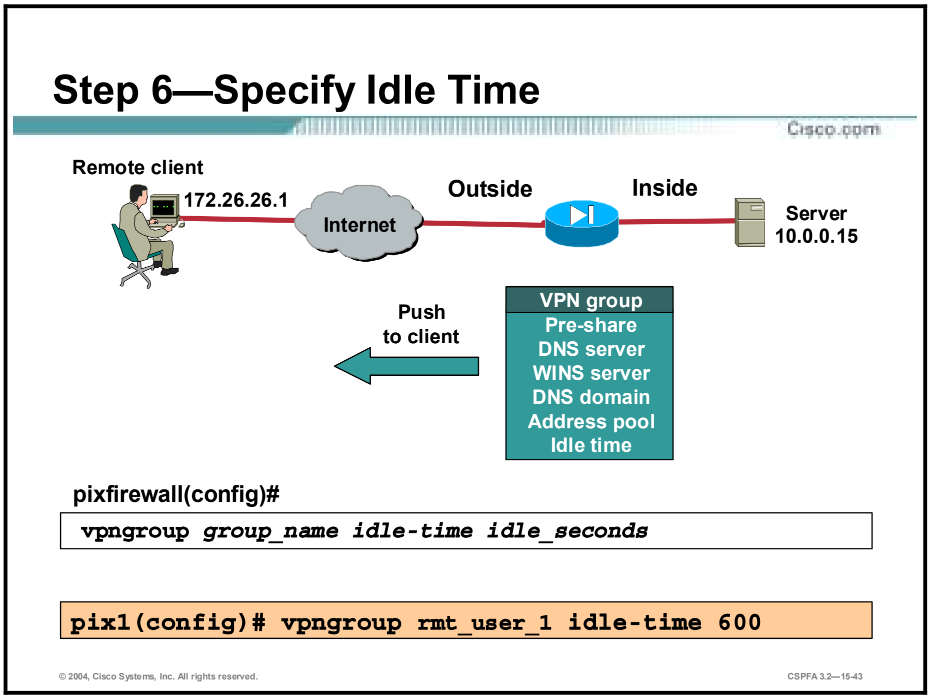
Step 5 Use the address-**pool** command to refer to an IP local pool address, which defines a range of addresses that will be used to allocate an internal IP address to a VPN client.

Use the address-**pool** command in the ISAKMP group configuration mode to define a local pool address.

The syntax for the address-**pool** command is as follows:

pool ***name***

name	Name of the local pool.



Step 6—Specify Idle Time

pixfirewall(config)#

```
vpngroup group_name idle-time idle_seconds
```

```
pix1(config)# vpngroup rmt_user_1 idle-time 600
```

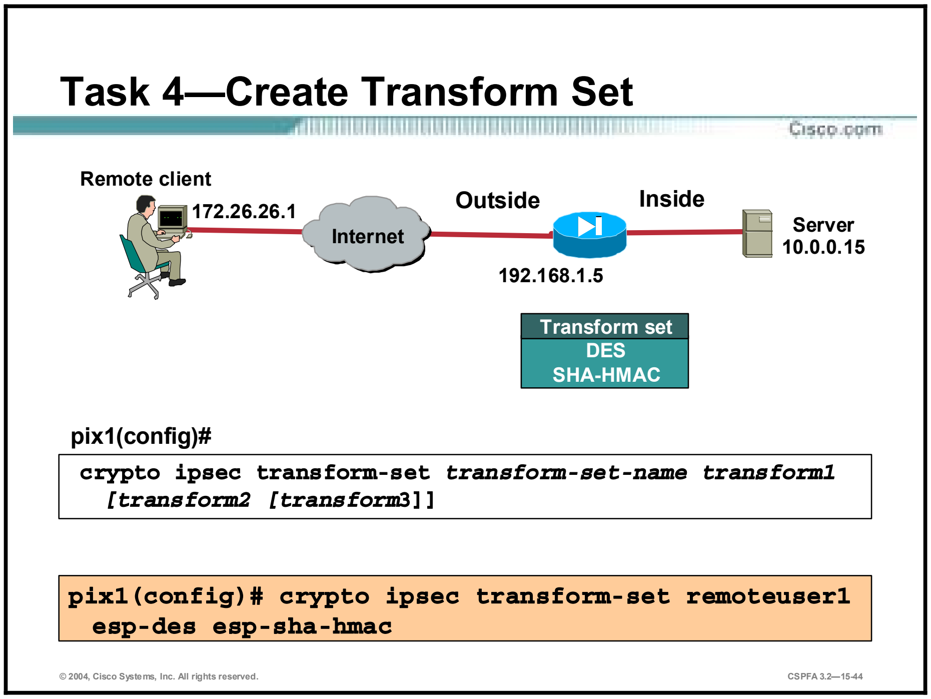
CSPFA 3.2—15-43

Step 6 Use the **idle-time** command to set the inactivity timeout for a Cisco VPN Client. When the inactivity timeout for a given VPN client or Easy VPN Remote device expires, the tunnel is terminated. The default inactivity timeout is 30 minutes.

The syntax for the **idle-time** command is as follows:

vpngroup *group_name* **idle-time** *idle_seconds*

Idle-seconds	The idle timeout in seconds, from 60 to 86400. The default is 1800 seconds (30 minutes).



Task 4—Create Transform Set

pix1(config)#

```
crypto ipsec transform-set transform-set-name transform1
  [transform2 [transform3]]
```

```
pix1(config)# crypto ipsec transform-set remoteuser1
  esp-des esp-sha-hmac
```

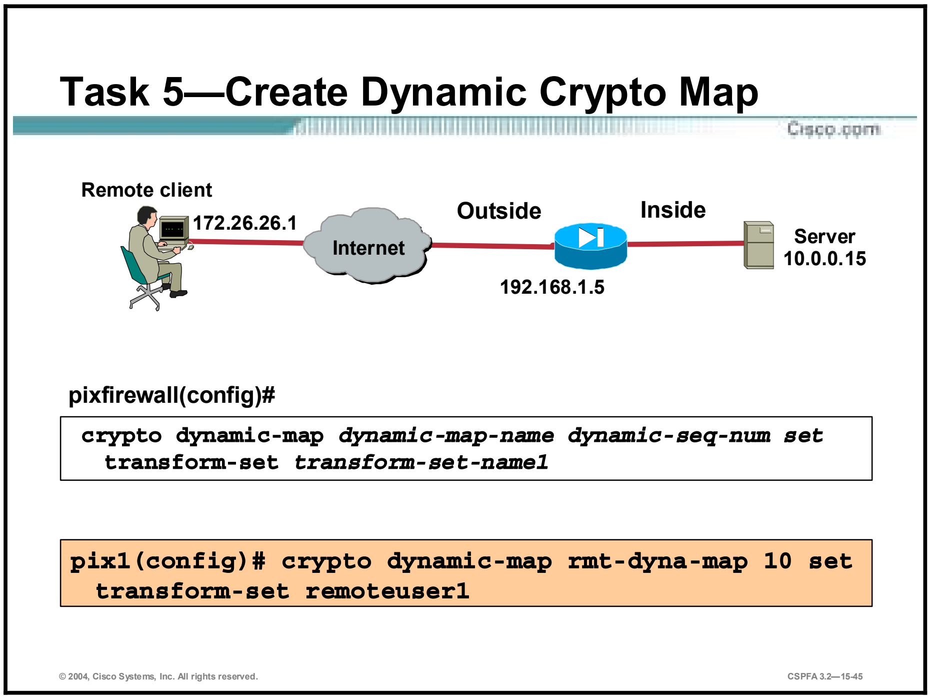
CSPFA 3.2—15-44

Specify which transform sets are allowed for the crypto map entry using the **crypto ipsec transform-set** command. When using this command, be sure to list multiple transform sets in order of priority (highest priority first). Note that this is the only configuration statement required in dynamic crypto map entries.

The syntax for the **crypto ipsec transform-set** command is as follows:

Crypto ipsec transform-set ***transform-set-name*** [***transform-set-name2…transform-set-name6***]

transform-set-name	Name of the transform set: ■ For an IPSec-manual crypto map entry, you can specify only one transform set. ■ For an IPSec-ISAKMP or dynamic crypto map entry, you can specify up to six transform sets.



Task 5—Create Dynamic Crypto Map

pixfirewall(config)#

```
crypto dynamic-map dynamic-map-name dynamic-seq-num set
  transform-set transform-set-name1
```

```
pix1(config)# crypto dynamic-map rmt-dyna-map 10 set
  transform-set remoteuser1
```

CSPFA 3.2—15-45

Create a dynamic crypto map entry and enter the crypto map configuration mode using the **crypto dynamic-map** command.

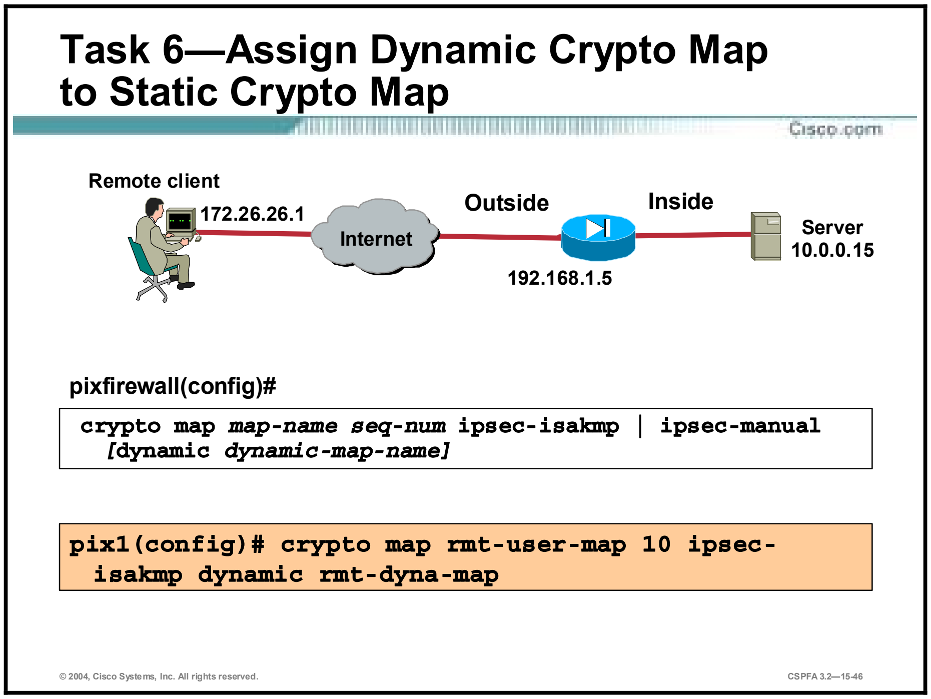
The syntax for the **crypto dynamic-map** command is as follows:

crypto dynamic-map ***dynamic-map-name dynamic-seq-num***

dynamic-map-name	Specifies the name of the dynamic crypto map set.
dynamic-seq-num	Specifies the number of the dynamic crypto map entry.

A dynamic crypto map entry is essentially a crypto map entry without all the parameters configured. It acts as a policy template where the missing parameters are later dynamically configured (as the result of an IPSec negotiation) to match a remote peer's requirements. This allows remote peers to exchange IPSec traffic with the router even if the router does not have a crypto map entry specifically configured to meet all of the remote peer's requirements.

Dynamic crypto maps are not used by the router to initiate new IPSec security associations with remote peers. Dynamic crypto maps are used when a remote peer tries to initiate an IPSec security association with the router. Dynamic crypto maps are also used in evaluating traffic.



Task 6—Assign Dynamic Crypto Map to Static Crypto Map

pixfirewall(config)#

```
crypto map map-name seq-num ipsec-isakmp | ipsec-manual
  [dynamic dynamic-map-name]
```

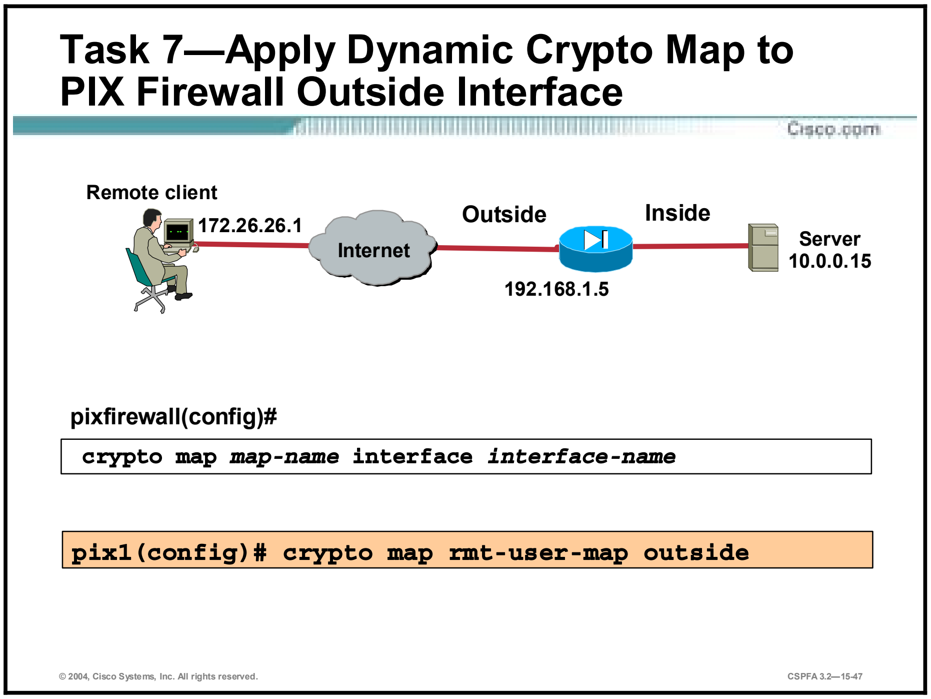
```
pix1(config)# crypto map rmt-user-map 10 ipsec-
  isakmp dynamic rmt-dyna-map
```

CSPFA 3.2—15-46

Add the dynamic crypto map to a static crypto map. The syntax for the command is as follows:

crypto map *map-name seq-num* **ipsec-isakmp** | **ipsec-manual** [**dynamic** *dynamic-map-name*]

dynamic-map-name	Specifies the name of the dynamic crypto map set.
map *map-name*	The name of the crypto map set.
seq-num	Specifies the number of the crypto map entry.



Task 7—Apply Dynamic Crypto Map to PIX Firewall Outside Interface

pixfirewall(config)#

```
crypto map map-name interface interface-name
```

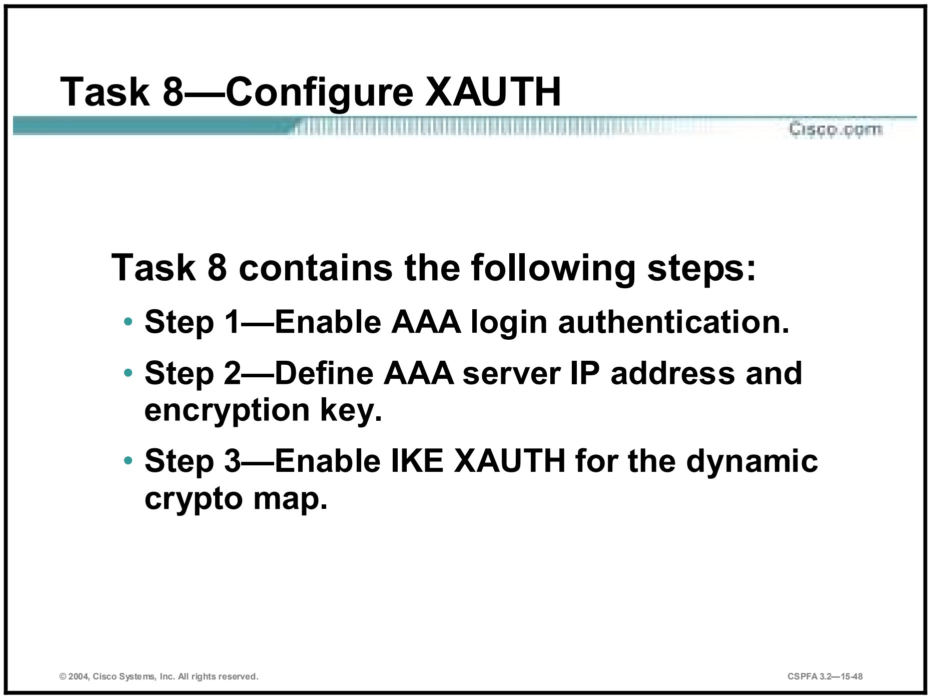
```
pix1(config)# crypto map rmt-user-map outside
```

CSPFA 3.2—15-47

Apply the crypto map to the PIX Firewall outside interface. The syntax for the command is as follows:

crypto map *map-name* **interface** *interface-name*

map *map-name*	The name of the crypto map set.
interface *interface-name*	Specify the identifying interface to be used by the PIX Firewall to identify itself to peers.



Task 8—Configure XAUTH

Task 8 contains the following steps:

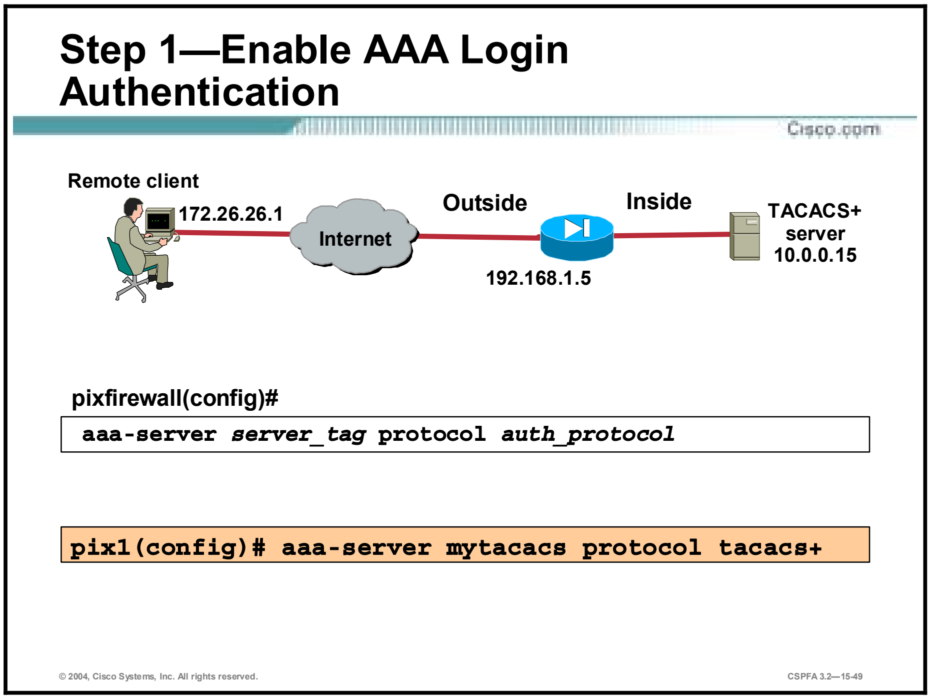
- Step 1—Enable AAA login authentication.
- Step 2—Define AAA server IP address and encryption key.
- Step 3—Enable IKE XAUTH for the dynamic crypto map.

Complete the following steps to configure XAUTH on your Easy VPN Server router:

Step 1 Enable AAA login authentication.

Step 2 Define AAA server IP address and encryption key.

Step 3 Enable IKE XAUTH for the crypto map.



Step 1—Enable AAA Login Authentication

Cisco.com

pixfirewall(config)#

```
aaa-server server_tag protocol auth_protocol
```

```
pix1(config)# aaa-server mytacacs protocol tacacs+
```

© 2004, Cisco Systems, Inc. All rights reserved.

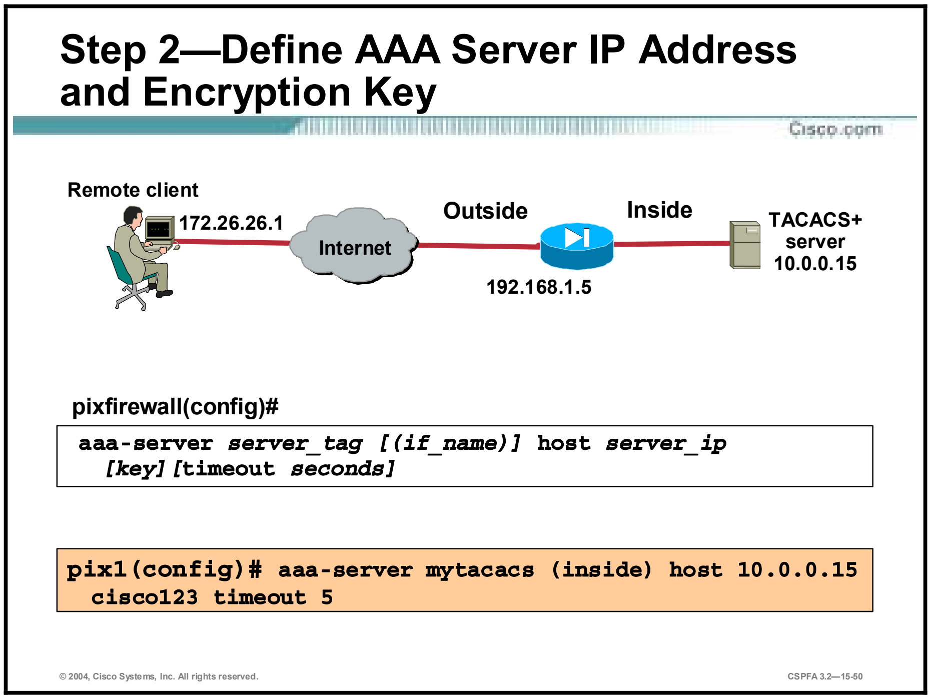
CSPFA 3.2—15-49

Step 1 Enable AAA login authentication using the **aaa server** command.

The syntax for the **aaa server** command is as follows:

aaa-server *server_tag* **protocol** *auth_protocol*

aaa-server ***server_tag***	An alphanumeric string which is the name of the server group. Use the *server_tag* in the **aaa** command to associate **aaa authentication** and **aaa accounting** command statements to a AAA server. Up to 14 server groups are permitted. However, **LOCAL** cannot used with **aaa-server** command because **LOCAL** is predefined by the PIX Firewall.
protocol ***auth_protocol***	The type of AAA server, either TACACS+ or RADIUS.



Step 2—Define AAA Server IP Address and Encryption Key

pixfirewall(config)#

```
aaa-server server_tag [(if_name)] host server_ip
  [key] [timeout seconds]
```

```
pix1(config)# aaa-server mytacacs (inside) host 10.0.0.15
  cisco123 timeout 5
```

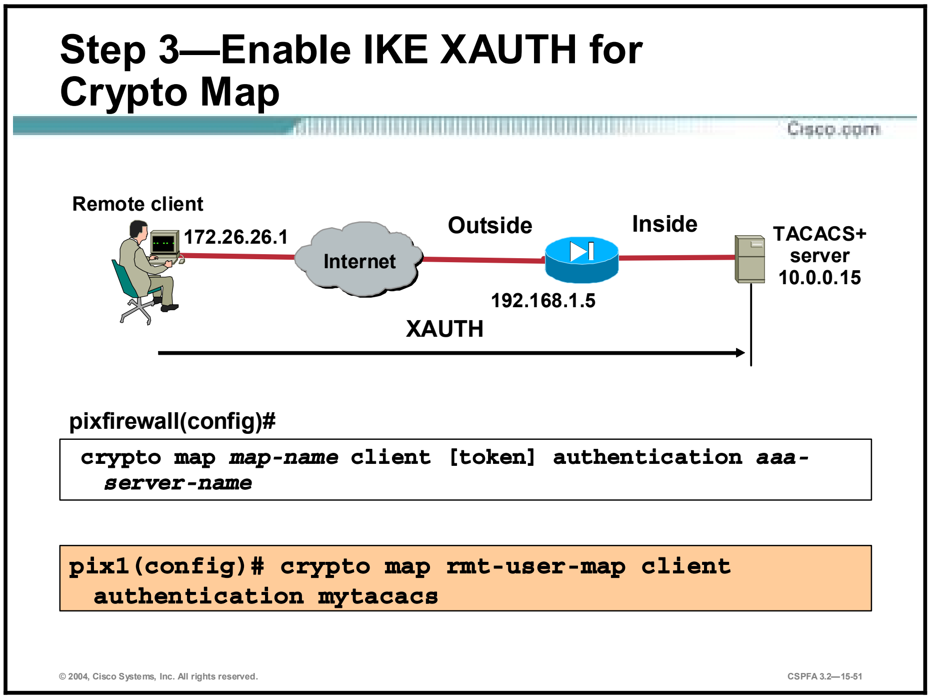
CSPFA 3.2—15-50

Step 2 Set the IP address of the AAA server and the encryption key using the **aaa-server** command.

The syntax for the **aaa-server** command is as follows:

aaa-server *server_tag* [(*if_name*)] **host** *server_ip [key*][**timeout** *seconds*]

host *server_ip*	The IP address of the TACACS+ or RADIUS server.
key	A case-sensitive, alphanumeric keyword of up to 127 characters that is the same value as the key on the TACACS+ server. Any characters entered past 127 are ignored. The key is used between the client and server for encrypting data between them. The key must be the same on both the client and server systems. Spaces are not permitted in the key, but other special characters are.
timeout *seconds*	The timeout interval for the request. This is the time after which the PIX Firewall gives up on the request to the primary AAA server. If there is a standby AAA server, the PIX Firewall will send the request to the backup server. The retransmit timeout is currently set to 10 seconds and is not user configurable.



Step 3—Enable IKE XAUTH for Crypto Map

pixfirewall(config)#

```
crypto map map-name client [token] authentication aaa-
  server-name
```

```
pix1(config)# crypto map rmt-user-map client
  authentication mytacacs
```

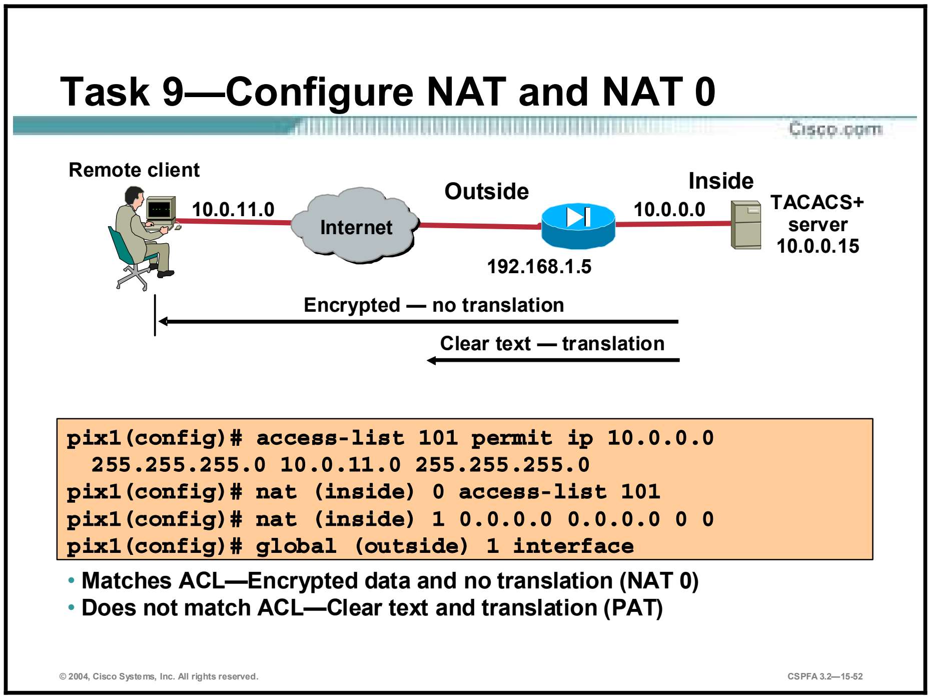
CSPFA 3.2—15-51

Step 3 Enable IKE XAUTH for the crypto map using the **crypto map** command.

The syntax for the **crypto map** command is as follows:

crypto map *map-name* **client** [token] **authentication** *aaa-server-name*

aaa-server- name	The name of the AAA server that will authenticate the user during IKE authentication. The AAA server options available are TACACS+, RADIUS, or LOCAL.
map-name	Name you assign to the crypto map set.



Task 9—Configure NAT and NAT 0

```
pix1(config)# access-list 101 permit ip 10.0.0.0
  255.255.255.0 10.0.11.0 255.255.255.0
pix1(config)# nat (inside) 0 access-list 101
pix1(config)# nat (inside) 1 0.0.0.0 0.0.0.0 0 0
pix1(config)# global (outside) 1 interface
```

- Matches ACL—Encrypted data and no translation (NAT 0)
- Does not match ACL—Clear text and translation (PAT)

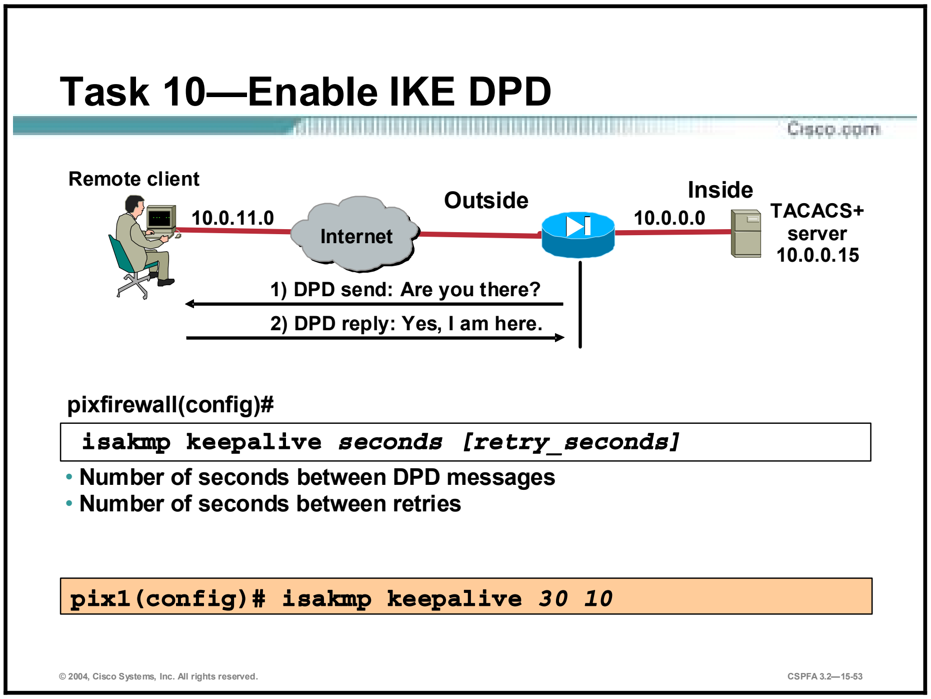
CSPFA 3.2—15-52

The last task is to define which traffic is encrypted and sent down the IPSec tunnel, and which traffic is translated and transmitted in clear text. The **NAT 0 access-list** command defines which traffic is encrypted but not translated. In the figure, traffic sourced from network 10.0.0.0/24 and destined for a host on 10.0.11.0/24 network is encrypted. The remaining traffic is translated, using NAT, to the IP address of the outside interface then transmitted in clear text.

The syntax for the **NAT 0 access-list** command is as follows:

nat [(*if_name*)] **0 access-list** *acl_id*

access-list	Associates **access-list** command statements with the **nat 0** command and exempts traffic that matches the ACL from NAT processing.
acl_id	The ACL name.
if_name	The name of the network interface, as specified by the **nameif** command, through which the hosts or network designated by *local_ip* are accessed.



Task 10—Enable IKE DPD

pixfirewall(config)#

```
isakmp keepalive seconds [retry_seconds]
```

- Number of seconds between DPD messages
- Number of seconds between retries

```
pix1(config)# isakmp keepalive 30 10
```

CSPFA 3.2—15-53

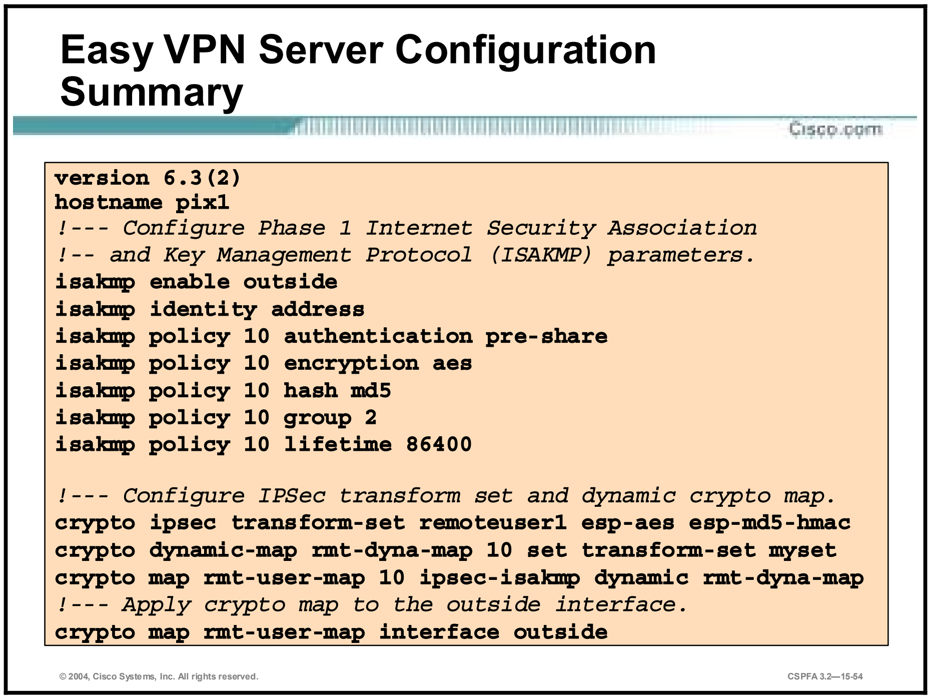
Dead peer detection (DPD) allows two IPSec peers to determine if the other is still “alive” during the lifetime of a VPN connection. DPD is useful because a host may reboot or the dialup link of a remote user may disconnect without notifying the peer that the VPN connection is gone away. When the IPSec host determines that a VPN connection no longer exists, it can notify the user, attempt to switch to another IPSec host, or clean up valuable resources that were allocated for the peer that no longer exists.

A DPD peer can send DPD messages, reply to DPD messages, or both. DPD messages are unidirectional and are automatically sent by Cisco VPN clients. Unlike the old-style IKE keepalives, DPD is not required on both peers. You can configure DPD on just the remote, just the headend, or both depending on the requirements. Use the **isakmp keepalive** command to enable a PIX Firewall gateway to send IKE DPD messages. You can specify the number of seconds between DPD messages, 30 seconds for example. You can also specify the number of seconds between retries if a DPD message fails, 10 seconds for example.

The syntax of the command is as follows:

isakmp keepalive *seconds* [*retry_seconds*]

isakmp keepalive *seconds*	The keepalive interval can be between 10 and 3600 seconds. The retry interval can be between 2 and 10 seconds, with the default being 2 seconds. The retry interval is the interval between retries after a keepalive response has not been received. You can specify the keepalive interval without specifying the retry interval, but cannot specify the retry interval without specifying the keepalive interval.
retry_seconds	Specifies the time interval before a keepalive message is sent if a keepalive response is not received from the previous request.

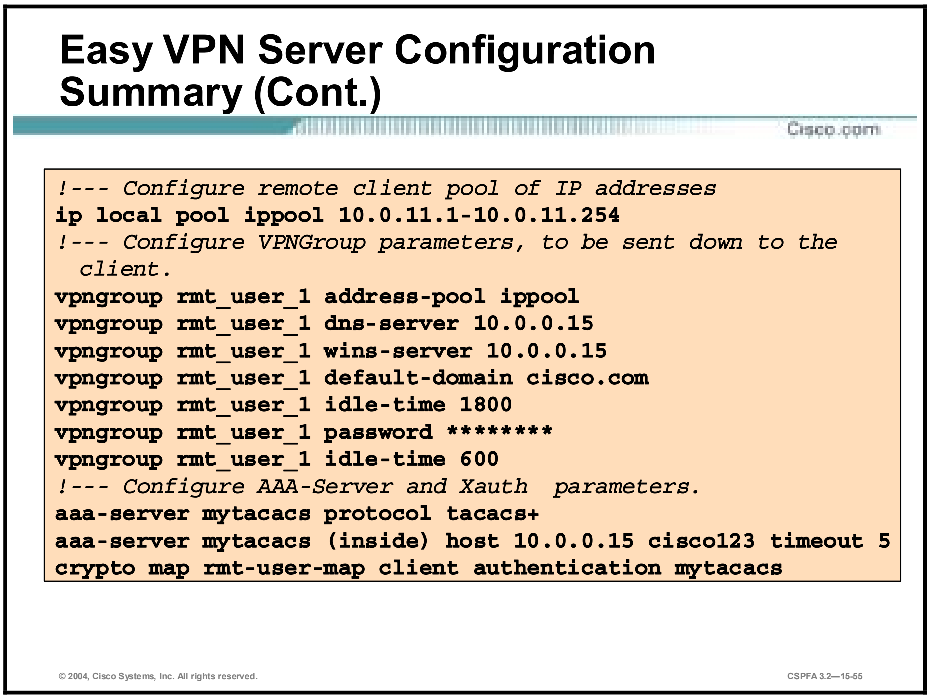


Easy VPN Server Configuration Summary

```
version 6.3(2)
hostname pix1
!--- Configure Phase 1 Internet Security Association
!-- and Key Management Protocol (ISAKMP) parameters.
isakmp enable outside
isakmp identity address
isakmp policy 10 authentication pre-share
isakmp policy 10 encryption aes
isakmp policy 10 hash md5
isakmp policy 10 group 2
isakmp policy 10 lifetime 86400

!--- Configure IPSec transform set and dynamic crypto map.
crypto ipsec transform-set remoteuser1 esp-aes esp-md5-hmac
crypto dynamic-map rmt-dyna-map 10 set transform-set myset
crypto map rmt-user-map 10 ipsec-isakmp dynamic rmt-dyna-map
!--- Apply crypto map to the outside interface.
crypto map rmt-user-map interface outside
```

CSPFA 3.2—15-54

The next few examples display the PIX Firewall Easy VPN Server configuration. The example in the figure here highlights the isakmp and crypto map parameter portion of the configuration.

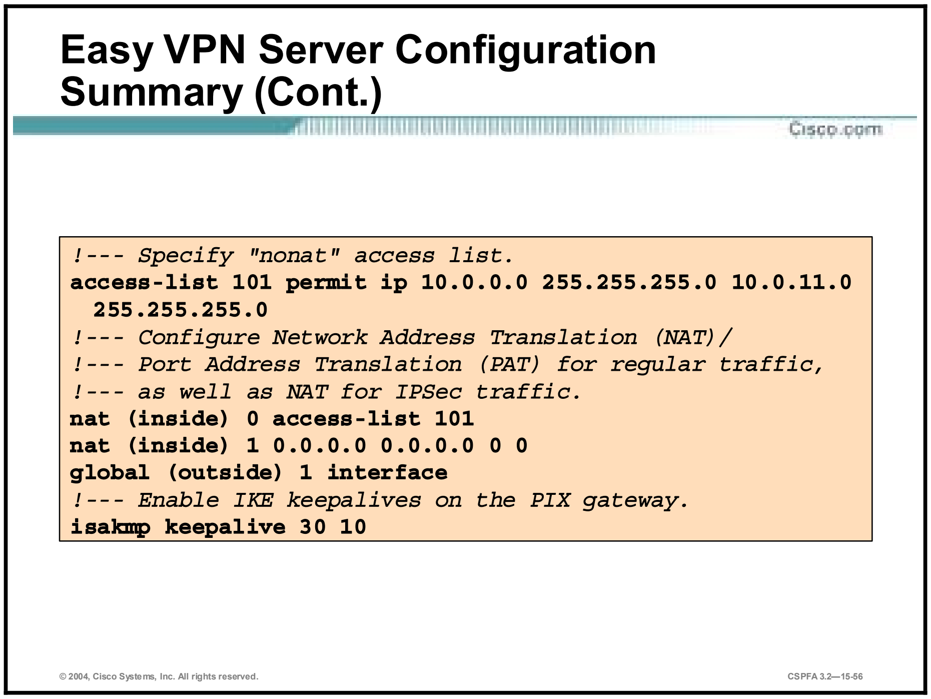


Easy VPN Server Configuration Summary (Cont.)

```
!--- Configure remote client pool of IP addresses
ip local pool ippool 10.0.11.1-10.0.11.254
!--- Configure VPNGroup parameters, to be sent down to the
  client.
vpngroup rmt_user_1 address-pool ippool
vpngroup rmt_user_1 dns-server 10.0.0.15
vpngroup rmt_user_1 wins-server 10.0.0.15
vpngroup rmt_user_1 default-domain cisco.com
vpngroup rmt_user_1 idle-time 1800
vpngroup rmt_user_1 password ********
vpngroup rmt_user_1 idle-time 600
!--- Configure AAA-Server and Xauth  parameters.
aaa-server mytacacs protocol tacacs+
aaa-server mytacacs (inside) host 10.0.0.15 cisco123 timeout 5
crypto map rmt-user-map client authentication mytacacs
```

CSPFA 3.2—15-55

This figure highlights the vpngroup and aaa-server parameter portion of the configuration.



Easy VPN Server Configuration Summary (Cont.)

```
!--- Specify "nonat" access list.
access-list 101 permit ip 10.0.0.0 255.255.255.0 10.0.11.0
  255.255.255.0
!--- Configure Network Address Translation (NAT)/
!--- Port Address Translation (PAT) for regular traffic,
!--- as well as NAT for IPSec traffic.
nat (inside) 0 access-list 101
nat (inside) 1 0.0.0.0 0.0.0.0 0 0
global (outside) 1 interface
!--- Enable IKE keepalives on the PIX gateway.
isakmp keepalive 30 10
```

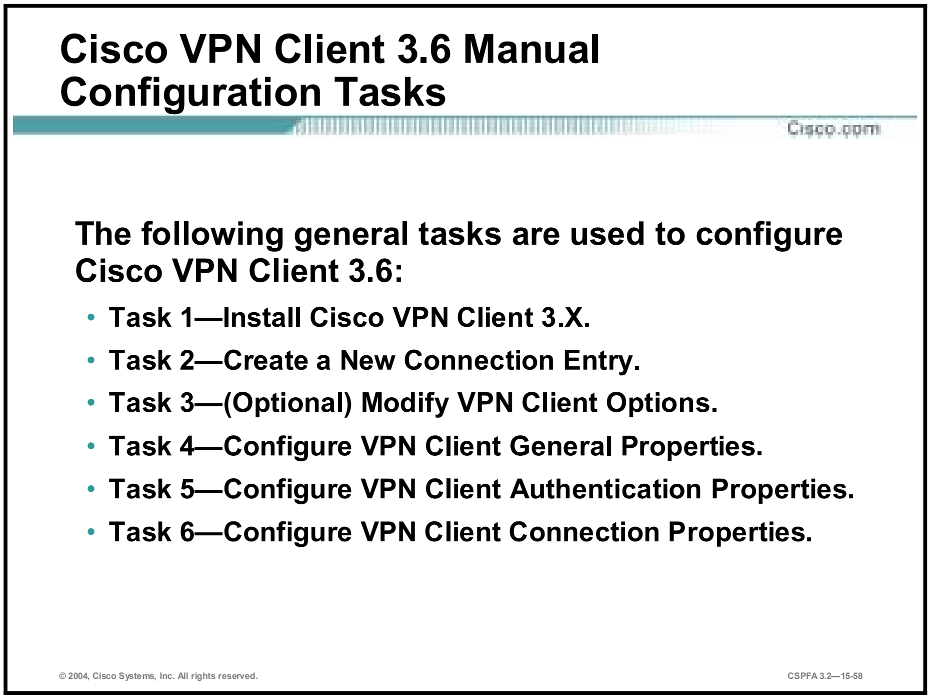
CSPFA 3.2—15-56

This figure highlights the NAT 0, NAT, and global parameter portion of the configuration. Any traffic between 10.0.0.0/24 and 10.0.11.0/24 will not be translated by the PIX Firewall. Any other traffic will be translated using the outside interface address of the PIX Firewall.

The PIX Firewall will also send DPD messages to the Cisco VPN Client every 30 seconds. If the client fails to respond, the PIX Firewall will resend the message after waiting 10 seconds.

Cisco VPN Client 3.6 Manual Configuration Tasks

This topic contains information regarding the installation and configuration of the Cisco VPN Client release 3.6.



Cisco VPN Client 3.6 Manual Configuration Tasks

The following general tasks are used to configure Cisco VPN Client 3.6:

- **Task 1—Install Cisco VPN Client 3.X.**
- **Task 2—Create a New Connection Entry.**
- **Task 3—(Optional) Modify VPN Client Options.**
- **Task 4—Configure VPN Client General Properties.**
- **Task 5—Configure VPN Client Authentication Properties.**
- **Task 6—Configure VPN Client Connection Properties.**

CSPFA 3.2—15-58

Use the following steps to install and configure the Cisco VPN Client:

Step 1 Install Cisco VPN Client 3.X.

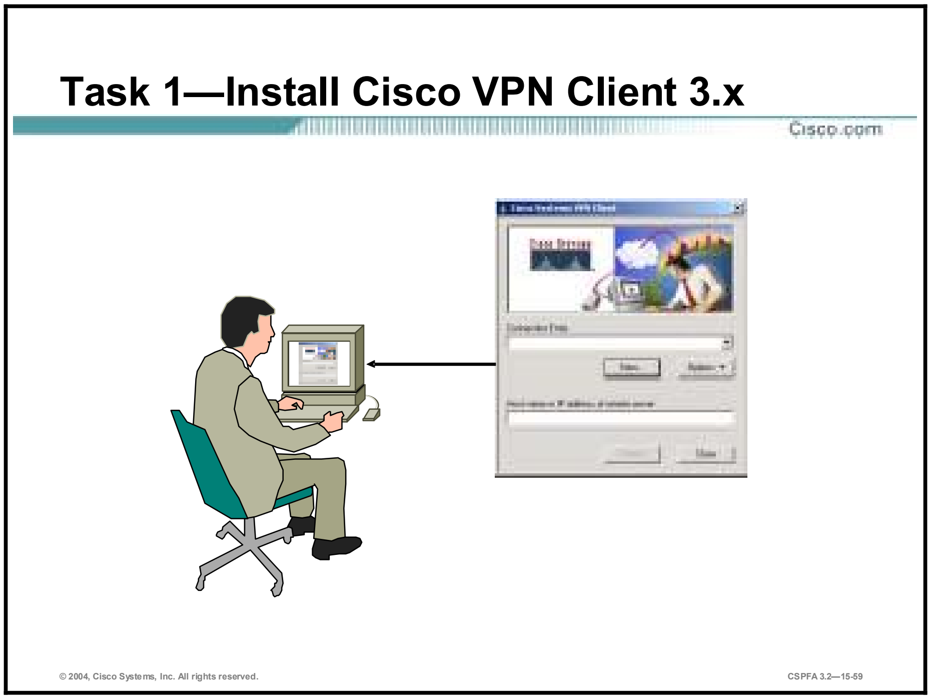
Step 2 Create a new connection entry.

Step 3 (Optional.) Modify VPN client options.

Step 4 Configure VPN client general properties.

Step 5 Configure VPN client authentication properties.

Step 6 Configure VPN client connection properties.



Task 1—Install Cisco VPN Client 3.x

Installation of the Cisco VPN Client varies slightly based on the type of operating system. Always review the installation instructions that come with the Cisco VPN Client before attempting any installation. Generally, installation of the Cisco VPN Client 3.6 involves the following steps (this example is based on installing the Cisco VPN Client on a Windows 2000 PC):

Step 1 Obtain a copy of the Cisco VPN Client application. The application can be downloaded from the Cisco website or by obtaining a Cisco VPN 3000 Series Concentrator CD-ROM.

Step 2 Locate and run the Cisco VPN Client **setup.exe** executable file. If this is the first time the Cisco VPN Client is installed, a window opens and displays the following message: “Do you want the installer to disable the IPSec Policy Agent?”

Step 3 Click **Yes** to disable the IPSec policy agent. The Welcome window opens.

Step 4 Read the Welcome window and click **Next**. The License Agreement window opens.

Step 5 Read the license agreement and click **Yes**. The Choose Destination Location window opens.

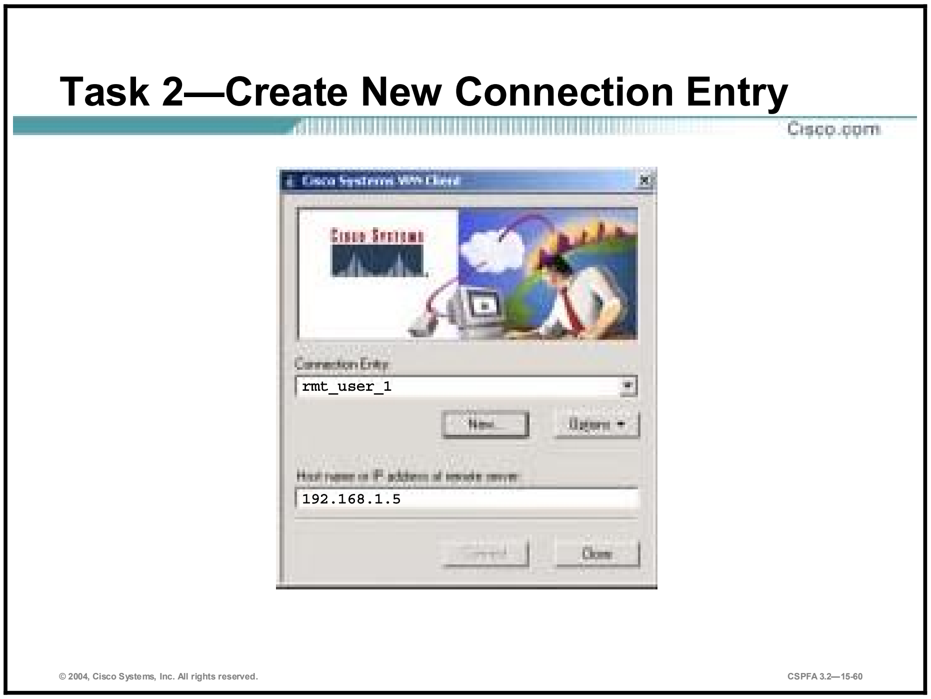
Step 6 Click **Next**. The Select Program Folder window opens.

Step 7 Accept the defaults by clicking **Next**. The Start Copying Files window opens.

Step 8 The files are copied to the hard disk drive of the PC and the InstallShield Wizard Complete window opens.

Step 9 Select **Yes, I want to restart my computer now** and click **Finish**. The PC restarts.

This completes the installation of the Cisco VPN Client.



Task 2—Create New Connection Entry

Connection Entry: rmt_user_1

Host name or IP address of remote server: 192.168.1.5

CSPFA 3.2—15-60

The Cisco VPN Client enables users to configure multiple connection entries. Multiple connection entries enable the user to build a list of possible network connection points. For example, a corporate telecommuter may want to connect to the sales office in Boston for sales data (the first connection entry), and then they may want to connect to the Austin factory for inventory data (a second connection entry). Each connection contains a specific entry name and remote server hostname or IP address.

Generally, creating a new connection entry involves the following steps (this example is based on creating new connection entries on a Windows 2000 PC):

Step 1 Choose **Start>Programs>Cisco Systems VPN Client>VPN Dialer**. The Cisco Systems VPN Client window opens.

Step 2 Click **New**. The New Connection Entry wizard opens.

Step 3 Enter a name for the new connection entry in the Name of the New Connection Entry field (for example, Boston Sales).

Step 4 Click **Next**.

Step 5 Enter the public interface IP address or hostname of the remote Easy VPN Server in the Remote Server field.

Step 6 Click **Next**.

Step 7 Select **Group Access Information** and complete the following sub-steps. The following entries are always case sensitive:

1. Enter a group name that matches a group on the Easy VPN Server.
2. Enter the group password.
3. Confirm the password.

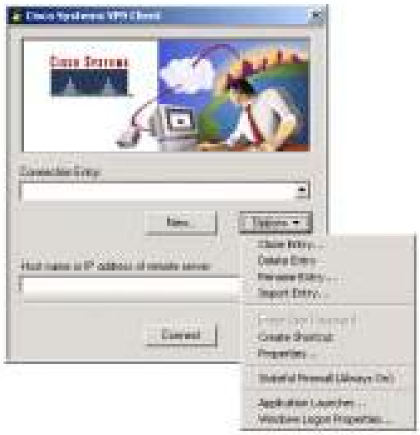
Step 8 Click **Next**.

Step 9 Click **Finish** and leave the Cisco Systems VPN Client window open.

You have successfully configured the network parameters for the Cisco VPN Client and created a new VPN connection entry.

Task 3—(Optional.) Modify Cisco VPN Client Options

Cisco.com



CSPFA 3.2—15-61

Several configuration options are available from the Options menu. This figure shows how to select the Options menu from the splash window.

The Options drop-down menu enables you to configure or change optional parameters. By clicking the **Options** button, the following options become available:

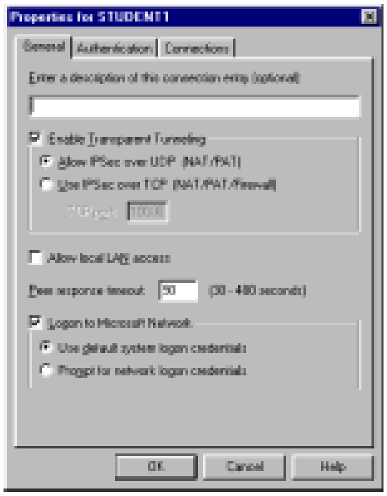
- Clone Entry—Enables you to copy a connection entry with all of its properties.
- Delete Entry—Enables you to delete a connection entry.
- Rename Entry—Enables you to rename a connection entry (not case-sensitive).
- Import Entry—Provides a file that will load the Cisco VPN Client parameters.
- Erase User Password—Eliminates a saved password. Erase User Password is available only when you have enabled Allow Password Storage under the mode configuration parameters for this group.
- Create Shortcut—Enables you to create a shortcut for your desktop.
- Properties—Enables you to configure or change the properties of the connection.
- Stateful Firewall (Always On)—Blocks all inbound traffic to the Cisco VPN Client that is not related to an outbound session. After the remote user enables the stateful firewall, it is always on.
- Application Launcher—Enables you to launch an application before establishing a connection. This is used in conjunction with Windows Login Properties. Windows Login Properties enables the Cisco VPN Client to make the connection to the Concentrator before the user logs in.

If you want to know what version of the Cisco VPN Client you have installed on your PC, right-click the Cisco VPN Dialer icon in the system tray, to view the Cisco VPN Client version.

Task 4—Configure Cisco VPN Client General Properties

Cisco.com

Win 95/98/ME



Enter a description of this connection entry (optional)

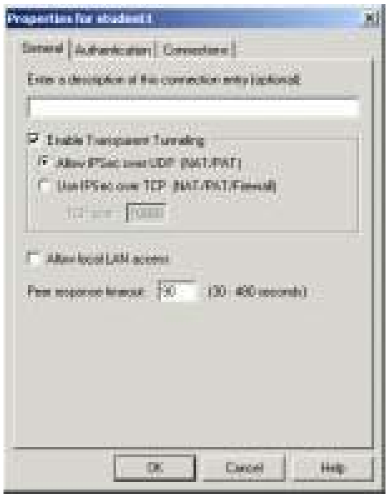
☑ Enable Transparent Tunneling
● Allow IPSec over UDP (NAT/PAT)
○ Use IPSec over TCP (NAT/PAT/Firewall)
TCP port: 10000

☐ Allow local LAN access

Peer response timeout: 90 (30 - 480 seconds)

☑ Logon to Microsoft Network
● Use default system logon credentials
○ Prompt for network logon credentials

Win-NT 4/2000/XP



Enter a description of this connection entry (optional)

☑ Enable Transparent Tunneling
● Allow IPSec over UDP (NAT/PAT)
○ Use IPSec over TCP (NAT/PAT/Firewall)
TCP port: 10000

☐ Allow local LAN access

Peer response timeout: 90 (30 - 480 seconds)

CSPFA 3.2—15-62

Choose **Properties** from the Options drop-down menu. The following three tabs are found under Properties:

- General tab—Enables IPSec through NAT, displays the status of the local LAN access feature, and selects Microsoft network logon options.
- Authentication tab—Configures the Cisco VPN Client's group or digital certificate information.
- Connections tab—Enables backup connections, and links the VPN connection to Dialup Networking phonebook entries.

The figure displays the General tab for both the Windows 95, 98, ME, and Windows NT, 2000, and XP operating systems.

There are two versions of General tab, depending on the operating system you are using: the Windows 95, 98, and Millennium Edition (ME) version or the Windows NT 4.0, 2000, and XP version. The Windows 95, 98, and ME version provides options for transparent tunneling, local LAN access, and Microsoft login options. The Windows NT 4.0, 2000, and XP version provides transparent tunneling and local LAN access only.

The functions of the General tab are as follows:

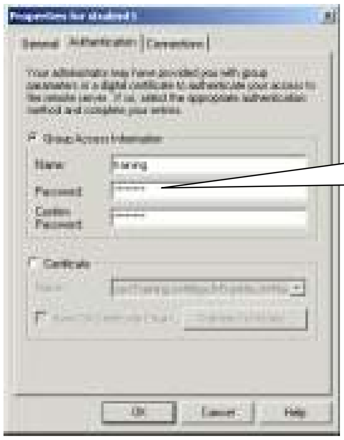
- Enable Transparent Tunneling check box—Works with Windows 95, 98, NT 4.0, 2000, and XP.
- Allow IPSec over UDP (NAT/PAT) radio button—Enables you to use the Cisco VPN Client to connect to the Easy VPN Server via UDP through a firewall or router that is running NAT. Both the Cisco VPN Client and Easy VPN Server must be enabled for this feature to work.
- Use IPSec over TCP (NAT/PAT/Firewall) radio button—Enables you to use the Cisco VPN Client to connect to the Easy VPN Server via TCP through a firewall or router that is

running NAT. Both the Cisco VPN Client and the Easy VPN Server must be enabled for this feature to work.

- Allow local LAN access check box—For security purposes, the user has the ability to disable local LAN access when using an insecure local LAN (for example, in a hotel). This option is not available when using an IOS-based Easy VPN Server or PIX Firewall as the VPN gateway.
- Peer response timeout field—The number of seconds to wait before the Cisco VPN Client decides that the peer is no longer active. The Cisco VPN Client continues to send DPD requests until it reaches the peer response timeout value.
- Logon to Microsoft Network check box—Works with Windows 95, 98, and ME only.
- Use default system logon credentials radio button—Use the logon username and password resident on your PC to log onto the Microsoft network (for example, student4).
- Prompt for network logon credentials radio button—If your logon username and password differ from the enterprise network, the enterprise network prompts you for the username and password.

Task 5—Configure Cisco VPN Client Authentication Properties

Cisco.com



The end user never sees this after the initial configuration

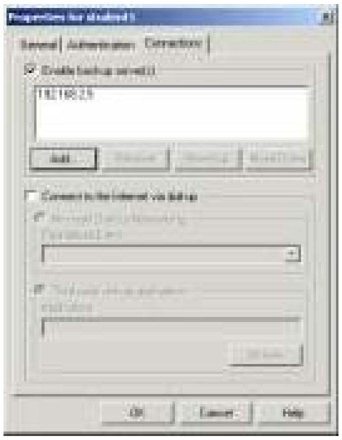
CSPFA 3.2—15-63

The Easy VPN Server and the Cisco VPN Client connection can be authenticated with either the group name and password, or digital certificates. The Authentication tab enables you to set your authentication information. You need to choose one method, group, or certificate via the radio buttons.

Within the Group Access information group box in the Authentication tab, enter the group name and password in the appropriate fields. The group name and password must match what is configured for this group within the Easy VPN Server. Entries are case-sensitive.

For certificates to be exchanged, the Certificate radio button must be selected. In the Name drop-down menu, any personal certificates loaded on your PC are listed. Choose the certificate to be exchanged with the Easy VPN Server during connection establishment. If no personal certificates are loaded in your PC, the drop-down menu is blank. Use the Validate Certificate button to check the validity of the Cisco VPN Clients' certificate.

Task 6—Configure Cisco VPN Client Connections Properties



☑ Enable backup server(s)

192.168.2.5

☐ Connect to the Internet via dial-up

CSPFA 3.2—15-64

The Connections tab defines backup networks and connection to the Internet via dialup networking.

An enterprise network may include one or more backup Easy VPN Servers to use if the primary Easy VPN Server gateway is not available. Select the **Enable backup server(s)** check box to enable this feature. Once selected, click **Add** to enter the IP address of the backup Easy VPN Server.

The Cisco VPN Client attempts to connect to the primary Easy VPN Server first. If that Easy VPN Server cannot be reached, the Cisco VPN Client accesses the backup list for the addresses of available backup Easy VPN Server.

Connecting to an enterprise network using a dialup connection is typically a two-step process:

Step 1 Use a dialup connection to your ISP.

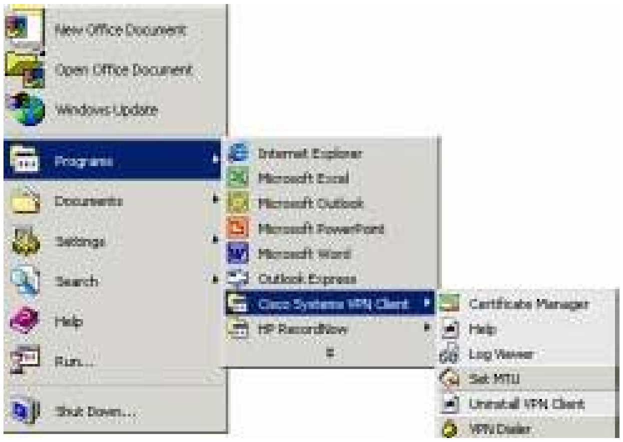
Step 2 Use the Cisco VPN Client to connect to the enterprise network.

Connecting to the Internet via dialup automatically launches the connection before making the VPN connection. It makes connecting to the ISP and Easy VPN Server an easy, one-step process. Select **Connect to the Internet via Dial-Up Networking** to enable the option.

Working with the Cisco VPN 3.6 Client

This topic contains information regarding the Cisco VPN Client program menus, log viewer, and status displays.

Cisco VPN Client Program Menu



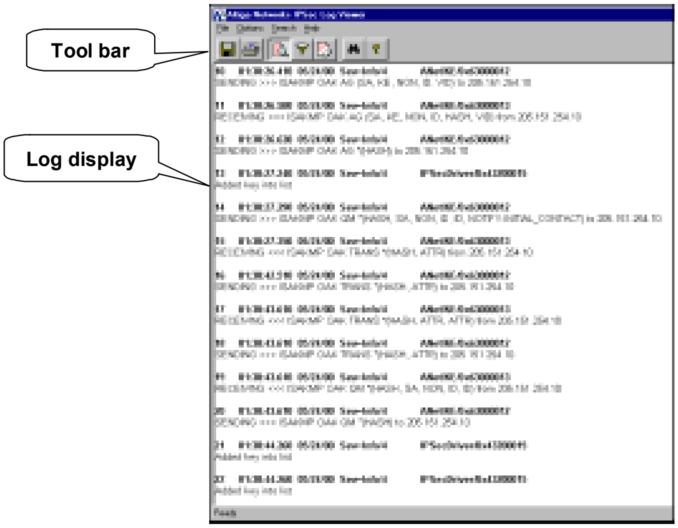
CSPFA 3.2—15-66

This figure displays the Cisco VPN Client program menu as viewed on a Windows 2000 PC.

After the Cisco VPN Client is installed, access the Cisco VPN Client program menu by choosing **Start>Programs>Cisco Systems VPN Client**. Under the Cisco VPN Client menu, a number of options are available:

- Certificate Manager—Enables you to enroll, import, export, verify, and view certificates.
- Help—Accesses the Cisco VPN Client help text. Help is also available by doing the following:
  - Press **F1** at any window while using the Cisco VPN Client.
  - Click the **Help** button.
  - Click on the logo in the title bar.
- Log Viewer—Displays the Cisco VPN Client event log.
- Set MTU—The Cisco VPN Client automatically sets the maximum transmission unit (MTU) size to approximately 1420 bytes. For unique applications, Set MTU can change the MTU size to fit a specific scenario.
- Uninstall VPN Client—Only one Cisco VPN Client can be loaded at a time. When upgrading, the old Cisco VPN Client must be uninstalled before the new Cisco VPN Client is installed. Choose **Uninstall VPN Client** to remove the old Cisco VPN Client.
- Cisco VPN Dialer—Initiates the Cisco VPN Client connection process by displaying the Cisco VPN Client splash window.

Cisco VPN Client Log Viewer



```
10   01:38:26.418 05/24/00 Sev=Info/4        IKE/0x63000013
SENDING >>> ISAKMP OAK AG (SA, KE, NON, ID, VID) to 205.151.254.10

11   01:38:26.388 05/24/00 Sev=Info/5        IKE/0x63000011
RECEIVING <<< ISAKMP OAK AG (SA, KE, NON, ID, HASH, VID) from 205.151.254.10

12   01:38:26.638 05/24/00 Sev=Info/4        IKE/0x63000013
SENDING >>> ISAKMP OAK AG *(HASH) to 205.151.254.10

13   01:38:27.248 05/24/00 Sev=Info/4        IPSecDriver/0x43000011
Added key into list

14   01:38:27.298 05/24/00 Sev=Info/4        IKE/0x63000013
SENDING >>> ISAKMP OAK QM *(HASH, SA, NON, ID, ID, NOTIFY:INITIAL_CONTACT) to 205.151.254.10

15   01:38:27.358 05/24/00 Sev=Info/5        IKE/0x63000011
RECEIVING <<< ISAKMP OAK TRANS *(HASH, ATTR) from 205.151.254.10

16   01:38:43.218 05/24/00 Sev=Info/4        IKE/0x63000013
SENDING >>> ISAKMP OAK TRANS *(HASH, ATTR) to 205.151.254.10

17   01:38:43.618 05/24/00 Sev=Info/5        IKE/0x63000011
RECEIVING <<< ISAKMP OAK TRANS *(HASH, ATTR, ATTR) from 205.151.254.10

18   01:38:43.618 05/24/00 Sev=Info/4        IKE/0x63000013
SENDING >>> ISAKMP OAK TRANS *(HASH, ATTR) to 205.151.254.10

19   01:38:43.618 05/24/00 Sev=Info/5        IKE/0x63000011
RECEIVING <<< ISAKMP OAK QM *(HASH, SA, NON, ID, ID) from 205.151.254.10

20   01:38:43.618 05/24/00 Sev=Info/4        IKE/0x63000013
SENDING >>> ISAKMP OAK QM *(HASH) to 205.151.254.10

21   01:38:44.268 05/24/00 Sev=Info/4        IPSecDriver/0x43000011
Added key into list

22   01:38:44.268 05/24/00 Sev=Info/4        IPSecDriver/0x43000011
Added key into list
Ready
```

CSPFA 3.2—15-67

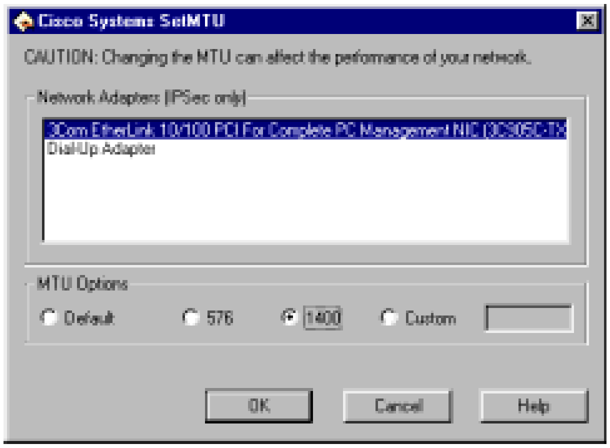
Examining the event log can help a network administrator diagnose problems with an IPSec connection between a Cisco VPN Client and an Easy VPN Server. The Log Viewer application collects event messages from all processes that contribute to the Cisco VPN Client-Easy VPN Server connection.

This figure displays the Log Viewer window with a sample Cisco VPN Client log file. From the toolbar, you can perform the following:

- Save the log file.
- Print the log file.
- Capture event messages to the log.
- Filter the events.
- Clear the event log.
- Search the event log.

Setting MTU Size

Cisco.com



Cisco Systems SetMTU

CAUTION: Changing the MTU can affect the performance of your network.

Network Adapters (IPSec only)

- 3Com EtherLink 10/100 PCI For Complete PC Management NIC (3C905C-TX
- Dial-Up Adapter

MTU Options

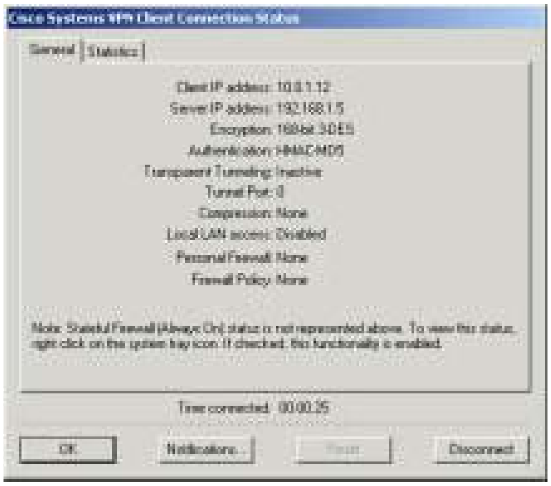
○ Default ○ 576 ● 1400 ○ Custom

CSPFA 3.2—15-68

This figure displays the SetMTU window, which is where you set the MTU size.

The Cisco VPN Client automatically sets the MTU size to approximately 1420 bytes. For unique applications where fragmentation is still an issue, SetMTU can change the MTU size to fit the specific scenario. In the Network Adapters (IPSec only) field, select the network adapter. In the example in the figure, 3Com EtherLink is selected. In the MTU Options group box, set the MTU option size by selecting the appropriate radio button. You must reboot for MTU changes to take effect.

Cisco VPN Client Connection Status—General Tab



Client IP address: 10.0.1.12
Server IP address: 192.168.1.5
Encryption: 168-bit 3-DES
Authentication: HMAC-MD5
Transparent Tunneling: Inactive
Tunnel Port: 0
Compression: None
Local LAN access: Disabled
Personal Firewall: None
Firewall Policy: None

Note: Stateful Firewall (Always On) status is not represented above. To view this status, right click on the system tray icon. If checked, this functionality is enabled.

Time connected: 00:00.25

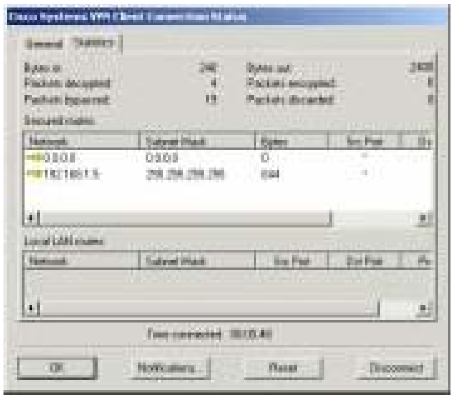
The Cisco Systems VPN Client Connection Status window contains up to three tabs: General, Statistics, and Firewall (Firewall is not shown in the figure as it is not currently supported by Cisco PIX Firewall Easy VPN). The figure displays the General tab.

The General tab provides IP security information, listing the IPSec parameters that govern this IPSec tunnel. The following information is available within the General tab:

- Client IP address—The IP address assigned to the Cisco VPN Client for the current session.
- Server IP address—The IP address of the Easy VPN Server device to which the Cisco VPN Client is connected.
- Encryption—The data encryption method for traffic through this tunnel.
- Authentication—The data or packet authentication method used for traffic through this tunnel.
- Transparent Tunneling—The status of transparent tunnel mode in the Cisco VPN Client (either active or inactive).
- Tunnel Port—If Transparent Mode is active; the tunnel port through which packets are passing is displayed. This field also identifies whether the Cisco VPN Client is sending packets through UDP or TCP. If Transparent Tunneling is inactive, then the value of Tunnel Port is zero.
- Compression—Displays whether data compression is in effect as well as the type of compression in use. Currently, the type of compression is LZS.
- Local LAN Access—Displays whether this parameter is enabled or disabled.
- Personal Firewall—Displays the name of the firewall in use on the Cisco VPN Client PC, such as the Cisco Integrated Client, Zone Labs ZoneAlarm, ZoneAlarm Pro, BlackICE Defender, and so on. This feature is not currently supported in IOS 12.2(8)T Cisco Easy VPN.

- Firewall Policy—Displays the firewall policy in use:
  - AYT (Are You There)—Not currently supported in Cisco PIX Firewall Easy VPN.
  - Central Protection Policy (CPP) or “policy pushed” as defined on the VPN gateway. Not currently supported in Cisco PIX Firewall Easy VPN.

Cisco VPN Client Connection Status—Statistics Tab



Bytes in: 240 Bytes out: 2400
Packets decrypted: 4 Packets encrypted: 8
Packets bypassed: 13 Packets discarded: 0

Secured routes:

Network	Subnet Mask	Bytes	Src Port
0.0.0.0	0.0.0.0	0	*
192.168.1.5	255.255.255.255	644	*

Local LAN routes:

Network	Subnet Mask	Src Port	Dst Port

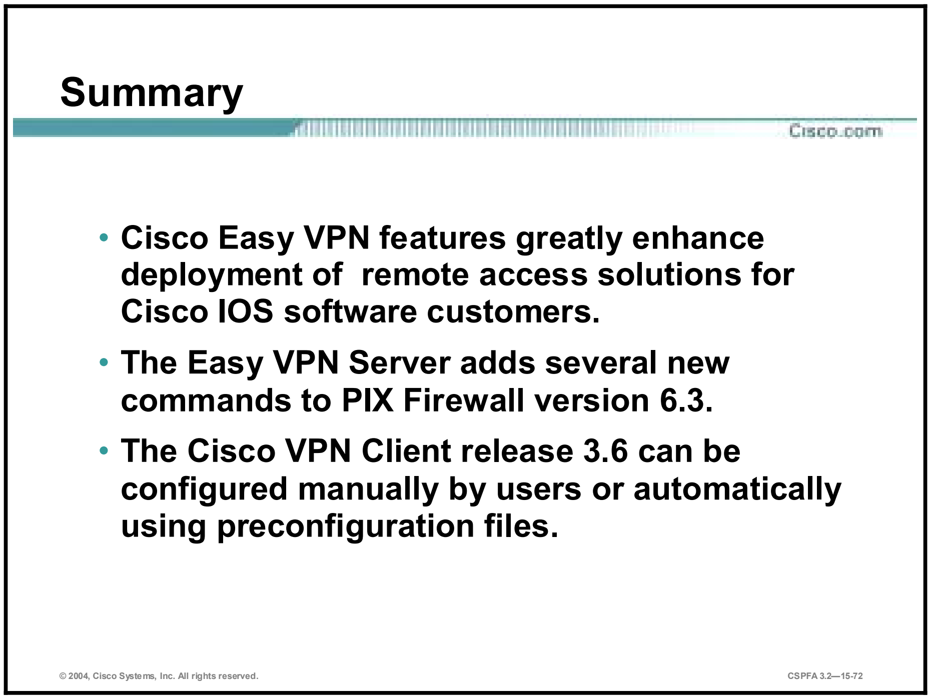
Time connected: 00:05:40

This figure displays the Statistics tab. The Statistics tab on the Connection Status window shows statistics for data packets that the Cisco VPN Client has processed during the current session, or since the statistics were reset. The following information is available in this window:

- Bytes in—The total amount of data received after a secure packet has been successfully decrypted.
- Bytes out—The total amount of encrypted data transmitted through the tunnel.
- Packets decrypted—The total number of data packets received on the port.
- Packets encrypted—The total number of secured data packets transmitted out of the port.
- Packets bypassed—The total number of data packets that the Cisco VPN Client did not process because they did not need to be encrypted. Local Address Resolution Protocols (ARPs) and Dynamic Host Configuration Protocols (DHCPs) fall into this category.
- Packets discarded—The total number of data packets that the Cisco VPN Client rejected because they did not come from the secure Easy VPN Server gateway.
- Secured Routes—The IP address of the private network with which this Cisco VPN Client has a secure connection.
- Local LAN Routes—If present, the Local LAN Routes box shows the network addresses of the networks you can access on your local LAN while you are connected to your organization's private network.

Summary

This topic summarizes what you have learned in this lesson.



Summary

- Cisco Easy VPN features greatly enhance deployment of remote access solutions for Cisco IOS software customers.
- The Easy VPN Server adds several new commands to PIX Firewall version 6.3.
- The Cisco VPN Client release 3.6 can be configured manually by users or automatically using preconfiguration files.

CSPFA 3.2—15-72

Lab Exercise—Configure a Secure VPN Using IPSec Between a PIX Firewall and a Cisco VPN Client

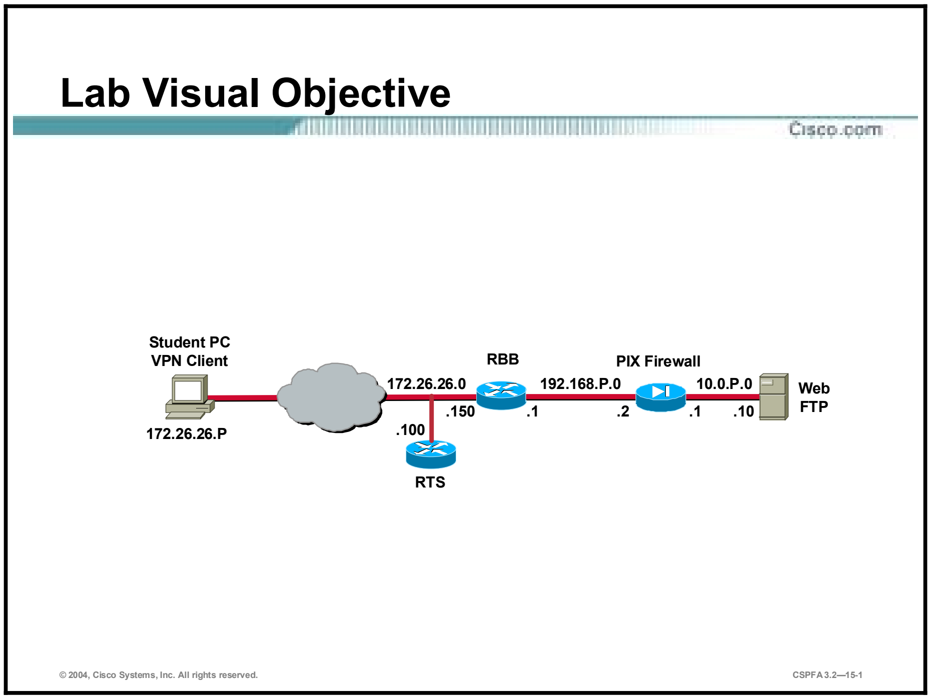
Objectives

In this lab exercise, you will complete the following tasks:

- Configure the student PC networking parameters.
- Configure the PIX Firewall.
- Verify your configuration.
- Install the Cisco VPN Client on your student PC.
- Configure the Cisco VPN Client.
- Verify the Cisco VPN Client properties.
- Launch the Cisco VPN Client.
- Verify the VPN connection.

Visual Objective

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—15-1

Scenario

Your company wants to implement a virtual private network (VPN) using remotely located Cisco VPN Clients terminating at your PIX Firewall. Your manager has asked you to configure both the remote VPN Client and the PIX Firewall to implement the following security policy:

- Users with Cisco VPN Clients can connect to an Internet service provider (ISP), and then have access to everything on the inside of the firewall. The Cisco VPN Clients access all machines on the inside using all ports and are authenticated before the IPSec tunnel is built. This traffic is encrypted.
- Users who are not running the client can connect to an internal web server using its statically-mapped global IP address. These users must be authenticated before the connection is built. This traffic is not encrypted.
- Traffic initiated by inside users does not go through the IPSec tunnel when the user connects to the Internet.

You must configure the following to implement this policy:

- Wildcard pre-shared keys
- Mode-config
- Extended authentication (XAUTH)

Task 1—Configure the Student PC Networking Parameters

Certain networking parameters must be configured before your student PC will operate in the lab environment. Complete the following steps to configure your student PC networking parameters. This procedure assumes that the Windows 2000 Server is operating with an active Network Interface Card (NIC):

Step 1 Change the IP address and default gateway of your student PC. Use the following configuration parameters:

— IP address—**172.26.26.P**
(where P = pod number)

— Subnet mask—**255.255.255.0**

— Default gateway—**172.26.26.150**

Step 2 Ping the backbone router's IP address. You should be able to ping the backbone router.

```
C:\> ping 172.26.26.150
Pinging 172.26.26.150 with 32 bytes of data:

Reply from 172.26.26.150: bytes=32 time<10ms TTL=128
Reply from 172.26.26.150: bytes=32 time<10ms TTL=128
Reply from 172.26.26.150: bytes=32 time<10ms TTL=128
Reply from 172.26.26.150: bytes=32 time<10ms TTL=128
```

Task 2—Configure the PIX Firewall

The instructor will provide you with the procedures for access to the PIX Firewall console port. After you access the PIX Firewall console port, enter configuration mode, and complete the following steps to configure the PIX Firewall:

Step 1 Create an access list that permits traffic from the inside network to hosts using addresses from mode-config pool:

```
pixP(config)#  access-list 101 permit ip 10.0.P.0 255.255.255.0
10.0.20+P.0 255.255.255.0
```

(where P = pod number)

Step 2 Instruct the PIX Firewall to bypass NAT for VPN traffic:

```
pixP(config)# nat (inside) 0 access-list 101
```

Step 3 Set up a pool of IP addresses that will dynamically be assigned to the Cisco VPN Clients via IKE mode configuration:

```
pixP(config)# ip local pool MYPOOL 10.0.20+P.1-10.0.20+P.254
```

(where P = pod number)

Step 4 Configure the PIX Firewall for TACACS+ by completing the following substeps:

1. Create a group tag called MYTACACS and assign the TACACS+ protocol to it:

```
pixP(config)# aaa-server MYTACACS protocol tacacs+
```

2. Assign the CSACS IP address and the encryption key **secretkey**:

```
pixP(config)# aaa-server MYTACACS (inside) host 10.0.P.10 secretkey
timeout 5
```

(where P = pod number)

3. Configure the PIX Firewall to require authentication for all inbound traffic:

```
pixP(config)# aaa authentication include any inbound 0 0 0 0 MYTACACS
```

Step 5 Enable the PIX Firewall to implicitly permit any packet from an IPSec tunnel, and bypass checking with an associated conduit or **access-group** command for IPSec connections:

```
pixP(config)# sysopt connection permit-ipsec
```

Step 6 Set up a transform set that will be used for the Cisco VPN Clients:

```
pixP(config)# crypto ipsec transform-set AAADES esp-des esp-md5-hmac
```

Step 7 Set up a dynamic crypto map to enable the Cisco VPN Clients to connect to the PIX Firewall:

```
pixP(config)# crypto dynamic-map DYNOMAP 10 set transform-set AAADES
```

Step 8 Create a crypto map, and assign the dynamic crypto map to it:

```
pixP(config)# crypto map VPNPEER 20 ipsec-isakmp dynamic DYNOMAP
```

Step 9 Configure Xauth to point to the RADIUS server:

```
pixP(config)# crypto map VPNPEER client authentication MYTACACS
```

Step 10 Apply the crypto map to the PIX Firewall interface:

```
pixP(config)# crypto map VPNPEER interface outside
```

Step 11 Enable IKE on the outside interface:

```
pixP(config)# isakmp enable outside
```

Step 12 Set the IKE identity:

```
pix(config)# isakmp identity address
```

Step 13 Configure the ISAKMP policy by completing the following substeps:

1. Configure a basic IKE policy using pre-shared keys for authentication:

```
pixP(config)# isakmp policy 10 authentication pre-share
```

2. Specify the encryption algorithm:

```
pixP(config)# isakmp policy 10 encryption des
```

3. Specify the hash algorithm:

```
pixP(config)# isakmp policy 10 hash md5
```

4. Specify the Diffie-Hellman (DH) by group identifier:

```
pixP(config)# isakmp policy 10 group 2
```

5. Specify the IKE Security Association's (SA's) lifetime:

```
pixP(config)# isakmp policy 10 lifetime 86400
```

Step 14 Configure the VPN group to support pushing mode configuration parameters to the Cisco VPN Client. The VPN group name of **training** must match the group name in the Cisco VPN Client. Complete the following substeps:

1. Configure the IP address pool name:

```
pixP(config)# vpngroup training address-pool MYPOOL
```

2. Configure the inactivity timeout in seconds:

```
pixP(config)# vpngroup training idle-time 1800
```

3. Configure the domain name:

```
pixP(config)# vpngroup training default-domain cisco.com
```

4. Configure the VPN group password:

```
pixP(config)# vpngroup training password training
```

Task 3—Verify Your Configuration

Complete the following steps to verify your PIX Firewall's configuration:

Step 1 Verify your IP local pool:

```
pixP(config)# sh ip local pool
Pool            Begin           End              Free    In use
MYPOOL          10.0.20+P.1     10.0.20+P.254    254        0
Available Addresses:
10.0.20+P.1
10.0.20+P.2
10.0.20+P.3
10.0.20+P.4
10.0.20+P.5
10.0.20+P.6
10.0.20+P.7
10.0.20+P.8
10.0.20+P.9
10.0.20+P.10
```

(where P = pod number)

Step 2 Verify your Network Address Translation (NAT) configuration:

```
pixP(config)# sh nat
nat (inside) 1 10.0.P.0 255.255.255.0 0 0
nat (inside) 0 access-list 101
```

(where P = pod number)

Step 3 Verify your authentication, authorization, and accounting (AAA) server configuration:

```
pixP(config)# show aaa-server
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server LOCAL protocol local
aaa-server MYTACACS protocol tacacs+
aaa-server MYTACACS (inside) host 10.0.P.10 secretkey timeout 5
```

(where P = pod number)

Step 4 Verify your crypto map:

```
pixP(config)# show crypto map
Crypto Map: "VPNPEER" interfaces: { outside }
        client authentication MYTACACS

Crypto Map "VPNPEER" 20 ipsec-isakmp
        Dynamic map template tag: DYNOMAP
```

Step 5 Verify your transform set:

```
pixP(config)# show crypto ipsec transform-set
Transform set AAADES: { esp-des esp-md5-hmac  }
```

```
      will negotiate = { Tunnel,  },
```

Step 6 Verify your IKE policy:

```
pixfirewall(config)# show isakmp policy
Protection suite of priority 10
        encryption algorithm:   DES - Data Encryption Standard (56 bit
                                   keys).
        hash algorithm:         Message Digest 5
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #2 (1024 bit)
        lifetime:               86400 seconds, no volume limit
Default protection suite
        encryption algorithm:   DES - Data Encryption Standard (56 bit
                                   keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Rivest-Shamir-Adleman Signature
        Diffie-Hellman group:   #1 (768 bit)
        lifetime:               86400 seconds, no volume limit
```

Step 7 Verify your VPN group configuration:

```
pixfirewall(config)# show vpngroup
vpngroup training address-pool MYPOOL
vpngroup training idle-time 1800
vpngroup training default-domain cisco.com
vpngroup training password ********
```

Task 4—Install the Cisco VPN Client on Your Student PC

In this lab exercise, the source files for the Cisco VPN Client already reside on the hard disk drive of the student PC. Complete the following steps to install the Cisco VPN Client:

Step 1 Open the Cisco VPN Client folder found on the student PC desktop.

Step 2 Double-click the **setup.exe** file in the Cisco VPN Client folder. The Cisco Systems VPN Client Setup window opens.

Step 3 Click **Next**. The License Agreement window opens.

Step 4 Read the license agreement and click **Yes**. You are prompted to choose a destination location.

Step 5 Accept the default destination folder by clicking **Next**. You are prompted to choose a program folder.

Step 6 Accept the defaults by clicking **Next**. The Start Copying Files window opens.

Step 7 The files are copied to the hard disk drive of the student PC, and the InstallShield Wizard Complete window opens.

Step 8 Select **Yes, I want to restart my computer now** and click **Finish**. The student PC restarts.

Step 9 Log in to the student PC.

Task 5—Configure the Cisco VPN Client

Use the following procedure to configure the networking parameters of the Cisco VPN Client. This procedure assumes Windows 2000 is already running.

Step 1 Choose **Start>Programs>Cisco Systems VPN Client>VPN Dialer**. The Cisco Systems VPN Client window opens.

Step 2 Click **New**. The New Connection Entry wizard opens.

Step 3 Enter **vpnpeerP** as the name for the new connection entry in the Name of the new connection entry field.

(where P = pod number)

Step 4 Click **Next**.

Step 5 Enter the PIX Firewall's public interface IP address, **192.168.P.2**, as the IP address of the server.

(where P = pod number)

Step 6 Click **Next**.

Step 7 Select **Group Access Information** and complete the following substeps. The following entries are always case sensitive. Use lowercase characters for this lab exercise.

1. Enter a group name: **training**.
2. Enter a group password: **training**.
3. Confirm the password: **training**.

Step 8 Click **Next**.

Step 9 Click **Finish** and leave the Cisco Systems VPN Client window open.

Task 6—Verify the Cisco VPN Client Properties

Complete the following steps to verify the Cisco VPN Client parameters you just configured:

Step 1 Ensure that the Cisco Systems VPN Client window is open. If the Cisco Systems VPN Client window is not open, choose **Start>Programs>Cisco Systems VPN Client>VPN Dialer**.

Step 2 Select **vpnpeerP** within the Connection Entry group box.

(where P = pod number)

Step 3 Verify that the IP address of the remote server is set to the PIX Firewall's public interface IP address, 192.168.P.2.

(where P = pod number)

Step 4 Click **Options**. A popup menu opens.

Step 5 Choose **Properties**. The Properties for vpnpeerP window opens.

(where P = pod number)

Step 6 Select the **General** tab and view the available options. Do not make any changes to the default settings.

Step 7 Select the **Authentication** tab and verify the spelling of the group name. If you needed to, you can edit the group name and password here.

Step 8 Select the **Connections** tab and view the available options. Do not make any changes to the default settings.

Step 9 Click **OK**.

Step 10 Close the Cisco Systems VPN Client window.

Task 7—Launch the Cisco VPN Client

Complete the following steps to launch the Cisco VPN Client on your student PC:

Step 1 Choose **Start>Programs>Cisco Systems VPN Client>VPN Dialer**.

Step 2 Verify that the Connection Entry is **vpnpeerP**.

(where P = pod number)

Step 3 Verify that the IP address of the remote server is set to the PIX Firewall's public interface IP address, **192.168.P.2**.

(where P = pod number)

Step 4 Click **Connect**. The Connection History window opens, and several messages flash by quickly. Complete the following substeps:

1. When prompted for a username, enter **studentP**.
   (where P = pod number)

2. When prompted to enter a password, enter **training**.

Step 5 Click **OK**. The following messages flash by quickly:

- Initializing the connection
- Contacting the security gateway at
- Authenticating user

The window closes and a Cisco VPN Dialer (lock) icon appears in the system tray.

Task 8—Verify the VPN Connection

Complete the following steps to verify the IPSec connection:

Step 1 Test access to the inside Web server from the remote client by completing the following substeps:

1. Open a web browser on the VPN Client PC.

2. Use the web browser to access the inside web server by entering: **http://10.0.P.10**.
   (where P = pod number)

3. The web server's home page should display.

Step 2 Double-click the Cisco VPN Dialer icon in the system tray and observe the IP address that was assigned to your student PC.

Step 3 Click the **Statistics** tab and view the information provided. Notice the number of packets encrypted and decrypted.

Step 4 Refresh your browser.

Step 5 Return to the Cisco Systems VPN Client Connection Status window and notice that the number of packets encrypted and decrypted has incremented.

Step 6 Click **OK** to close the window.

Step 7 Disconnect your VPN Dialer session using the system tray Cisco VPN Dialer icon.

Step 8 Remove the crypto map from the PIX Firewall's outside interface:

```
pixP(config)# no crypto map VPNPEER interface outside
```

Step 9 Clear your IPSec SAs with the **clear crypto sa** command:

```
pixP(config)# clear crypto sa
```

Step 10 Remove all isakmp command statements from your configuration with the **clear isakmp** command:

```
pixP(config)# clear isakmp
```

Step 11 Remove all parameters entered through the crypto map command with the **clear crypto map** command:

```
pixP(config)# clear crypto map
```

Step 12 Remove the **sysopt** command statements from your configuration with the **clear sysopt** command:

```
pixP(config)# clear sysopt
```

Step 13 Remove ACL 101 from your configuration:

```
pixP(config)# clear access-list 101
```

16

Easy VPN Remote—Small Office/Home Office

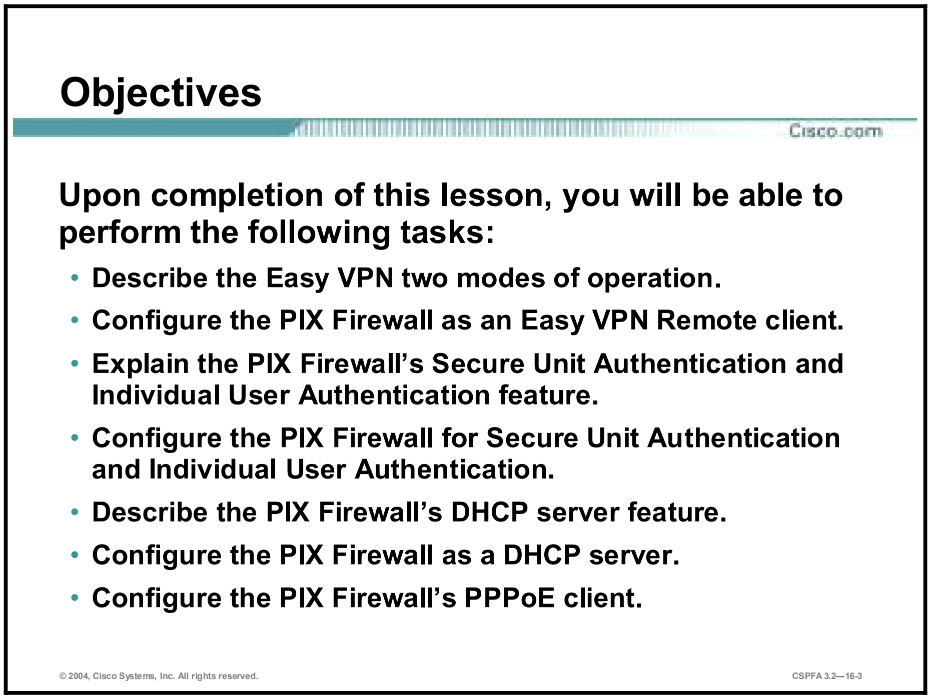
Overview

This lesson includes the following topics:

- Objectives
- PIX Easy VPN Remote feature overview
- Easy VPN Remote configuration
- PPPoE and the PIX Firewall
- DHCP server configuration
- Summary

Objectives

This topic lists the lesson’s objectives.



Objectives

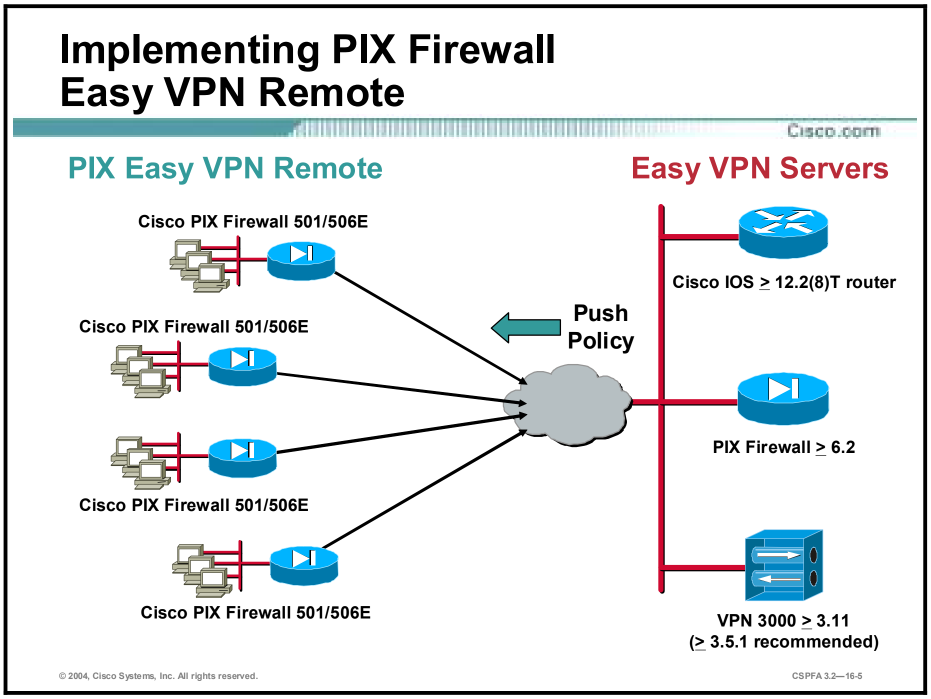
Upon completion of this lesson, you will be able to perform the following tasks:

- **Describe the Easy VPN two modes of operation.**
- **Configure the PIX Firewall as an Easy VPN Remote client.**
- **Explain the PIX Firewall’s Secure Unit Authentication and Individual User Authentication feature.**
- **Configure the PIX Firewall for Secure Unit Authentication and Individual User Authentication.**
- **Describe the PIX Firewall’s DHCP server feature.**
- **Configure the PIX Firewall as a DHCP server.**
- **Configure the PIX Firewall’s PPPoE client.**

CSPFA 3.2—16-3

PIX Easy VPN Remote Feature Overview

This topic provides an overview of the PIX Firewall Easy VPN Remote feature.



Implementing PIX Firewall Easy VPN Remote

Cisco.com

PIX Easy VPN Remote

Easy VPN Servers

Cisco PIX Firewall 501/506E

Cisco PIX Firewall 501/506E

Cisco PIX Firewall 501/506E

Cisco PIX Firewall 501/506E

Push Policy

Cisco IOS ≥ 12.2(8)T router

PIX Firewall ≥ 6.2

VPN 3000 ≥ 3.11
(≥ 3.5.1 recommended)

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—16-5

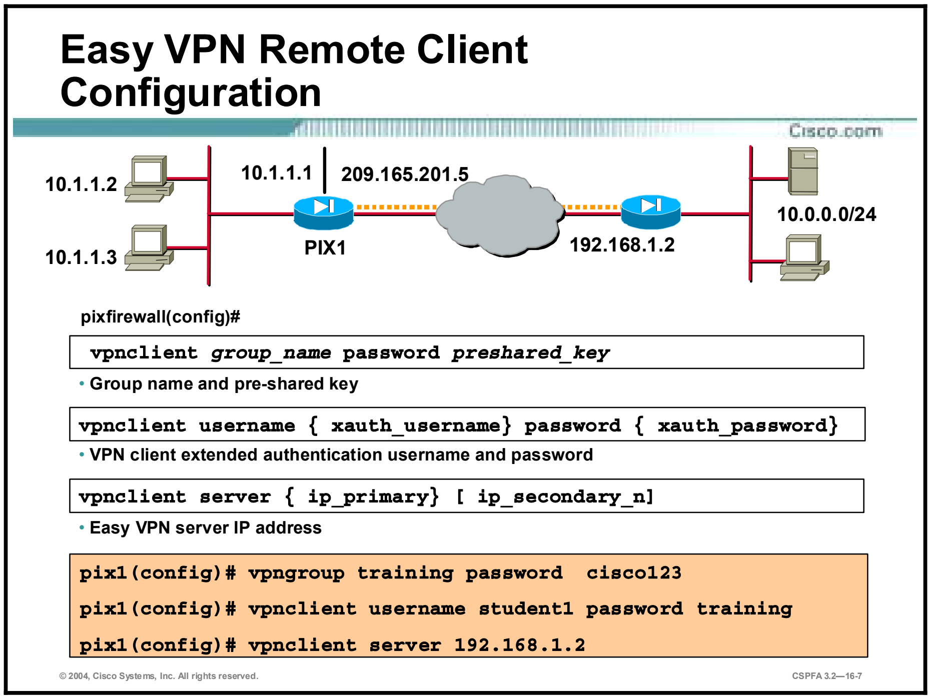
You can use any PIX Firewall unit running version 6.2 or higher as a Cisco Easy Virtual Private Network (VPN) Server or an Easy VPN Remote. Using the PIX Firewall as an Easy VPN Server lets you configure your VPN policy in a single location on the Easy VPN Server. After configuring your VPN policy, Easy VPN Server can push VPN policy configuration to multiple Easy VPN Remote devices, which greatly simplifies configuration and administration.

When you are using PIX Firewall Software Version 6.2 and higher, you can use a PIX Firewall 501 or PIX 506/506E as an Easy VPN Remote device when connecting to an Easy VPN Server, such as a Cisco VPN 3000 Concentrator, Cisco IOS router, or another PIX Firewall. Easy VPN Remote device functionality, sometimes called a “hardware client,” allows the PIX Firewall to establish a VPN tunnel to the Easy VPN Server. Hosts running on the LAN behind the PIX Firewall can connect through the Easy VPN Remote without individually running any VPN client software.

Each Easy VPN Remote device is assigned to a group. The administrator use the **vpngroup** command to associate security policy attributes with a VPN group name. As Easy VPN Remote devices establish a VPN tunnel to the Easy VPN Server, the attributes associated with their group are pushed to the Easy VPN Remote device.

Easy VPN Remote Configuration

This topic explains some basic commands that are useful for configuring the PIX Firewall.



Easy VPN Remote Client Configuration

pixfirewall(config)#

```
vpnclient group_name password preshared_key
```

- Group name and pre-shared key

```
vpnclient username { xauth_username} password { xauth_password}
```

- VPN client extended authentication username and password

```
vpnclient server { ip_primary} [ ip_secondary_n]
```

- Easy VPN server IP address

```
pix1(config)# vpngroup training password  cisco123
pix1(config)# vpnclient username student1 password training
pix1(config)# vpnclient server 192.168.1.2
```

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—16-7

The Easy VPN Server controls the policy enforced on the PIX Firewall Easy VPN Remote device. However, to establish the initial connection to the Easy VPN Server, you must complete some configuration locally. You can perform this configuration by using Cisco PIX Device Manager (PDM) or by using the command-line interface as described in the following points:

- If the Easy VPN remote uses pre-shared keys, enter the following command:

  ```
  vpnclient vpngroup { groupname} password { preshared_key}
  ```

  Replace *groupname* with an alphanumeric identifier for the VPN group. Replace *preshared_key* with the encryption key to use for securing communications to the Easy VPN Server.

- If the Easy VPN Server uses extended authentication (XAUTH) to authenticate the PIX Firewall client, enter the following command:

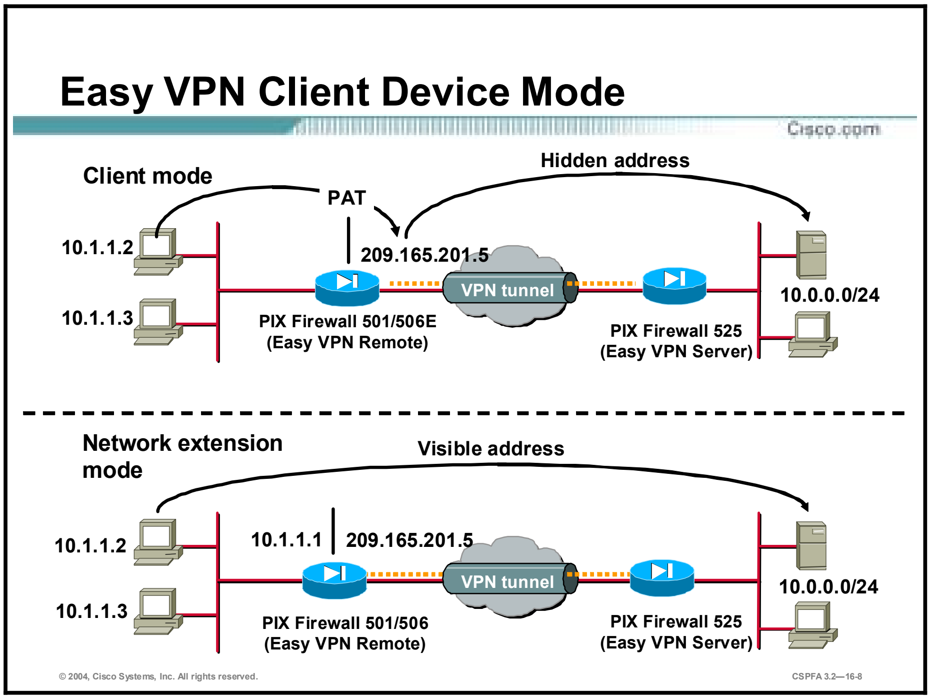
  ```
  vpnclient username { xauth_username} password { xauth_password}
  ```

  Replace *xauth_username* with the username assigned for XAUTH. Replace *xauth_password* with the password assigned for XAUTH.

- Identify the remote Easy VPN Server by entering the following command:

  ```
  vpnclient server { ip_primary} [ ip_secondary_n]
  ```

  Replace *ip_primary* with the IP address of the primary Easy VPN Server. Replace *ip_secondary_n* with the IP address of one or more Easy VPN Servers. A maximum of eleven Easy VPN Servers are supported (one primary and up to ten secondary).



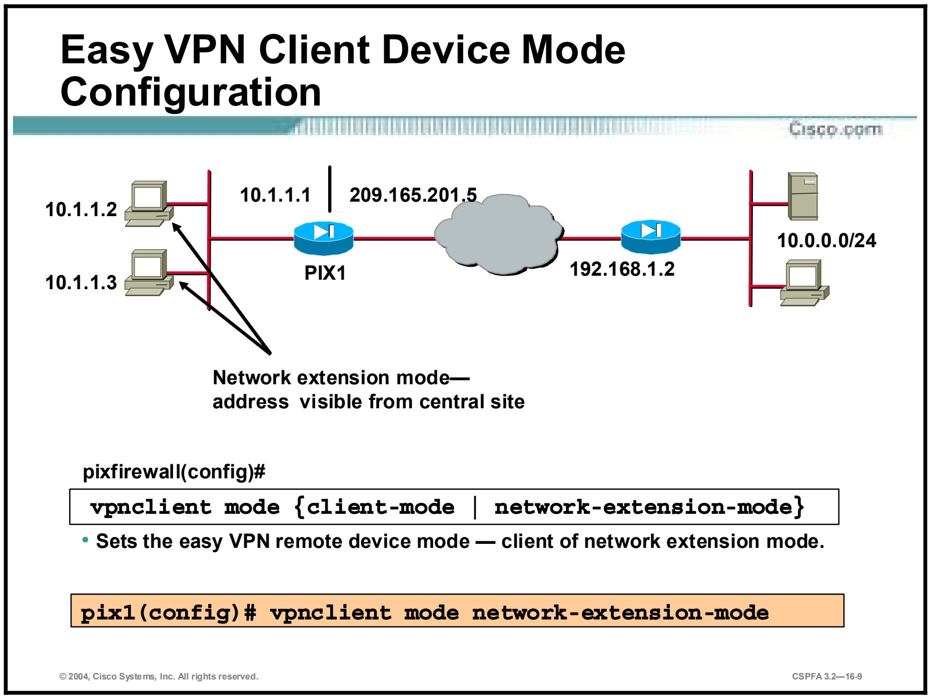
Easy VPN Client Device Mode

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—16-8

Set the Easy VPN Remote device to one of two modes, client mode or network extension mode. In client mode, the remote PIX Firewall applies PAT to all client IP addresses connected to the inside interface. In the example in the figure, when PC 10.1.1.2 attempts connect to the server at the central site, the remote PIX Firewall translates the original PC IP address and port number using the IP address and a port number of the outside interface, port address translation. Due to the translation, the PC's IP address is not visible from the central site.

The other option is network extension mode (NEM). With NEM, the IP address of the inside PCs are received without change at the central site. In this instance, the PCs IP address is visible from the central site. In the example in the figure, the remote inside PC makes a connection to a server on the central site. The original PC IP address, 10.1.1.2, is not translated by the remote PIX Firewall.



Easy VPN Client Device Mode Configuration

10.1.1.2
10.1.1.3
10.1.1.1
209.165.201.5
PIX1
192.168.1.2
10.0.0.0/24

Network extension mode—
address visible from central site

pixfirewall(config)#

```
vpnclient mode {client-mode | network-extension-mode}
```

- Sets the easy VPN remote device mode — client of network extension mode.

```
pix1(config)# vpnclient mode network-extension-mode
```

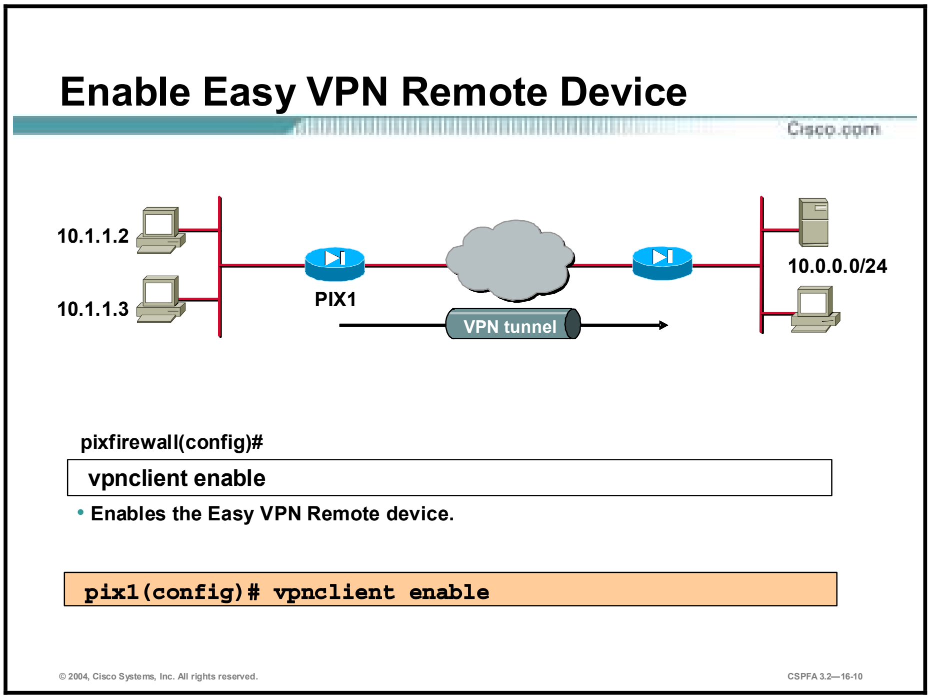
CSPFA 3.2—16-9

Set the Easy VPN Remote device mode by entering the following command:

vpnclient mode {client-mode | network-extension-mode}

Client mode applies NAT to all IP addresses of clients connected to the inside (higher security) interface of the PIX Firewall.

Network extension mode—This option does not apply Network Address Translation (NAT) to any IP addresses of clients on the inside (higher security) interface of the PIX Firewall.



Enable Easy VPN Remote Device

Cisco.com
10.1.1.2
10.1.1.3
PIX1
VPN tunnel
10.0.0.0/24

pixfirewall(config)#

```
vpnclient enable
```

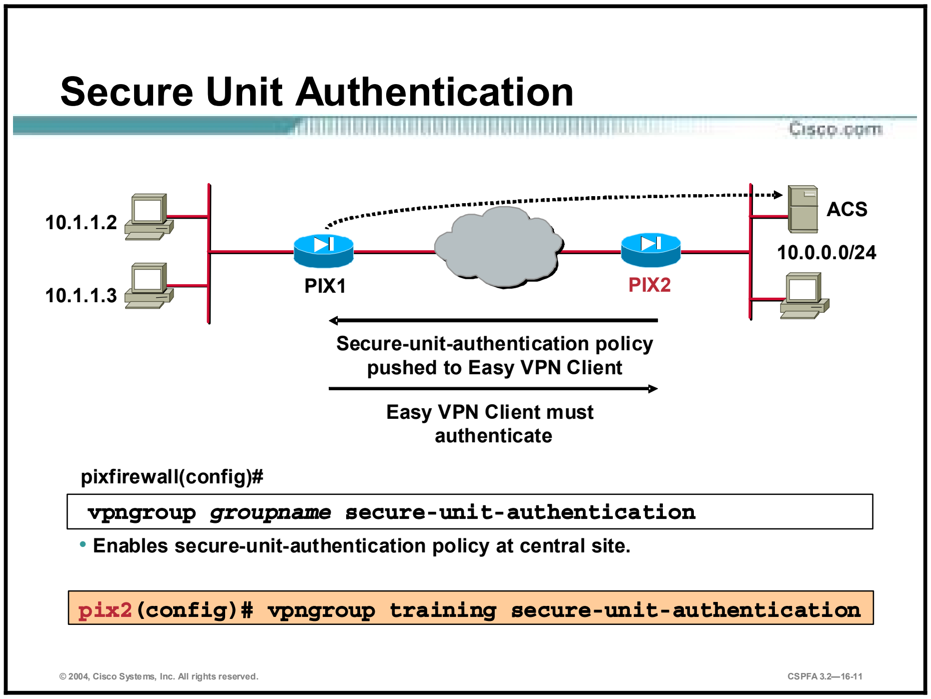
- Enables the Easy VPN Remote device.

```
pix1(config)# vpnclient enable
```

CSPFA 3.2—16-10

Enable the Easy VPN Remote device by entering the following command:

```
vpnclient enable
```



Secure Unit Authentication

Secure-unit-authentication policy pushed to Easy VPN Client

Easy VPN Client must authenticate

pixfirewall(config)#

```
vpngroup groupname secure-unit-authentication
```

- Enables secure-unit-authentication policy at central site.

```
pix2(config)# vpngroup training secure-unit-authentication
```

CSPFA 3.2—16-11

Secure Unit Authentication (SUA) is a feature introduced with PIX Firewall Software Version 6.3 to improve security when using a PIX Firewall as an Easy VPN Remote device. With SUA, one-time passwords, two-factor authentication, and similar authentication schemes can be used to authenticate the remote PIX Firewall before establishing a VPN tunnel to an Easy VPN Server. SUA is configured as part of the VPN policy on the Easy VPN Server and cannot be configured directly on the Easy VPN Remote device. After connecting to the Easy VPN Server, the Easy VPN Remote device downloads the VPN policy, which then enables or disables SUA.

When SUA is disabled and the PIX Firewall is in network extension mode, a connection is automatically initiated. When SUA is disabled with client mode, the connection is automatically initiated whenever any traffic is sent through the PIX Firewall to a network protected by the Easy VPN Server.

When SUA is enabled, static credentials included in the local configuration of the Easy VPN Remote device are ignored. A connection request is initiated as soon as an HTTP request is sent from the remote network to the network protected by the Easy VPN Server. All other traffic to the network protected by the Easy VPN Server is dropped until a VPN tunnel is established. You can also initiate a connection request from the command line interface (CLI) of the Easy VPN Remote device.

After SUA is enabled and before a VPN tunnel is established, any HTTP request to the network protected by the Easy VPN Server is redirected to the URL as follows:

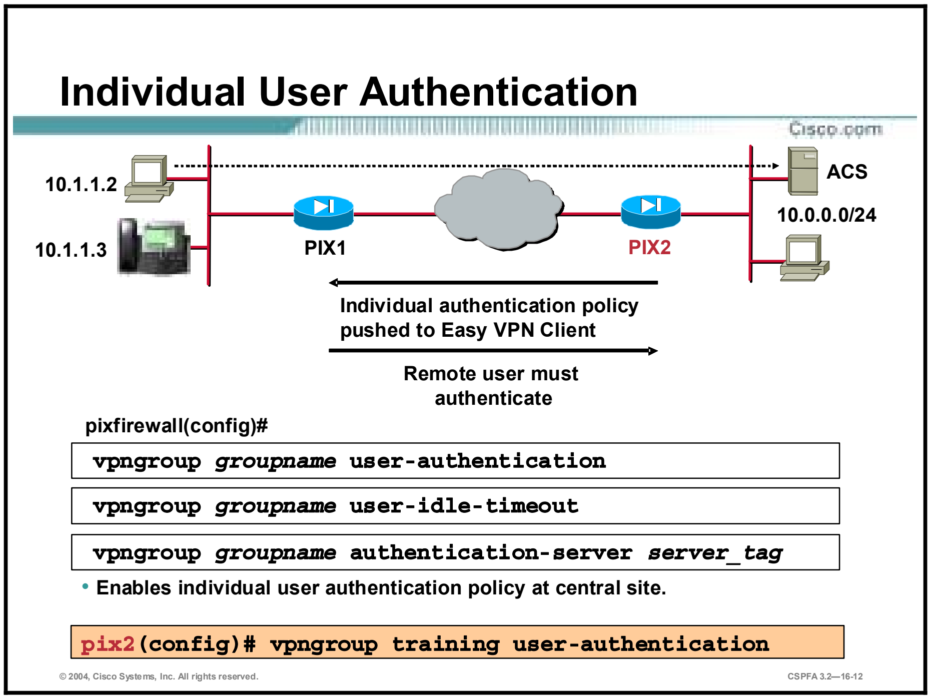
```
https:// inside-ipaddr/vpnclient/connstatus.html
```

Where *inside-ipaddr* is replaced by the IP address of the inside interface of the PIX Firewall used as the Easy VPN Remote device. You can activate the connection by manually entering this URL in the Address or Location box of a browser, and you can use this URL to check the status of the VPN tunnel. This URL provides a page containing a Connect link that displays an authentication page. If authentication is successful, the VPN tunnel is established. After the VPN tunnel is established, other users on the network protected by the Easy VPN Remote device can access the network protected by the Easy VPN Server without further authentication. If you want to control access by individual users, you can implement Individual User Authentication (IUA).

Enable SUA by entering the following command at the Easy VPN Server:

```
vpngroup groupname secure-unit-authentication
```

Replace *groupname* with an alphanumeric identifier for the VPN group using SUA.



Individual User Authentication

Cisco.com

10.1.1.2

10.1.1.3

PIX1

PIX2

ACS

10.0.0.0/24

Individual authentication policy pushed to Easy VPN Client

Remote user must authenticate

pixfirewall(config)#

```
vpngroup groupname user-authentication
```

```
vpngroup groupname user-idle-timeout
```

```
vpngroup groupname authentication-server server_tag
```

- Enables individual user authentication policy at central site.

```
pix2(config)# vpngroup training user-authentication
```

CSPFA 3.2—16-12

Individual User Authentication (IUA) causes clients on the inside network of the Easy VPN Remote to be individually authenticated based on the IP address of the inside client. IUA supports authentication with both static and dynamic password mechanisms.

IUA is enabled by means of the downloaded VPN policy and it cannot be configured locally. When IUA is enabled, each user on the network protected by the Easy VPN Remote device is prompted for a user name and password when trying to initiate a connection. A PIX Firewall acting as an Easy VPN Server downloads the contact information for the authentication, authorization, and accounting (AAA) server to the Easy VPN Remote device, which sends each authentication request directly to the AAA server. A PIX Firewall Easy VPN Server performs proxy authentication to the AAA server. The Easy VPN Remote device sends each authentication request to the Easy VPN Server.

- IUA supports individually authenticating clients on the inside network of the Easy VPN Remote, based on the IP address of each inside client. IUA supports both static and one-time password (OTP) authentication mechanisms. IUA is enabled by means of the downloaded VPN policy and it cannot be configured locally. To enable IUA on a PIX Firewall used as the Easy VPN Server, enter the following command:

  vpngroup *groupname* user-authentication

  This command enables individual user authentication for the VPN group identified by *groupname*.

- To specify the length of time that a VPN tunnel can remain open without user activity, enter the following command:

  vpngroup *groupname* user-idle-timeout { hh: mm: ss}

  This command specifies the length of time for the specified VPN group in hours, minutes, and seconds (hh:mm:ss).

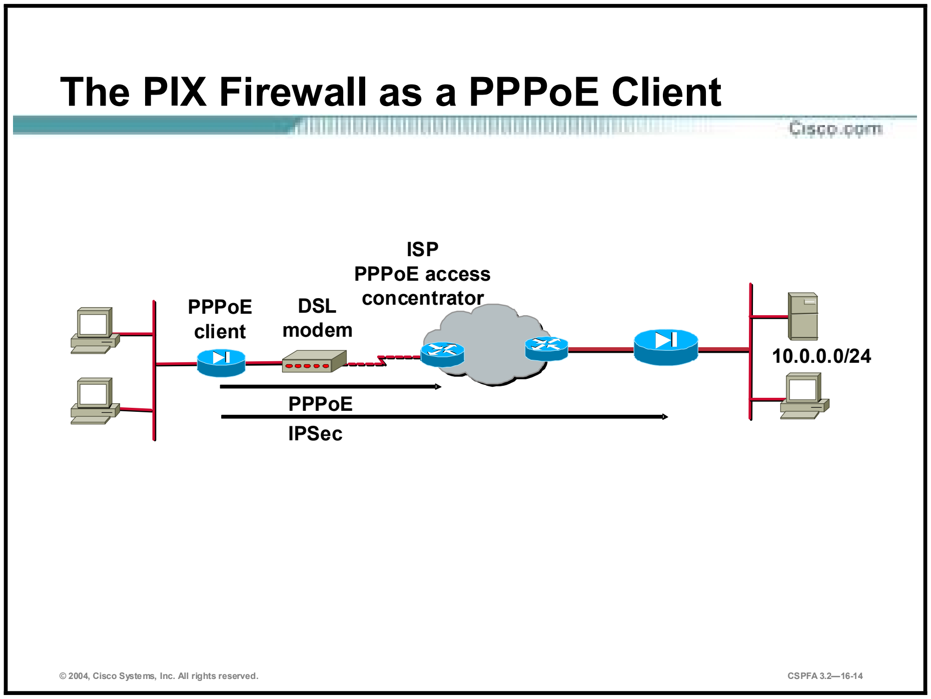
- To specify the AAA server to use for IUA on a PIX Firewall being used as the Easy VPN Server, enter the following command:

  ```
  vpngroup groupname authentication-server server_tag
  ```

  This command specifies the AAA server identified by *server_tag* for the VPN group identified by *groupnam*e.

PPPoE and the PIX Firewall

This topic explains how the PIX Firewall works with Point-to-Point Protocol over Ethernet (PPPoE).



The PIX Firewall as a PPPoE Client

CSPFA 3.2—16-14

Beginning with software version 6.2, the PIX Firewall can be configured as a PPPoE client. This makes it compatible with broadband offerings that require PPPoE usage. Many Internet service providers (ISPs) deploy PPPoE because it supports high-speed broadband access using their existing remote access infrastructure and because it is easy for customers to use.

Broadband connections such as digital subscriber line (DSL), cable modem, and fixed wireless deliver high-speed, "always on" connections at a low cost. The PIX Firewall enables you to secure these broadband Internet connections.

PPPoE combines two widely accepted standards, Ethernet and Point-to-Point Protocol (PPP), to provide an authenticated method of assigning IP addresses to client systems. PPPoE is composed of two main phases:

- Active discovery phase—In this phase, the PPPoE client locates a PPPoE server, called an access concentrator (AC). A session identification is assigned and the PPPoE layer is established.
- PPP session phase—In this phase, PPP options are negotiated and authentication is performed. After the link setup is completed, PPPoE functions as a Layer 2 encapsulation method, allowing data to be transferred over the PPP link within PPPoE headers.

After it is configured, the PIX Firewall's PPPoE client automatically connects to a service provider's AC without user intervention. The maximum transmission unit (MTU) size is automatically set to 1492 bytes, the correct value to allow PPPoE to be transmitted in an Ethernet frame. All traffic flowing to, from, and through the interface is then encapsulated with

PPPoE/PPP headers. The PIX Firewall also detects session termination and automatically attempts to reconnect.

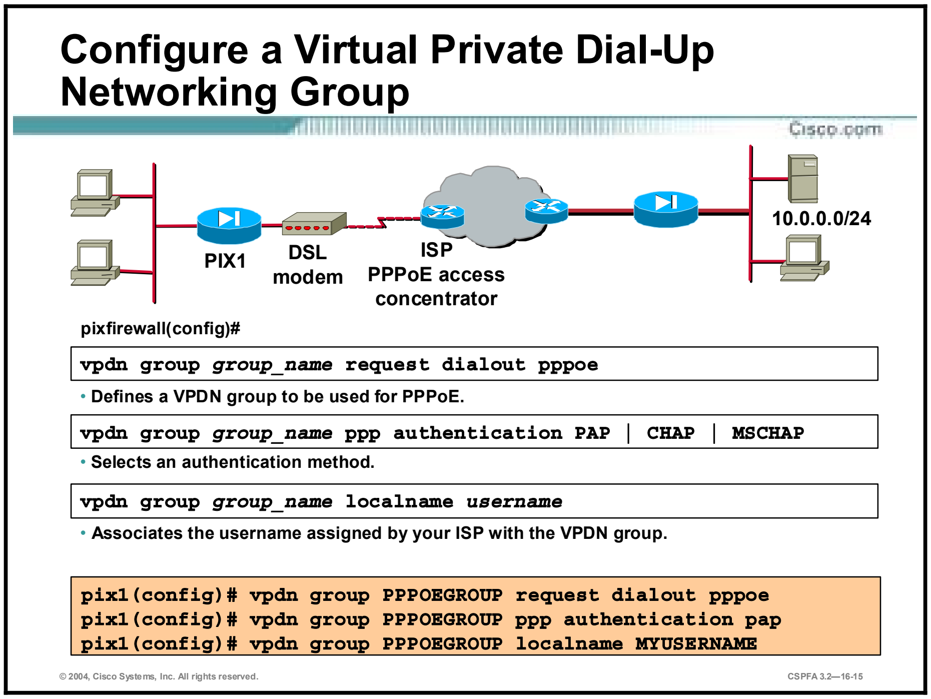
The PIX Firewall PPPoE client can operate in environments where other PIX Firewall features are being used. For example, the following features function as usual:

- NAT on traffic to or from the outside interface or over a VPN
- URL and content filtering before transmission to or from the outside interface
- Application of firewall rules on traffic before transmission to or from the outside interface or over a VPN

If your ISP's PPPoE server distributes configuration parameters such as Domain Name System (DNS) and Windows Internet Name Service (WINS) addresses along with the IP addresses it assigns to its clients, the PIX Firewall's PPPoE client can retrieve these parameters and automatically pass them along to its Dynamic Host Configuration Protocol (DHCP) clients. The PIX Firewall must be configured with the dhcpd auto_config option for this to work. Although the PIX Firewall's DHCP server feature functions normally with the PPPoE client enabled, its DHCP and PPPoE client features are mutually exclusive. When you configure the DHCP client on the outside interface, the PPPoE client is automatically disabled. The converse of this configuration statement is also true; when the PPPoE client is configured, the DHCP client is automatically disabled.

With PPPoE support, the PIX Firewall is able to provide telecommuters, branch offices and small businesses with firewall, VPN, and intrusion protection.

Note The PIX Firewall's PPPoE client is not interoperable with failover, Layer 2 Tunneling Protocol (L2TP), or Point-to-Point Tunneling Protocol (PPTP).



Configure a Virtual Private Dial-Up Networking Group

pixfirewall(config)#

```
vpdn group group_name request dialout pppoe
```

- Defines a VPDN group to be used for PPPoE.

```
vpdn group group_name ppp authentication PAP | CHAP | MSCHAP
```

- Selects an authentication method.

```
vpdn group group_name localname username
```

- Associates the username assigned by your ISP with the VPDN group.

```
pix1(config)# vpdn group PPPOEGROUP request dialout pppoe
pix1(config)# vpdn group PPPOEGROUP ppp authentication pap
pix1(config)# vpdn group PPPOEGROUP localname MYUSERNAME
```

CSPFA 3.2—16-15

Five steps are needed to configure the PIX Firewall PPPoE client. The first four steps require the use of the **vpdn** command as follows:

Step 1 Use the **vpdn group** command to define a VPDN group to be used for PPPoE.

Step 2 If your ISP requires authentication, use the **vpdn group** command to select one of the following authentication protocols:

- **PAP**—Password Authentication Protocol
- **CHAP**—Challenge Handshake Authentication Protocol
- **MS-CHAP**—Microsoft Challenge Handshake Authentication Protocol

Note ISPs that use CHAP or MS-CHAP may refer to the username as the remote system name, and may refer to the password as the CHAP secret.

Step 3 Use the **vpdn group** command to associate the username assigned by your ISP to the VPDN group.

The syntaxes for the **vpdn** commands are as follows:

vpdn group ***group_name*** **request dialout pppoe**

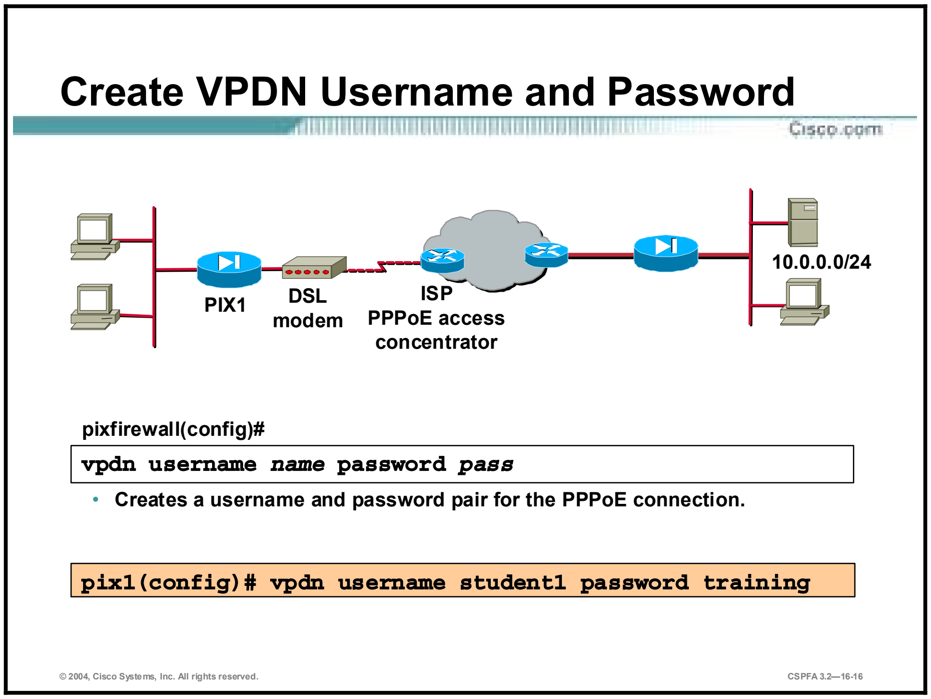
vpdn group ***group_name*** **ppp authentication PAP | CHAP | MSCHAP**

vpdn group ***group_name*** **localname** ***username***

clear vpdn [group | username | tunnel [all | [id ***tunnel_id*****]]]**

group_name	Descriptive name for the group.
local name *username*	Username assigned by the ISP.

username ***name***	Local username. However, when used as a **clear** command option, username removes all **vpdn username** commands from the configuration.
pass	Password assigned by the ISP.
group	Removes all **vpdn group** commands from the configuration.
tunnel	Removes one or more L2TP or PPTP tunnels from the configuration. This option is not used with PPPoE.
all	Removes all L2TP or PPTP tunnels from the configuration. This option is not used with PPPoE.
id	PPPoE session identifier.
session_id	Unique session identifier.



Create VPDN Username and Password

pixfirewall(config)#

```
vpdn username name password pass
```

- Creates a username and password pair for the PPPoE connection.

```
pix1(config)# vpdn username student1 password training
```

CSPFA 3.2—16-16

Step 4 Use the **vpdn username** command to create a username and password pair for the PPPoE connection. This username and password combination is used to authenticate the PIX Firewall to the AC. The username must be a username that is already associated with the VPDN group specified for PPPoE.

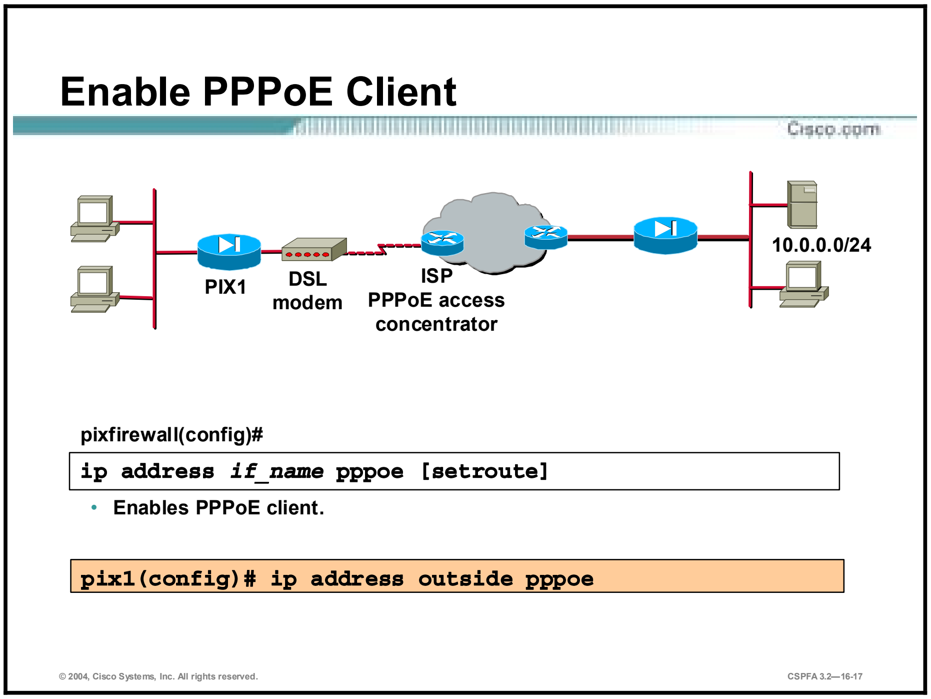
The **clear vpdn** command removes all **vpdn** commands from the configuration. The **clear vpdn group** command removes all **vpdn group** commands from the configuration, and the **clear vpdn username** command removes all **vpdn username** commands from the configuration.

The syntaxes for the **vpdn** commands are as follows:

vpdn username ***name*** **password** ***pass***

clear vpdn [group | username | tunnel [all | [id ***tunnel_id*****]]]**

username ***name***	Local username. However, when used as a **clear** command option, username removes all **vpdn username** commands from the configuration.
pass	Password assigned by the ISP.



Enable PPPoE Client

pixfirewall(config)#

```
ip address if_name pppoe [setroute]
```

- Enables PPPoE client.

```
pix1(config)# ip address outside pppoe
```

CSPFA 3.2—16-17

Step 5 PPPoE client functionality is disabled by default. Use the **ip address pppoe** command to enable PPPoE on the PIX Firewall.

The syntax for the **ip address pppoe** command is as follows:

ip address ***if_name*** **pppoe [setroute]**

if_name	The name of the outside interface of the PIX Firewall.
setroute	Tells the PIX Firewall to set the default route using the default gateway parameter the DHCP or PPPoE server returns.

Re-enter the **ip address outside pppoe** command to clear and restart a PPPoE session. This shuts down the current session and starts a new one.

The PPPoE client is only supported on the outside interface of the PIX Firewall. PPPoE is not supported in conjunction with DHCP because with PPPoE the IP address is assigned by PPP. The **setroute** option causes a default route to be created if no default route exists. The default router will be the address of the access concentrator. If you use the setroute option when a default route is already set in the configuration, the PIX Firewall does not establish PPPoE. This is because it cannot overwrite an existing default gateway with one supplied by PPPoE. If you wish to use the default route from a PPPoE server, erase the default route in the configuration. The maximum transmission unit (MTU) size is automatically set to 1492 bytes, which is the correct value to allow PPPoE transmission within an Ethernet frame.

You can also enable PPPoE by manually entering the IP address, using the command in the following format:

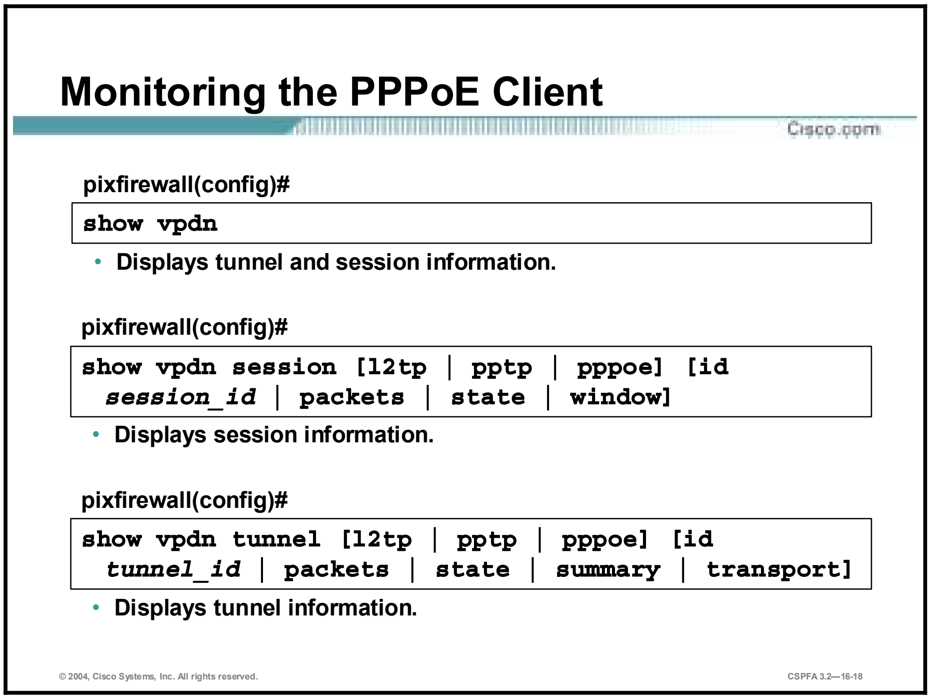
```
ip address ifname ipaddress mask pppoe
```

This command causes the PIX Firewall to use the specified address instead of negotiating with the PPPoE server to assign an address dynamically. To use this command, replace *ifname* with the name of the outside interface of the PIX Firewall connected to the PPPoE server. Replace *ipaddress* and *mask* with the IP address and subnet mask assigned to your PIX Firewall.

The syntax for the **ip address pppoe** command is as follows:

ip address *if_name ip_address netmask* pppoe [setroute]

if_name	The name of the outside interface of the PIX Firewall.
setroute	Tells the PIX Firewall to set the default route using the default gateway parameter the DHCP or PPPoE server returns.
ip_address	The IP address assigned to the PIX Firewall's outside interface.
netmask	The subnet mask assigned to the PIX Firewall's outside interface.



Monitoring the PPPoE Client

Cisco.com

pixfirewall(config)#

```
show vpdn
```

- Displays tunnel and session information.

pixfirewall(config)#

```
show vpdn session [l2tp | pptp | pppoe] [id
  session_id | packets | state | window]
```

- Displays session information.

pixfirewall(config)#

```
show vpdn tunnel [l2tp | pptp | pppoe] [id
  tunnel_id | packets | state | summary | transport]
```

- Displays tunnel information.

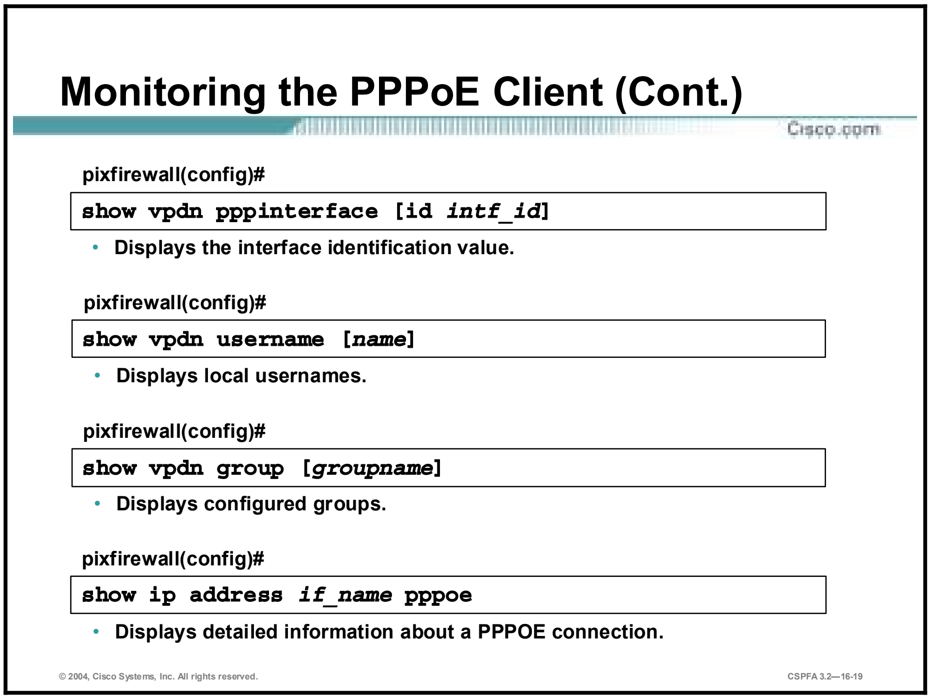
© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—16-18

The **show vpdn** command displays PPPoE tunnel and session information. To view only session information, use the **show vpdn session** command. To view only tunnel information, use the **show vpdn tunnel** command.

The following example shows the kind of information provided by the **show vpdn** commands:

```
pix1# sh vpdn
Tunnel id 0, 1 active sessions
   time since change 65862 secs
   Remote Internet Address 172.31.31.1
  Local Internet Address 192.168.10.2
   6 packets sent, 6 received, 84 bytes sent, 0 received
Remote Internet Address is 10.0.0.1
   Session state is SESSION_UP
    Time since event change 65865 secs, interface outside
    PPP interface id is 1
    6 packets sent, 6 received, 84 bytes sent, 0 received
pix1# sh vpdn session
PPPoE Session Information (Total tunnels=1 sessions=1)
Remote Internet Address is 172.31.31.1
 Session state is SESSION_UP
  Time since event change 65887 secs, interface outside
  PPP interface id is 1
  6 packets sent, 6 received, 84 bytes sent, 0 received
pix1# sh vpdn tunnel
PPPoE Tunnel Information (Total tunnels=1 sessions=1)
```

```
Tunnel id 0, 1 active sessions
   time since change 65901 secs
   Remote Internet Address 172.31.31.1
   Local Internet Address 192.168.10.2
   6 packets sent, 6 received, 84 bytes sent, 0 received
```



Monitoring the PPPoE Client (Cont.)

pixfirewall(config)#

```
show vpdn pppinterface [id intf_id]
```

- Displays the interface identification value.

pixfirewall(config)#

```
show vpdn username [name]
```

- Displays local usernames.

pixfirewall(config)#

```
show vpdn group [groupname]
```

- Displays configured groups.

pixfirewall(config)#

```
show ip address if_name pppoe
```

- Displays detailed information about a PPPOE connection.

CSPFA 3.2—16-19

You can use the **show vpdn pppinterface** command when a PPPoE session is established to view the address of the AC. When the PIX Firewall is unable to find an AC, the address of the AC is displayed as 0.0.0.0.

Use the **show vpdn username** command to view local usernames. Use **the show vpdn group** command to view your configured VPDN groups.

When a PPPoE session is established, you can use the **show ip address outside pppoe** command to view the IP address assigned by the PPPoE server.

The syntaxes for the **show vpdn** commands are as follows:

show vpdn

show vpdn session [l2tp | pptp | pppoe] [id *session_id* | packets | state | window]

show vpdn tunnel [l2tp | pptp | pppoe] [id *tunnel_id* | packets | state | summary | transport]

show vpdn pppinterface [id *intf_id*]

show vpdn username [*name*]

show vpdn group [*groupname*]

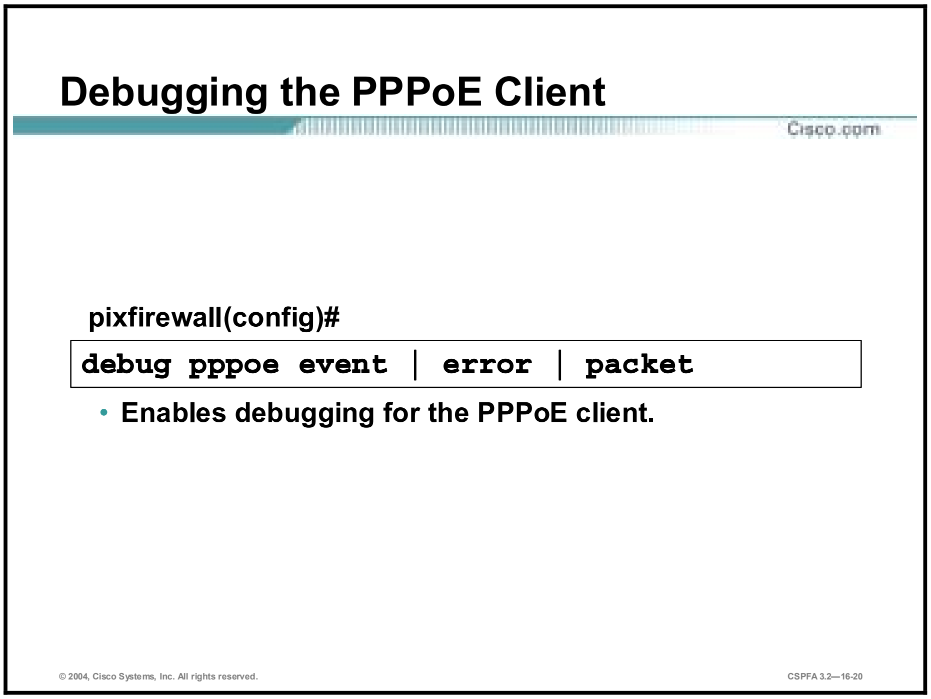
session_id	A unique session identifier.
tunnel_id	A unique tunnel identifier.
intf_id	A unique identifier for a PPP virtual interface. A PPP virtual interface is created for each PPTP or PPPoE tunnel.
name	The local username.

group_name	The VPDN group name. The VPDN group_name is an ASCII string to denote a VPDN group. The maximum length is 63 characters.

The syntax for the **show ip address pppoe** command is as follows:

show ip address *if_name* pppoe

if_name	The internal or external interface name designated by the **nameif** command.



Debugging the PPPoE Client

pixfirewall(config)#

```
debug pppoe event | error | packet
```

- **Enables debugging for the PPPoE client.**

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—16-20

Use the **debug pppoe** command to enable debugging for the PPPoE client.

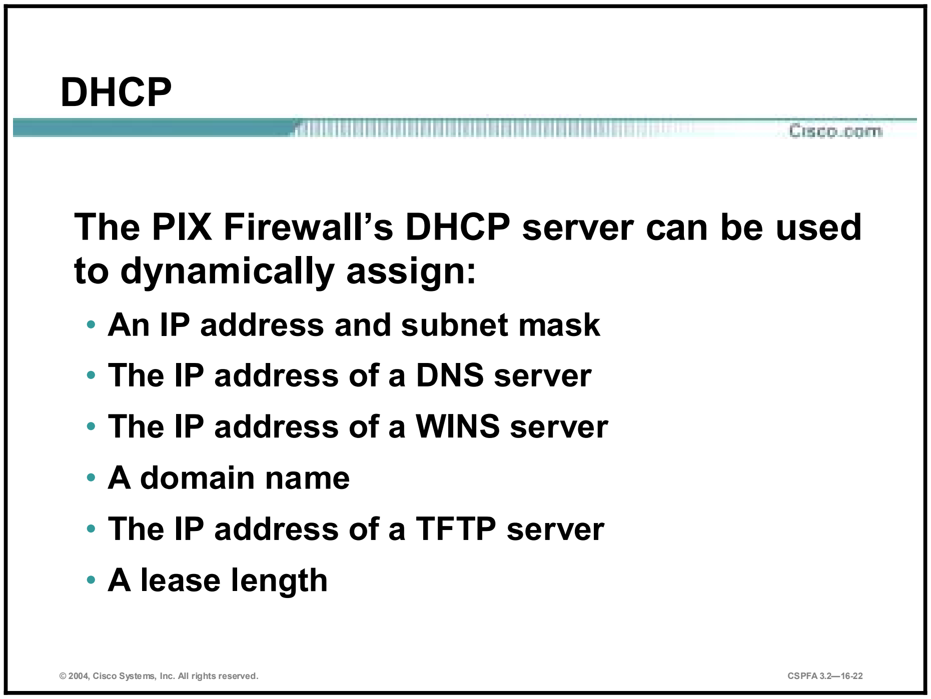
The syntax for the **debug pppoe** command is as follows:

debug pppoe event | error | packet

event	Displays protocol event information.
error	Displays error messages.
packet	Displays packet information.

DHCP Server Configuration

This topic explains how to configure the PIX Firewall's DHCP server.



DHCP

Cisco.com

The PIX Firewall's DHCP server can be used to dynamically assign:

- **An IP address and subnet mask**
- **The IP address of a DNS server**
- **The IP address of a WINS server**
- **A domain name**
- **The IP address of a TFTP server**
- **A lease length**

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—16-22

DHCP provides automatic allocation of reusable network addresses on a TCP/IP network. This provides ease of administration and dramatically reduces the margin of human error. Without DHCP, IP addresses must be manually entered at each computer or device that requires an IP address.

DHCP can also distribute other configuration parameters such as DNS and WINS server addresses and domain names. The host that distributes the addresses and configuration parameters to DHCP clients is called a DHCP server. A DHCP client is any host using DHCP to obtain configuration parameters.

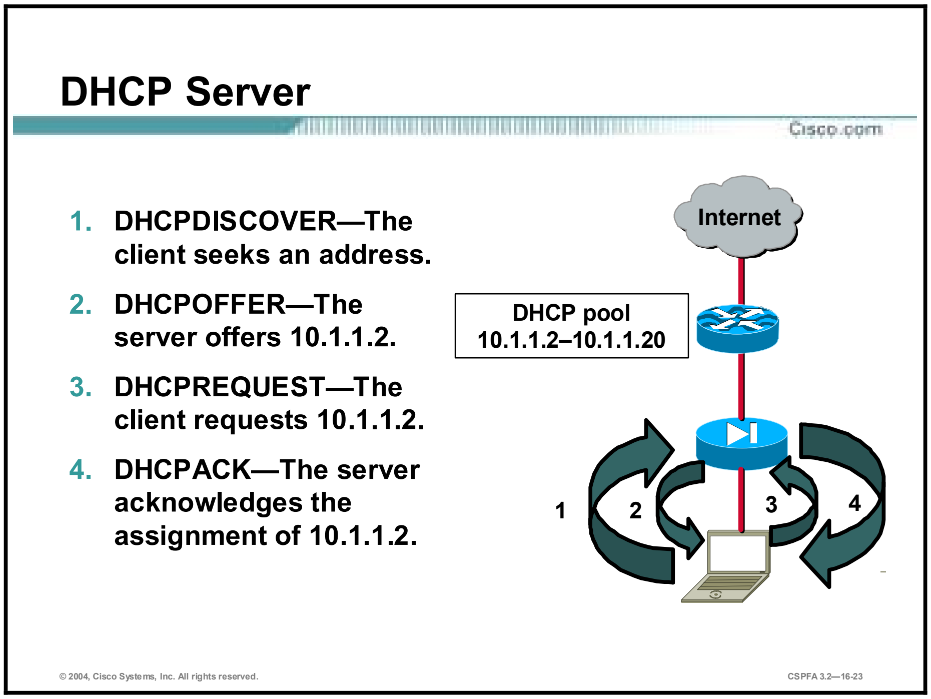
Because DHCP traffic consists of broadcasts and a significant goal of router configuration is to control unnecessary proliferation of broadcast packets, it may be necessary to enable forwarding of DHCP broadcast packets on routers that lie between your DHCP server and its clients. Use the **ip helper-address** interface configuration command to have the Cisco IOS software forward these broadcasts. The address specified in the command should be that of the DHCP server.

Note WINS registers Network Basic Input/Output System (NetBIOS) computer names and resolves them to IP addresses.

Any PIX Firewall that runs version 5.2 or higher supports a DHCP server and client. In a network environment secured by a PIX Firewall, PC clients connect to the PIX Firewall and establish network connections to access an enterprise or corporate network. As a DHCP server, the PIX Firewall provides these PCs (its DHCP clients) the networking parameters necessary for accessing the enterprise or corporate network, and once inside the network, the PIX

Firewall provides the network services to use, such as the DNS server. As a DHCP client, the PIX Firewall is able to obtain an IP address, subnet mask, and, optionally, a default route from a DHCP server.

Currently, the PIX Firewall can distribute configuration parameters only to clients that are physically connected to the subnet of its inside interface.



DHCP Server

Cisco.com

1. DHCPDISCOVER—The client seeks an address.
2. DHCPOFFER—The server offers 10.1.1.2.
3. DHCPREQUEST—The client requests 10.1.1.2.
4. DHCPACK—The server acknowledges the assignment of 10.1.1.2.

CSPFA 3.2—16-23

DHCP communication consists of several broadcast messages passed between the DHCP client and DHCP server. The following events occur during this exchange:

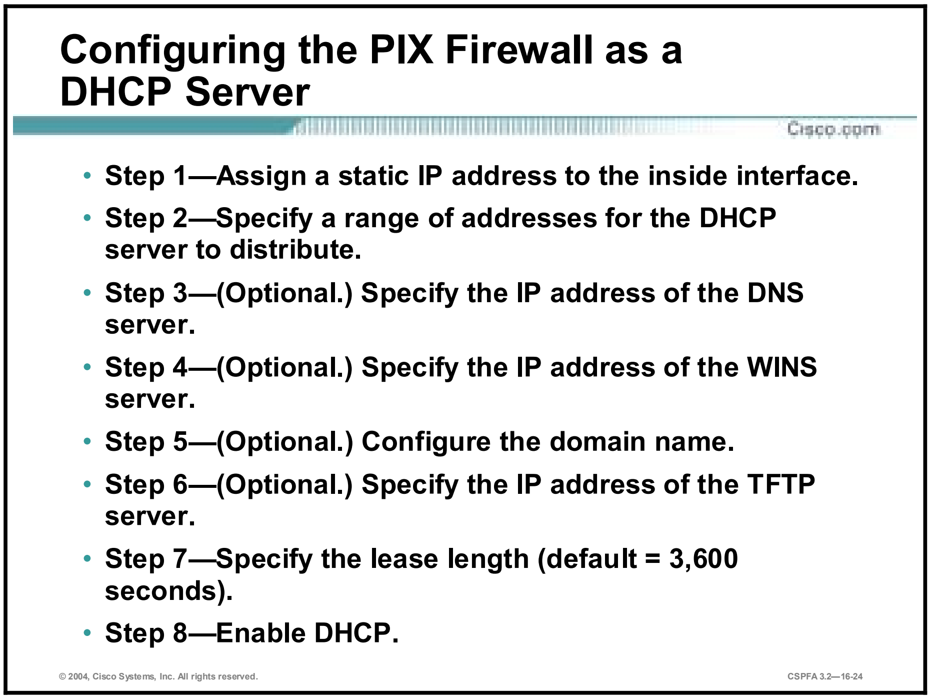
1. The client broadcasts a DHCPDISCOVER message on its local physical subnet to locate available DHCP servers.

2. Any reachable DHCP server may respond with a DHCPOFFER message that includes an available network address and other configuration parameters.

3. Based on the configuration parameters offered in the DHCPOFFER messages, the client chooses one server from which to request configuration parameters. The client broadcasts a DHCPREQUEST message requesting the offered parameters from one server and implicitly declining offers from all others.

4. The server selected in the DHCPREQUEST message responds with a DHCPACK message containing the configuration parameters for the requesting client. If the selected server has since become unable to satisfy the DHCPREQUEST (for example, in case the requested network address has already been allocated) the server responds with a DHCPNAK message. The client receives either the DHCPNAK or the DHCPACK containing the configuration parameters.

Note The PIX Firewall DHCP server does not support BOOTP requests and failover configurations.



Configuring the PIX Firewall as a DHCP Server

- **Step 1—Assign a static IP address to the inside interface.**
- **Step 2—Specify a range of addresses for the DHCP server to distribute.**
- **Step 3—(Optional.) Specify the IP address of the DNS server.**
- **Step 4—(Optional.) Specify the IP address of the WINS server.**
- **Step 5—(Optional.) Configure the domain name.**
- **Step 6—(Optional.) Specify the IP address of the TFTP server.**
- **Step 7—Specify the lease length (default = 3,600 seconds).**
- **Step 8—Enable DHCP.**

CSPFA 3.2—16-24

Complete the following steps to enable DHCP server support on the PIX Firewall:

Step 1 Assign a static IP address to the inside interface by using the **ip address** command. (covered in a previous lesson)

Step 2 Specify a range of addresses for the DHCP server to distribute by using the **dhcpd address** command.

Step 3 Specify the IP address of the DNS server that the client will use by using the **dhcpd dns** command. This step is optional.

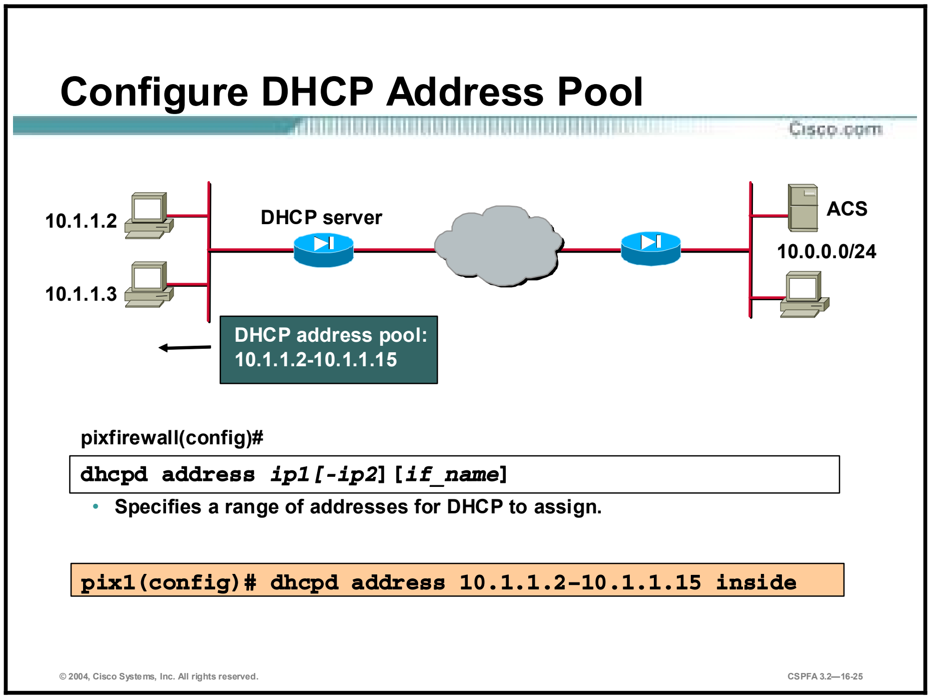
Step 4 Specify the IP address of the WINS server the client will use by using the **dhcpd wins** command. This step is also optional.

Step 5 Configure the domain name the client will use by using the **dhcpd domain** command. This step is optional.

Step 6 Specify the IP address of the TFTP server. This step is also optional.

Step 7 Specify the lease length to grant the client by using the **dhcpd lease** command.

Step 8 Enable the DHCP daemon within the PIX Firewall to listen for DHCP client requests on the enabled interface by using the **dhcpd enable** command.



Configure DHCP Address Pool

pixfirewall(config)#

```
dhcpd address ip1[-ip2] [if_name]
```

- Specifies a range of addresses for DHCP to assign.

```
pix1(config)# dhcpd address 10.1.1.2-10.1.1.15 inside
```

CSPFA 3.2—16-25

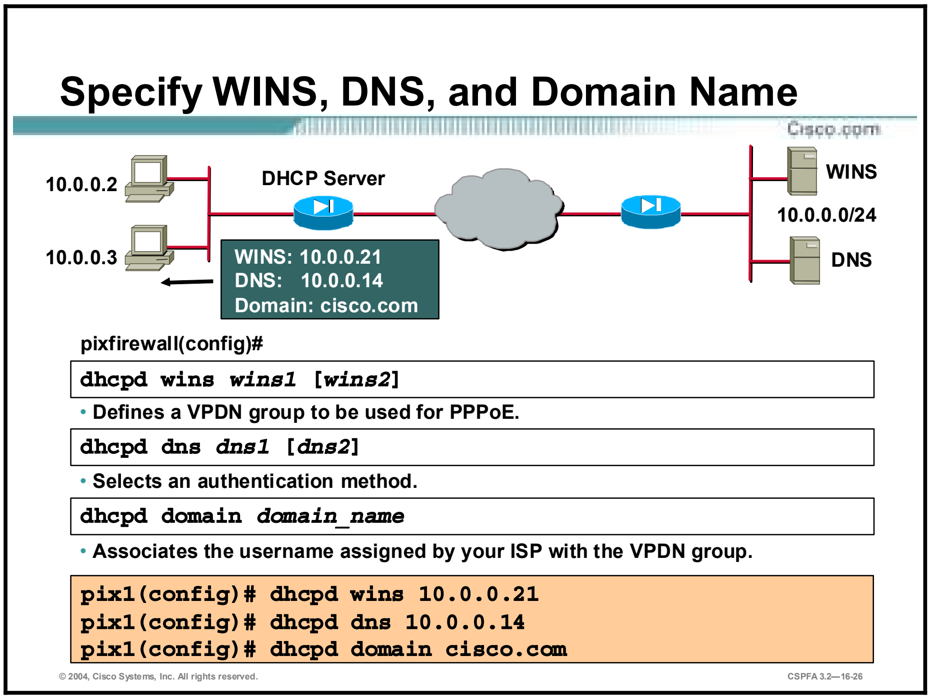
The **dhcpd address** command specifies the range of IP addresses for the server to distribute. The address pool of a PIX Firewall DHCP server must be within the same subnet as the PIX Firewall interface that is enabled. In other words, the client must be physically connected to the subnet of a PIX Firewall interface. Up to 256 addresses can be included in the pool. The default for the PIX Firewall interface name is the inside interface, which is the only interface currently supported. The **no dhcpd address** command removes the DHCP server address pool.

The syntax for the **dhcpd address** command is as follows:

dhcpd address ***ip1*** **[-*ip2*] [*if_name*]**

address *ip1 [ip2]*	The IP pool address range. The size of the pool is limited to 256 addresses.
if_name	Name of the PIX Firewall interface. The default is the inside interface. The PIX Firewall DHCP server daemon can only be enabled on the inside interface.

Note The DHCP address pool is limited to 32 addresses for a PIX Firewall 501 with a 10-user license. With the 50-user license, 128 addresses are supported.



Specify WINS, DNS, and Domain Name

pixfirewall(config)#

```
dhcpd wins wins1 [wins2]
```

- Defines a VPDN group to be used for PPPoE.

```
dhcpd dns dns1 [dns2]
```

- Selects an authentication method.

```
dhcpd domain domain_name
```

- Associates the username assigned by your ISP with the VPDN group.

```
pix1(config)# dhcpd wins 10.0.0.21
pix1(config)# dhcpd dns 10.0.0.14
pix1(config)# dhcpd domain cisco.com
```

CSPFA 3.2—16-26

The **dhcpd dns** command specifies the IP address of the DNS server for DHCP clients. Up to two DNS servers can be specified with this command. Use the **no dhcpd dns** command to remove the DNS IP addresses from your configuration.

The syntax for the **dhcpd dns** command is as follows:

dhcpd dns *dns1 [dns2]*

dns ***dns1 [dns2]***	The IP addresses of the DNS servers for the DHCP client. The second server address is optional.

Specify the IP addresses or addresses of the WINS server or servers the client will use. You can specify up to two WINS servers.

The syntax for the **dhcpd dns** command is as follows:

dhcpd wins *wins1 [wins2]*

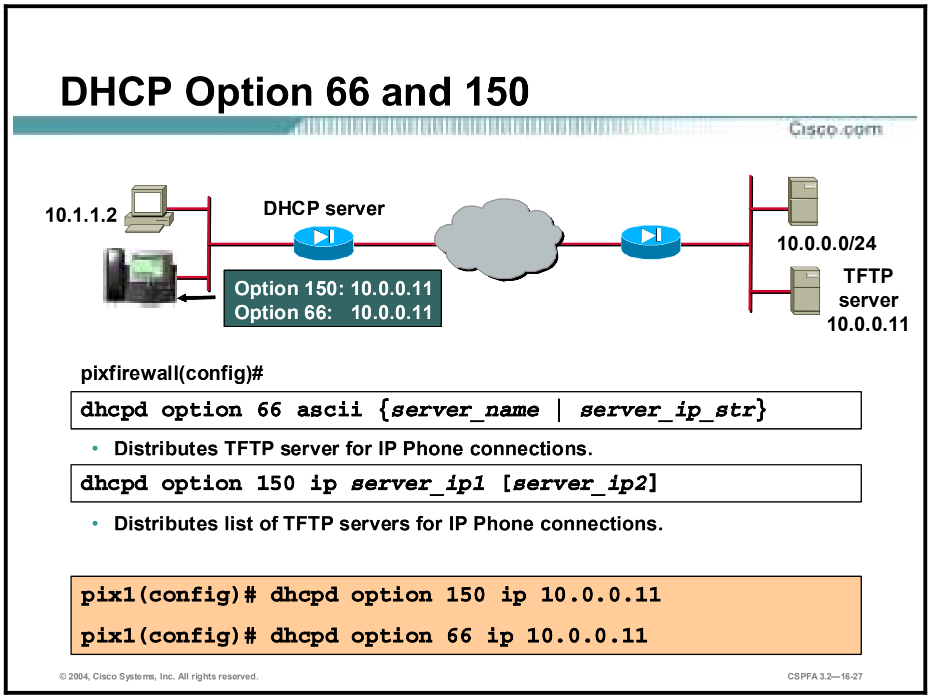
wins ***wins1 [wins2]***	The IP addresses of the Microsoft NetBIOS name servers (WINS server). The second server address is optional.

Specify the domain name the client will use.

The syntax for the **dhcpd domain** ***domain_name*** command is as follows:

dhcpd domain *domain_name*

domain ***domain_name***	The DNS domain name. For example, example.com.



DHCP Option 66 and 150

Cisco.com
10.1.1.2
DHCP server
Option 150: 10.0.0.11
Option 66:   10.0.0.11
10.0.0.0/24
TFTP
server
10.0.0.11

pixfirewall(config)#

```
dhcpd option 66 ascii {server_name | server_ip_str}
```

- Distributes TFTP server for IP Phone connections.

```
dhcpd option 150 ip server_ip1 [server_ip2]
```

- Distributes list of TFTP servers for IP Phone connections.

```
pix1(config)# dhcpd option 150 ip 10.0.0.11
pix1(config)# dhcpd option 66 ip 10.0.0.11
```

CSPFA 3.2—16-27

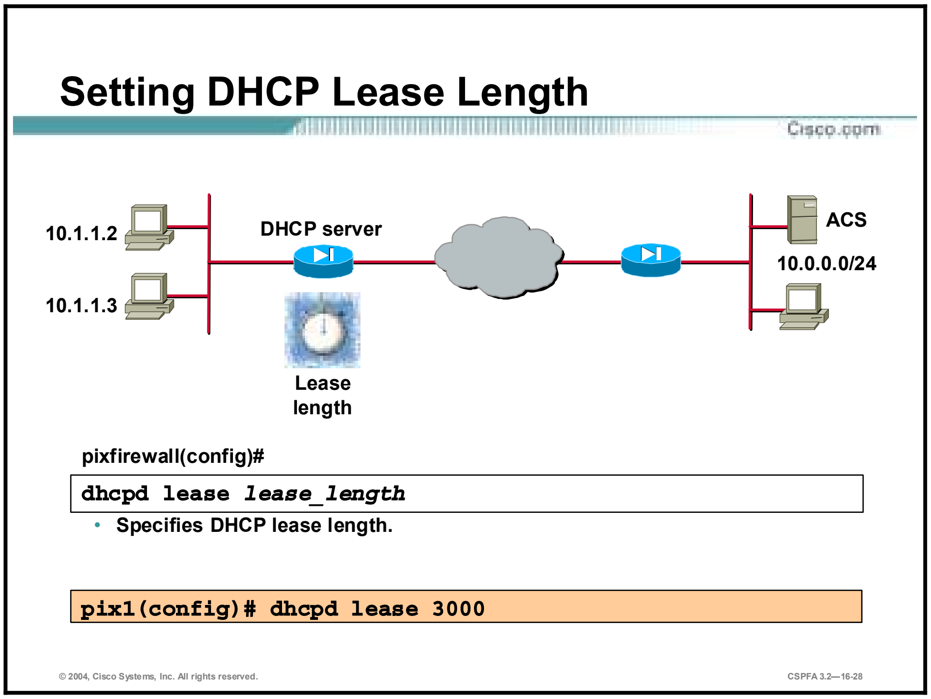
The **dhcpd option** commands enable the PIX Firewall's DHCP server to distribute the IP address of a TFTP server to serve DHCP clients. These options are useful for IP Phones, which may need to obtain configuration files from a TFTP server. With the **dhcpd option 66** command, the PIX Firewall distributes the IP address of a single TFTP server. With the **dhcpd option 150** command, it distributes a list of TFTP servers. You can remove the **dhcpd option** commands by using their **no** forms.

The syntax for the **dhcpd option** commands is as follows:

dhcpd option 66 ascii {*server_name* | *server_ip_str*}

dhcpd option 150 ip *server_ip1* [*server_ip2*]

server_ip(1,2)	Specifies the IP addresses of a TFTP server.
server_ip_str	Any combination of characters used to identify the server. For example, this could have the form of an IP address, such as 1.1.1.1, but is treated as a character string.
server_name	Specifies an ASCII character string representing the TFTP server.



Setting DHCP Lease Length

pixfirewall(config)#

```
dhcpd lease lease_length
```

- Specifies DHCP lease length.

```
pix1(config)# dhcpd lease 3000
```

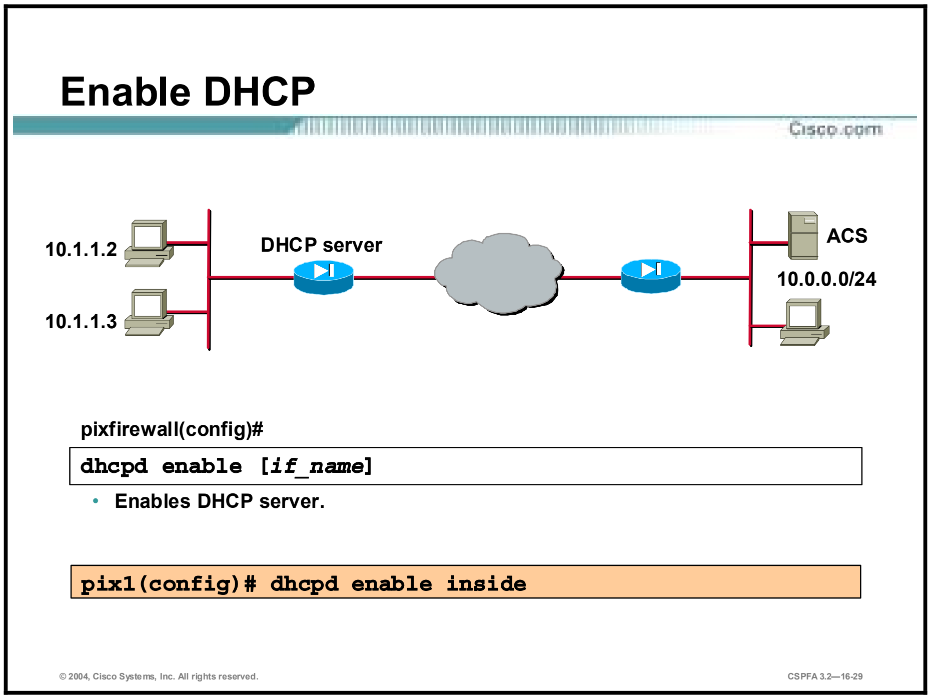
CSPFA 3.2—16-28

The **dhcpd lease** command specifies the amount of time in seconds that the client can use the assigned IP address. The default is 3600 seconds. The minimum lease length is 300 seconds, and the maximum lease length is 2,147,483,647 seconds.

The syntax for the **dhcpd lease** command is as follows:

dhcpd lease *lease_length*

lease ***lease_length***	The length of the lease in seconds granted to the DHCP client from the DHCP server. The lease indicates how long the client can use the assigned IP address. The default is 3,600 seconds. The minimum lease length is 300 seconds, and the maximum is 2,147,483,647 seconds.



Enable DHCP

pixfirewall(config)#

```
dhcpd enable [if_name]
```

- Enables DHCP server.

```
pix1(config)# dhcpd enable inside
```

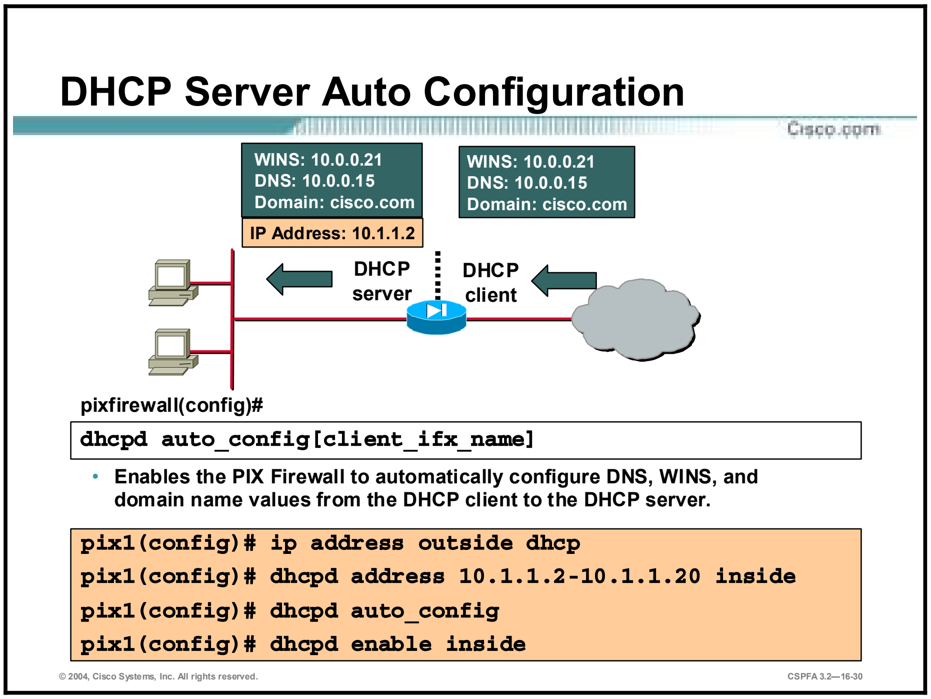
CSPFA 3.2—16-29

Enable the DHCP daemon within the PIX Firewall to listen for DHCP client requests on the enabled interface by executing the **dhcpd enable** command. Currently, you can only enable the DHCP server feature on the inside interface, which is the default. Use the **no** form of the command to disable the DHCP daemon.

The syntax for the **dhcpd enable** command is as follows:

dhcpd enable [*if_name*]

if_name	Name of the PIX Firewall interface. The default is the inside interface. The DHCP server daemon can only be enabled on the inside interface.



DHCP Server Auto Configuration

Cisco.com

WINS: 10.0.0.21
DNS: 10.0.0.15
Domain: cisco.com

IP Address: 10.1.1.2

WINS: 10.0.0.21
DNS: 10.0.0.15
Domain: cisco.com

DHCP server

DHCP client

pixfirewall(config)#

```
dhcpd auto_config[client_ifx_name]
```

- Enables the PIX Firewall to automatically configure DNS, WINS, and domain name values from the DHCP client to the DHCP server.

```
pix1(config)# ip address outside dhcp
pix1(config)# dhcpd address 10.1.1.2-10.1.1.20 inside
pix1(config)# dhcpd auto_config
pix1(config)# dhcpd enable inside
```

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—16-30

The PIX Firewall can be a DHCP server, a DHCP client, or a DHCP server and client simultaneously. DHCP server and client support enables you to automatically leverage the DNS, WINS, and domain name values obtained by the PIX Firewall DHCP client for use by the hosts served by the PIX Firewall’s DHCP server.

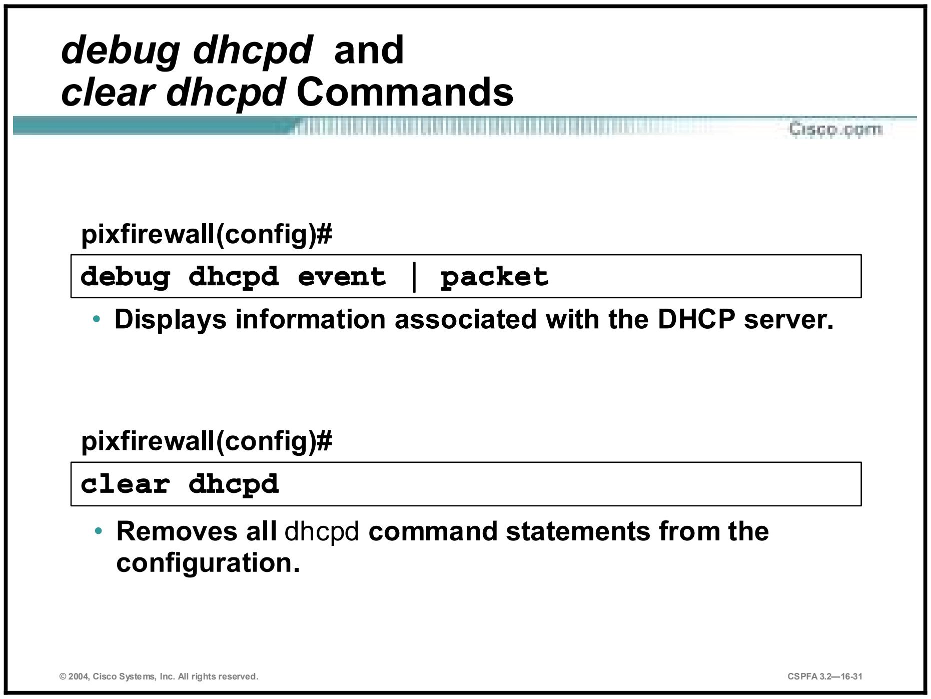
Use the **dhcpd auto_config** command to enable the PIX Firewall to automatically pass configuration parameters it receives from a DHCP server to its own DHCP clients. DHCP must be enabled with the **dhcpd enable** command in order to use the **dhcpd auto_config** command. Use the **no dhcpd auto_config** command to disable the auto_config feature.

The syntax for the **dhcp auto_config** command is as follows:

dhcpd auto_config [*client_ifx_name*]

no dhcpd auto_config [*client_ifx_name*]

auto_config	Enables the PIX Firewall to automatically configure DNS, WINS, and domain name values from the DHCP client to the DHCP server.
client_ifx_name	Supports only the outside interface at this time. When more interfaces are supported, this argument will specify which interface supports the DHCP auto_config feature.



debug dhcpd and *clear dhcpd* Commands

pixfirewall(config)#

```
debug dhcpd event | packet
```

- **Displays information associated with the DHCP server.**

pixfirewall(config)#

```
clear dhcpd
```

- **Removes all** dhcpd **command statements from the configuration.**

CSPFA 3.2—16-31

The **debug dhcpd** command displays information associated with the DHCP server. Use the **debug dhcpd event** command to display event information about the DHCP server, and use the **debug dhcpd packet** command to display packet information about the DHCP server. Use the **no** form of the **debug dhcpd** command to disable debugging.

The syntax for the **debug dhcpd** command is as follows:

debug dhcpd event | packet

dhcpd event	Displays event information associated with the DHCP server.
dhcpd packet	Displays packet information associated with the DHCP server.

The **clear dhcpd** command can be used to clear all **dhcpd** commands, and binding and statistics information. Use the **clear dhcpd** command with no options to remove all **dhcpd** command statements from the configuration.

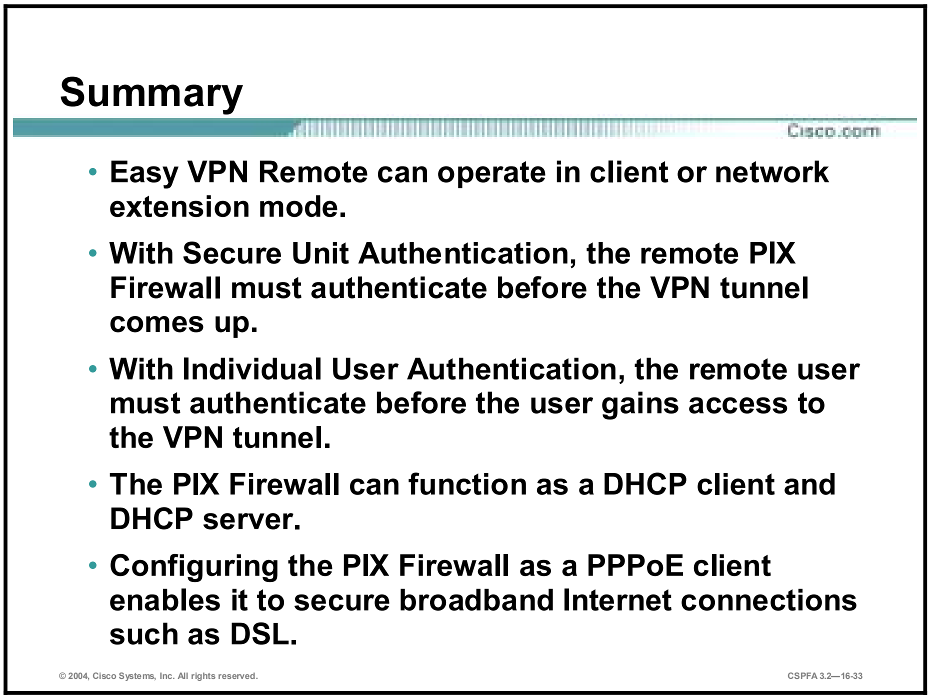
The syntax for the **clear dhcpd** command is as follows:

clear dhcpd [binding | statistics]

bindings	The binding information for a given server IP address and its associated client hardware address and lease length.
statistics	Statistical information, such as address pool, number of bindings, malformed messages, sent messages, and received messages.

Summary

This topic summarizes what you learned in this lesson.



Summary

Cisco.com

- Easy VPN Remote can operate in client or network extension mode.
- With Secure Unit Authentication, the remote PIX Firewall must authenticate before the VPN tunnel comes up.
- With Individual User Authentication, the remote user must authenticate before the user gains access to the VPN tunnel.
- The PIX Firewall can function as a DHCP client and DHCP server.
- Configuring the PIX Firewall as a PPPoE client enables it to secure broadband Internet connections such as DSL.

CSPFA 3.2—16-33

17

System Maintenance

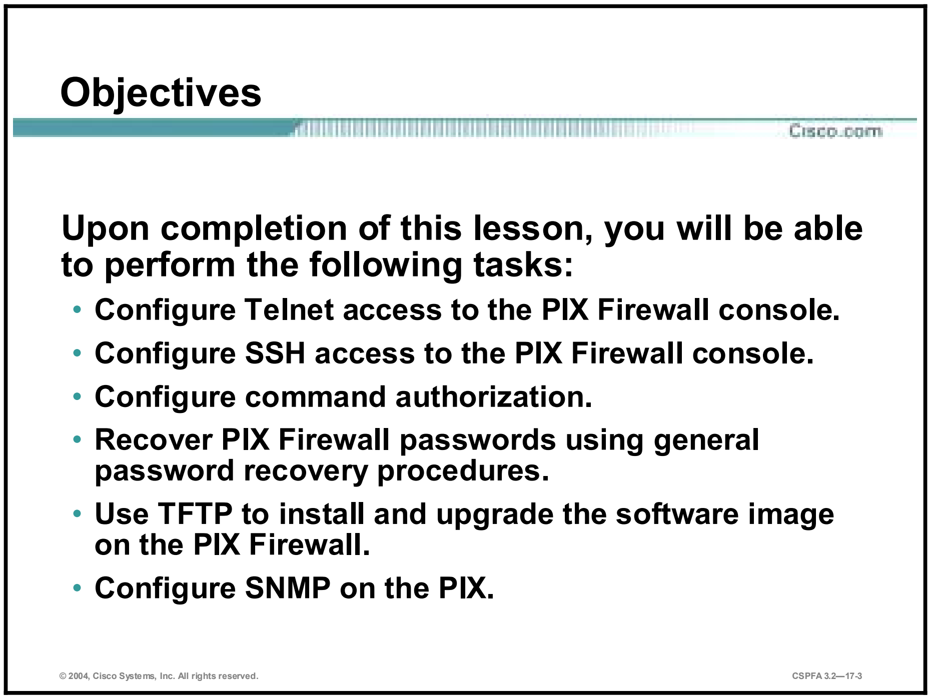
Overview

This lesson includes the following topics:

- Objectives
- Remote access
- Command authorization
- SNMP
- Management tools
- Activation keys
- Password recovery and image upgrade
- Summary
- Lab exercise

Objectives

This topic lists the lesson’s objectives.



Objectives

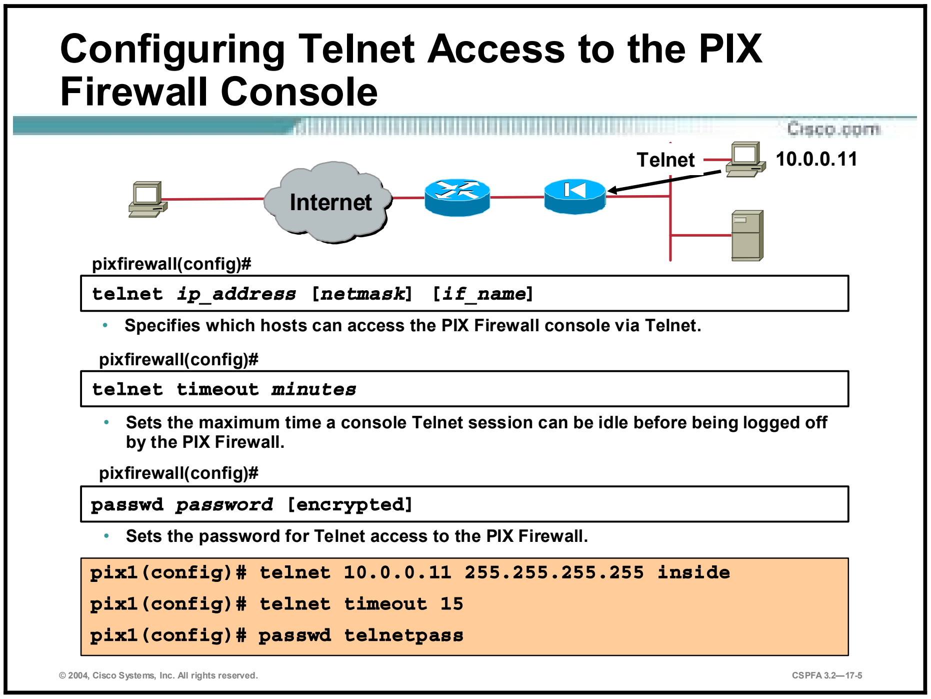
Upon completion of this lesson, you will be able to perform the following tasks:

- **Configure Telnet access to the PIX Firewall console.**
- **Configure SSH access to the PIX Firewall console.**
- **Configure command authorization.**
- **Recover PIX Firewall passwords using general password recovery procedures.**
- **Use TFTP to install and upgrade the software image on the PIX Firewall.**
- **Configure SNMP on the PIX.**

CSPFA 3.2—17-3

Remote Access

This topic explains the how to configure a secure remote connection to your PIX Firewall's console.



Configuring Telnet Access to the PIX Firewall Console

pixfirewall(config)#

```
telnet ip_address [netmask] [if_name]
```

- Specifies which hosts can access the PIX Firewall console via Telnet.

pixfirewall(config)#

```
telnet timeout minutes
```

- Sets the maximum time a console Telnet session can be idle before being logged off by the PIX Firewall.

pixfirewall(config)#

```
passwd password [encrypted]
```

- Sets the password for Telnet access to the PIX Firewall.

```
pix1(config)# telnet 10.0.0.11 255.255.255.255 inside
pix1(config)# telnet timeout 15
pix1(config)# passwd telnetpass
```

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—17-5

The serial console permits a single user to configure the PIX Firewall, but often this is not convenient for a site with more than one administrator. By configuring console access via Telnet, you can enable up to five hosts or networks to access the PIX Firewall console simultaneously.

You can enable Telnet to the PIX Firewall on all interfaces. However, the PIX Firewall requires that all Telnet traffic to the outside interface be IPSec protected. To enable a Telnet session to the outside interface, configure IPSec on the outside interface to include IP traffic generated by the PIX Firewall, and enable Telnet on the outside interface.

The following are the Telnet configuration commands:

- **telnet**—Specifies which hosts can access the PIX Firewall console via Telnet. Up to 16 hosts or networks can be specified.
- **telnet timeout**—Sets the maximum time a console Telnet session can be idle before being logged off by the PIX Firewall. The default is five minutes.
- **passwd**—Sets the password for Telnet access to the PIX Firewall. The default value is cisco. The **show passwd** command displays the Telnet password, and the **clear passwd** command removes it.

The syntaxes for the Telnet configuration commands are as follows:

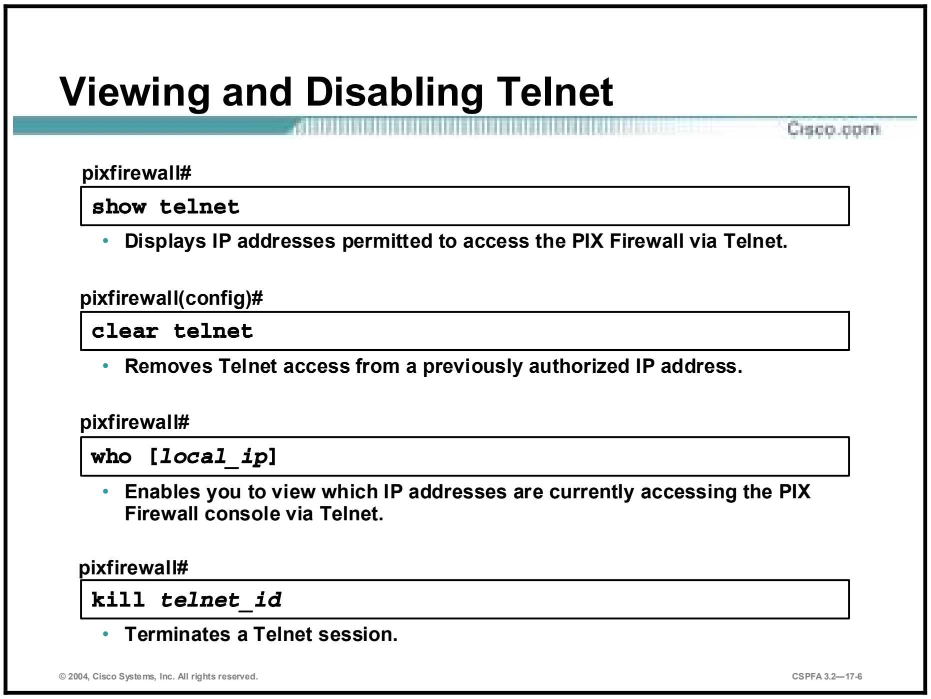
telnet ***ip_address*** **[*****netmask*****] [*****if_name*****]**

telnet timeout ***minutes***

passwd ***password*** **[encrypted]**

ip_address	An IP address of a host or network that can access the PIX Firewall Telnet console. If an interface name is not specified, the address is assumed to be on an internal interface. The PIX Firewall automatically verifies the IP address against the IP addresses specified by the **ip address** commands to ensure that the address you specify is on an internal interface. If an interface name is specified, the PIX Firewall only checks the host against the interface you specify.
netmask	The bit mask of *ip_address*. To limit access to a single IP address, use 255 in each octet (for example, 255.255.255.255). If you do not specify the netmask, it defaults to 255.255.255.255 regardless of the class of *local_ip*. Do not use the subnetwork mask of the internal network. The netmask is only a bit mask for the IP address in *ip_address*.
if_name	If IPSec is operating, the PIX Firewall enables you to specify an unsecure interface name, typically, the outside interface. At a minimum, the **crypto map** command must be configured to specify an interface name with the **telnet** command.
minutes	The number of minutes that a Telnet session can be idle before being closed by the PIX Firewall. The default is 5 minutes. The range is 1 to 60 minutes.
password	A case-sensitive password of up to 16 alphanumeric and special characters. Any character can be used in the password except a question mark and a space.
encrypted	Specifies that the password you entered is already encrypted. The password you specify with the encrypted option must be 16 characters in length.

In the figure, host 10.0.0.11 on the internal interface is allowed to access the PIX Firewall console via Telnet using the password telnetpass. If the Telnet session is idle more than fifteen minutes, the PIX Firewall closes it.



Viewing and Disabling Telnet

pixfirewall#

```
show telnet
```

- Displays IP addresses permitted to access the PIX Firewall via Telnet.

pixfirewall(config)#

```
clear telnet
```

- Removes Telnet access from a previously authorized IP address.

pixfirewall#

```
who [local_ip]
```

- Enables you to view which IP addresses are currently accessing the PIX Firewall console via Telnet.

pixfirewall#

```
kill telnet_id
```

- Terminates a Telnet session.

CSPFA 3.2—17-6

The following commands enable you to view and clear Telnet configuration and Telnet sessions:

- **show telnet**—Displays the current list of IP addresses authorized to access the PIX Firewall via Telnet.
- **clear telnet** and **no telnet**—Remove Telnet access from a previously authorized IP address.
- **who**—Enables you to view which IP addresses are currently accessing the PIX Firewall console via Telnet.
- **kill**—Terminates a Telnet session. When you kill a Telnet session, the PIX Firewall lets any active commands terminate and then drops the connection without warning the user.

The syntaxes for these commands are as follows:

show telnet

clear telnet [*ip_address* [*netmask*] [*if_name*]]

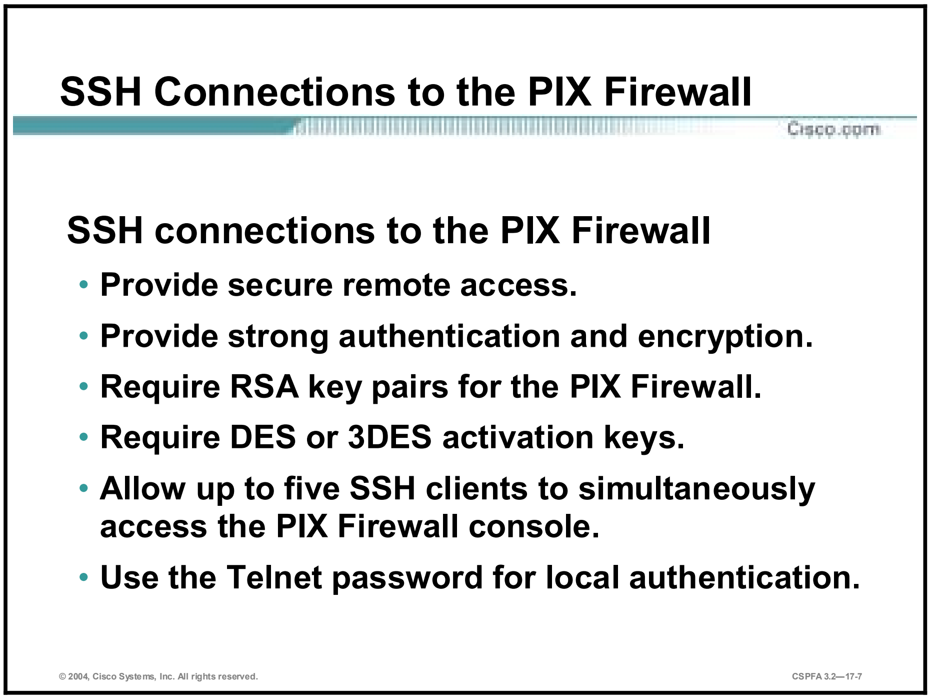
no telnet [*ip_address* [*netmask*] [*if_name*]]

who [*local_ip*]

kill *telnet_id*

ip_address	An IP address of a host or network that can access the PIX Firewall Telnet console. If an interface name is not specified, the address is assumed to be on an internal interface. The PIX Firewall automatically verifies the IP address against the IP addresses specified by the **ip address** commands to ensure that the address you specify is on an internal interface. If an interface name is specified, the PIX Firewall only checks the host against the interface you specify.

netmask	The bit mask of *ip_address*. To limit access to a single IP address, use 255 in each octet (for example, 255.255.255.255). If you do not specify the netmask, it defaults to 255.255.255.255 regardless of the class of *local_ip*. Do not use the subnetwork mask of the internal network. The netmask is only a bit mask for the IP address in *ip_address*.
if_name	If IPSec is operating, the PIX Firewall enables you to specify an unsecure interface name, typically, the outside interface. At a minimum, the **crypto map** command must be configured to specify an interface name with the **telnet** command.
local_ip	An optional internal IP address to limit the listing to one IP address or to a network IP address.
telnet_id	The Telnet session identification.



SSH Connections to the PIX Firewall

SSH connections to the PIX Firewall

- Provide secure remote access.
- Provide strong authentication and encryption.
- Require RSA key pairs for the PIX Firewall.
- Require DES or 3DES activation keys.
- Allow up to five SSH clients to simultaneously access the PIX Firewall console.
- Use the Telnet password for local authentication.

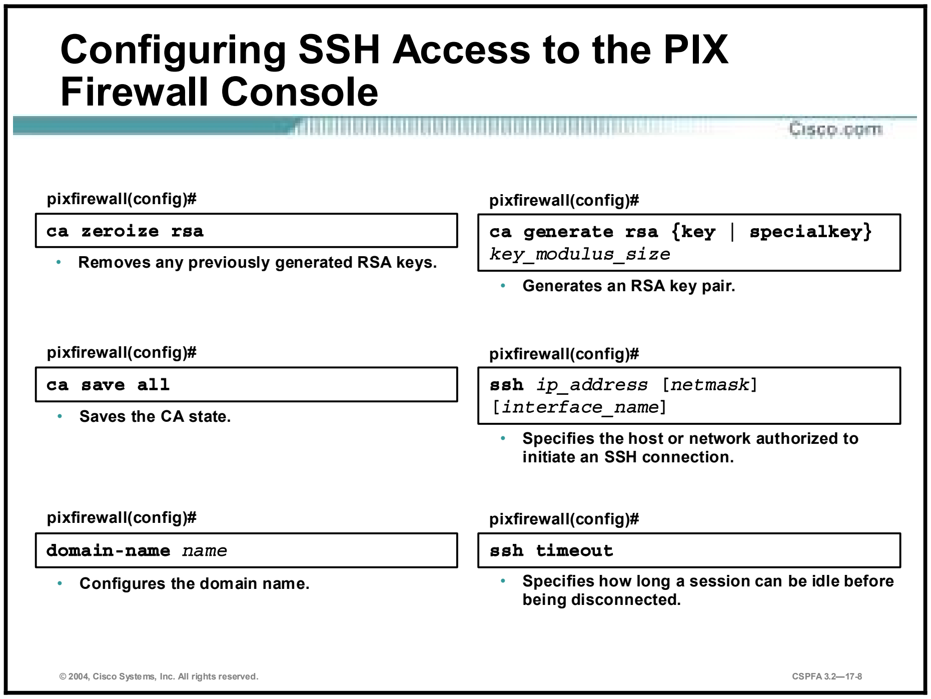
CSPFA 3.2—17-7

Secure Shell (SSH) provides another option for remote management of the PIX Firewall. SSH provides a higher degree of security than Telnet, which provides lower-layer encryption and application security. The PIX Firewall supports the SSH remote functionality, as provided in SSH version 1, which provides strong authentication and encryption capabilities. SSH, an application running on top of a reliable transport layer such as TCP, supports logging onto another computer over a network, executing commands remotely, and moving files from one host to another.

Both ends of an SSH connection are authenticated, and passwords are protected by being encrypted. Since SSH uses Rivest, Shamir, and Adleman (RSA) public key cryptography, an Internet encryption and authentication system, you must generate an RSA key pair for the PIX Firewall before clients can connect to the PIX Firewall console. Your PIX Firewall must also have a Data Encryption Standard (DES) or Triple-Data Encryption Standard (3DES) activation key.

The PIX Firewall allows up to five SSH clients to simultaneously access its console. You can define specific hosts or networks that are authorized to initiate an SSH connection to the PIX Firewall, as well as how long a session can remain idle before being disconnected.

Note The PIX Firewall SSH implementation provides a secure remote shell session without IPSec, and only functions as a server, which means that the PIX Firewall cannot initiate SSH connections.



Configuring SSH Access to the PIX Firewall Console

pixfirewall(config)#

```
ca zeroize rsa
```

- Removes any previously generated RSA keys.

pixfirewall(config)#

```
ca generate rsa {key | specialkey}
key_modulus_size
```

- Generates an RSA key pair.

pixfirewall(config)#

```
ca save all
```

- Saves the CA state.

pixfirewall(config)#

```
ssh ip_address [netmask]
[interface_name]
```

- Specifies the host or network authorized to initiate an SSH connection.

pixfirewall(config)#

```
domain-name name
```

- Configures the domain name.

pixfirewall(config)#

```
ssh timeout
```

- Specifies how long a session can be idle before being disconnected.

CSPFA 3.2—17-8

Complete the following steps to configure an SSH connection to your PIX Firewall:

Step 1 Obtain an SSH client and install it on the system from which you want to establish the SSH connection. The following SSH v1.x clients are available for download:

- Tera Term Pro SSH v1.x client—For Windows 3.1, Windows CE, Windows 95, and Windows NT 4.0. You also need to download the Tera Term Secure Shell (TTSSH) security enhancement for Tera Term Pro. TTSSH provides a zip file that you copy to your system. Extract the zipped files into the same folder in which you installed Tera Term Pro.
- SSH v1.x client for Linux, Solaris, OpenBSD, AIX, IRIX, HP/UX, FreeBSD, and NetBSD.
- Nifty Telnet 1.1 SSH client—For Macintosh.

Note SSH v1.x and v2 are entirely different protocols and are not compatible. Make sure that you download a client that supports SSH v1.x.

Step 2 Use the **ca zeroize rsa** command to delete any previously created RSA keys.

Step 3 Use the **ca save all** command to save the Certificate Authority (CA) state and complete the erasure of the old RSA key pair.

Step 4 Use the **domain-name** command to configure the domain name.

Step 5 Use the **ca generate rsa key** command to generate an RSA key pair to use to encrypt SSH sessions.

Step 6 Use the **ca save all** command to save the keys to Flash memory.

Step 7 Use the **ssh ip_address** command to specify the host or network authorized to initiate an SSH connection to the PIX Firewall. Use the no keyword to remove this command from the configuration.

Step 8 Use the **ssh timeout** command to specify the duration in minutes that a session can be idle before being disconnected. The default duration is five minutes.

Note The password used to perform local authentication is the same as the one used for Telnet access. The default for this password is cisco. To change it, use the **passwd** command.

The syntaxes for the **ca** commands are as follows:

ca zeroize rsa

ca save all

ca generate rsa key ***modulus***

modulus	The size of the key modulus, which is between 512 and 2048 bits. Choosing a size greater than 1024 bits may cause key generation to take a few minutes.

The syntax for the **domain-name** command is as follows:

domain-name ***name***

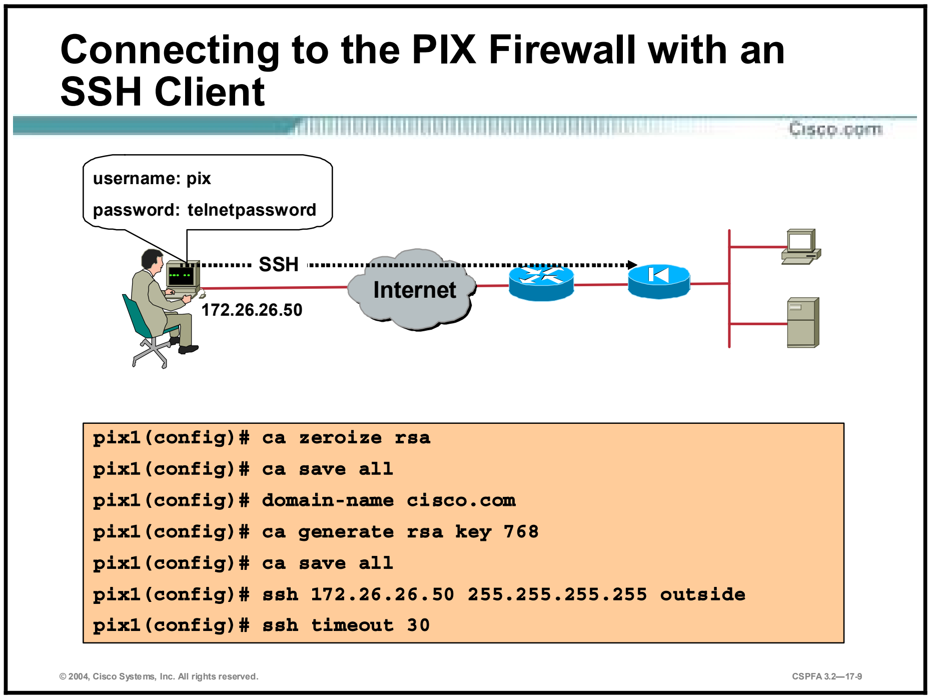
name	The name of the domain.

The syntaxes for the **ssh** configuration commands are as follows:

ssh ***ip_address*** **[*netmask*] [*interface_name*]**

ssh timeout ***mm***

ip_address	IP address of the host or network authorized to initiate an SSH connection to the PIX Firewall.
netmask	Network mask for ip_address. If you do not specify a netmask, the default is 255.255.255.255 regardless of the class of *ip_address*.
interface_name	PIX Firewall interface name on which the host or network initiating the SSH connection resides.
mm	The duration in minutes that a session can be idle before being disconnected. The default duration is 5 minutes. The allowable range is from 1 to 60 minutes.



Connecting to the PIX Firewall with an SSH Client

Cisco.com
username: pix
password: telnetpassword
SSH
172.26.26.50
Internet

```
pix1(config)# ca zeroize rsa
pix1(config)# ca save all
pix1(config)# domain-name cisco.com
pix1(config)# ca generate rsa key 768
pix1(config)# ca save all
pix1(config)# ssh 172.26.26.50 255.255.255.255 outside
pix1(config)# ssh timeout 30
```

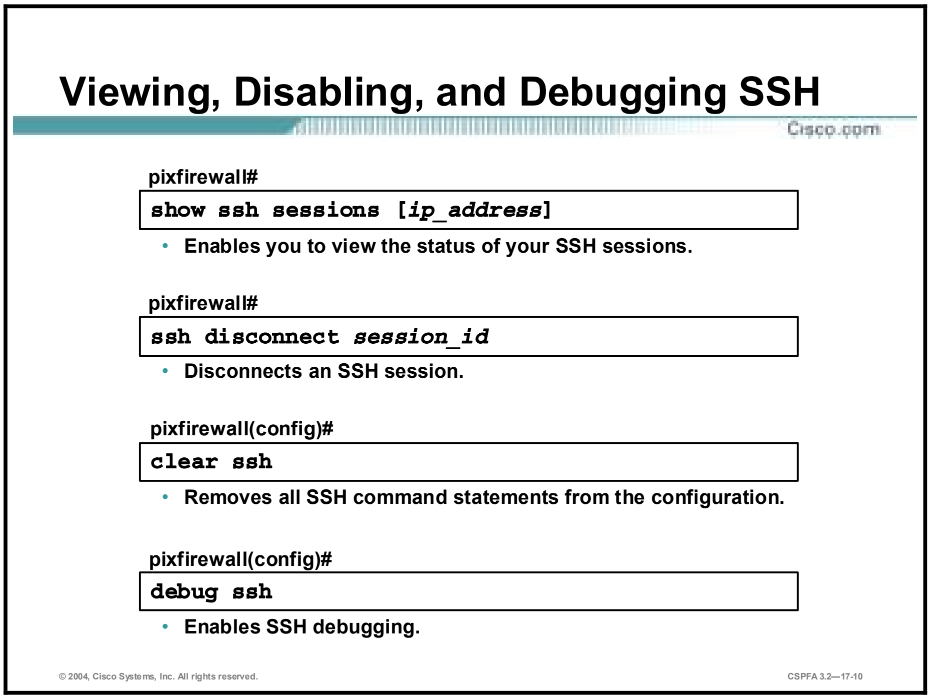
CSPFA 3.2—17-9

To establish an SSH connection to your PIX Firewall console, enter the username **pix** and the Telnet password at the SSH client. When starting an SSH session, the PIX Firewall displays a dot (.) on the console before the SSH user authentication prompt appears, as follows:

```
pixfirewall(config)# .
```

The display of the dot does not affect the functionality of SSH. The dot appears at the console when generating a server key or decrypting a message using private keys during SSH key exchange before user authentication occurs. These tasks can take up to two minutes or longer. The dot is a progress indicator that verifies that the PIX Firewall is busy and has not hung.

In the figure, an RSA key pair is generated for the PIX Firewall using the default key modulus size of 768. Host 172.26.26.50 is authorized to initiate an SSH connection to the PIX Firewall.



Viewing, Disabling, and Debugging SSH

pixfirewall#

```
show ssh sessions [ip_address]
```

- Enables you to view the status of your SSH sessions.

pixfirewall#

```
ssh disconnect session_id
```

- Disconnects an SSH session.

pixfirewall(config)#

```
clear ssh
```

- Removes all SSH command statements from the configuration.

pixfirewall(config)#

```
debug ssh
```

- Enables SSH debugging.

CSPFA 3.2—17-10

Use the **show ssh sessions** command to list all active SSH sessions on the PIX Firewall. The **ssh disconnect** command enables you to disconnect a specific session.

The **show ssh sessions** command provides the following display:

```
Session ID      Client IP       Version Encryption     State
Username
    0           172.16.25.15    1.5     3DES           4       -
    1           172.26.26.50    1.5     DES            6       pix
    2           172.16.25.11    1.5     3DES           4       -
```

The following are the different parts of the **show ssh sessions** command display:

- Session ID column—A unique number that identifies an SSH session.
- Client IP column—The IP address of the system running an SSH client.
- Version column—Lists the protocol version number that the SSH client supports.
- Encryption column—Lists the type of encryption the SSH client is using.
- State column—Lists the progress the client is making as it interacts with the PIX Firewall.
- Username column—Lists the login username that has been authenticated for the session. The “pix” username appears when authentication other than authentication, authorization, and accounting (AAA) is used.

Use the **clear ssh** command to remove all **ssh** command statements from the configuration, and use the **no ssh** command to remove selected **ssh** command statements. The **debug ssh** command displays information and error messages associated with the **ssh** command.

The syntaxes for these SSH commands are as follows:

show ssh sessions [*ip_address*]

ssh disconnect *session_id*

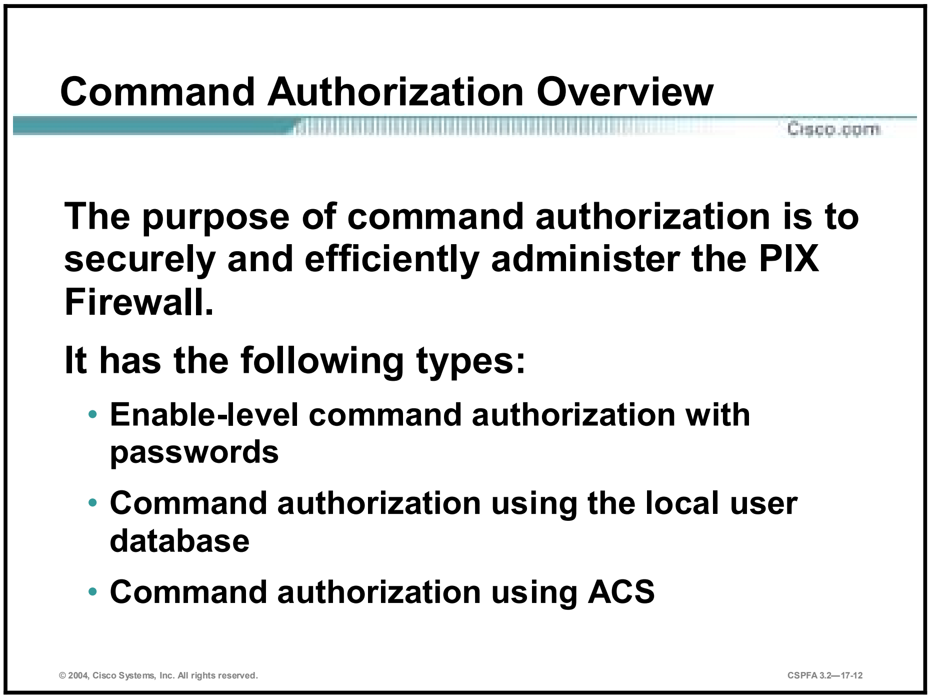
clear ssh

ssh debug

ip_address	The IP address of a system running an SSH client.
session_id	Identifier for the specific session that you want to disconnect.

Command Authorization

This topic explains how to configure and use local user authentication and command authorization.



Command Authorization Overview

The purpose of command authorization is to securely and efficiently administer the PIX Firewall.

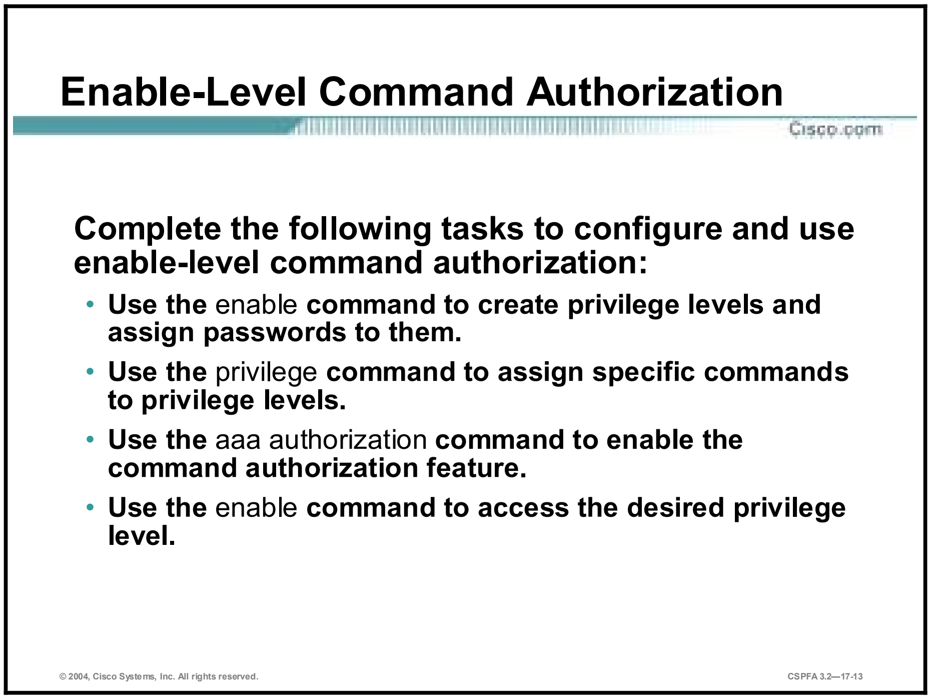
It has the following types:

- **Enable-level command authorization with passwords**
- **Command authorization using the local user database**
- **Command authorization using ACS**

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—17-12

Command authorization is a way of facilitating and controlling administration of the PIX Firewall. There are three types of command authorizations that can be used to control which users execute certain commands:

- Enable-level command authorization with passwords
- Command authorization using the local user database
- Command authorization using Access Control Server (ACS)



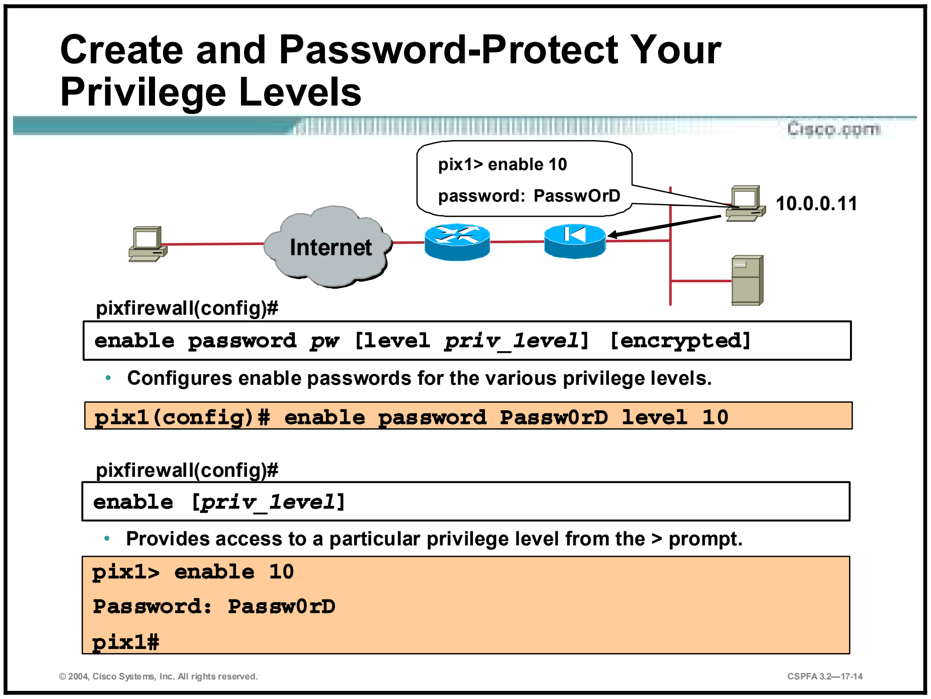
Enable-Level Command Authorization

Complete the following tasks to configure and use enable-level command authorization:

- **Use the** enable **command to create privilege levels and assign passwords to them.**
- **Use the** privilege **command to assign specific commands to privilege levels.**
- **Use the** aaa authorization **command to enable the command authorization feature.**
- **Use the** enable **command to access the desired privilege level.**

CSPFA 3.2—17-13

The first type of command authorization, enable level with passwords, allows you to use the **enable** command with the priv_level option to access a PIX Firewall privilege level and then use any command assigned to that privilege level or to a lower privilege level. To configure this type of command authorization, you must create and password-protect your privilege levels, assign privilege levels to commands, and enable the command authorization feature.



Create and Password-Protect Your Privilege Levels

Cisco.com
pix1> enable 10
password: PasswOrD
10.0.0.11
Internet

pixfirewall(config)#

```
enable password pw [level priv_1evel] [encrypted]
```

- Configures enable passwords for the various privilege levels.

```
pix1(config)# enable password Passw0rD level 10
```

pixfirewall(config)#

```
enable [priv_1evel]
```

- Provides access to a particular privilege level from the > prompt.

```
pix1> enable 10
Password: Passw0rD
pix1#
```

CSPFA 3.2—17-14

The PIX Firewall supports up to sixteen privilege levels: levels zero through fifteen. You can create privilege levels and secure them by using the **enable password** command. You can then gain access to a particular privilege level from the > prompt by entering the **enable** command with a privilege level designation and entering the password for that level when prompted.

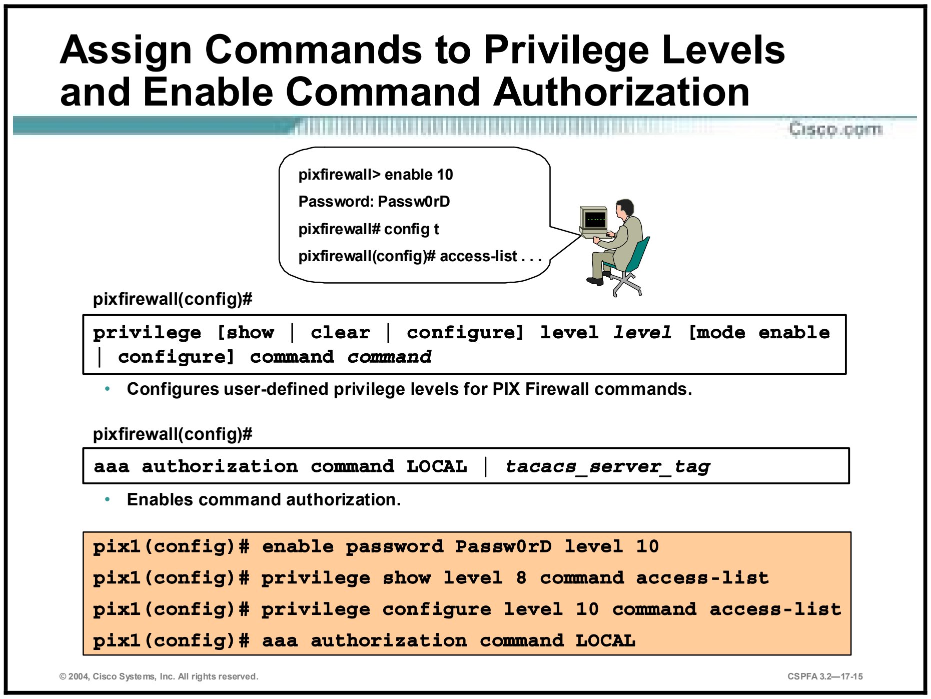
After you are inside a privilege level, you can execute the commands assigned to that level as well as commands assigned to lower privilege levels. For example, from privilege level 15, you can execute every command because this is the highest privilege level. If you do not specify a privilege level when entering enable mode, the default of 15 is used. Therefore, creating a strong password for level 15 is important.

The syntaxes for the **enable** commands when used with command authorization are as follows:

enable [*priv_1evel*]

enable password *pw* [level *priv_1evel*] [encrypted]

priv_level	Enables a privilege level, from 0 to 15.
pw	The privilege level password string.
encrypted	Specifies that the provided password is already encrypted.



Assign Commands to Privilege Levels and Enable Command Authorization

Cisco.com

```
pixfirewall> enable 10
Password: Passw0rD
pixfirewall# config t
pixfirewall(config)# access-list . . .
```

pixfirewall(config)#

```
privilege [show | clear | configure] level level [mode enable
| configure] command command
```

- Configures user-defined privilege levels for PIX Firewall commands.

pixfirewall(config)#

```
aaa authorization command LOCAL | tacacs_server_tag
```

- Enables command authorization.

```
pix1(config)# enable password Passw0rD level 10
pix1(config)# privilege show level 8 command access-list
pix1(config)# privilege configure level 10 command access-list
pix1(config)# aaa authorization command LOCAL
```

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—17-15

To assign commands to privilege levels, use the **privilege** command. Replace the level argument with the privilege level, and replace the command argument with the command you want to assign to the specified level. You can use the show, clear, or configure parameter to optionally set the privilege level for the show, clear, or configure command modifiers of the specified command. The **privilege** command can be removed by using the no keyword.

In the figure, privilege levels are set for the different command modifiers of the **access-list** command. The first **privilege** command entry sets the privilege level of **show access-list** (the show modifier of command **access-list**) to 8. The second **privilege** command entry sets the privilege level of the configure modifier to 10. The **aaa authorization command LOCAL** command is then used to enable command authorization. The user knows the highest privilege level to which the **access-list** command is assigned and also knows the level's password. The user is therefore able to view and create access control lists (ACLs) by entering level 10.

Use the **privilege** command without a show, clear, or configure parameter to set the privilege level for all the modifiers of the command. For example, to set the privilege level of all modifiers of the **access-list** command to a single privilege level of 10, enter the following command:

```
privilege level 10  command access-list
```

For commands that are available in multiple modes, use the mode parameter to specify the mode in which the privilege level applies. Do not use the mode parameter for commands that are not mode-specific.

The syntax of the **privilege** command is as follows:

privilege [show | clear | configure] level *level* [mode enable | configure] command *command*

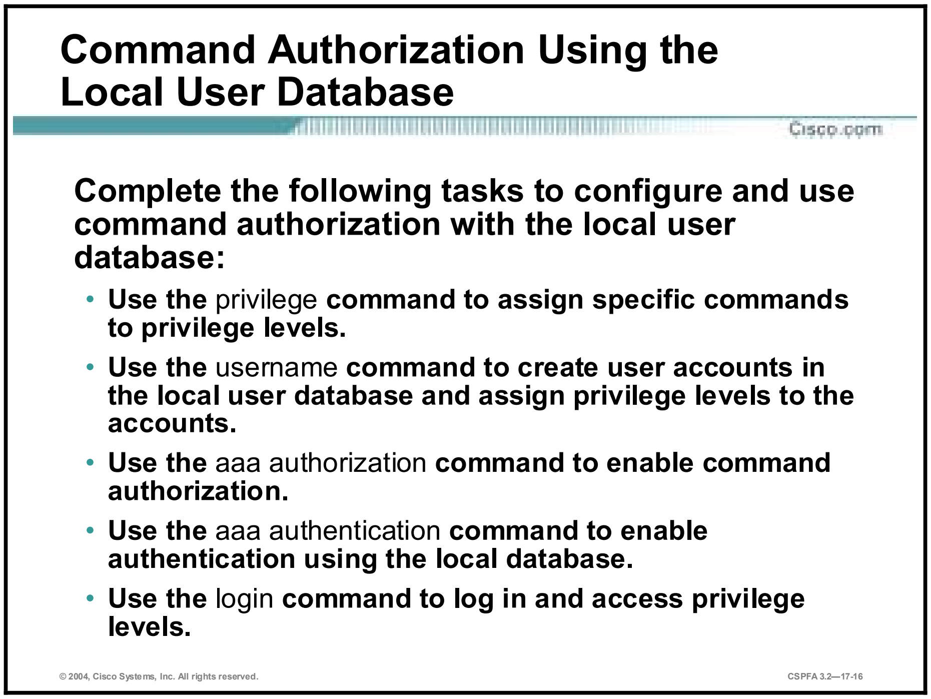
show	Sets the privilege level for the **show** command corresponding to the command specified.

clear	Sets the privilege level for the **clear** command corresponding to the command specified.
configure	Sets the privilege level for the **configure** command corresponding to the command specified.
level	Specifies the privilege level, from 0 to 15. (Lower numbers are lower privilege levels).
mode	Specifies the mode in which the privilege level applies.
enable	For commands with both enable and configure modes, this indicates that the level is for the enable mode of the command.
configure	Sets the privilege level for the **configure** command corresponding to the command specified.
command	The command on which to set the privilege level.

The syntax for the **aaa authorization** command for use with command authorization is as follows:

aaa authorization command LOCAL | ***tacacs_server_tag***

command	Specifies command authorization.
LOCAL	Specifies that the PIX Firewall local user database is used for local command authorization (using privilege levels).
tacacs_server_tag	Specifies that a Terminal Access Controller Access Control System (TACACS) user authentication server is used.



Command Authorization Using the Local User Database

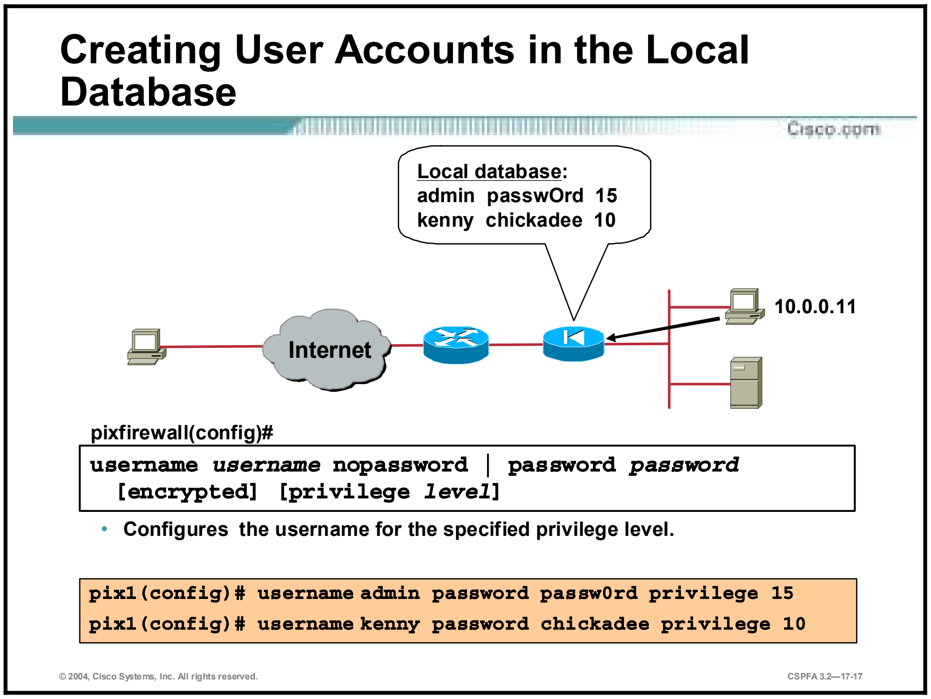
Cisco.com

Complete the following tasks to configure and use command authorization with the local user database:

- **Use the** privilege **command to assign specific commands to privilege levels.**
- **Use the** username **command to create user accounts in the local user database and assign privilege levels to the accounts.**
- **Use the** aaa authorization **command to enable command authorization.**
- **Use the** aaa authentication **command to enable authentication using the local database.**
- **Use the** login **command to log in and access privilege levels.**

CSPFA 3.2—17-16

A second way of controlling which commands users can execute is to configure command authorization using the local user database. After assigning commands to privilege levels with the **privilege** command, use the **username** command to define user accounts and their privilege levels in the local PIX Firewall user database. You can define as many user accounts as you need. After defining the user accounts, enable command authorization with the **aaa authorization** command. To enable a direct username and password prompt, enable authentication via the local user database by entering the **aaa authentication enable console LOCAL** command.



Creating User Accounts in the Local Database

Cisco.com

Local database:
admin passwOrd 15
kenny chickadee 10

pixfirewall(config)#

```
username username nopassword | password password
  [encrypted] [privilege level]
```

- Configures the username for the specified privilege level.

```
pix1(config)# username admin password passw0rd privilege 15
pix1(config)# username kenny password chickadee privilege 10
```

CSPFA 3.2—17-17

Use the **username** command to create user accounts in the local user database. You can create a password for the user or you can use the nopassword keyword to create a user account with no password. Use the encrypted keyword if the password you are supplying is already encrypted, and use the privilege keyword to assign a privilege level to the user.

To delete an existing user account, use the **no username** command. To remove all the entries from the user database, enter the **clear username** command.

In the figure, the **username** command assigns a privilege level of 15 to the user account admin and a privilege level of 10 to the user account kenny.

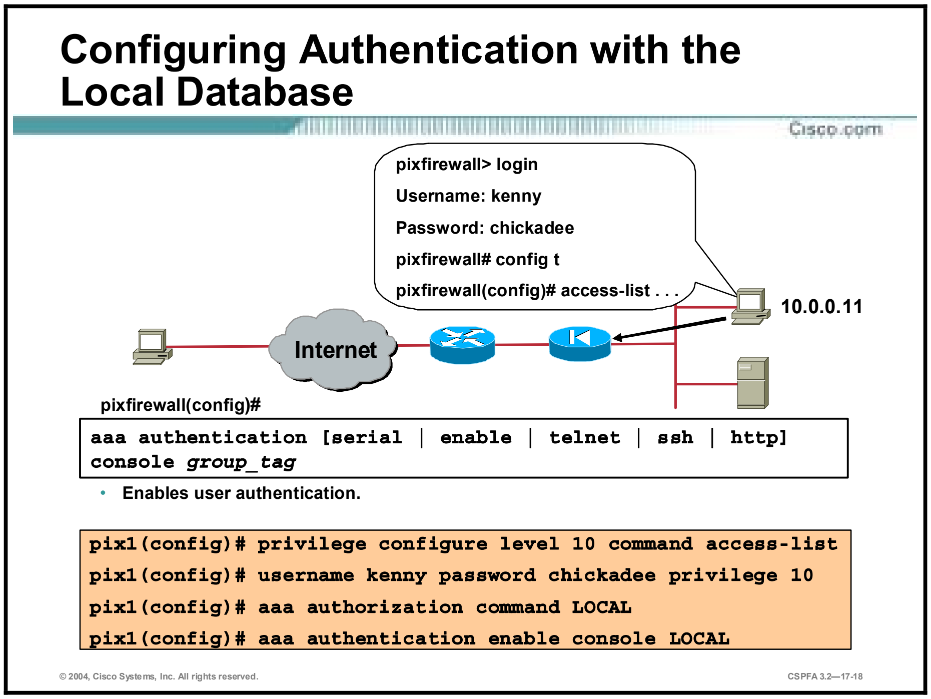
The syntaxes for the **username** commands are as follows:

username ***username*** **nopassword | password** ***password*** **[encrypted] [privilege** ***level*****]**

no username ***username***

clear username

username	The username assigned to the user account.
nopassword	Used to create a user account with no password.
password	The password assigned to the user account.
encrypted	Specifies that the password you are supplying is already encrypted.
level	Specifies the privilege level for the user account.



Configuring Authentication with the Local Database

```
pixfirewall> login
Username: kenny
Password: chickadee
pixfirewall# config t
pixfirewall(config)# access-list . . .
```

pixfirewall(config)#

```
aaa authentication [serial | enable | telnet | ssh | http]
console group_tag
```

- Enables user authentication.

```
pix1(config)# privilege configure level 10 command access-list
pix1(config)# username kenny password chickadee privilege 10
pix1(config)# aaa authorization command LOCAL
pix1(config)# aaa authentication enable console LOCAL
```

CSPFA 3.2—17-18

After you enter the **aaa authentication enable console LOCAL** command, the console prompts for a username and password when you execute the **enable** or **login** command. You can log into the PIX Firewall using the LOCAL internal user database. After you enter the **aaa authentication enable console LOCAL** command, you are no longer able to access the PIX Firewall using the **enable** command with the priv_level option.

When users log into the PIX Firewall, they can enter any command assigned to their privilege level or to lower privilege levels. For example, a user account with a privilege level of 15 can access every command because this is the highest privilege level. A user account with a privilege level of 0 can only access the commands assigned to level 0.

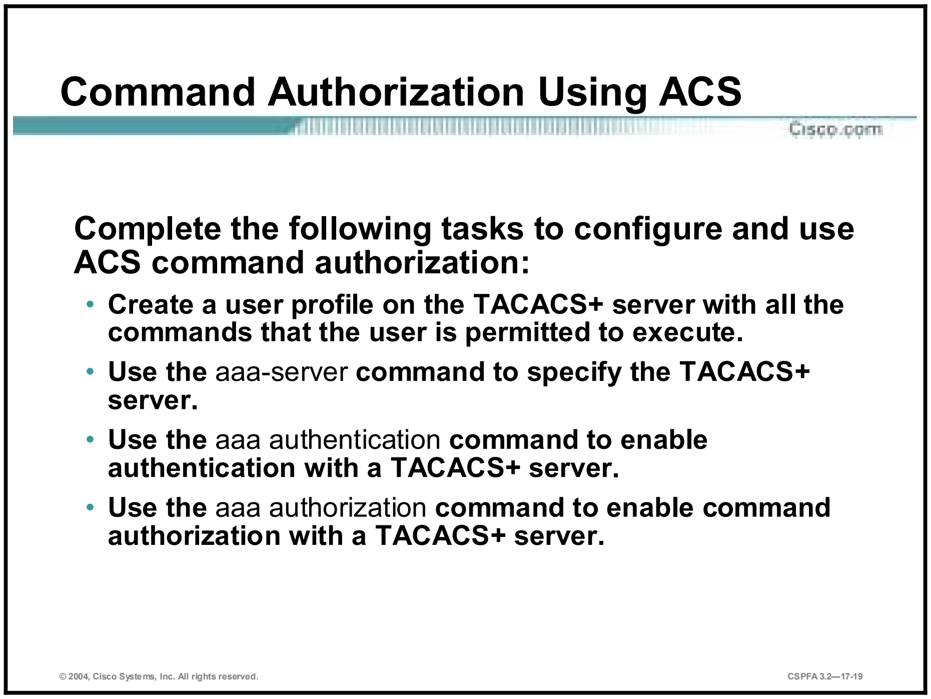
The syntax for the **aaa authentication** command when used with command authorization is as follows:

aaa authentication [serial | enable | telnet | ssh | http] console *group_tag*

serial	Access verification for the PIX Firewall's serial console.
enable	Access verification for the PIX Firewall's privilege mode.
telnet	Access verification for the Telnet access to the PIX Firewall console.
ssh	Access verification for the SSH access to the PIX Firewall console.
http	Access verification for the HTTP access to the PIX Firewall (via PDM).
console	Specifies that access to the PIX Firewall console requires authentication.
group_tag	Enter LOCAL for this parameter to use the local user authentication database.

Note The **logout** command logs you out of the currently logged in user account.

Note The LOCAL database can be used only for controlling access to the PIX Firewall, not for controlling access through the PIX Firewall.



Command Authorization Using ACS

Complete the following tasks to configure and use ACS command authorization:

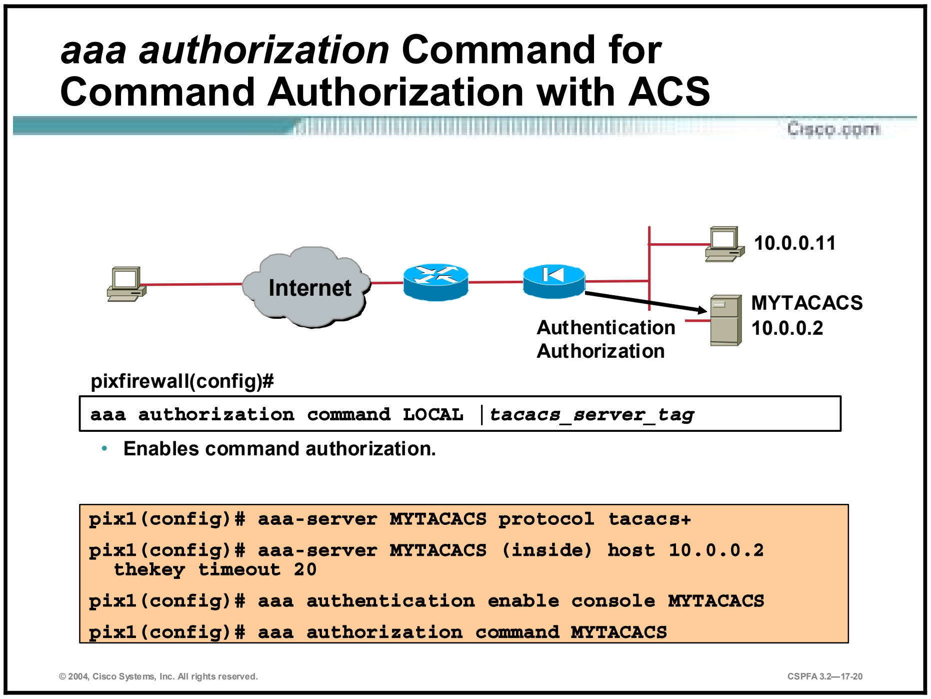
- **Create a user profile on the TACACS+ server with all the commands that the user is permitted to execute.**
- **Use the** aaa-server **command to specify the TACACS+ server.**
- **Use the** aaa authentication **command to enable authentication with a TACACS+ server.**
- **Use the** aaa authorization **command to enable command authorization with a TACACS+ server.**

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—17-19

Only enable authorization with ACS if you are absolutely sure that you have fulfilled the following requirements:

- You have created entries for enable_1, enable_15, and any other levels to which you have assigned commands.
- If you are enabling authentication with usernames:
  - You have a user profile on the Terminal Access Controller Access Control System Plus (TACACS+) server with all the commands that the user is permitted to execute.
  - You have tested authentication with the TACACS+ server.
- You are logged in as a user with the necessary privileges.
- Your TACACS+ system is completely stable and reliable. The necessary level of reliability typically requires that you have a fully redundant TACACS+ server system and fully redundant connectivity to the PIX Firewall.

When configuring the command authorization feature, do not save your configuration until you are sure it works the way you want. If you get locked out of your PIX Firewall, you can usually recover access by simply reloading it. There is more on PIX Firewall lockout later in this lesson.



aaa authorization Command for Command Authorization with ACS

pixfirewall(config)#

```
aaa authorization command LOCAL |tacacs_server_tag
```

- Enables command authorization.

```
pix1(config)# aaa-server MYTACACS protocol tacacs+
pix1(config)# aaa-server MYTACACS (inside) host 10.0.0.2
  thekey timeout 20
pix1(config)# aaa authentication enable console MYTACACS
pix1(config)# aaa authorization command MYTACACS
```

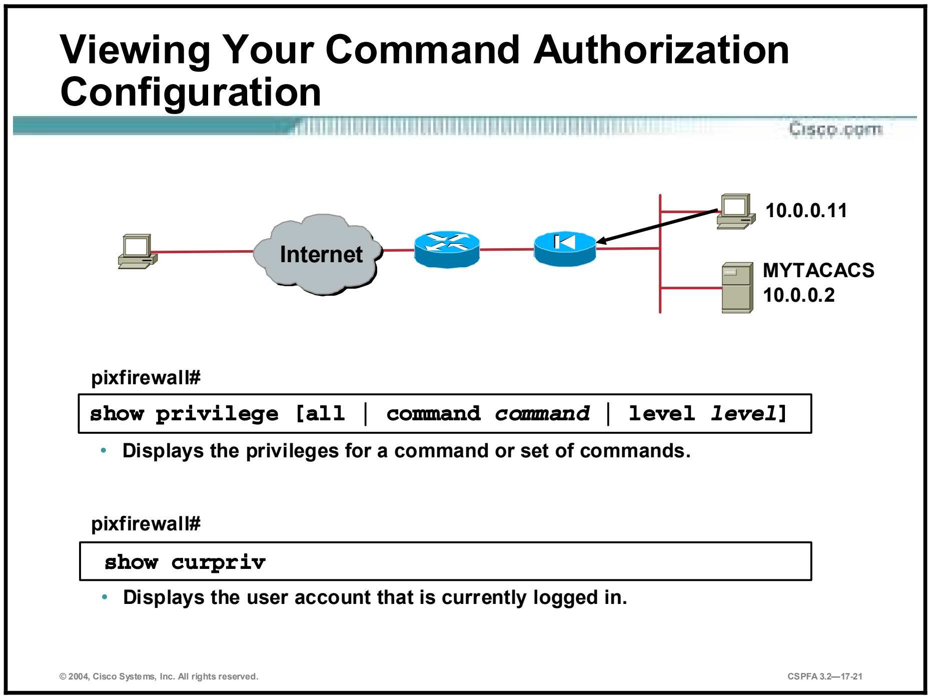
CSPFA 3.2—17-20

Use the **aaa authorization** command to enable command authorization with a TACACS+ server. You must also use the **aaa-server** command to create the tacacs_server_tag.

The syntax for the **aaa authorization** command, when used to configure command authorization using ACS, is as follows:

aaa authorization command LOCAL | ***tacacs_server_tag***

command	Specifies command authorization.
LOCAL	Instructs the PIX Firewall to use its own local user database for command authorization.
tacacs_server_tag	Instructs the PIX Firewall to use the specified TACACS+ server for command authorization.



Viewing Your Command Authorization Configuration

pixfirewall#

```
show privilege [all | command command | level level]
```

- Displays the privileges for a command or set of commands.

pixfirewall#

```
show curpriv
```

- Displays the user account that is currently logged in.

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—17-21

To view the command assignments for each privilege level, use the **show privilege all** command. The system displays the current assignment of each command line interface (CLI) command to a privilege level. The following example illustrates the first part of the display:

```
pix(config)# show privilege all
privilege show level 15 command aaa
privilege clear level 15 command aaa
privilege configure level 15 command aaa
privilege show level 15 command aaa-server
privilege clear level 15 command aaa-server
privilege configure level 15 command aaa-server
privilege show level 15 command access-group
privilege clear level 15 command access-group
privilege configure level 15 command access-group
privilege show level 15 command access-list
privilege clear level 15 command access-list
privilege configure level 15 command access-list
privilege show level 15 command activation-key
privilege configure level 15 command activation-key
```

Use the **show privilege level** command with the level option to display the command assignments for a specific privilege level. Use the **show privilege command** command to display the privilege level assignment of a specific command.

To view the user account that is currently logged in, enter the **show curpriv** command. The system displays the current user name and privilege level, as follows:

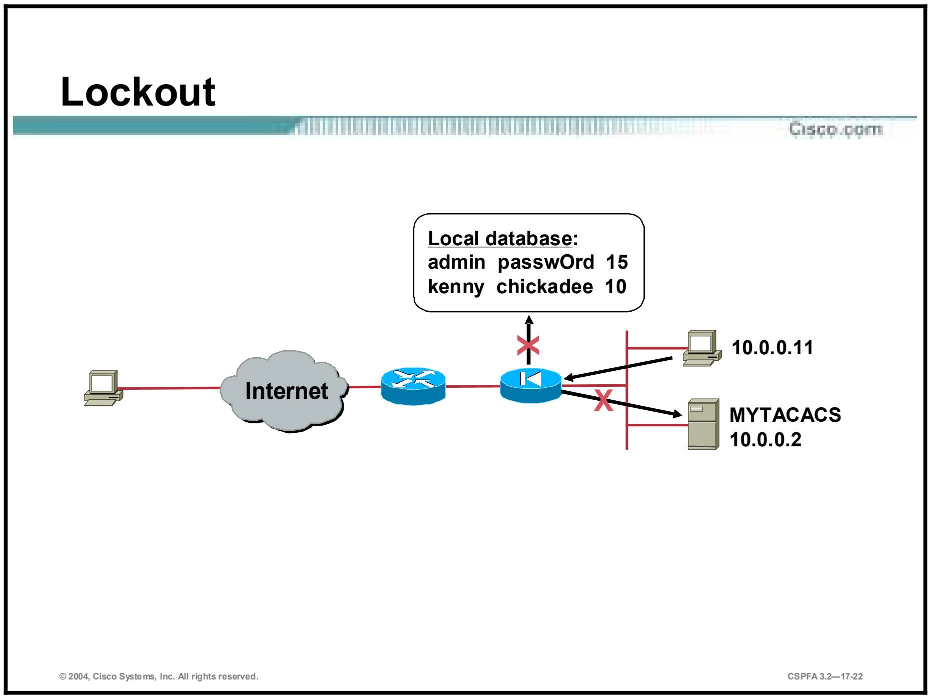
```
pix(config)# show curpriv
Username:admin
Current privilege level: 15
Current Mode/s:P_PRIV.
```

The syntaxes for the **show** commands are as follows:

show privilege [all | command *command* | level *level*]

show curpriv

level	The privilege level for which you want to display the command assignments.
command	The command for which you want to display the assigned privilege level.



Lockout

Local database:
admin passwOrd 15
kenny chickadee 10

CSPFA 3.2—17-22

When you are configuring the command authorization feature, do not save your configuration until you are sure it works the way you want. If you get locked out of your PIX Firewall, you can usually recover access by simply reloading it. If you have already saved your configuration and you find that you configured authentication using the LOCAL database but did not configure any usernames, you created a lockout problem. You can also encounter a lockout problem by configuring command authorization using a TACACS+ server if the TACACS+ server is unavailable, down, or misconfigured.

If you cannot recover access to the PIX Firewall by restarting your PIX Firewall, use your web browser to access the following website: http://www.cisco.com/warp/customer/110/34.shtml.

This website provides a downloadable file with instructions for using it to remove the lines in the PIX Firewall configuration that enable authentication and cause the lockout problem. If there are Telnet or console **aaa authentication** commands in PIX Firewall Software Versions 6.2 and greater, the system will also prompt to remove these.

Note If you have configured AAA on the PIX Firewall, and the AAA server is down, you can access the PIX Firewall by entering the Telnet password initially, and then "pix" as the username and the enable password (enable password password) for the password. If there is no enable password in the PIX Firewall configuration, enter "pix" for the username and press ENTER. If the enable and Telnet passwords are set but not known, you will need to continue with the password recovery process.

The PIX Password Lockout Utility is based on the PIX Firewall software version you are running. Use one of the following files, depending on the PIX Firewall software version you are running:

- np63.bin (6.3 version)
- np62.bin (6.2 version)

- np61.bin (6.1 version)
- np60.bin (6.0 version)
- np53.bin (5.3 version)
- np52.bin (5.2 version)
- np51.bin (5.1 version)

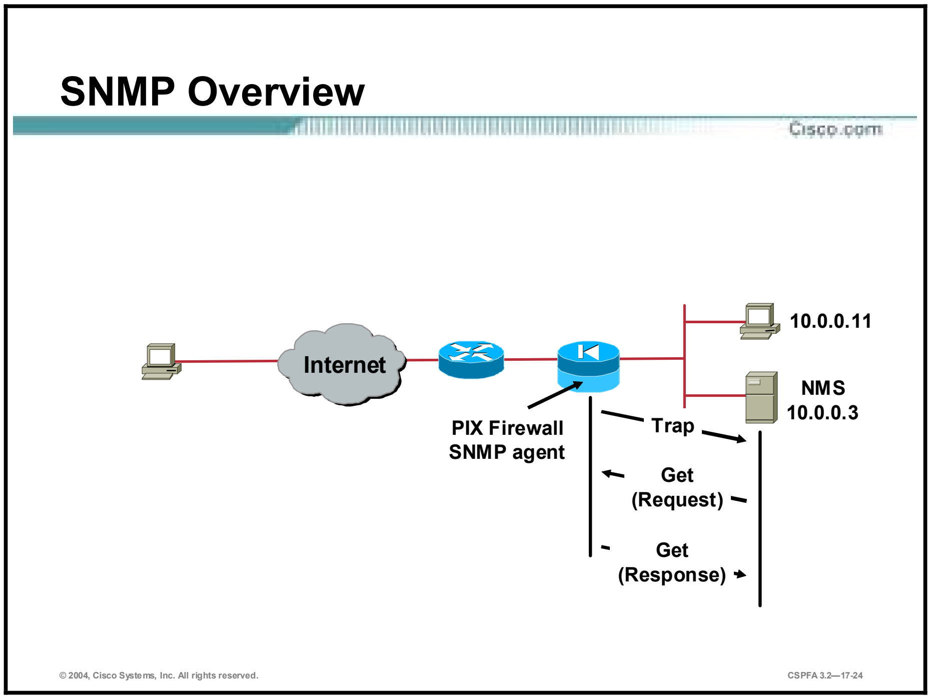
You can encounter a different type of lockout problem if you use the **aaa authorization** command and ***tacacs_server_tag*** argument, and you are not logged in as the correct user. For every command you enter, the PIX Firewall displays the following message:

```
Command Authorization failed
```

This occurs because the TACACS+ server does not have a user profile for the user account that you used for logging in. To prevent this problem, make sure that the TACACS+ server has all the users configured with the commands that they can execute. Also make sure that you are logged in as a user with the required profile on the TACACS+ server.

SNMP

This topic explains how to use the Simple Network Management Protocol (SNMP) to monitor your PIX Firewall and how to permit SNMP through the PIX Firewall for the management and monitoring of other network devices.



SNMP Overview

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—17-24

SNMP is an application-layer protocol designed to facilitate the exchange of management information between network devices. By using SNMP-transported data, such as packets per second and network error rates, network administrators can remotely manage and monitor network devices. As shown in the figure, devices managed by SNMP send information to a management server from which an administrator manages and monitors the device.

You can use SNMP to monitor system events on the PIX Firewall. SNMP events can be read, but information on the PIX Firewall cannot be changed with SNMP. You can also enable SNMP through the PIX Firewall so that any device can be managed and monitored by a management server on a PIX Firewall interface other than that on which it resides.

In order to understand SNMP support in the PIX Firewall, it is important to understand the following SNMP-related terminology:

- Managed devices—Hardware devices such as computers, routers, and terminal servers that are connected to networks.
- Agents—Software modules that reside in managed devices. Agents collect and store management information such as the number of error packets received by a network element.
- Managed object—A manageable component of a managed device. A managed object might be hardware, configuration parameters, or performance statistics of a device. For example, a list of currently active TCP circuits in a particular host computer is a managed object. Managed objects differ from variables, which are particular object instances. Whereas a

managed object might be a list of currently active TCP circuits in a particular host, an object instance is a single active TCP circuit in a particular host. Managed objects have object identifiers (OIDs).

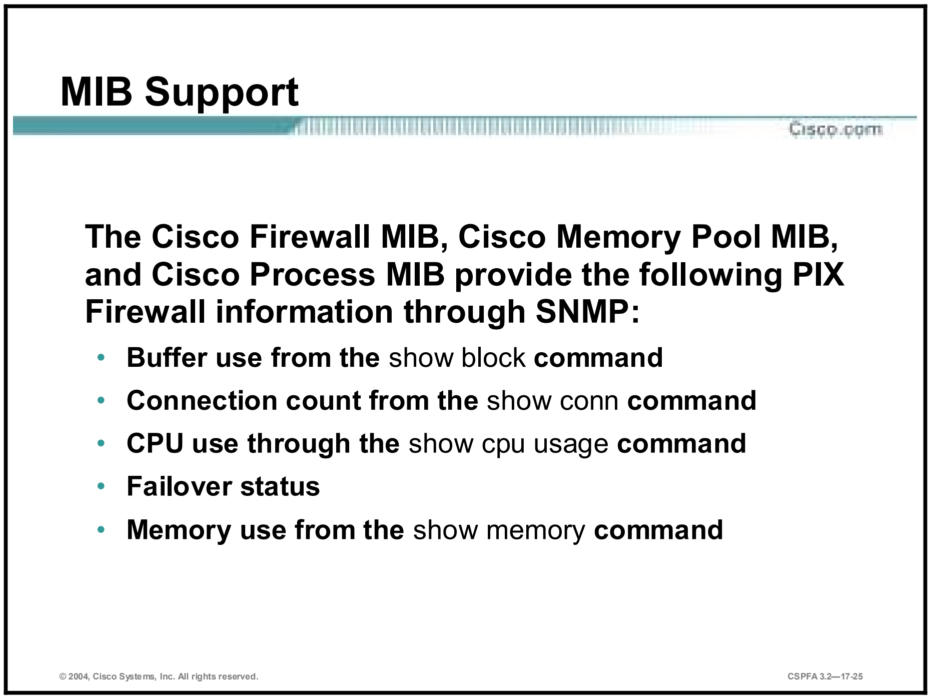
- Management information base (MIB)—A collection of managed objects. A MIB can be depicted as an abstract tree with an unnamed root. Individual data items make up the leaves of the tree. These leaves have object identifiers called OIDs that uniquely identify or name them. OIDs are like telephone numbers; they are organized hierarchically with specific digits assigned by different organizations. An OID is written as a sequence of sub-identifiers, starting with the tree root in dotted decimal notation.
- Network management stations (NMSs)—A device that executes management applications that monitor and control network elements. The NMS is the console through which the network administrator performs network management functions. It is usually a computer with a fast CPU, mega pixel color display, substantial memory, and abundant disk space.
- Trap—An event notification sent from an agent to the NMS. A trap is one of four types of interaction between an NMS and a managed device. Traps are unsolicited comments from the managed device to the NMS for certain events, such as link up, link down, and Syslog event generated. Traps are defined in MIB files.
- Community—A string value that provides a simple kind of password protection for communications between an SNMP agent and the SNMP NMS. The common default string is “public”.

SNMP is a request and response protocol. It uses the following operations:

- Get—Enables the NMS to retrieve an object instance from an agent
- GetNext—Enables the NMS to retrieve the next object instance from a table or list within an agent
- GetBulk—Used in place of the GetNext operation to simplify acquiring large amounts of related information
- Set—Enables the NMS to set values for object instances within an agent; not supported in the PIX Firewall
- Trap—Used by the agent to asynchronously inform the NMS of an event
- Inform—Enables one NMS to send trap information to another

Note Cisco SNMP agents communicate successfully with all SNMP-compliant NMSs, including those of Sun Microsystems (SunNet Manager), IBM (NetView/6000), and Hewlett-Packard (OpenView).

In the figure, the NMS uses a Get operation to request management information contained in an agent on the PIX Firewall. Within the Get Request, the NMS includes a complete OID so that the agent knows exactly what is being sought. The Get Response from the PIX Firewall SNMP agent contains a variable binding containing the same OID and the data associated with it. In an unrelated communication, the PIX Firewall sends a trap to the NMS because some urgent condition has occurred.



MIB Support

Cisco.com

The Cisco Firewall MIB, Cisco Memory Pool MIB, and Cisco Process MIB provide the following PIX Firewall information through SNMP:

- **Buffer use from the** show block **command**
- **Connection count from the** show conn **command**
- **CPU use through the** show cpu usage **command**
- **Failover status**
- **Memory use from the** show memory **command**

CSPFA 3.2—17-25

A MIB is a collection of information that is organized hierarchically. MIBs are accessed using a network-management protocol such as SNMP. They are comprised of managed objects and are identified by object identifiers.

A managed object is one of any number of specific characteristics of a managed device. An example of a managed object is cpmCPUTotal5sec, which is the Overall CPU busy percentage in the last five-second period. An object identifier uniquely identifies a managed object in the MIB hierarchy. The MIB hierarchy can be depicted as a tree with a nameless root, the levels of which are assigned by different organizations.

The top-level MIB object identifiers belong to different standards organizations, while lower-level object identifiers are allocated by associated organizations. Vendors can define private branches that include managed objects for their own products. Cisco appears in the MIB tree as the number 9. The PIX Firewall's managed object cpmCPUTotal5sec can be uniquely identified by its OID, 1.3.6.1.4.1.9.9.389.1.1.1.3. If you start at the top of the tree and move downward through the branches as directed by the digits in the OID, the first 9 you encounter is the Cisco branch. The number 389 is a PIX 506 Firewall. This shows that cpmCPUTotal5sec is a Cisco managed object.

Note See the MIB tree and further information on SNMP at the following site: http://www.cisco.com/univercd/cc/td/doc/product/webscale/css/advcfggd/snmp.htm

The PIX Firewall supports the following MIBs:

- PIX Firewall Software Versions 4.0 until 5.1: System and Interface groups of MIB-II (see RFC 1213) but not the AT, ICMP, TCP, UDP, EGP, transmission, IP, or SNMP groups CISCO-SYSLOG-MIB-V1SMI.my
- PIX Firewall Software Versions 5.1.X and later: Previous MIBs and CISCO-MEMORY-POOL-MIB.my and the cfwSystem branch of the CISCO-FIREWALL-MIB.my

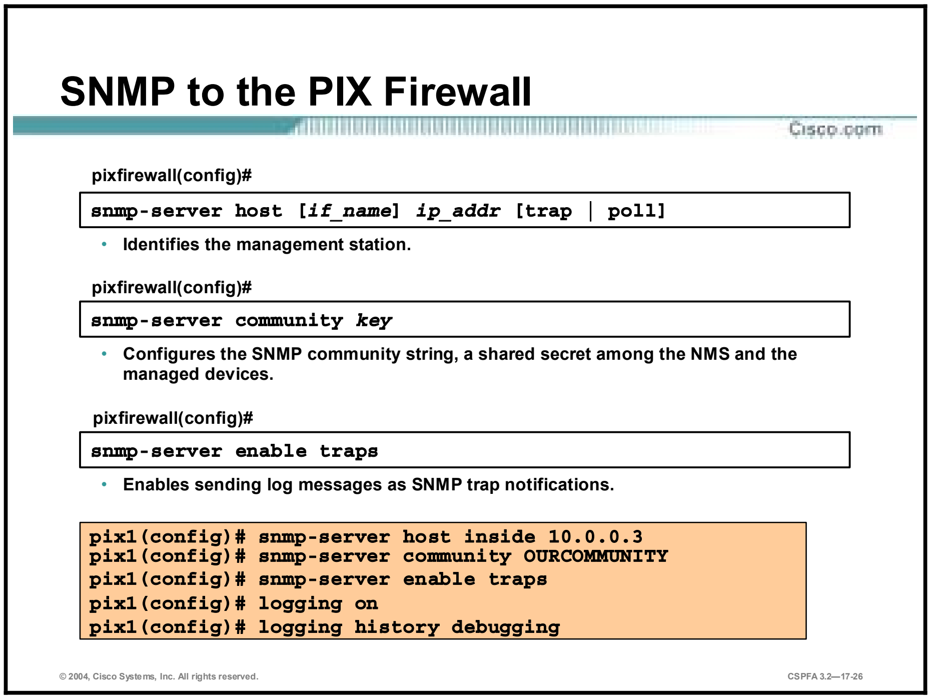
- PIX Firewall Software Versions 5.2.X and later: Previous MIBs and the ipAddrTable of the IP group
- PIX Firewall Software Versions 6.0.X and later: Previous MIBs and modification of the MIB-II OID to identify PIX Firewall by model (and enable CiscoWorks CiscoView 5.2 support). The new OIDs are found in the CISCO-PRODUCTS-MIB; for example, the PIX 515 Firewall has the OID 1.3.6.1.4.1.9.1.390
- PIX Firewall Software Versions 6.2.x and later: Previous MIBs and CISCO-PROCESS-MIB-V1SMI.my

The Cisco Firewall MIB, Cisco Memory Pool MIB, Cisco Process MIB provide the following PIX Firewall information through SNMP:

- Buffer usage from the show block command
- Connection count from the show conn command
- CPU usage through the show cpu usage command
- Failover status
- Memory usage from the show memory command

Note The supported section of the PROCESS MIB is the cpmCPUTotalTable branch of the cpmCPU branch of the ciscoProcessMIBObjects branch. There is no support for the ciscoProcessMIBNotifications branch, ciscoProcessMIBconformance branch, or the two tables, cpmProcessTable and cpmProcessExtTable, in the cpmProcess branch of the ciscoProcessMIBObjects branch of the MIB.

Note An SNMP OID for the PIX Firewall displays in SNMP event traps sent from the PIX Firewall. The model-specific OIDs are found in the CISCO-PRODUCTS-MIB.



SNMP to the PIX Firewall

pixfirewall(config)#

```
snmp-server host [if_name] ip_addr [trap | poll]
```

- Identifies the management station.

pixfirewall(config)#

```
snmp-server community key
```

- Configures the SNMP community string, a shared secret among the NMS and the managed devices.

pixfirewall(config)#

```
snmp-server enable traps
```

- Enables sending log messages as SNMP trap notifications.

```
pix1(config)# snmp-server host inside 10.0.0.3
pix1(config)# snmp-server community OURCOMMUNITY
pix1(config)# snmp-server enable traps
pix1(config)# logging on
pix1(config)# logging history debugging
```

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—17-26

The PIX Firewall can generate the following traps and send them to an NMS. Each trap is listed with its respective MIB file.

- authenticationFailure—CISCO-GENERAL-TRAPS.my
- clogNotificationsSent—CISCO-SYSLOG-MIB-V1SMI.my
- coldStart—CISCO-GENERAL-TRAPS.my
- linkDown—CISCO-GENERAL-TRAPS.my
- linkUp—CISCO-GENERAL-TRAPS.my
- warmStart—CISCO-GENERAL-TRAPS.my

To enable the PIX Firewall to receive SNMP requests from an NMS and send traps to an NMS, complete the following steps:

Step 1 Identify the IP address of the SNMP management station with the **snmp-server host** command. If the trap option of the **snmp-server host** command is used, the PIX Firewall sends traps to the NMS, but the NMS cannot query the PIX Firewall. If the poll option is used, the NMS is allowed to poll but the PIX Firewall does not send traps to it.

Step 2 Set the snmp-server options for location, contact, and the community password as required. If you only want to receive SNMP requests send the cold start, link up, and link down generic traps, no further configuration is required. The **snmp-server community** command is required. It specifies a secret to be shared among the NMS and the network nodes being managed. The PIX Firewall uses the community key to determine if an incoming SNMP request is valid and responds only to requests that contain a valid community string.

Step 3 Add an **snmp-server enable traps** command statement. This and all **snmp-server** commands can be removed by using their no forms. The **show snmp-server** command displays all of your SNMP configuration.

Step 4 Set the logging level with the **logging history** command. This sets the severity level for SNMP Syslog messages.

```
logging history debugging
```

Note It is recommended that you use the debugging level during initial setup and during testing. Thereafter, set the level from debugging to a lower value for production use.

Step 5 Start sending Syslog traps to the management station with the **logging on** command.

In the example in the figure, the PIX Firewall receives SNMP requests and sends traps to host 10.0.0.11. The PIX Firewall and host 10.0.0.11 are members of the community MYCOMMUNITY.

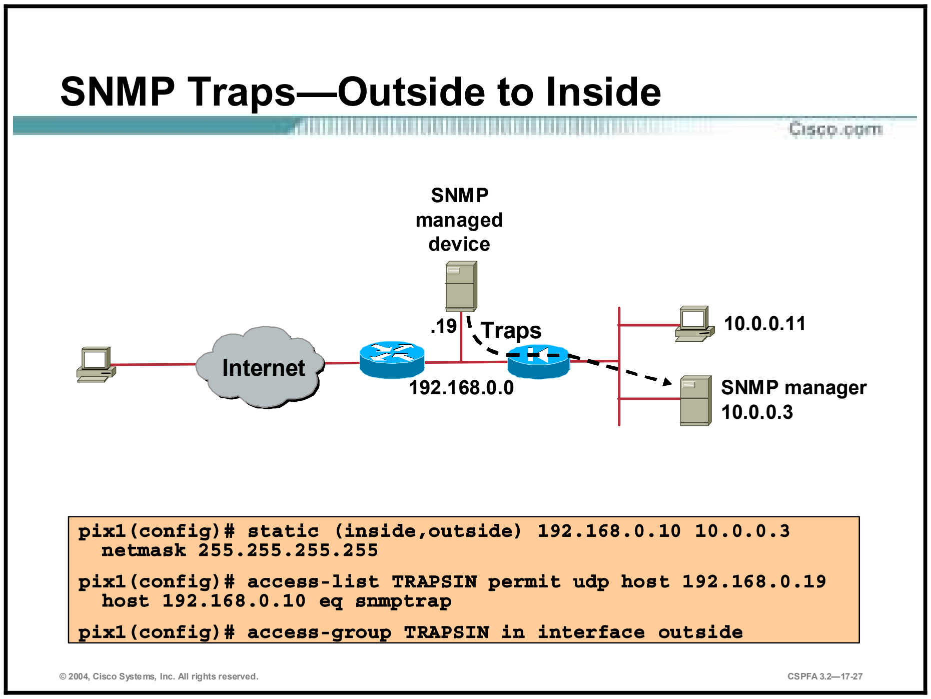
The syntaxes for the **snmp-server** commands are as follows:

snmp-server host [*if_name*] *ip_addr* [trap | poll]

snmp-server community *key*

snmp-server enable traps

if_name	The interface name where the SNMP management station resides.
ip_addr	The IP address of a host to which SNMP traps should be sent and from which the SNMP requests come.
trap	Specifies that this host is not allowed to poll. Only traps are sent.
poll	Specifies that this host is allowed to poll, but traps are not sent. The default allows both traps and polls to be acted upon.
key	The password key value in use at the SNMP management station. The key is a case-sensitive value up to 32 characters in length. Spaces are not permitted. The default is public if a key is not set. Consequently, it is important to specify a new key value for security reasons.



SNMP Traps—Outside to Inside

```
pix1(config)# static (inside,outside) 192.168.0.10 10.0.0.3
  netmask 255.255.255.255
pix1(config)# access-list TRAPSIN permit udp host 192.168.0.19
  host 192.168.0.10 eq snmptrap
pix1(config)# access-group TRAPSIN in interface outside
```

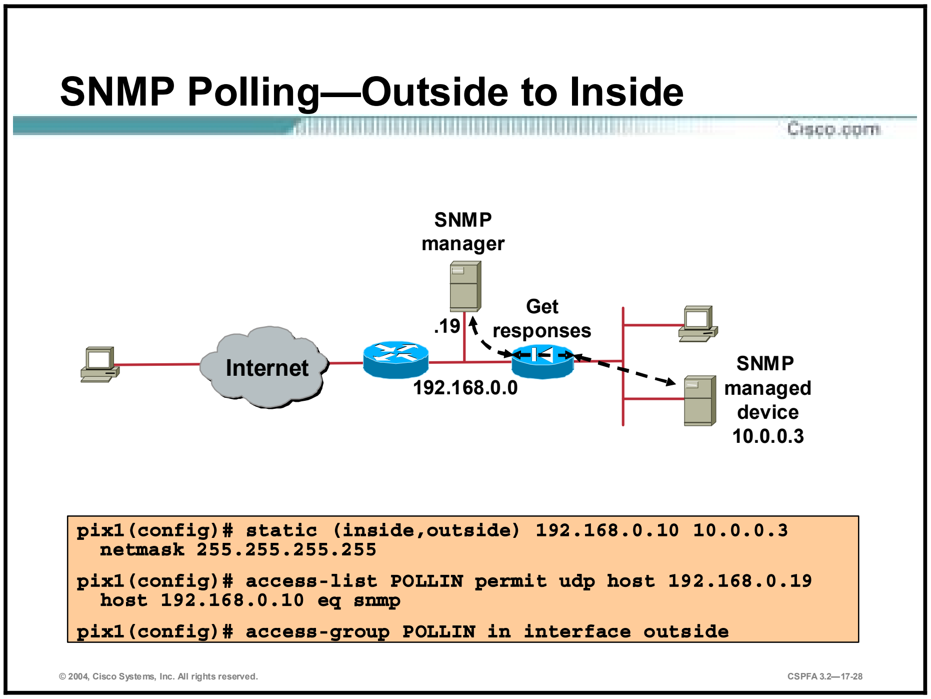
CSPFA 3.2—17-27

To enable SNMP through the PIX Firewall so that network devices other than the PIX Firewall can be managed and monitored, use the appropriate **static** and **access-list** commands, as shown in the following examples:

- Traps outside to inside—To allow traps from outside host 192.168.0.19 to inside management system 10.0.0.3, use the following commands:

```
static (inside,outside) 192.168.0.10 10.0.0.3 netmask
255.255.255.255

access-list TRAPSIN permit udp host 192.168.0.19 host 192.168.0.10
eq snmptrap

access-group TRAPSIN in interface outside
```

- Traps inside to outside—In the absence of outbound ACLs, outbound traffic is allowed by default.



SNMP Polling—Outside to Inside

```
pix1(config)# static (inside,outside) 192.168.0.10 10.0.0.3
  netmask 255.255.255.255
pix1(config)# access-list POLLIN permit udp host 192.168.0.19
  host 192.168.0.10 eq snmp
pix1(config)# access-group POLLIN in interface outside
```

CSPFA 3.2—17-28

To enable SNMP through the PIX Firewall so that network devices other than the PIX Firewall can be managed and monitored, use the appropriate **static** and **access-list** commands as shown in the following examples:

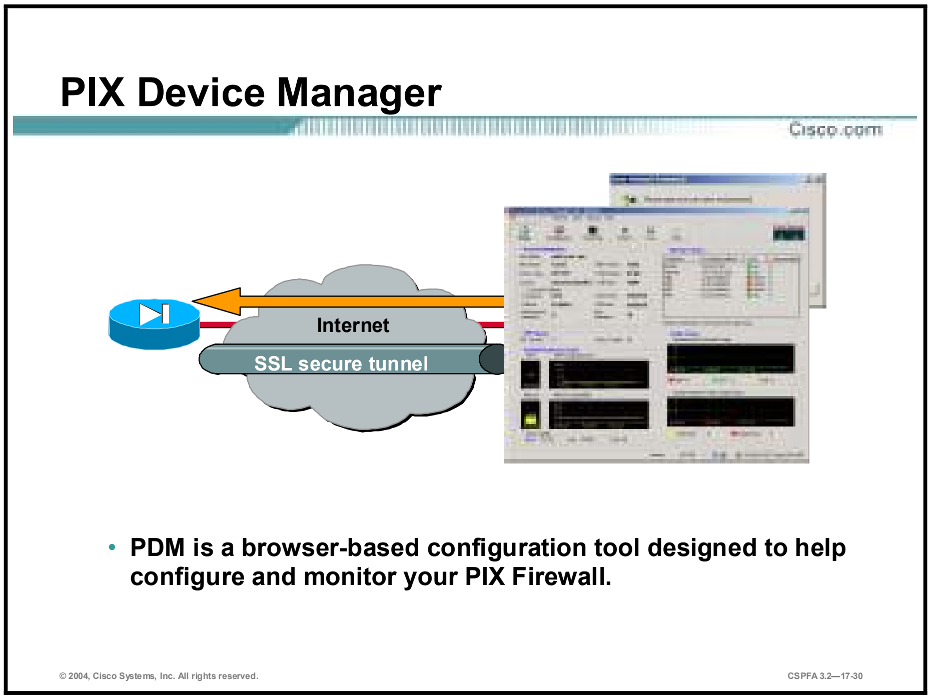
- Polling outside to inside—To allow polling from the NMS, outside host 192.168.0.19, to inside host 10.0.0.3, use the following configuration:

  ```
  static (inside,outside) 192.168.0.10 10.0.0.3 netmask
  255.255.255.255
  
  access-list POLLIN permit udp host 192.168.0.19 host 192.168.0.10
  eq snmp
  
  access-group POLLIN in interface outside
  ```

- Polling inside to outside—In the absence of outbound ACLs, outbound traffic is allowed by default.

Management Tools

This topic gives an overview of the tools available for managing your PIX Firewalls.



PIX Device Manager

- **PDM is a browser-based configuration tool designed to help configure and monitor your PIX Firewall.**

CSPFA 3.2—17-30

Cisco PIX Device Manager (PDM) is a browser-based configuration tool designed to help you graphically set up, configure, and monitor your PIX Firewall, without requiring an extensive knowledge of the PIX Firewall CLI.

PDM monitors and configures a single PIX Firewall. You can point your browser to more than one PIX Firewall and administer several PIX Firewalls from a single workstation.

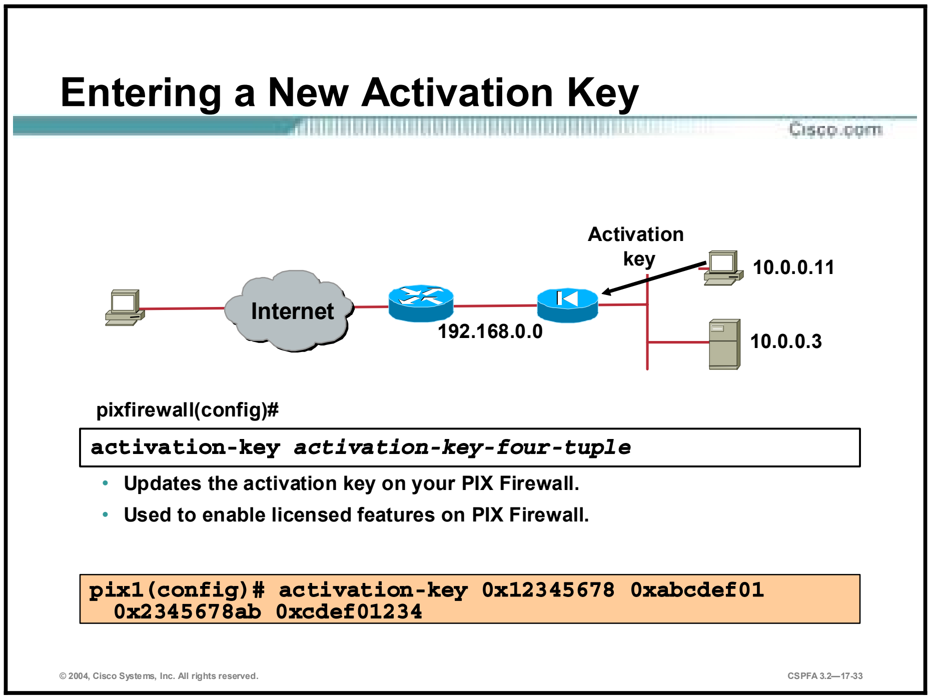
Management Center for Firewalls

Cisco.com

While PDM addresses single device management and the Firewall Management Center (Firewall MC) addresses mid-level, multidevice management. The Firewall MC uses terminology and concepts consistent with PDM and has a look and feel consistent with PDM as well.

Activation Keys

This topic explains how to upgrade an activation key.



Entering a New Activation Key

pixfirewall(config)#

```
activation-key activation-key-four-tuple
```

- Updates the activation key on your PIX Firewall.
- Used to enable licensed features on PIX Firewall.

```
pix1(config)# activation-key 0x12345678 0xabcdef01
  0x2345678ab 0xcdef01234
```

CSPFA 3.2—17-33

You can upgrade the license for your PIX Firewall from the CLI. Before entering the activation key, ensure that the image in Flash and the running image are the same. You can do this by rebooting the PIX Firewall before entering the new activation key. You will also need to reboot the PIX Firewall after entering the new activation key for the change to take effect.

Use the **activation-key** command to enter an activation key. In this command, replace ***activation-key-four-tuple*** with the activation key you obtained with your new license as follows:

```
activation-key 0x12345678 0xabcdef01 0x2345678ab 0xcdef01234
```

The leading “0x” hexadecimal indicator is optional. If it is omitted, the parameter is assumed to be a hexadecimal number, as in the following example:

```
activation-key   12345678   abcdef01   2345678ab   cdef01234
```

After you enter the activation key, the system displays the following output when the activation key has been successfully changed:

```
pixfirewall(config)# activation-key 0x01234567 0x89abcdef01 0x23456789
0xabcdef01
Serial Number: 12345678 (0xbc614e)


Flash activation key:  0x01234567 0x89abcdef01 0x23456789 0xabcdef01
Licensed Features:
Failover:        Enabled
```

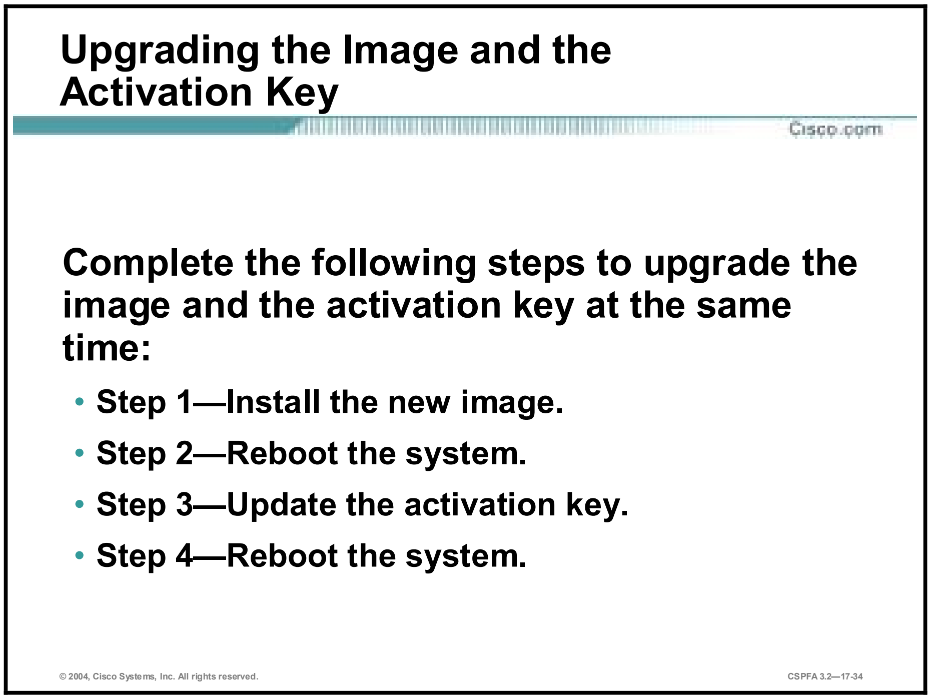
```
VPN-DES:        Enabled
VPN-3DES:       Enabled
Maximum Interfaces: 10
Cut-through Proxy:  Enabled
Guards:             Enabled
URL-filtering:      Enabled
Inside Hosts:       Unlimited
Throughput:         Unlimited
IKE peers:          Unlimited

The flash activation key has been modified.
The flash activation key is now DIFFERENT than the running key.
The flash activation key will be used when the unit is reloaded.
pixfirewall(config)#
```

The syntax for the **activation-key** command is as follows:

activation-key ***activation-key-four-tuple***

activation-key-four-tuple	The activation key you obtained with your new license.

Reload the PIX Firewall to activate the Flash activation key.



Upgrading the Image and the Activation Key

Complete the following steps to upgrade the image and the activation key at the same time:

- **Step 1—Install the new image.**
- **Step 2—Reboot the system.**
- **Step 3—Update the activation key.**
- **Step 4—Reboot the system.**

CSPFA 3.2—17-34

If you are upgrading the image to a newer version and the activation key is also being changed, reboot the system twice, as shown in the following procedure:

Step 1 Install the new image.

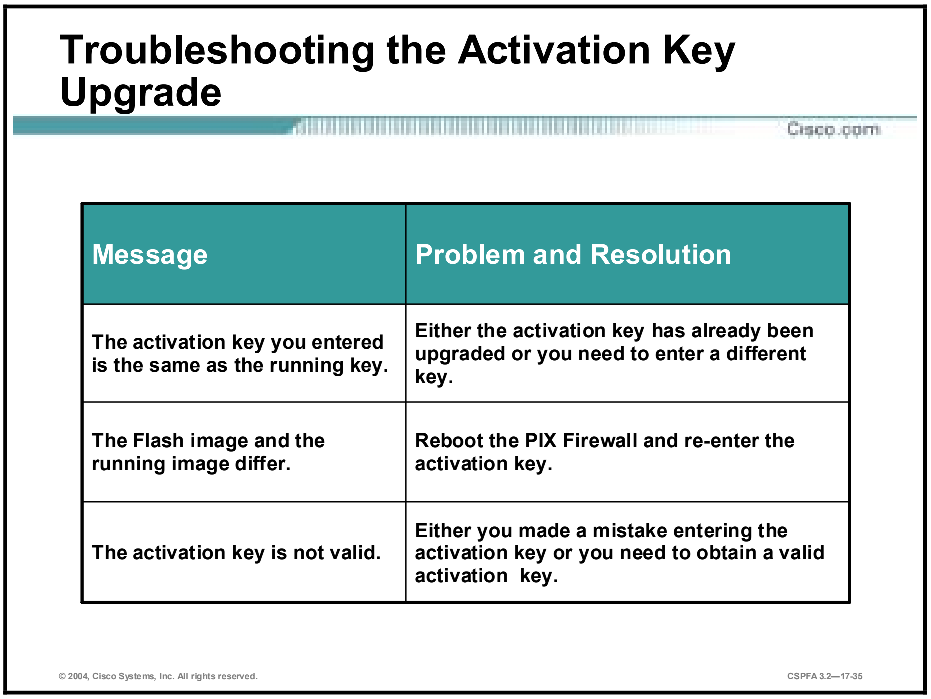
Step 2 Reboot the system.

Step 3 Update the activation key.

Step 4 Reboot the system.

After the key update is complete, the system is reloaded a second time, so the updated licensing scheme can take effect.

If you are downgrading an image, you only need to reboot once, after installing the new image. In this situation, the old key is both verified and changed with the current image.



Troubleshooting the Activation Key Upgrade

Message	Problem and Resolution
The activation key you entered is the same as the running key.	Either the activation key has already been upgraded or you need to enter a different key.
The Flash image and the running image differ.	Reboot the PIX Firewall and re-enter the activation key.
The activation key is not valid.	Either you made a mistake entering the activation key or you need to obtain a valid activation key.

CSPFA 3.2—17-35

To view your current activation key, enter the **show activation-key** command. The following examples show the output from this command under different circumstances.

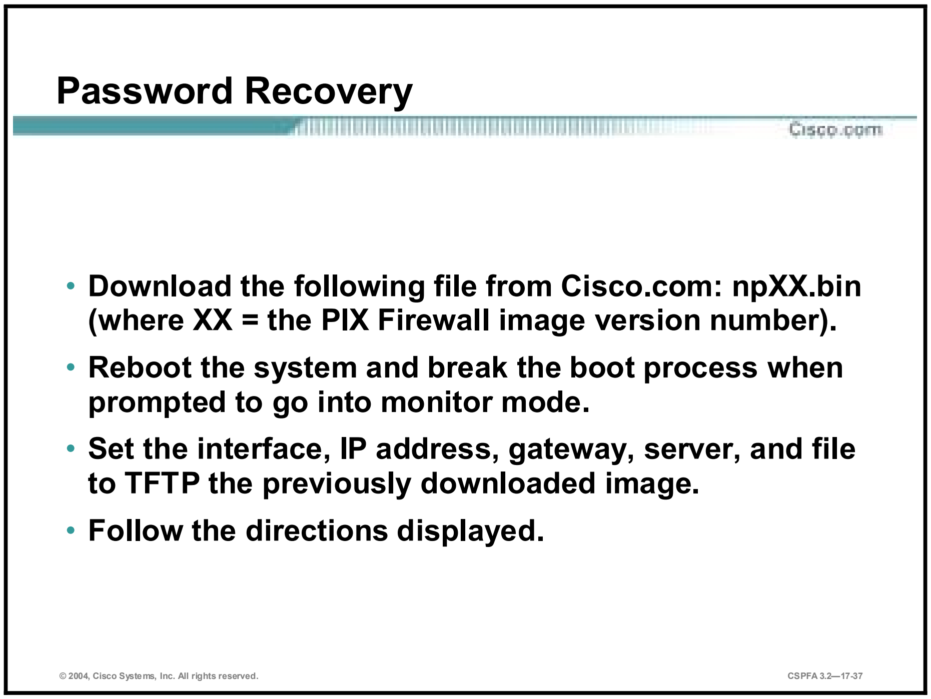
```
pixfirewall(config)# show activation-key

Serial Number: 12345678 (0xbc614e)

Running activation key: 0xe02888da 0x4ba7bed6 0xf1c123ae 0xffd8624e
Licensed Features:
Failover:       Enabled
VPN-DES:        Enabled
VPN-3DES:       Enabled
Maximum Interfaces: 6
Cut-through Proxy:  Enabled
Guards:         Enabled
Websense:       Enabled
Throughput:     Unlimited
ISAKMP peers:   Unlimited

The flash activation key is the SAME as the running key.
pixfirewall(config)# show activation-key

Serial Number: 12345678 (0xbc614e)

Running activation key: 0xe02888da 0x4ba7bed6 0xf1c123ae 0xffd8624e
```

```
Licensed Features:
Failover:       Enabled
VPN-DES:        Enabled
VPN-3DES:       Enabled
Maximum Interfaces:  8
Cut-through Proxy:  Enabled
Guards:         Enabled
URL-filtering:  Enabled
Throughput:     Unlimited
ISAKMP peers:   Unlimited

Flash activation key: 0xe02388da 0x5ca7bed2 0xf1c123ae 0xffd8624t
Licensed Features:
Failover:       Enabled
VPN-DES:        Enabled
VPN-3DES:       Disabled
Maximum Interfaces:  8
Cut-through Proxy:   Enabled
Guards:         Enabled
URL-filtering:  Enabled
Throughput:     Unlimited
ISAKMP peers:   Unlimited

The flash activation key is DIFFERENT than the running key.
The flash activation key takes effect after the next reload.
pixfirewall(config)# show activation-key

Serial Number: 12345678 (0xbc614e)

Running activation key: 0xe02888da 0x4ba7bed6 0xf1c123ae 0xffd8624e
Licensed Features:
Failover:       Enabled
VPN-DES:        Enabled
VPN-3DES:       Enabled
Maximum Interfaces:  6
Cut-through Proxy:   Enabled
Guards:         Enabled
Websense:       Enabled
Throughput:     Unlimited
ISAKMP peers:   Unlimited
```

```
The flash image is DIFFERENT than the running image.

The two images must be the same in order to examine the flash
activation key.

pixfirewall(config)#
```

Password Recovery and Image Upgrade

This topic explains how to perform password recovery and upgrade the PIX Firewall image.



Password Recovery

Cisco.com

- Download the following file from Cisco.com: npXX.bin (where XX = the PIX Firewall image version number).
- Reboot the system and break the boot process when prompted to go into monitor mode.
- Set the interface, IP address, gateway, server, and file to TFTP the previously downloaded image.
- Follow the directions displayed.

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—17-37

Password recovery for PIX Firewall models 501, 506, 515, 525, and 535 requires a TFTP server. To perform a password recovery using TFTP, complete the following steps:

Step 1 Download the file for the PIX Firewall software version you are running from Cisco.com. Each version requires a different file. You will need a Cisco.com login to download this data.

Step 2 Move the binary file you just downloaded to the TFTP home folder on your TFTP server.

Step 3 Reboot your PIX Firewall and interrupt the boot process to enter monitor mode. To do this, you must press the **Escape** key or send a break character.

Step 4 Specify the PIX Firewall interface to use for TFTP:

```
monitor> interface [num]
```

Step 5 Specify the PIX Firewall interface's IP address:

```
monitor> address [IP_address]
```

Step 6 Specify the default gateway (if needed):

```
monitor> gateway [IP_address]
```

Step 7 Verify connectivity to the TFTP server:

```
monitor> ping [server_address]
```

Step 8 Name the server:

```
monitor> server [IP_address]
```

Step 9 Name the image filename:

```
monitor> file [name]
```

Step 10 Start the TFTP process:

```
monitor> tftp
```

Step 11 When prompted, enter **y** to erase the password:

```
Do you wish to erase the passwords? [yn] y
Passwords have been erased
```

The system automatically erases the password and starts rebooting.

Note To obtain the files needed for password recovery, go to: http://www.cisco.com/warp/public/110/34.shtml#files.

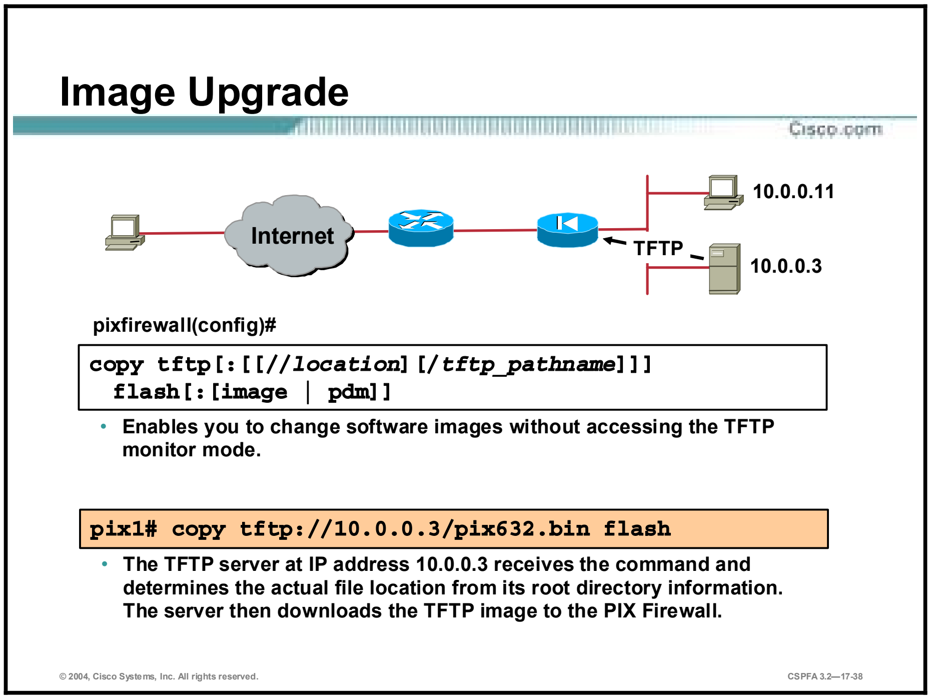


Image Upgrade

pixfirewall(config)#

```
copy tftp[:[[//location] [/tftp_pathname]]]
  flash[:[image | pdm]]
```

- Enables you to change software images without accessing the TFTP monitor mode.

```
pix1# copy tftp://10.0.0.3/pix632.bin flash
```

- The TFTP server at IP address 10.0.0.3 receives the command and determines the actual file location from its root directory information. The server then downloads the TFTP image to the PIX Firewall.

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—17-38

The **copy tftp flash** command enables you to change software images without accessing the TFTP monitor mode. You can use this command to download a software image via TFTP with any PIX Firewall model running version 5.1 or later. The image you download is made available to the PIX Firewall on the next reload.

Be sure to configure your TFTP server to point to the image you wish to download. For example, to download the pix611.bin file from the D: partition on a Windows system whose IP address is 10.0.0.3, you would access the Cisco TFTP Server View>Options menu and enter the filename path in the TFTP server root directory edit box (for example, D:\pix_images). Then, to copy the file to the PIX Firewall, use the following command: **copy tftp://10.0.0.3/pix632.bin flash**.

The TFTP server receives the command and determines the actual file location from its root directory information. The server then downloads the TFTP image to the PIX Firewall.

Note Your TFTP server must be open when you enter the **copy tftp** command on the PIX Firewall.

The syntax for the **copy tftp flash** command is as follows:

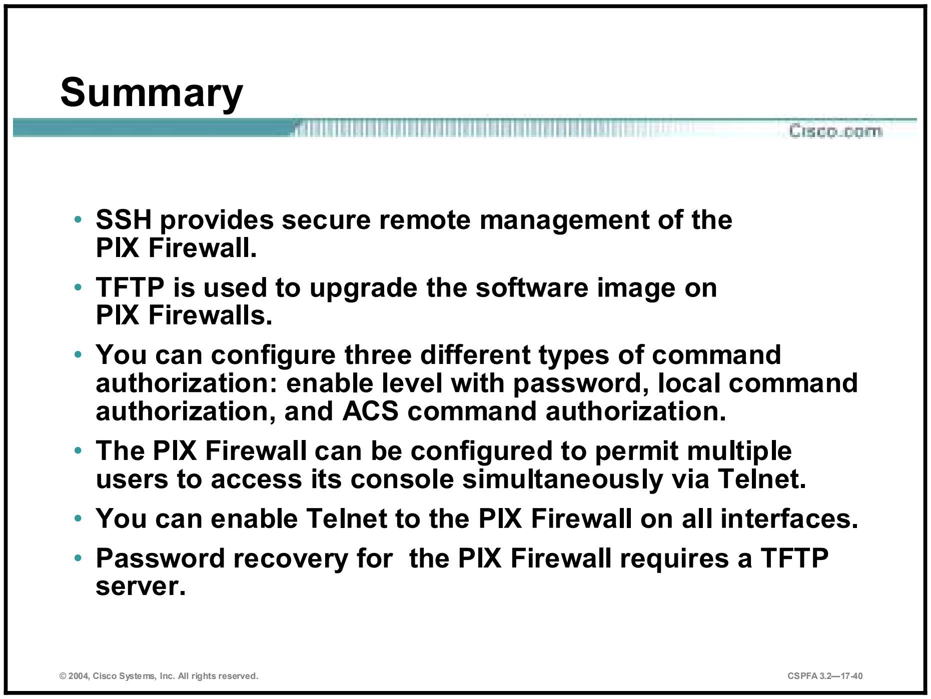
copy tftp[:[[//*location*] [/*pathname*]]] flash[:[image | pdm]]

copy tftp flash	Enables you to download Flash memory software images via TFTP without using monitor mode.
location	Specifies the IP address or name of the server on which the TFTP server resides.

pathname	Specifies any directory names in the path to the file as well as the actual file name. The PIX Firewall must know how to reach this location via its routing table information, which is determined by the **ip address** command, the **route** command, or RIP, depending upon your configuration.
image	Enables you to download the selected PIX Firewall image to Flash memory. An image you download is made available to the PIX Firewall on the next reboot.
pdm	Enables you to download the selected PDM image files to Flash memory. These files are available to the PIX Firewall immediately, without a reboot.

Summary

This topic summarizes the information you learned and the tasks you completed in this lesson.



Summary

Cisco.com

- SSH provides secure remote management of the PIX Firewall.
- TFTP is used to upgrade the software image on PIX Firewalls.
- You can configure three different types of command authorization: enable level with password, local command authorization, and ACS command authorization.
- The PIX Firewall can be configured to permit multiple users to access its console simultaneously via Telnet.
- You can enable Telnet to the PIX Firewall on all interfaces.
- Password recovery for the PIX Firewall requires a TFTP server.

CSPFA 3.2—17-40

Lab Exercise—System Maintenance

Complete the following lab exercise to practice what you have learned in this lesson.

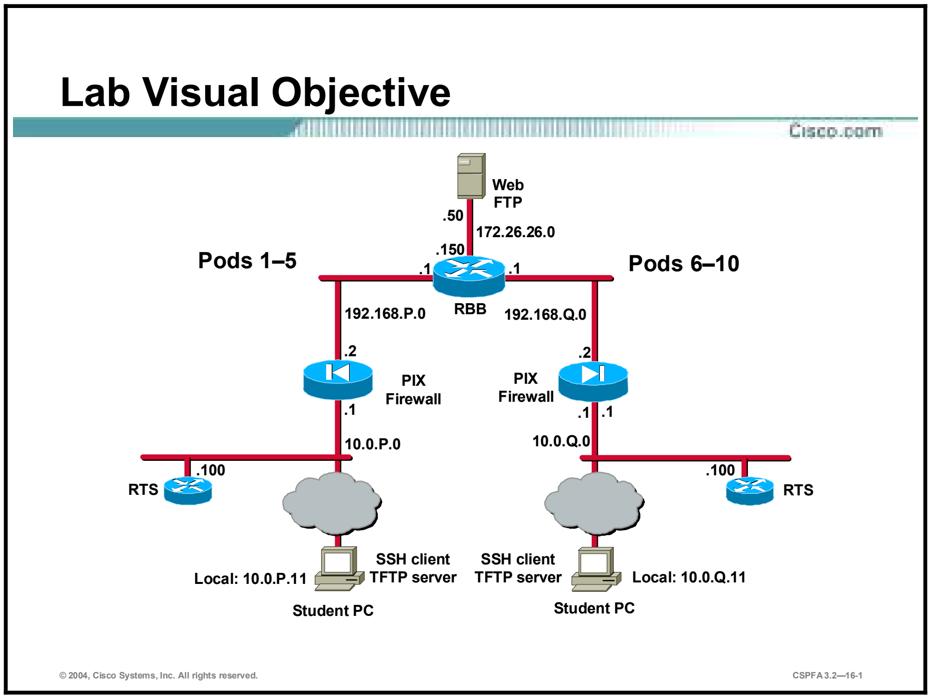
Objectives

In this lab exercise, you will complete the following tasks:

- Configure enable-level command authorization with passwords.
- Test enable-level command authorization.
- Generate an RSA key pair for encrypted SSH sessions.
- Establish an SSH connection to your PIX Firewall.
- Configure command authorization using the local user database.
- Test command authorization using the local user database.
- Perform a password recovery.
- Upgrade the PIX Firewall software image.

Visual Objective

The following figure displays the lab topology for your classroom environment.



Lab Visual Objective

CSPFA 3.2—16-1

Setup

Before starting this lab exercise, restore the original IP address and default gateway to your student PC. Ensure that your student PC is set up as follows:

Subhead	Student PC
Hostname	NTP
IP address	10.0.P.11
Netmask	255.255.255.0
Default gateway	10.0.P.1

(where P = pod number)

Task 1—Configure Enable-Level Command Authorization with Passwords

Complete the following steps to enable command authorization with privileged mode passwords:

Step 1 Set privilege level 10 for enable mode's **configure** command:

```
pixP(config)# privilege configure level 10 mode enable command
configure
```

Step 2 Set privilege level 10 for the **nameif** command:

```
pixP(config)# privilege level 10 command nameif
```

Step 3 Set privilege level 12 for the **interface** command:

```
pixP(config)# privilege level 12 command interface
```

Step 4 Assign an enable password for privileged level 15:

```
pixP(config)# enable password prmode15
```

Step 5 Assign an enable password to privileged level 10:

```
pixP(config)# enable password prmode10 level 10
```

Step 6 Assign an enable password to privileged level 12:

```
pixP(config)# enable password prmode12 level 12
```

Step 7 Enable command authorization by entering the following command:

```
pixP(config)# aaa authorization command LOCAL
```

Step 8 Exit configuration mode:

```
pixP(config)# exit
pixP#
```

Step 9 Exit privileged mode:

```
pixP# exit

Logoff

Type help or '?' for a list of available commands.
pixP>
```

Task 2—Test Enable-Level Command Authorization

Complete the following steps to test the command authorization you configured in Task 1:

Step 1 Enter privileged mode level 12. When prompted for a password, enter **prmode12**.

```
pixP> enable 12
Password:
pixP#
```

Step 2 Enter configuration mode:

```
pixP# config t
```

Step 3 Verify that you can use the **nameif** command:

```
pixP(config)# nameif e3 PRIVTEST sec30
```

Step 4 View your configuration:

```
pixP(config)# show nameif
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
nameif ethernet3 PRIVTEST security30
nameif ethernet4 intf4 security20
nameif ethernet5 intf5 security25
```

Step 5 Verify that you can use the **interface** command:

```
pixP(config)# interface e3 100full
```

Step 6 View your configuration:

```
pixP(config)# show interface
interface ethernet0 "outside" is up, line protocol is up
  Hardware is i82559 ethernet, address is 0003.e300.483a
  IP address 192.168.P.2, subnet mask 255.255.255.0
  MTU 1500 bytes, BW 100000 Kbit full duplex
        10640 packets input, 1374788 bytes, 0 no buffer
        Received 7179 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        3458 packets output, 348972 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128)
        software(0/6)
        output queue (curr/max blocks): hardware (0/9) software (0/2)
interface ethernet1 "inside" is up, line protocol is up
  Hardware is i82559 ethernet, address is 0003.e300.483b
  IP address 10.0.P.1, subnet mask 255.255.255.0
  MTU 1500 bytes, BW 100000 Kbit full duplex
        11119 packets input, 1438842 bytes, 0 no buffer
        Received 7554 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        4153 packets output, 390555 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128)
        software(0/4)
        output queue (curr/max blocks): hardware (0/15) software
        (0/14)
interface ethernet2 "dmz" is up, line protocol is up
  Hardware is i82558 ethernet, address is 00e0.b602.3387
  IP address 172.16.P.1, subnet mask 255.255.255.0
  MTU 1500 bytes, BW 100000 Kbit full duplex
        7024 packets input, 1050994 bytes, 0 no buffer
        Received 6991 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        98 packets output, 41652 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128)
        software(0/2)
        output queue (curr/max blocks): hardware (0/9) software (0/1)
interface ethernet3 "PRIVTEST" is up, line protocol is down
  Hardware is i82558 ethernet, address is 00e0.b602.3386
  IP address 172.17.P.1, subnet mask 255.255.255.0
  MTU 1500 bytes, BW 100000 Kbit full duplex
        2510 packets input, 317593 bytes, 0 no buffer
        Received 134 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        2384 packets output, 223276 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware
        (128/128)software(0/38)
        output queue (curr/max blocks): hardware (0/35) software (0/1)
interface ethernet4 "intf4" is administratively down, line protocol is
down
  Hardware is i82558 ethernet, address is 00e0.b602.3385
  IP address 127.0.0.1, subnet mask 255.255.255.255
  MTU 1500 bytes, BW 10000 Kbit half duplex
        0 packets input, 0 bytes, 0 no buffer
        Received 0 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        0 packets output, 0 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128)software
        (0/0)
        output queue (curr/max blocks): hardware (0/0) software (0/0)
interface ethernet5 "intf5" is administratively down, line protocol is
down
  Hardware is i82558 ethernet, address is 00e0.b602.3384
  IP address 127.0.0.1, subnet mask 255.255.255.255
  MTU 1500 bytes, BW 10000 Kbit half duplex
```

```
        0 packets input, 0 bytes, 0 no buffer
        Received 0 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        0 packets output, 0 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128)software
        (0/0)
        output queue (curr/max blocks): hardware (0/0) software (0/0)
```

(where P = pod number)

Step 7 Exit configuration mode:

```
pixP(config)# exit
pixP#
```

Step 8 Exit privileged mode:

```
pixP# exit

Logoff

Type help or '?' for a list of available commands.

pixP>
```

Step 9 Enter privileged mode level 10. When prompted for a password, enter **prmode10**:

```
pixP> enable 10
Password:
pixP#
```

Step 10 Enter configuration mode:

```
pixP# config t
pixP(config)#
```

Step 11 Verify that you can use the **nameif** command:

```
pixP(config)# nameif e4 PRIVTEST2 sec35
```

Step 12 View your configuration:

```
pixP(config)# show nameif
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
nameif ethernet3 PRIVTEST security30
nameif ethernet4 PRIVTEST2 security35
nameif ethernet5 intf5 security25
```

Step 13 Try to use the **interface** command:

```
pixP(config)# interface e4 100full
Command authorization failed.
```

Step 14 Exit configuration mode:

```
pixP(config)# exit
pixP#
```

Step 15 Exit privileged mode:

```
pixP# exit

Logoff

Type help or '?' for a list of available commands.

pixP>
```

Step 16 Enter privileged mode. When prompted for a password, enter **prmode15**.

```
pixP> enable
Password:
pixP#
```

Step 17 Enter configuration mode:

```
pixP# config t
pixP(config)#
```

Task 3—Generate an RSA Key Pair for Encrypted SSH Sessions

Complete the following steps to generate an RSA key pair to encrypt the SSH terminal session:

Step 1 Delete any previously created RSA keys:

```
pixP(config)# ca zeroize rsa
```

Step 2 Save the CA state to complete the erasure of the old RSA key pair:

```
pixP(config)# ca save all
```

Step 3 Configure the domain name:

```
pixP(config)# domain-name cisco.com
```

Step 4 Generate an RSA key pair to use to encrypt SSH sessions:

```
pixP(config)# ca generate rsa key 1024
For <key_modulus_size> >= 1024, key generation could
  take up to several minutes. Please wait.
```

Step 5 Save the keys to Flash memory:

```
pixP(config)# ca save all
```

Step 6 View your public key:

```
pixP(config)# sh ca mypubkey rsa
% Key pair was generated at: 18:34:29 UTC Apr 17 2002
Key name: pixP.cisco.com
 Usage: General Purpose Key
 Key Data:
  30819f30 0d06092a 864886f7 0d010101 05000381 8d003081 89028181 00bc43bf
  33d9c65d e508b6df ecf71e37 5574a21d 56185faf cbb9fe14 5a345222 42cd2927
  604fd719 a58d4f82 dc382fc4 ae037d15 f4f11ca8 06020c8d 5cd350d1 9bf19457
  a6dc1a86 f1e101ae 842b0281 f42f38c5 c8e5c095 711ac751 f28d693f ffdcb40f
  2892169e 90be60dd 15c2fdc9 b8bda690 e55b29bf 670ed794 30e9c012 5f020301 0001
```

(where P = pod number)

Task 4—Establish an SSH Connection to Your PIX Firewall

Complete the following steps to securely connect to your PIX Firewall via SSH:

Step 1 Enable SSH debugging:

```
pixP(config)# debug ssh
SSH debugging on
```

Step 2 Grant SSH access to your inside subnet:

```
pixP(config)# ssh 10.0.P.0 255.255.255.0 inside
```

(where P = pod number)

Step 3 Set the SSH inactivity timeout to 30 minutes:

```
pixP(config)# ssh timeout 30
```

Step 4 Minimize, but do not close, your Telnet session window. Double-click the **Shortcut to ttssh.exe** icon on your desktop.

Step 5 Enter **10.0.P.1**, the IP address of the PIX Firewall's inside interface, in the Host drop-down menu within the TCP/IP group box.

Step 6 Select the **SSH** radio button.

Step 7 Click **OK**. The Security Warning window opens.

Step 8 Select **Add this new key to the known hosts lists**.

Step 9 Click **Continue**. The SSH Authentication window opens.

Note If you get a message saying the known hosts file does not exist, click **OK** and continue with Step 10.

Step 10 Enter **pix** as the username and **cisco** as the password. Click **OK**.

Step 11 Enter privileged mode. When prompted for a password, enter prmode15.

```
pixP>enable
```

```
Password:

pixP#
```

Step 12 Enter configuration mode:

```
pixP# config t

pixP(config)#
```

Step 13 To view the status your SSH session, enter the following command:

```
pixP(config)# show ssh sessions

Session ID      Client IP       Version Encryption      State
Username

    0           insidehost      1.5     3DES            6       pix
```

Step 14 Disconnect your SSH session:

```
pixP(config)# ssh disconnect 0
```

Step 15 Click **OK** in the TTSSH window.

Step 16 Return to your Telnet session window, and change the PIX Firewall's Telnet password from cisco to sshpass:

```
pixP(config)# passwd sshpass
```

Step 17 Exit configuration mode:

```
pixP(config)# exit

pixP#
```

Step 18 Exit privileged mode:

```
pixP# exit

Logoff

Type help or '?' for a list of available commands.

pixP>
```

Step 19 Minimize your Telnet window. Do not close it.

Note Leave this Telnet session open throughout the rest of this lab exercise.

Step 20 Establish another SSH session to your PIX Firewall. When prompted to authenticate, enter **pix** as the username and **sshpass** as the password.

Task 5—Configure Command Authorization Using the Local User Database

Complete the following steps to configure local user authentication via a secure SSH session:

Step 1 Enter privileged mode. When prompted for a password, enter **prmode15**.

```
pixP>enable

Password:

pixP#
```

Step 2 Enter configuration mode:

```
pixP# config t
pixP(config)#
```

Step 3 Create three user accounts in the local database:

```
pixP(config)# username user10 password user10pass privilege 10
pixP(config)# username user12 password user12pass privilege 12
pixP(config)# username admin password adminpass privilege 15
```

Step 4 Enable authentication using the LOCAL database:

```
pixP(config)# aaa authentication enable console LOCAL
```

Step 5 Disconnect your SSH session.

Task 6—Test Command Authorization Using the Local User Database

Complete the following steps to test command authorization with local user authentication:

Step 1 Return to your Telnet session.

Step 2 Enter privileged mode. When prompted for a username, enter **user12**. When prompted for a password, enter **user12pass**.

```
pixP> enable
Username:
Password:
pixP#
```

Step 3 Enter configuration mode:

```
pixP# config t
pixP(config)#
```

Step 4 View the user account that is currently logged in:

```
pixP(config)# show curpriv
Username : user12
Current privilege level : 12
Current Mode/s : P_PRIV P_CONF
```

Step 5 Verify that you can use the **nameif** command by attempting to change Ethernet 4’s name and security level:

```
pixP(config)# nameif e4 BOB sec36
```

Step 6 View your configuration:

```
pixP(config)# show nameif
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
nameif ethernet3 PRIVTEST security30
nameif ethernet4 BOB security36
nameif ethernet5 intf5 security25
```

Step 7 Verify that you can user the **interface** command:

```
pixP(config)# interface e4 100full
```

Step 8 View your configuration:

```
pixP(config)# show int
interface ethernet0 "outside" is up, line protocol is up
  Hardware is i82559 ethernet, address is 0003.e300.486a
  IP address 192.168.P.2, subnet mask 255.255.255.0
  MTU 1500 bytes, BW 100000 Kbit full duplex
        0 packets input, 0 bytes, 0 no buffer
        Received 0 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        0 packets output, 0 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128)software
        (0/0)
        output queue (curr/max blocks): hardware (0/0) software (0/0)
interface ethernet1 "inside" is up, line protocol is up
  Hardware is i82559 ethernet, address is 0003.e300.486b
  IP address 10.0.P.1, subnet mask 255.255.255.0
  MTU 1500 bytes, BW 100000 Kbit full duplex
        6197 packets input, 597517 bytes, 0 no buffer
        Received 2231 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        4698 packets output, 356441 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128)software
        (0/5)
        output queue (curr/max blocks): hardware (1/3) software (0/2)
interface ethernet2 "dmz" is up, line protocol is up
  Hardware is i82558 ethernet, address is 00e0.b602.375b
  IP address 172.16.P.1, subnet mask 255.255.255.0
  MTU 1500 bytes, BW 100000 Kbit full duplex
        1890 packets input, 280534 bytes, 0 no buffer
        Received 1890 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        0 packets output, 0 bytes, 0 underruns
```

```
    0 output errors, 0 collisions, 0 interface resets
    0 babbles, 0 late collisions, 0 deferred
    0 lost carrier, 0 no carrier
    input queue (curr/max blocks): hardware (128/128)software
    (0/3)
    output queue (curr/max blocks): hardware (0/0) software (0/0)
interface ethernet3 "PRIVTEST" is up, line protocol is down
 Hardware is i82558 ethernet, address is 00e0.b602.375a
 IP address 127.0.0.1, subnet mask 255.255.255.255
 MTU 1500 bytes, BW 100000 Kbit full duplex
    0 packets input, 0 bytes, 0 no buffer
    Received 0 broadcasts, 0 runts, 0 giants
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    0 packets output, 0 bytes, 0 underruns
    0 output errors, 0 collisions, 0 interface resets
    0 babbles, 0 late collisions, 0 deferred
    0 lost carrier, 0 no carrier
    input queue (curr/max blocks): hardware (128/128)software
    (0/0)
    output queue (curr/max blocks): hardware (0/0) software (0/0)
interface ethernet4 "BOB" is up, line protocol is down
 Hardware is i82558 ethernet, address is 00e0.b602.3759
 IP address 127.0.0.1, subnet mask 255.255.255.255
 MTU 1500 bytes, BW 100000 Kbit full duplex
    0 packets input, 0 bytes, 0 no buffer
    Received 0 broadcasts, 0 runts, 0 giants
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    0 packets output, 0 bytes, 0 underruns
    0 output errors, 0 collisions, 0 interface resets
    0 babbles, 0 late collisions, 0 deferred
    0 lost carrier, 0 no carrier
    input queue (curr/max blocks): hardware (128/128)software
    (0/0)
    output queue (curr/max blocks): hardware (0/0) software (0/0)
interface ethernet5 "intf5" is up, line protocol is down
 Hardware is i82558 ethernet, address is 00e0.b602.3758
 IP address 127.0.0.1, subnet mask 255.255.255.255
 MTU 1500 bytes, BW 100000 Kbit full duplex
    0 packets input, 0 bytes, 0 no buffer
    Received 0 broadcasts, 0 runts, 0 giants
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    0 packets output, 0 bytes, 0 underruns
    0 output errors, 0 collisions, 0 interface resets
```

```
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128)software
        (0/0)
        output queue (curr/max blocks): hardware (0/0) software (0/0)
```

(where P = pod number)

Step 9 Try to create a static mapping for DMZ host 172.16.P.4:

```
pixP(config)# static (dmz,outside) 192.168.P.18 172.16.P.4 netmask
255.255.255.255
Command authorization failed
```

(where P = pod number)

Step 10 Log out of the user12 account:

```
pixP(config)# logout
Logoff

Type help or '?' for a list of available commands.

pixP>
```

Step 11 Log in to user 10's account. When prompted for a username, enter **user10**. When prompted for a password, enter **user10pass**.

```
pixP>login
Username:
Password:
pixP#
```

Step 12 Enter configuration mode:

```
pixP# config t
pixP(config)#
```

Step 13 Verify that you can use the **nameif** command by creating a name and security level for Ethernet 5:

```
pixP(config)# nameif e5 ALICE sec60
```

Step 14 View your configuration:

```
pixP(config)# show nameif
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
nameif ethernet3 PRIVTEST security30
nameif ethernet4 BOB security36
nameif ethernet5 ALICE security60
```

Step 15 Try to use the **interface** command to enable Ethernet 5 for 100 Mbps Ethernet full duplex communication:

```
pixP(config)# interface e5 100full
Command authorization failed
```

Step 16 Log out of the user10 account:

```
pixP(config)# logout

Logoff

Type help or '?' for a list of available commands.

pixP>
```

Step 17 Log in to user admin's account. When prompted for a username, enter **admin**. When prompted for a password, enter **adminpass**.

```
pixP>login
Username:
Password:
pixP#
```

Step 18 Enter configuration mode:

```
pixP# config t
pixP(config)#
```

Step 19 Clear your AAA configuration:

```
pixP(config)# clear aaa
```

Step 20 Save your configuration:

```
pixP(config)# write mem
Building configuration...
Cryptochecksum: a2d046eb daa27d65 f4a7a65f cdb3b13d
[OK]
```

Task 7—Perform a Password Recovery

Note If you are doing the lab exercises remotely using the PIX 520 Firewall, this task cannot be done.

Complete the following steps to perform a password recovery for the PIX Firewall model 515:

Step 1 Open and minimize the TFTP server on your desktop.

Step 2 Clear the translation table:

```
pixP(config)# clear xlate
```

Step 3 Create an enable password for entering into privileged mode:

```
pixP(config)# enable password badpassword
```

Step 4 Save your configuration:

```
pixP(config)# write memory
Building configuration...
Cryptochecksum: e18c684e d86c9171 9f63acf0 f64a8b43
[OK]
```

Step 5 Log out of the admin account:

```
pixP(config)# logout

Logoff

Type help or '?' for a list of available commands.

pixP>
```

Step 6 Attempt to enter privileged mode with the old password, **prmode15**:

```
pixP> enable
Password:
Invalid password:
```

Step 7 Enter privileged mode with the new password, **badpassword**:

```
Password:
pixP#
```

Step 8 Reboot your PIX Firewall and interrupt the boot process to enter monitor mode. To do this, press the Escape key or send a break character.

```
pixP# reload
```

Step 9 Specify the PIX Firewall interface to use for TFTP:

```
monitor> int 1
```

Step 10 Specify the PIX Firewall interface's IP address:

```
monitor> address 10.0.P.1
```

(where P = pod number)

Step 11 Verify connectivity to the TFTP server:

```
monitor> ping 10.0.P.11
```

(where P = pod number)

Step 12 Name the server:

```
monitor> server 10.0.P.11
```

(where P = pod number)

Step 13 Name the image filename:

```
monitor> file np63.bin
```

Step 14 Start the TFTP process:

```
monitor> tftp
```

```
tftp
np63.bin@10.0.P.11...............................................



Received 73728 bytes


Cisco Secure PIX Firewall password tool (3.0) #0: Wed Mar 27 11:02:16
PST 2002

Flash=i28F640J5 @ 0x300

BIOS Flash=AT29C257 @ 0xd8000
```

(where P = pod number)

Step 15 When prompted, press **y** to erase the password:

```
Do you wish to erase the passwords? [yn] y

The following lines will be removed from the configuration:

        enable password GlFe5rCOwv2JUi5H level 5 encrypted

        enable password .7P6WvOReYzHKnus level 10 encrypted

        enable password tgGMO76/Nf26X5Lv encrypted

        passwd w.UT.4mPsVA418Ij encrypted


Do you want to remove the commands listed above from the
configuration? [yn]

Please enter a y or n.
```

Step 16 When prompted to remove the commands listed from the configuration, press **y**:

```
Do you want to remove the commands listed above from the
configuration? [yn] y

Passwords have been erased.
```

The system automatically erases the passwords and starts rebooting.

Step 17 Verify that the password **badpassword** has been erased by entering privileged mode on your PIX Firewall:

```
pix> enable

password: <Enter>

pixP#
```

Task 8—Upgrade the PIX Firewall Software Image

Complete the following steps to load the PIX 515 Firewall image using TFTP:

Step 1 Use the **copy tftp flash** command to load the image file pix631.bin:

```
pixP# copy tftp://10.0.P.11/pix631.bin flash:image
```

(where P = pod number)

Step 2 After the PIX Firewall has received the image from the TFTP server and you see the message "Image installed," reload the PIX Firewall. When prompted to confirm, press **Enter**.

```
pixP# reload
Proceed with reload? [confirm] <Enter>
```

Step 3 Enter the **show version** command to verify that you have loaded PIX Firewall Software Version 6.3(1):

```
pixP> show version
Cisco PIX Firewall Version 6.3(1)
Cisco PIX Device Manager Version 3.0(1)

Compiled on Wed 19-Mar-03 15:14 by morlee

pixP up 34 mins 52 secs

Hardware:   PIX-515, 64 MB RAM, CPU Pentium 200 MHz
Flash i28F640J5 @ 0x300, 16MB
BIOS Flash AT29C257 @ 0xfffd8000, 32KB

0: ethernet0: address is 0003.e300.486a, irq 10
1: ethernet1: address is 0003.e300.486b, irq 7
2: ethernet2: address is 00e0.b602.375b, irq 11
3: ethernet3: address is 00e0.b602.375a, irq 11
4: ethernet4: address is 00e0.b602.3759, irq 11
5: ethernet5: address is 00e0.b602.3758, irq 11
Licensed Features:
Failover:           Enabled
VPN-DES:            Enabled
VPN-3DES:           Enabled
Maximum Interfaces: 6
Cut-through Proxy:  Enabled
Guards:             Enabled
URL-filtering:      Enabled
Inside Hosts:       Unlimited
Throughput:         Unlimited
IKE peers:          Unlimited

This PIX has an Unrestricted <UR> license.

Serial Number: 480430946 (0x1ca2cb62)
Running Activation Key: 0xf4e352a3 0xef857686 0x468be692 0xbd984b0b
Configuration last modified by  at 18:20:17.510 UTC Thu Sep 18 2003
```

18

Cisco PIX Device Manager

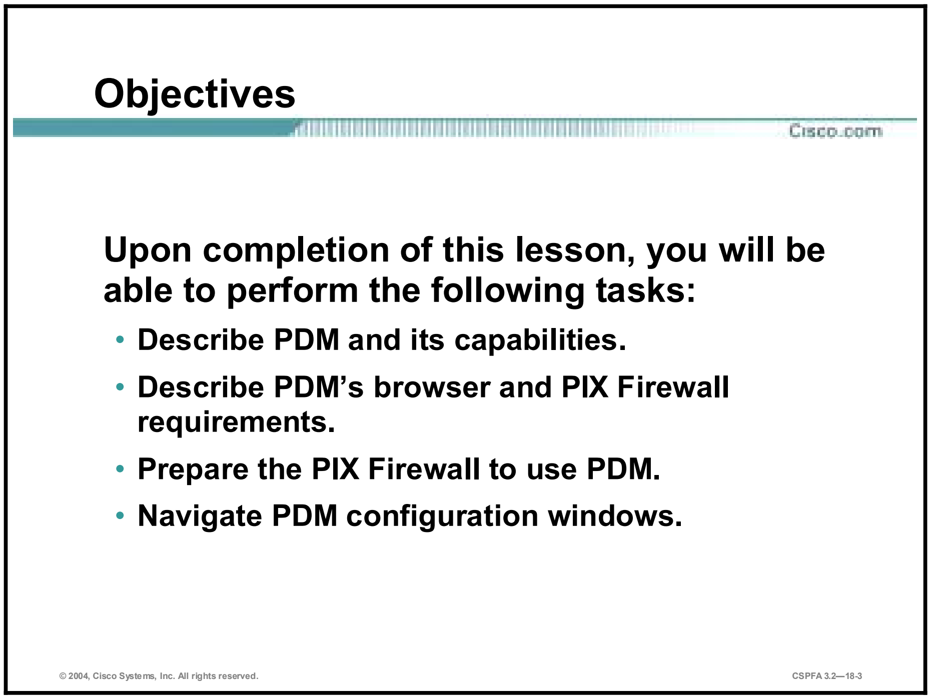
Overview

This lesson includes the following topics:

- Objectives
- PDM overview
- PDM operating requirements
- Prepare for PDM
- Using PDM to configure the PIX Firewall
- Using PDM to create site-to-site VPNs
- Using PDM to create remote access VPNs
- Summary
- Lab exercise

Objectives

This topic lists the lesson's objectives.

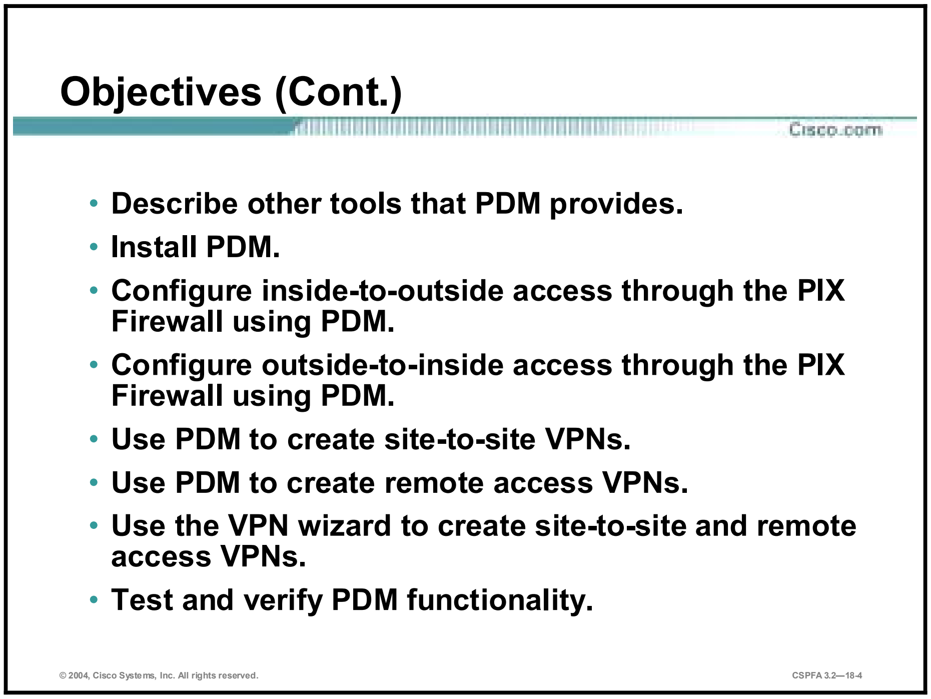


Objectives

Upon completion of this lesson, you will be able to perform the following tasks:

- Describe PDM and its capabilities.
- Describe PDM's browser and PIX Firewall requirements.
- Prepare the PIX Firewall to use PDM.
- Navigate PDM configuration windows.

CSPFA 3.2—18-3



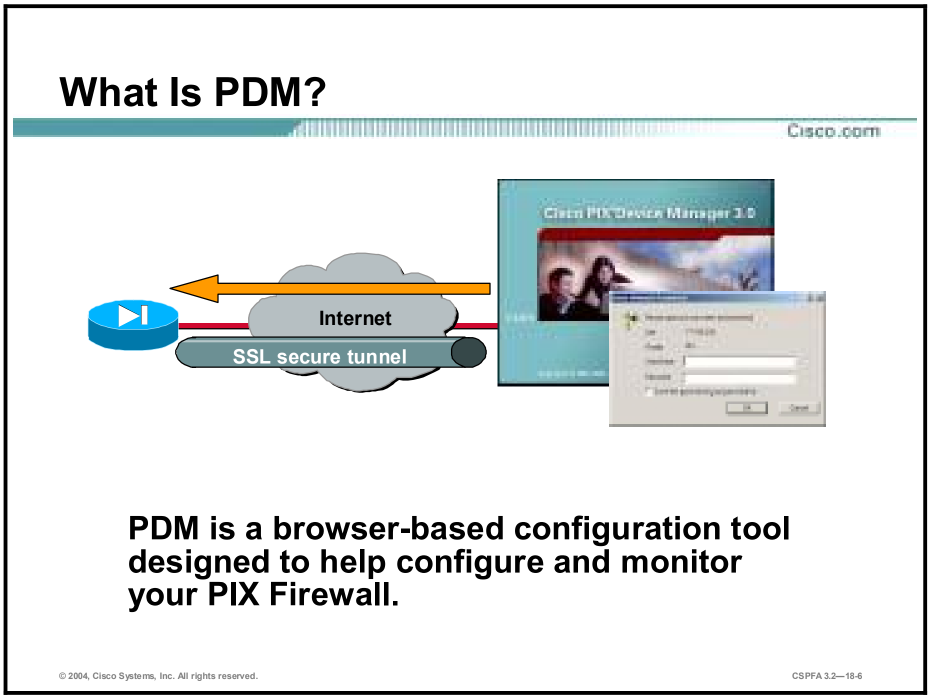
Objectives (Cont.)

- Describe other tools that PDM provides.
- Install PDM.
- Configure inside-to-outside access through the PIX Firewall using PDM.
- Configure outside-to-inside access through the PIX Firewall using PDM.
- Use PDM to create site-to-site VPNs.
- Use PDM to create remote access VPNs.
- Use the VPN wizard to create site-to-site and remote access VPNs.
- Test and verify PDM functionality.

CSPFA 3.2—18-4

PDM Overview

This topic provides an overview of the Cisco PIX Device Manager (PDM) and its limitations.



What Is PDM?

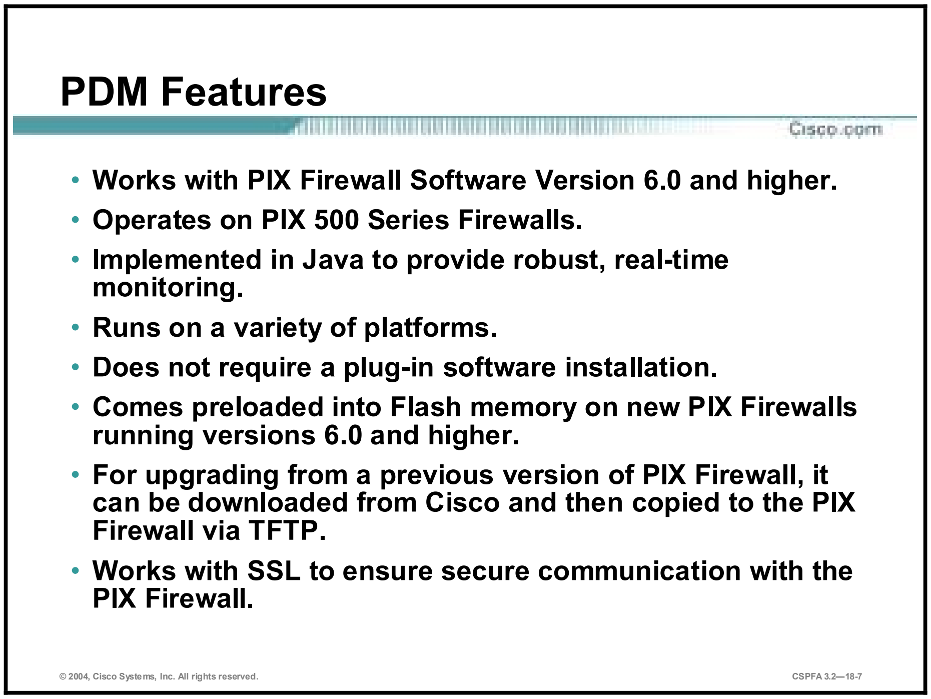
PDM is a browser-based configuration tool designed to help configure and monitor your PIX Firewall.

CSPFA 3.2—18-6

PDM is a browser-based configuration tool designed to help you set up, configure, and monitor your Cisco PIX Firewall graphically, without requiring an extensive knowledge of the PIX Firewall command line interface (CLI).

PDM monitors and configures a single PIX Firewall. You can use PDM to create a new configuration. You can use PDM to monitor and maintain current PIX Firewalls. You can point your browser to more than one PIX Firewall and administer several PIX Firewall from a single workstation.

Note PDM can also be used to configure and monitor the Firewall Services Module on a Cisco Catalyst 6500 switch. This use of PDM is discussed later in the course.



PDM Features

Cisco.com

- Works with PIX Firewall Software Version 6.0 and higher.
- Operates on PIX 500 Series Firewalls.
- Implemented in Java to provide robust, real-time monitoring.
- Runs on a variety of platforms.
- Does not require a plug-in software installation.
- Comes preloaded into Flash memory on new PIX Firewalls running versions 6.0 and higher.
- For upgrading from a previous version of PIX Firewall, it can be downloaded from Cisco and then copied to the PIX Firewall via TFTP.
- Works with SSL to ensure secure communication with the PIX Firewall.

CSPFA 3.2—18-7

PDM is secure, versatile, and easy to use. It works with PIX 500 Series Firewalls and runs on a variety of platforms.

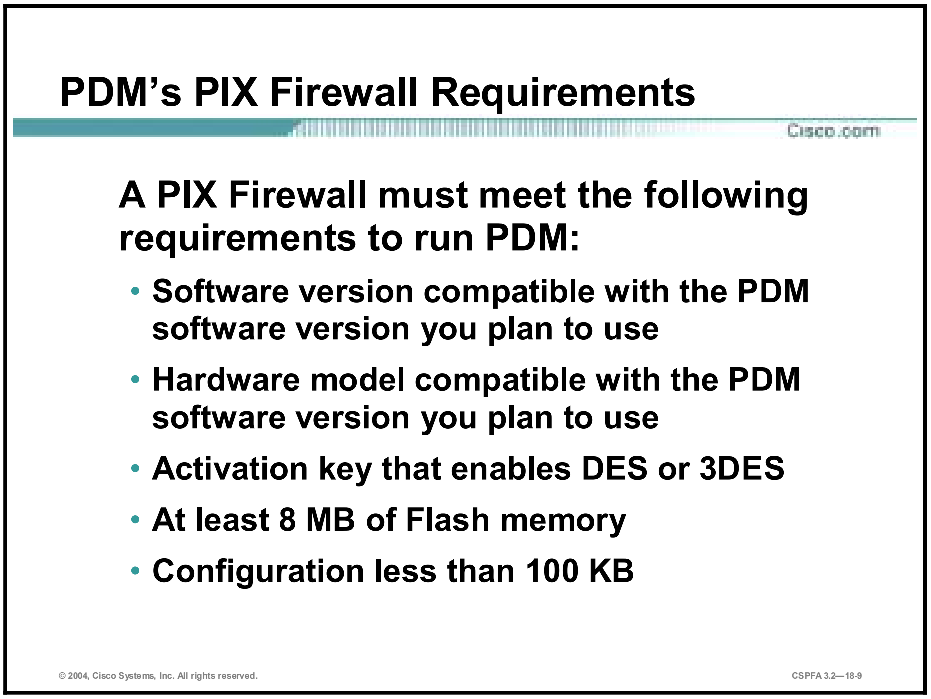
PDM enables you to securely configure and monitor your PIX Firewall remotely. Its ability to work with the Secure Socket Layer (SSL) protocol ensures that communication with the PIX Firewall is secure, and because it is implemented in Java, it is able to provide robust, real-time monitoring.

PDM works with PIX Firewall Software Versions 6.0 and higher and comes preloaded into Flash memory on new PIX Firewalls running Software Versions 6.0 and higher. If you are upgrading from a previous version of the PIX Firewall, you can download PDM from Cisco and then copy it to the PIX Firewall via TFTP.

PDM runs on Windows, Sun Solaris, and Linux platforms and requires no plug-ins or complex software installations. The PDM applet uploads to your workstation when you access the PIX Firewall from your browser.

PDM Operating Requirements

This topic discusses the system requirements for PDM.



PDM's PIX Firewall Requirements

Cisco.com

A PIX Firewall must meet the following requirements to run PDM:

- Software version compatible with the PDM software version you plan to use
- Hardware model compatible with the PDM software version you plan to use
- Activation key that enables DES or 3DES
- At least 8 MB of Flash memory
- Configuration less than 100 KB

CSPFA 3.2—18-9

A PIX Firewall must meet the following requirements to run PDM:

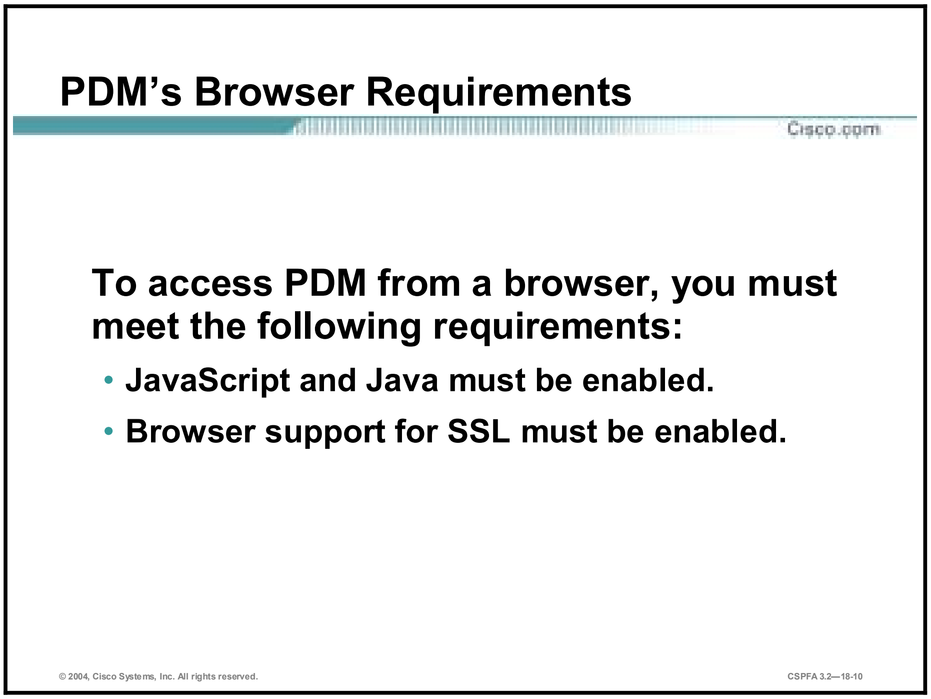
Note New PIX Firewalls that contain version 6.0 also have a preinstalled Data Encryption Standard (DES) activation key. If you are using a new PIX Firewall, you have all the requirements discussed in this topic and you can continue to the next topic.

- You must have an activation key that enables DES or the more secure Triple-Data Encryption Standard (3DES), which PDM requires for support of the SSL protocol. If your PIX Firewall is not enabled for DES, you can have a new activation key sent to you by completing the form at the following web site: http://www.cisco.com/kobayashi/sw-center/internet/pix-56bit-license-request.shtml.
- Verify that your PIX Firewall meets all requirements listed in the release notes for the PIX Firewall software version you are using.
- Verify that your PIX Firewall hardware model, PIX Firewall software version, and PDM version are compatible. Refer to the table below to ensure compatibility. You can download PIX Firewall software and the PDM software from the following web site: http://www.cisco.com/cgi-bin/tablebuild.pl/pix.

PDM Version	PIX Firewall Software Version	PIX Firewall Model
1.0	6.0 or 6.1	506, 515, 520, 525, 535
1.1	6.0 or 6.1	506, 515, 520, 525, 535
2.0	6.2	501,506/506E, 515/515E, 520, 525, 535

PDM Version	PIX Firewall Software Version	PIX Firewall Model
2.1	6.2	501, 506/506E, 515/515E, 520, 525, 535
3.0	6.3	501, 506/506E, 515/515E, 520, 525, 535

- You must have at least 8 MB of Flash memory on the PIX 501 and 506/506E Firewalls.
- You must have at least 16 MB of Flash memory on the PIX 515/515E, 520, 525, and 535 Firewalls.
- Ensure that your configuration is less than 100 KB (approximately 1,500 lines). Configurations over 100 KB cause PDM performance degradation.



PDM's Browser Requirements

Cisco.com

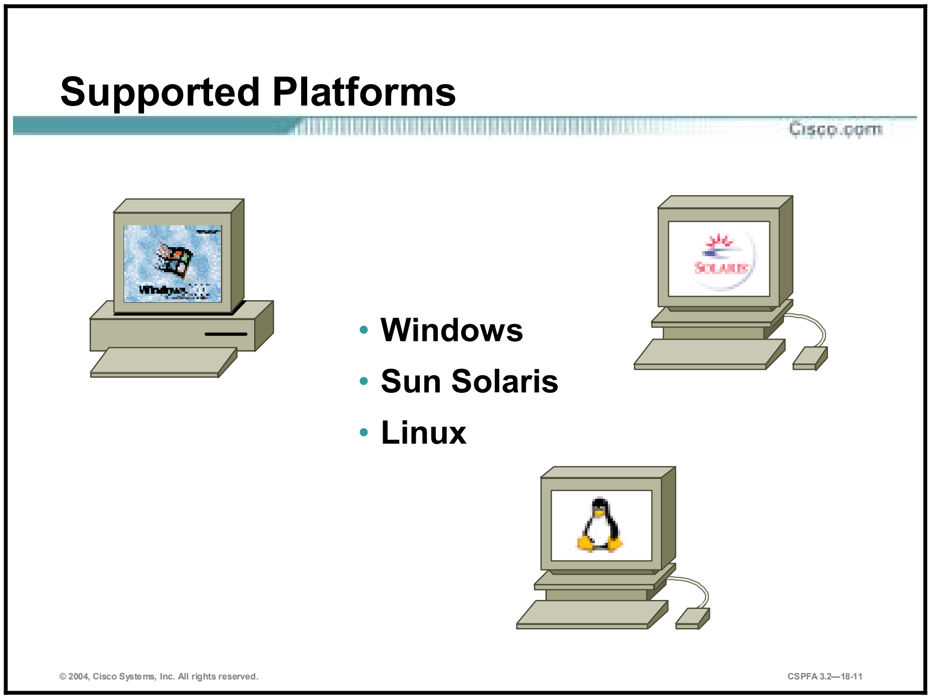
To access PDM from a browser, you must meet the following requirements:

- **JavaScript and Java must be enabled.**
- **Browser support for SSL must be enabled.**

CSPFA 3.2—18-10

To access PDM from a browser, you must meet the following requirements:

- JavaScript and Java must be enabled. If these are not enabled, PDM helps you enable them. If you are using Microsoft Internet Explorer, your Java Development Kit (JDK) version should be 1.1.4 or higher. To check which version you have, launch PDM. In the main PDM menu, click **Help>About Cisco PIX Device Manager**. When the About PDM information window opens, it displays your browser specifications in a table, including your JDK version. If you have an older JDK version, you can use the latest Java Virtual Machine (JVM) to enable Java to run on your computer. Download the product named Virtual Machine from Microsoft.
- Browser support for SSL must be enabled. The supported versions of Internet Explorer and Netscape Navigator support SSL without requiring additional configuration.



Supported Platforms

Cisco.com

- Windows
- Sun Solaris
- Linux

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—18-11

PDM can operate in browsers running on Windows, SUN, Solaris, or Linux operating systems. The requirements for each operating system follow:

Windows Requirements

The following requirements apply to the use of PDM with Windows:

- Windows 2000 (Service Pack 3), Windows NT 4.0 (Service Pack 4 and higher), Windows 98, Windows ME, or Windows XP.
- The supported browsers are Internet Explorer 5.5 or higher, and Netscape Communicator 4.7x or 7.0x. PDM does not support Netscape 6.x.
- Any Pentium or Pentium-compatible processor running at 450 MHz or higher.
- At least 256 MB of RAM.
- An 1024 x 768 pixel display with at least 256 colors.
- PDM does not support use on Windows 3.1 or Windows 95.

Note The use of virus checking software may dramatically increase the time required to start PDM. This is especially true for Netscape Communicator on any Windows platform or Windows 2000 running any browser.

SUN Solaris Requirements

The following requirements apply to the use of PDM with Sun SPARC:

- Sun Solaris 2.8 or 2.9 running CDE window manager
- SPARC microprocessor
- Netscape Communicator 4.78

- At least 128 MB of RAM
- An 1024 x 768 pixel display with at least 256 colors

Note PDM does not support Solaris on IBM PCs.

Linux Requirements

The following requirements apply to the use of PDM with Linux:

- Red Hat Linux 7.0, 7.1, 7.2, or 7.3 or 8.0 running the GNOME or KDE 2.0 desktop environment
- Netscape Communicator 4.7x on Red Hat 7.x. or Mozilla 1.0.1 on Red Hat 8.0
- At least 128 MB of RAM
- An 1024 x 768 pixel display with at least 256 colors

General Guidelines

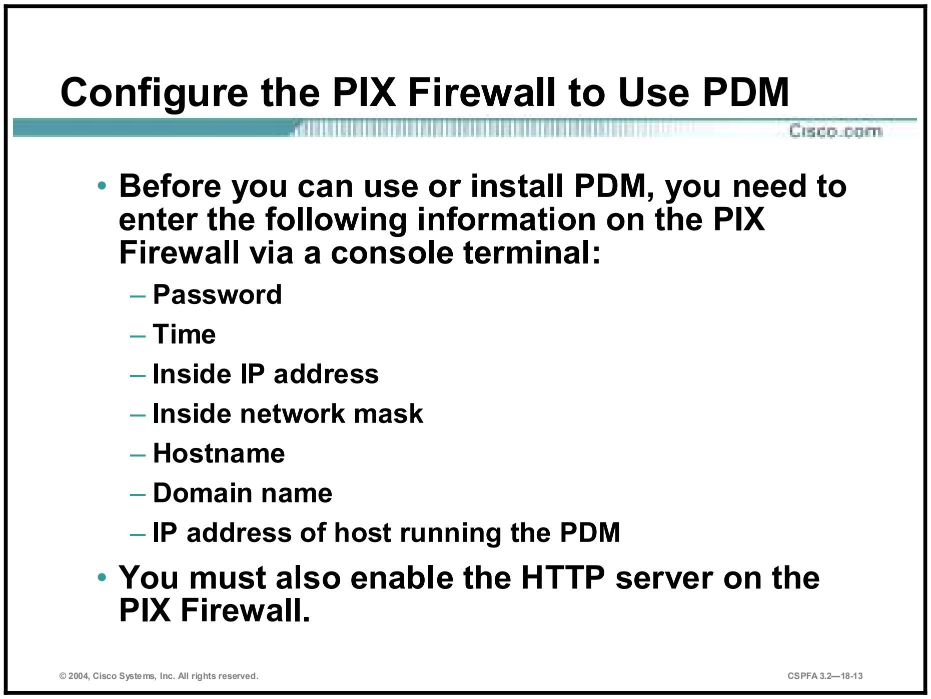
The following are a few general guidelines for workstations running PDM:

- You can run several PDM sessions on a single workstation. The maximum number of PDM sessions you can run varies depending on your workstation's resources such as memory, CPU speed, and browser type.
- The time required to download the PDM applet can be greatly affected by the speed of the link between your workstation and the PIX Firewall. A minimum 56-Kbps link speed is required; however, 384 Kbps or higher is recommended. After the PDM applet is loaded on your workstation, the link speed impact on PDM operation is negligible.
- The use of virus-checking software may dramatically increase the time required to start PDM. This is especially true for Netscape Communicator on any Windows platform or Windows 2000 running any browser.

If your workstation's resources are running low, you should close and reopen your browser before launching PDM.

Prepare for PDM

This topic details the configuration of the PIX Firewall to enable the use of PDM.



Configure the PIX Firewall to Use PDM

Cisco.com

- Before you can use or install PDM, you need to enter the following information on the PIX Firewall via a console terminal:
  - Password
  - Time
  - Inside IP address
  - Inside network mask
  - Hostname
  - Domain name
  - IP address of host running the PDM
- You must also enable the HTTP server on the PIX Firewall.

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—18-13

The PIX Firewall must be configured with the following information before you can install or use PDM. You can either preconfigure a new PIX Firewall through the interactive prompts, which appear after the PIX Firewall boots, or you can enter the commands shown below each information item.

- Enable Password—Enter an alphanumeric password to protect the PIX Firewall's privileged mode. The alphanumeric password can be up to 16 characters in length. You must use this password to log in to PDM. The command syntax for enabling a password is as follows:

  ```
  enable password password [encrypted]
  ```

- Time—Set the PIX Firewall clock to Universal Coordinated Time (UTC, also known as Greenwich Mean Time, or GMT). For example, if you are in the Pacific Daylight Savings time zone, set the clock 7 hours ahead of your local time to set the clock to UTC. Enter the year, month, day, and time. Enter the UTC time in 24-hour time as hour:minutes:seconds. The command syntax for setting the clock is as follows:

  ```
  clock set hh:mm:ss day month year
  ```

- Inside IP address—Specify the IP address of the PIX Firewall's inside interface. Ensure that this IP address is unique on the network and not used by any other computer or network device, such as a router. The command syntax for setting an inside IP address is as follows:

  ```
  ip address if_name ip_address [netmask]
  ```

- Inside network mask—Specify the network mask for the inside interface. An example mask is 255.255.255.0. You can also specify a subnetted mask (for example, 255.255.255.224).

Do not use all 255s, such as 255.255.255.255; this prevents traffic from passing on the interface. Use the **ip address** command shown above to set the inside network mask.

- Hostname—Specify up to 16 characters as a name for the PIX Firewall. The command syntax for setting a hostname is as follows:

```
hostname newname
```

- Domain name—Specify the domain name for the PIX Firewall. The command syntax for enabling domain name is as follows:

```
domain-name name
```

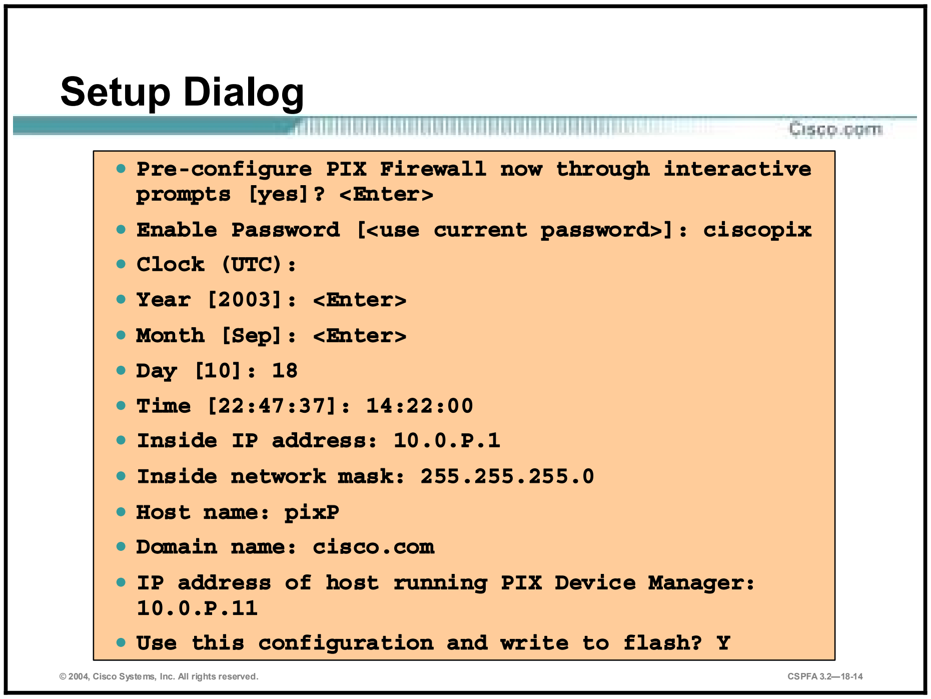
- IP address of the host running PDM—Specify the IP address of the workstation that will access PDM from its browser. The command syntax for granting permission for a host to connect to the PIX Firewall with SSL is as follows:

```
http ip_address [netmask] [if_name]
```

- HTTP Server—Enable the HTTP server on the PIX Firewall with the **http server enable** command. You must also use the **http ip_address** command to specify the host or network authorized to initiate an HTTP connection to the PIX Firewall.

If you are installing PDM on a PIX Firewall with an existing configuration, you may need to restructure your configuration from the PIX Firewall CLI before installing PDM in order to obtain full PDM capability. There are certain commands that PDM does not support in a configuration. If these commands are present in your configuration, you will only have access to the Monitoring tab. This is because PDM handles each PIX Firewall command in one of the following ways, each of which is explained in detail in the document titled “PDM Support for PIX Firewall CLI Commands” on Cisco.com:

- Parse and allow changes (supported commands)
- Parse and only permit access to the Monitoring tab (unsupported commands)
- Parse without allowing changes (commands PDM does not understand but handles without preventing further configuration)
- Only display in the unparsable command list (commands PDM does not understand but handles without preventing further configuration)



Setup Dialog

Cisco.com

- Pre-configure PIX Firewall now through interactive prompts [yes]? <Enter>
- Enable Password [<use current password>]: ciscopix
- Clock (UTC):
- Year [2003]: <Enter>
- Month [Sep]: <Enter>
- Day [10]: 18
- Time [22:47:37]: 14:22:00
- Inside IP address: 10.0.P.1
- Inside network mask: 255.255.255.0
- Host name: pixP
- Domain name: cisco.com
- IP address of host running PIX Device Manager: 10.0.P.11
- Use this configuration and write to flash? Y

CSPFA 3.2—18-14

An unconfigured PIX Firewall starts in an interactive setup dialog to enable you to perform the initial configuration required to use PDM. You can also access the setup dialog by entering **setup** at the configuration mode prompt.

The dialog asks for several responses, including the inside IP address, network mask, hostname, domain name and PDM host. The hostname and domain name are used to generate the default certificate for the SSL connection.

The example in the figure shows how to respond to the **setup** command prompts. Pressing the **Enter** key instead of entering a value at the prompt accepts the default value within the brackets. You must fill in any fields that show no default values, and change default values as necessary. After the configuration is written to Flash memory, your PIX Firewall is ready to start PDM.

Note The clock must be set for PDM to generate a valid certification. Set the PIX Firewall clock to Universal Coordinated Time (also known as Greenwich Mean Time).

The following table explains each prompt in the setup dialog:

- Enable password—Enables you to specify an enable password for this PIX Firewall.
- Clock (UTC)—Enables you to set the PIX Firewall clock to Universal Coordinated Time (also known as Greenwich Mean Time).
- Year [system year]—Enables you to specify the current year, or return to the default year stored in the host computer.
- Month [system month]—Enables you to specify the current month, or return to the default month stored in the host computer.
- Day [system day]—Enables you to specify the current day, or return to the default day stored in the host computer.

- Time [system time]—Enables you to specify the current time in hh:mm:ss format, or return to the default time stored in the host computer.
- Inside IP address—The network interface IP address of the PIX Firewall.
- Inside network mask—A network mask that applies to the inside IP address. Use 0.0.0.0 to specify a default route. The 0.0.0.0 netmask can be abbreviated as 0.
- Host name—The hostname you want to display in the PIX Firewall command line prompt.
- Domain name—The DNS domain name of the network on which the PIX Firewall runs (for example, cisco.com).
- IP address of host running PIX Device Manager—IP address on which PDM connects to the PIX Firewall.
- Use this configuration and write to flash?—Enables you to store the new configuration to Flash memory. It is the same as the **write memory** command. If the answer is yes, the inside interface is enabled and the requested configuration is written to Flash memory. If the user answers anything else, the setup dialog repeats using the values already entered as the defaults for the questions.

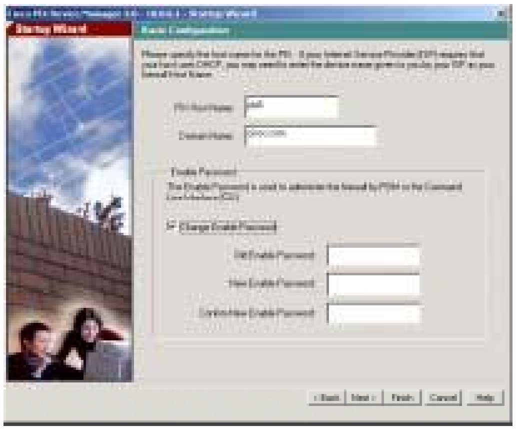
Using PDM to Configure the PIX Firewall

This topic discusses getting started with PDM and the layout of the product.

Startup Wizard

Cisco.com

The PDM Startup Wizard enables you to easily perform basic configuration of the PIX Firewall.



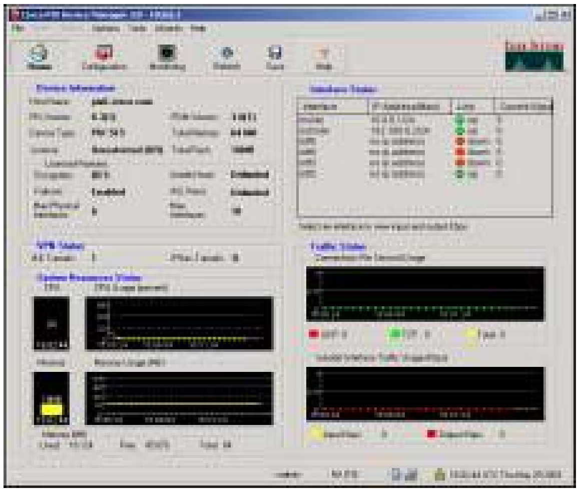
CSPFA 3.2—18-16

The PDM Startup Wizard is an easy way to begin the process of configuring your PIX Firewall. The wizard steps you through such tasks as the following:

- Enabling the PIX Firewall interfaces
- Assigning IP addresses to the interfaces
- Configuring a hostname and password
- Configuring Point-to-Point Protocol over Ethernet (PPPoE)
- Configuring Auto Update
- Configuring Network Address Translation (NAT) and Port Address Translation (PAT)
- Configuring the Dynamic Host Configuration Protocol (DHCP) server

The Startup Wizard can be run at any time by choosing **Tools>Startup Wizard**.

PDM Home Window



Main toolbar

Device information

VPN status

System resources

Interface status

Traffic status

The PDM Home window enables the administrator to view important information about the PIX Firewall, such as the status of the interfaces, the version running, licensing information, and performance. Many of the details available on the PDM Home window are available elsewhere in PDM, but the Home window provides a useful and quick way to see how the PIX Firewall is running. All information on the Home window is updated every ten seconds, except for the Device Information. The administrator can access the Home window any time by clicking the Home button on the main toolbar.

The following sections are included in the PDM Home window:

- Main toolbar—Provides quick access to the Home window, configuration panels, PDM monitoring, and context-sensitive help. The administrator can also save the running configuration to Flash memory by clicking the Save button, or reload the running configuration from Flash by clicking the Refresh button.
- Device Information—Displays the hostname, PIX Firewall version, device type, license, PDM version, total memory, and total Flash.
- VPN Status—Displays the status of virtual private network (VPN) tunnels, if they are configured.
- System Resources Status—Displays CPU and memory usage.
- Interface Status Interface—Displays the interface, IP address and mask, and link status.
- Traffic Status—Displays the number of TCP and UDP connections that occur each second. Their sum is displayed as the total number of connections. The “outside” Interface Traffic Usage area displays the traffic going through the outside interface in kilobits per second.

Overall Layout

Cisco.com

PDM consists of five major configuration areas:

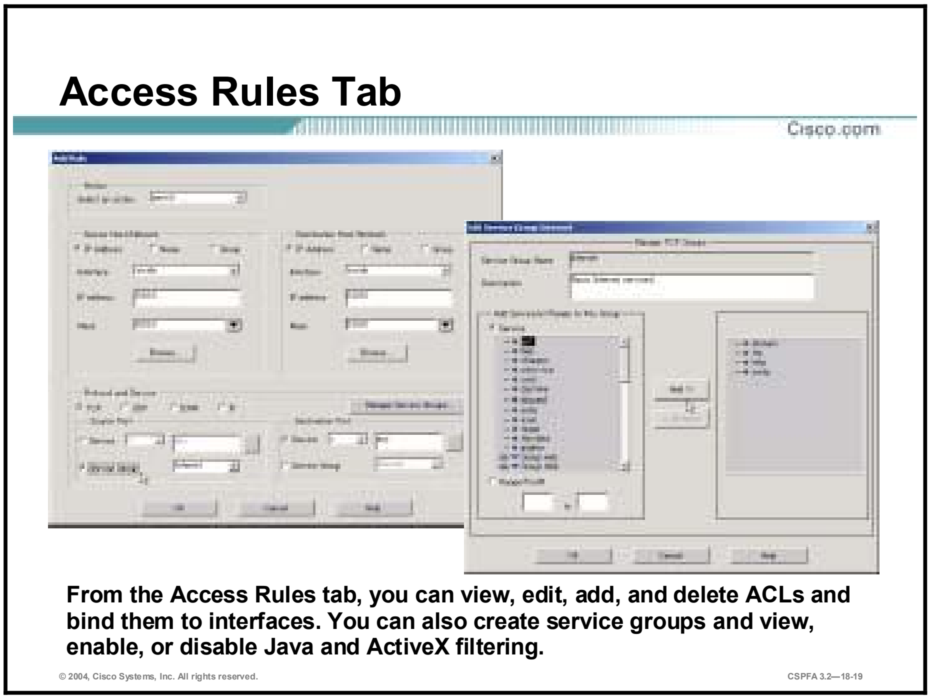
- **Access Rules**
- **Translation Rules**
- **VPN**
- **Hosts/Networks**
- **System Properties**

CSPFA 3.2—18-18

PDM consists of five tabs, which enable you to configure various aspects of the product:

- Access Rules—Shows your entire network security policy
- Translation Rules—Enables you to view all the address translation rules applied to your network
- VPN—Enables you to create VPNs using IPSec
- Hosts/Networks—Enables you to view, edit, add to, or delete from the list of hosts and networks defined for the selected interface
- System Properties—Enables you to configure many aspects of the PIX Firewall

The remainder of this topic covers each configuration area in detail.



Access Rules Tab

Cisco.com

From the Access Rules tab, you can view, edit, add, and delete ACLs and bind them to interfaces. You can also create service groups and view, enable, or disable Java and ActiveX filtering.

CSPFA 3.2—18-19

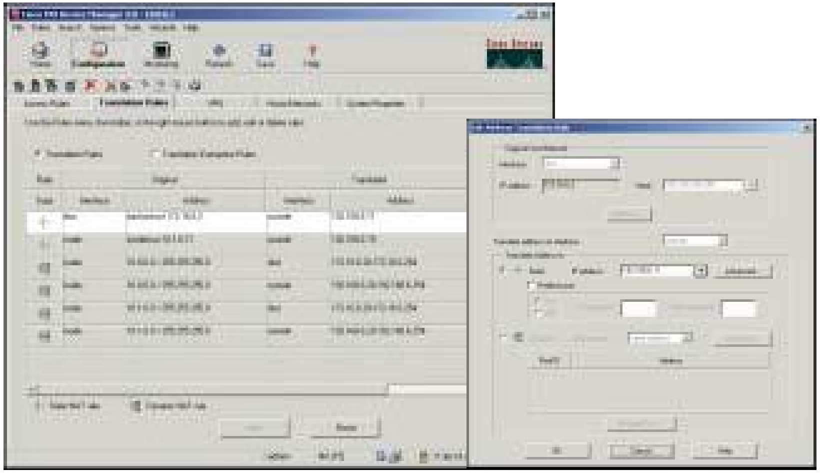
The Access Rules tab shows your entire network security policy expressed in rules. This tab includes a panel for Access Rules, as well as for AAA Rules and Filter Rules. When you click the **Access Rules** option button, this tab lets you define access control lists (ACLs), and in a PIX Firewall that supports it, to control the access of a specific host or network to another host/network, including the protocol or port that can be used.

This tab also enables you to define authentication, authorization, or accounting rules, and filter rules for ActiveX and Java. The configuration edits you perform on the Access Rules tab are captured by PDM but are not sent to the PIX Firewall until you click **Apply**. This applies to all configuration performed with PDM, including those performed in the Translation Rules tab, the Hosts/Networks tab, and the System Properties tab. Always click **Apply** to send your configuration edits to the PIX Firewall. Also remember, it is very important to save your configuration to Flash memory by choosing **File>Write Configuration to Flash** from the main menu or clicking the **Save** icon in the toolbar.

Note You can also use the Access Rules tab to create object groups and apply them to ACLs.

Translation Rules Tab

Cisco.com



From the Translation Rules tab, you can view, edit, create, and delete static and dynamic address translation rules.

CSPFA 3.2—18-20

The Translation Rules tab lets you view all the address translation rules or NAT exemption rules applied to your network. Before you can designate access and translation rules for your network, you must first define each host or server for which a rule will apply. To do this, select the **Hosts/Networks** tab to define hosts and networks.

When you are working in either the Access Rules or the Translation Rules tabs, you can access the task menus used for modifying rules three ways:

- The PDM toolbar
- The Rules menu
- Right-clicking anywhere in the rules table

Note The order in which you apply translation rules can affect the way the rules operate. PDM lists the static translations first and then the dynamic translations. When processing NAT, the PIX Firewall first translates the static translations in the order they are configured. You can use the Insert Before or Insert After command from the Rules menu to determine the order in which static translations are processed. Because dynamically translated rules are processed on a best-match basis, the option to insert a rule before or after a dynamic translation is disabled.

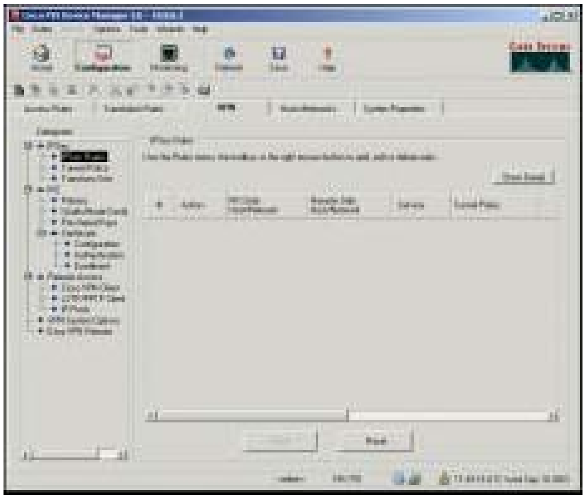
The Manage Global Address Pools window enables you to create global address pools to be used by NAT. From this window, you can also view or delete existing global pools. You can access the Manage Global Address Pools window from the Manage Pools button on the Translation Rules tab.

Remember that it is necessary to run NAT even if you have routable IP addresses on your secure networks. This is a unique feature of the PIX Firewall. You can do this by translating the IP address to itself on the outside.

VPN Tab

Cisco.com

From the VPN tab, you can create site-to-site and remote access VPNs.



CSPFA 3.2—18-21

From the VPN tab, you can create site-to-site or remote access VPNs. This is covered in depth later in the lesson.

Hosts/Networks Tab

Cisco.com

From the Hosts/Networks tab, you can view, edit, add, or delete hosts, networks, and network groups.

CSPFA 3.2—18-22

The PDM requires that you define any host or network that you intend to use in ACLs and translation rules. These hosts or networks are organized below the interface from which they are reachable. When defining either type of rule, you can reference a host or network by clicking the **Browse** button in the appropriate add or edit rule window. Additionally, you can reference the host or network by name if a name is defined for that host or network. It is recommended that you name all hosts and networks.

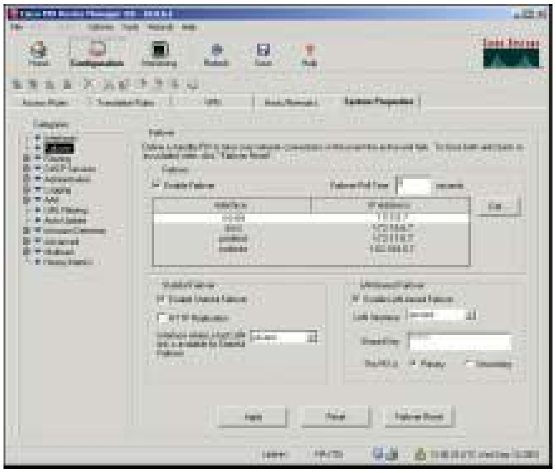
In addition to defining the basic information for these hosts or networks, you can define route settings and translation rules (NAT) for any host or network. You can also configure route settings in the Static Route panel on the System Properties tab and translation rules on the Translation Rules tab. These different configuration options accomplish the same results. The Hosts/Networks tab provides another view to modify these settings on a per host and per network basis.

The information provided in this window enables the basic identification information for that host or network. This includes values for the IP address, netmask, interface, and name of the host or network. PDM uses the name and IP address and netmask pair to resolve references to this host or network in the source and destination conditions of access rules and in translation rules. PDM uses the interface value to apply access and translation rules that reference this host or network to the correct interface. The interface delivers network packets to the host or network; therefore, it enforces the rules that reference that host or network.

System Properties Tab

From the System Properties tab, you can configure such features as the following:

- Interfaces
- Failover
- Routing
- User accounts for command authorization
- DHCP server
- Privilege level for command authorization
- Logging
- AAA
- URL filtering
- Remote management
- Intrusion detection
- Turbo ACLs
- Multicast



CSPFA 3.2—18-23

The System Properties tab enables you to configure many aspects of the PIX Firewall, including the following:

- Interfaces   In addition to their names, the Interfaces panel displays and enables you to edit additional configuration information required for each interface. You can configure a PIX Firewall interface with a static IP address, VLAN ID, or you can configure it to use DHCP or PPPoE.

Note Your configuration edits are captured by PDM but not sent to the PIX Firewall until **Apply to PIX** is clicked.

- Failover—Enables you to enable, disable, and configure serial and LAN-based failover and stateful failover.
- Routing   The routing panel is divided into the following four sections dealing with different routing configurations:
  - — RIP
  - — Static routes
  - — Proxy ARPs
  - — OSPF
- DHCP Services—The DHCP Services panel enables you to configure the PIX Firewall as a DHCP server or configure the PIX Firewall as a DHCP relay agent. You cannot configure both simultaneously on the same PIX Firewall.
- PIX Administration Users   This panel enables you to create local user accounts.
- PIX Administration—This panel contains the following sections:
  - — Device
  - — Password

  - Authentication/Authorization
  - User Accounts
  - Banner
  - Console
  - PDM/HTTPS
  - Telnet
  - Secure Shell
  - Management Access
  - SNMP
  - ICMP
  - TFTP Server
  - Clock
  - NTP
- Logging—This panel is divided into the following sections:
  - Logging Setup
  - PDM Logging
  - Syslog
  - Others
- AAA—This panel contains the following sections:
  - URL Filtering
  - Auto Update
- Intrusion Detection This panel is divided into the following two sections:
  - IDS Policy
  - IDS Signatures
- Advanced This panel is made up of the five panels listed below, with the FixUp panel having further selections nested beneath it.
  - Fixup
    - CTIQBE
    - ESP-IKE
    - FTP
    - H.323 H225
    - H.323 RAS
    - HTTP
    - ICMP Error
    - ILS
    - MGCP
    - PPTP

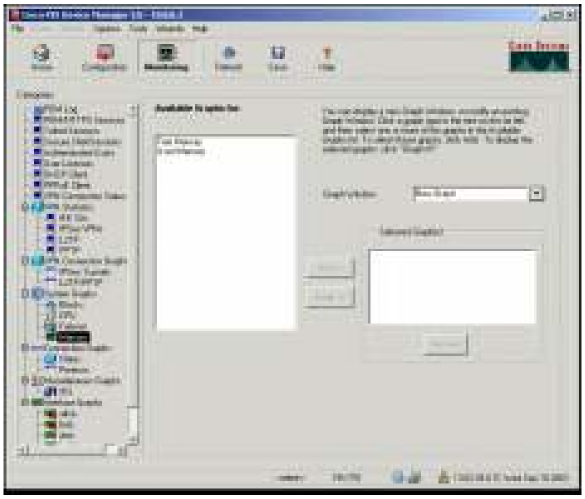
    - RSH
    - RTSP
    - SIP over TCP
    - SIP over UDP
    - Skinny
    - SMTP
    - SQL*Net
  - Anti-Spoofing
  - Fragment
  - TCP Options
  - Timeouts
  - Turbo Access Rules
- Multicast
  - Stub Multicast Routing
  - IGMP
  - MRoute
- History Metrics   This panel enables the PIX Firewall to keep a history of many statistics, which can be displayed by PDM through the Monitoring tab.

Note If PDM History Metrics is not enabled, the only view available in the Monitoring tab is the "Real-time" view. PDM History Metrics is enabled by default.

Monitoring Button

Cisco.com

The Monitoring button enables you to monitor per-interface statistics, such as packet counts and bit rates, for each enabled interface on the PIX Firewall.

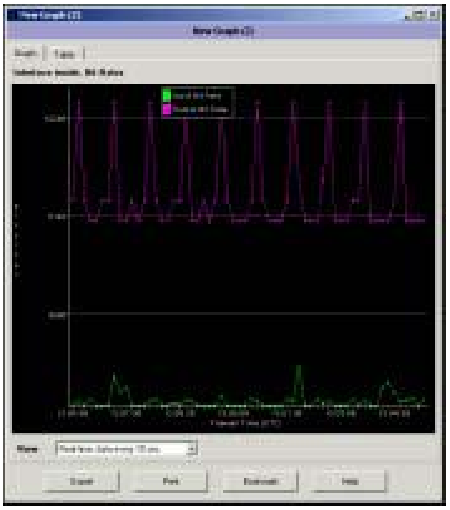


CSPFA 3.2—18-24

Many different items can be monitored using PDM, including but not limited to the following:

- PDM log
- Secure Shell (SSH) sessions
- Telnet console settings
- PDM users
- VPN statistics
- System performance graphs
- Connection graphs
- Interface graphs

Interface Graphs Panel

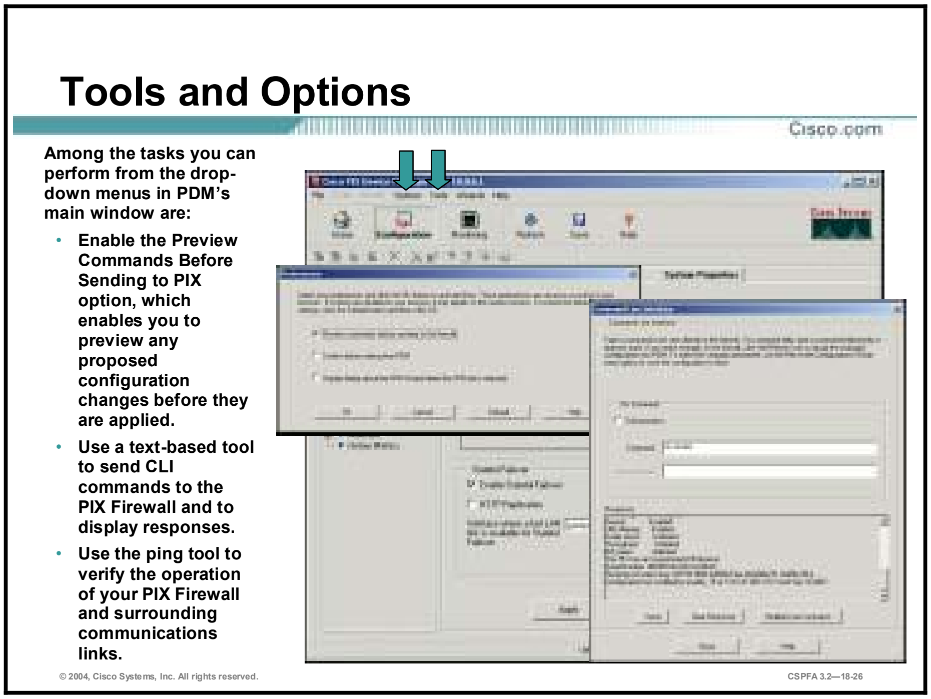


The Interface Graphs panel enables you to monitor per-interface statistics, such as bit rates, for each enabled interface on the PIX Firewall.

The Interface Graphs panel enables you to monitor per-interface statistics, such as packet counts and bit rates, for each enabled interface on the PIX Firewall.

The list of graphs available is the same for every interface. Each graph can be viewed as a line graph and in table form. Each graph can also be viewed with different time horizons.

Note If an interface is not enabled using the Interfaces panel under the System Properties panel, no graphs will be available for that interface.



Tools and Options

Among the tasks you can perform from the drop-down menus in PDM's main window are:

- Enable the Preview Commands Before Sending to PIX option, which enables you to preview any proposed configuration changes before they are applied.
- Use a text-based tool to send CLI commands to the PIX Firewall and to display responses.
- Use the ping tool to verify the operation of your PIX Firewall and surrounding communications links.

© 2004, Cisco Systems, Inc. All rights reserved.

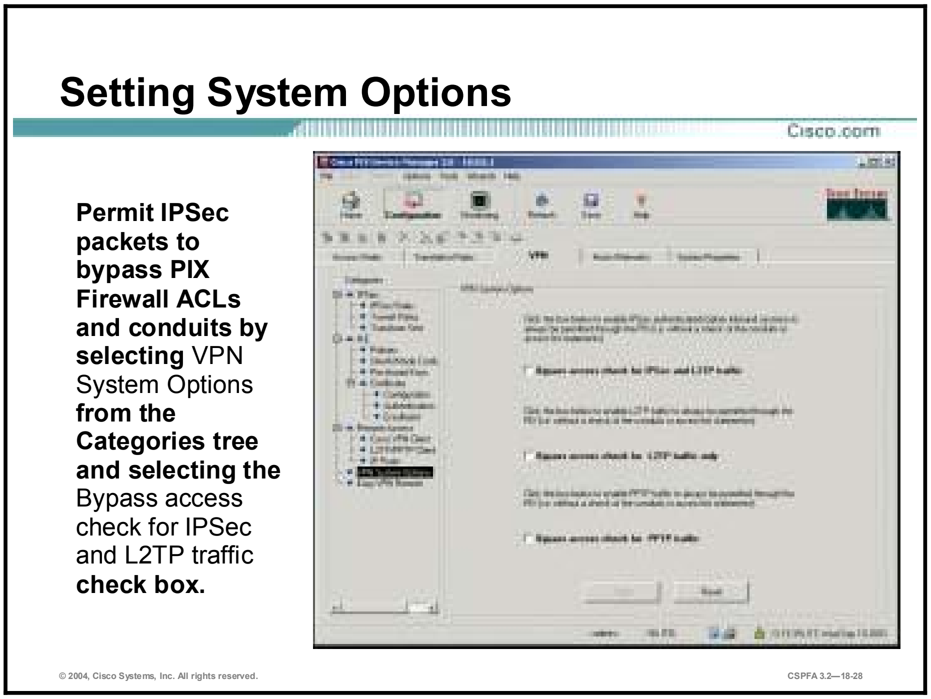
CSPFA 3.2—18-26

Choosing **Options>Preferences>Preview Commands Before Sending to PIX** enables you to preview any commands generated by any panel before they are sent to the PIX Firewall.

Select **Tools>Command Line Interface** to enter CLI commands to be sent to the PIX Firewall. You can also access the ping tool from the tools menu by selecting **Tool>Ping**.

Using PDM to Create Site-to-Site VPNs

This topic explains how to use PDM to create site-to-site VPNs.



Setting System Options

Cisco.com

Permit IPSec packets to bypass PIX Firewall ACLs and conduits by selecting VPN System Options **from the Categories tree and selecting the** Bypass access check for IPSec and L2TP traffic **check box.**

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—18-28

To begin creating your VPN, implicitly permit IPSec packets to bypass PIX Firewall ACLs by selecting **VPN System Options** from the Categories tree and selecting the **Bypass access check for IPSec traffic and L2TP traffic** check box.

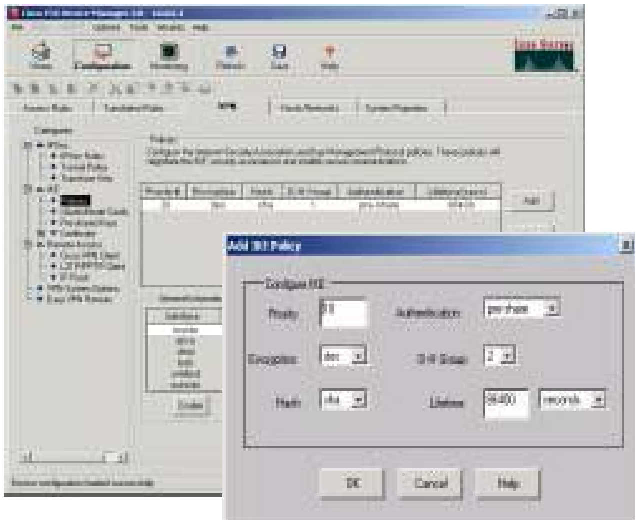
System options are used to tune the PIX Firewall security features. All of the system options are not enabled by default and must be explicitly enabled.

As shown in the figure, the system options pertaining to IPSec are as follows:

- Bypass access check for IPSec and L2TP traffic—This command enables IPSec authenticated/cipher inbound sessions to always be permitted. This permits IPSec traffic to pass through the firewall without a check of the **access-list** command statements. The **sysopt connection permit-ipsec** command is written to the firewall. Because Layer 2 Tunneling Protocol (L2TP) traffic can only come from IPSec, the **sysopt connection permit-ipsec** command allows L2TP traffic to pass as well.
  - Bypass access check for L2TP traffic only—Specifying this command in the firewall configuration permits L2TP traffic to pass through the firewall without a check of the **access-list** command statements. The **sysopt connection permit-l2tp** command is written to the firewall.
  - Bypass access check for L2TP traffic only—Specifying this command in the firewall configuration permits L2TP traffic to pass through the firewall without a check of the **access-list** command statements. The **sysopt connection permit-l2tp** command is written to the firewall.

Configuring IKE

Select Policies **from the IKE branch of the Categories tree, and click** Add **to create an IKE policy.**



Add IKE Policy

Configure IKE

Priority:

Authentication: pre-share

Encryption: des

D-H Group: 2

Hash: sha

Lifetime: 86400 seconds

CSPFA 3.2—18-29

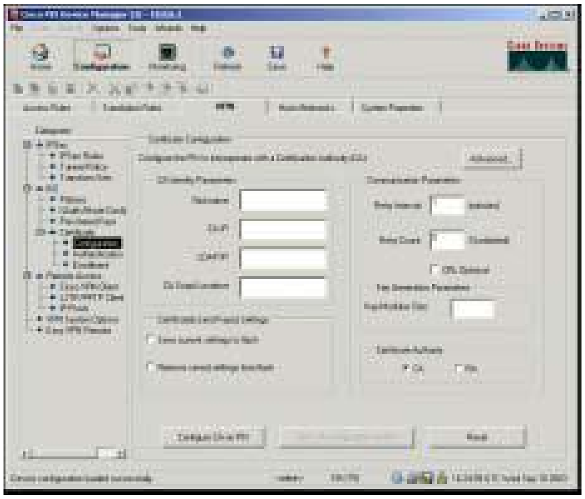
Internet Key Exchange (IKE)-related screens are grouped under the IKE branch of the Categories tree. You can view your Internet Security Association Key Management Protocol (ISAKMP) policies by selecting **Policies** from the ISAKMP branch. Add, edit, or delete policies by clicking the buttons on the right side of the screen.

Clicking **Add** opens a window where you can configure an IKE policy, including encryption algorithm, hash algorithm, Diffie-Hellman (DH) group, Security Association (SA) lifetime, and authentication method. There is a separate section for pre-shared keys. If you select pre-share as the authentication type, you need to specify the key and identify the peer that will be sharing this key with your PIX Firewall.

Certificates

Cisco.com

Select Certificate> Configuration **from the IKE branch of the Categories tree to configure CAs.**



CSPFA 3.2—18-30

PDM supports the configuration of Certificate Authority (CA) interoperability with IPSec. This enables the PIX Firewall and the CA to communicate so that the PIX Firewall can obtain and use digital certificates from the CA.

The PIX Firewall currently supports CA servers from VeriSign, Entrust, Baltimore Technologies, and Microsoft. You must ensure that the PIX Firewall clock is set to GMT, month, day, and year before configuring the CA. If the clock is set incorrectly, the CA may reject certificates based on the incorrect timestamps. Also, the lifetime of the certificate and the Certificate Revocation List (CRL) is checked in GMT time.

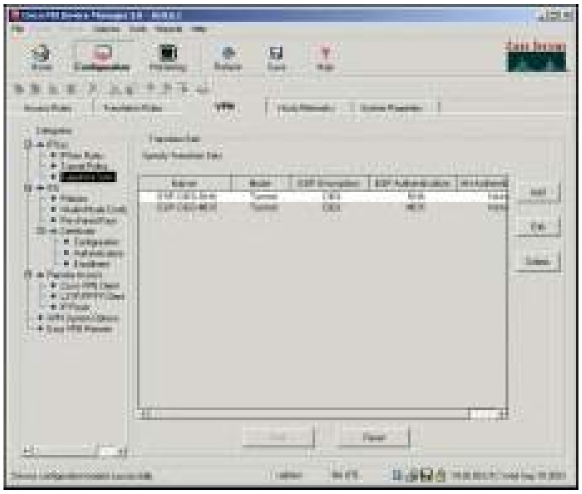
The PDM window for configuring a CA is accessible from the IKE branch of the Categories tree. The following parameters can be configured in this window:

- Nickname—Name given to the CA.
- CA IP—The IP address of the CA server.
- LDAP IP—The IP address of the Lightweight Directory Access Protocol (LDAP) server.
- CA Script Location—The location of the CA script. The default location and script on the CA server is /cgi-bin/pkiclient.exe. If the CA administrator has not put the CGI script in this location, provide the location and the name of the script in the **ca identity** command. A PIX Firewall uses a subset of the HTTP protocol to contact the CA, so it must identify a particular cgi-bin script to handle CA requests.
- Retry Interval—The number of minutes the PIX Firewall waits before resending a certificate request to the CA when it does not receive a response from the previous request. Specify 1 to 60 minutes. By default, the PIX Firewall retries every minute.
- Retry Count—Number of times the PIX Firewall will resend a certificate request when it does not receive a certificate from the CA from the previous request. Specify 1 to 100. The default is 0, which indicates it will keep retrying.
- Key Modulus—The size of the key modulus, which is between 512 and 2048 bits.

- Certificate Authority—Indicates whether to contact the CA or Registration Authority (RA). Some CA systems provide an RA, which the PIX Firewall contacts instead of the CA.

Configuring Transform Sets

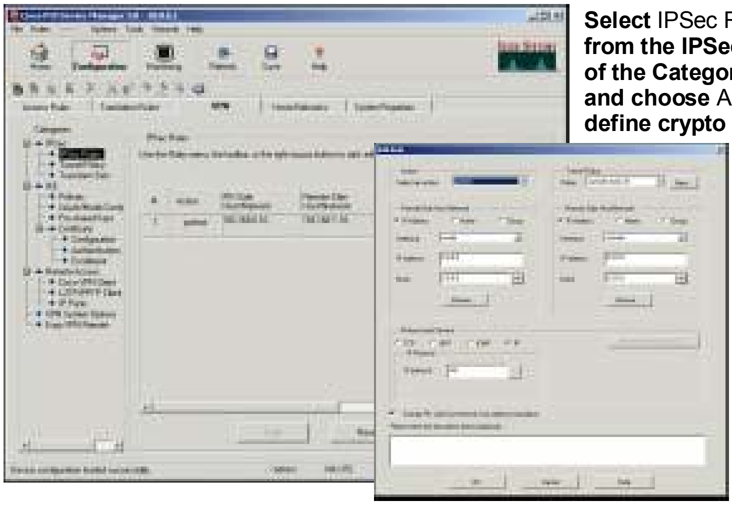
Select Transform Sets **from the IPSec branch of the Categories tree to configure transform sets.**



To view Transform Sets, click the **VPN** tab and then click **IPSec>Transform Sets** in the Categories area. The Transform Sets panel lets you define transform sets that can be applied to tunnel policies. A transform set specifies the IPSec protocol, encryption algorithm, and hash algorithm to use on traffic matching the IPSec policy. Your configuration edits are captured by PDM but are not sent to the firewall unit until you click Apply. The screen that appears enables you to view, add, edit, and delete transform sets and covers following CLI commands:

- **crypto ipsec** ***transform-set-name transform1*** **[*****transform2*** **[*****transform3*****]]**
- **crypto ipsec** ***transform-set-name*** **mode transport**

Creating an IPSec Rule



Select IPSec Rules **from the IPSec branch of the Categories tree and choose** Add **to define crypto ACLs.**

CSPFA 3.2—18-32

IPSec-related screens are grouped under the IPSec branch of the Categories tree. To select traffic to be protected by IPSec, select **IPSec Rules** from the IPSec branch. Right-click within the IPSec Rules table and choose **Add** from the drop-down menu. The Add Rules panel opens. This panel consists of a table similar to the Access Rule panel. This is where you select the traffic to be protected. When applied to the PIX Firewall, the rule you create is implemented by attaching an ACL to a crypto map.

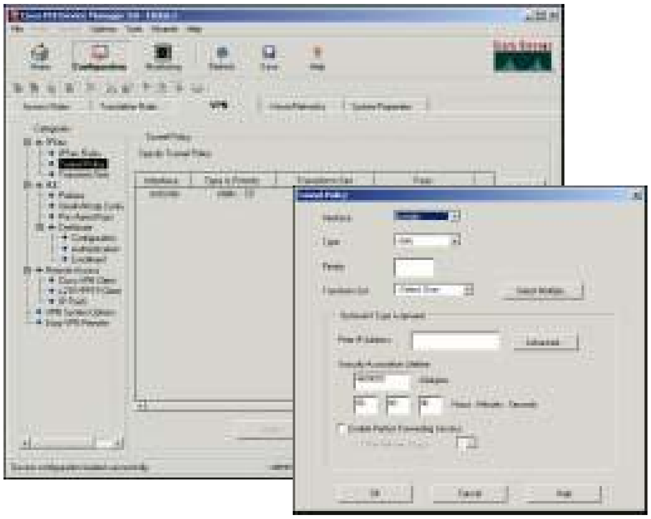
In the Add Rule panel, the action choices are protect and do not protect, rather than permit or deny. Use the Firewall Side Host/Network and Remote Side Host/Network group boxes to select the traffic to be protected. Like the Access Rule table, PDM displays the real IP addresses of the hosts and networks. The ACL, however, may show translated addresses depending on your current NAT configuration and where the crypto map is applied.

At the bottom of the panel, is the Exempt from address translation check box. If this check box is selected, PDM generates **nat 0 match acl** commands to allow IPSec traffic to bypass the NAT engine. This check box is selected by default.

Use the Tunnel Policy group box in the top right corner to attach your traffic selection rule to a Tunnel Policy. This is the equivalent of attaching an ACL to a crypto map. The New button provides a shortcut for defining a new tunnel policy without leaving the Add Rule panel.

You can add, edit, and delete traffic selector rules by using the main Rules menu or the buttons in the toolbar or by right-clicking within the table. You can also cut, copy, and paste rules between the Access Rule tables, NAT Exemption Rule table, and IPSec rule table.

Creating a Crypto Map



Select Tunnel Policy **from the IPSec branch of the Categories tree to create a crypto map.**

CSPFA 3.2—18-33

Use the Tunnel Policy window to create crypto maps. Tunnel policies are the equivalent of crypto maps in the PIX Firewall CLI. When you select **Tunnel Policy** from the IPSec branch of the Categories tree, the Tunnel Policies table appears displaying your currently configured tunnels. You can add, edit, or delete policies by clicking the buttons on the right side of the screen.

The figure shows the window that appears when you click **Add** in the Tunnel Policy window. This is the window in which you define or modify tunnel policies. As you create a tunnel policy, you must bind it to an interface. The PIX Firewall CLI enables you to configure a crypto map and then apply it to an interface. PDM hides the concept of the crypto map. It does not support crypto maps that are not applied to any interface. If such a map exists in the configuration, PDM parses and ignores it.

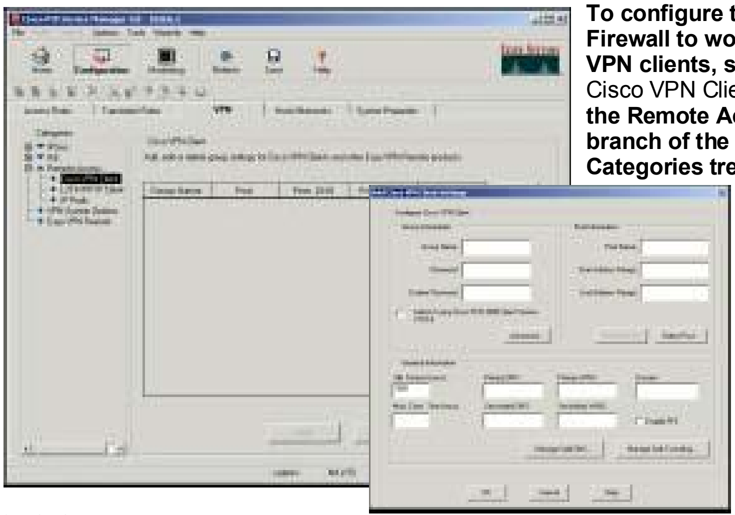
A tunnel policy can have more than one transform set and more than one peer. This is the purpose of the Select the **Multiple** button beside the Transform Set field and the **Advanced** button beside the Peer IP Address field. Each opens a window where you can configure multiple values per field.

Using PDM to Create Remote Access VPNs

This topic gives an overview of how PDM configures the PIX Firewall-supported VPN clients.

Configuring the PIX Firewall for VPN Clients

Cisco.com



To configure the PIX Firewall to work with VPN clients, select Cisco VPN Client **from the Remote Access branch of the Categories tree.**

CSPFA 3.2—18-35

The PIX Firewall supports the following VPN clients. PDM supports all of them except the Cisco VPN Client version 1.x.

- Cisco VPN 3000 Client version 2.5 or 2.6
- Cisco VPN Client version 3.0 and higher
- Cisco VPN 3002 Hardware Client
- Windows 2000 Client

To configure support for VPN clients, select **Cisco VPN Client** from the Remote Access branch of the Categories tree. In the CLI, the **vpngroup** command set configures Cisco VPN 3000 Client policy attributes to be associated with a VPN group name and downloaded to the Cisco VPN 3000 clients that are part of the given group. The same VPN group name is configured in PDM's Edit Client Settings window shown in the figure.

The following parameters, which appear in the General Info group box within the Add Client Settings window, consist of information pertaining to the pool of local addresses assigned to this VPN group:

Note	To add a new pool of addresses, expand the **Remote Access** branch of the Categories tree and select **IP Pools**. To delete the association of an address pool from a group, click **Deselect Pool** in the Edit Client Settings window.

- Group Name—A descriptive group name to be used for this Cisco VPN Client 3.x client group.

- — Pool—The IKE pool created in VPN>Remote Access>IP Pools. This pool determines the range of local addresses that will be assigned to the VPN clients.

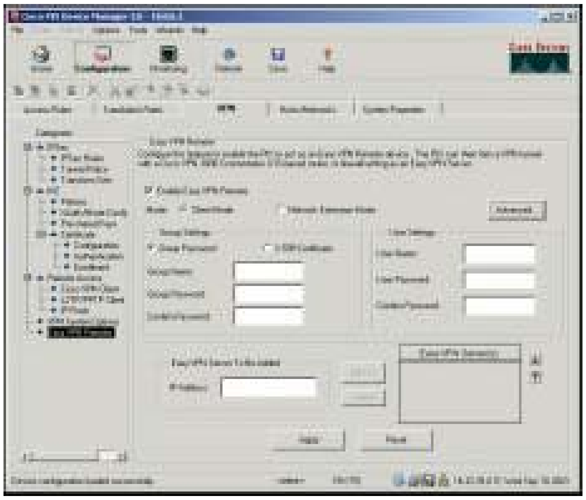
- Idle Timeout (secs)—Sets the inactivity timeout for a client. When the inactivity timeout for all IPSec SAs have expired for a given VPN client, the tunnel is terminated. The default inactivity timeout is 1800 seconds (30 minutes).
- Max. Conn. Time (secs)—Sets the maximum connection time for a client. When the maximum connection time is reached for a given VPN client, the tunnel is terminated. This means the connection between the client and the PIX Firewall will have to be re-established. The default maximum connection time is set to an unlimited amount of time.
- Primary DNS—Enables the PIX Firewall to download the IP address of a primary Domain Name System (DNS) server to the client as part of an IKE negotiation.
- Secondary DNS—Enables the PIX Firewall to download the IP address of a secondary DNS server to the client as part of an IKE negotiation.
- Primary WINS—Enables the PIX Firewall to download the IP address of a primary Windows Internet Name Service (WINS) server to the client as part of an IKE negotiation.
- Secondary WINS—Enables the PIX Firewall to download the IP address of a secondary WINS server to the client as part of an IKE negotiation.
- Domain—Specifies a domain name to enable the PIX Firewall to download a default domain name to a client of IKE negotiation.

Select the **Enable PFS** check box to enable Perfect Forward Secrecy (PFS). Clicking the **Manage Split DNS** or **Manage Split Tunneling** button opens a window to configure split DNS or split tunneling.

Easy VPN Remote Device Setting

Cisco.com

To configure the PIX Firewall as a Easy VPN Remote client, select Easy VPN Remote **from the Categories tree.**



© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—18-36

The Easy VPN Remote panel lets you connect your firewall to a VPN concentrator or headend as a VPN client. To configure your PIX Firewall as a Easy VPN remote client, select Easy VPN Remote Client from the Categories tree.

You can use PDM to configure PIX Firewall as a VPN client when connecting to a VPN headend, such as a Cisco VPN 3000 Concentrator, a Cisco IOS-based router, or PIX Firewall acting as an Easy VPN Remote Server. This functionality, sometimes called a hardware client, let the firewall act as a VPN client and to establish a VPN tunnel to the VPN headend. Hosts running on the LAN behind the firewall can connect through the VPN headend without individually running any VPN client software. You must select locally one of the following modes of operation when you enable your firewall VPN client:

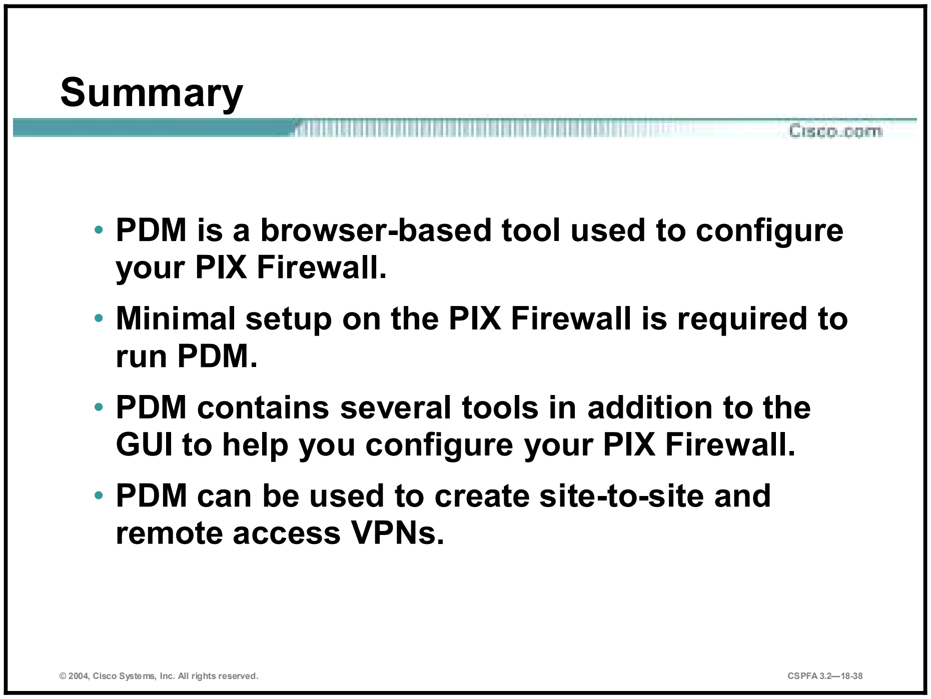
- Client mode—This option applies NAT to all IP addresses of clients connected to the inside (higher security) interface of the firewall. To use this mode, you must also enable the DHCP server on the inside interface.
- Network Extension Mode—This option does not apply NAT to any IP addresses of clients on the inside (higher security) interface of the firewall. In network extension mode, the IP addresses of clients on the inside interface are received without change at the VPN headend. If these addresses are registered with the Network Information Center (NIC), they may be forwarded to the public Internet without further processing. Otherwise, they may be translated by the VPN headend or forwarded to a private network without translation.
- Group Settings—for authentication, you must either specify a group name and password or use X.509 Certificate. The following boxes apply to Group Password only:
  - Group Name—specify the alphanumeric group name of the VPN client as determined by the VPN concentrator or headend.
  - Group Password—specify the group password. This is sometimes referred to as the pre-shared key. It is recommended that it be at least eight alphanumeric characters with no special (not alphanumeric) characters, such as @, ! or *.
  - Confirm Password—Retype the group password.

- User Name—Specify the alphanumeric username of the VPN client as determined by the VPN concentrator or headend. This is optional, and is used when the VPN concentrator or headend is using Extended Authentication (XAUTH).
- User Password—Specify the user password. Review the password policy of your organization.
- Confirm Password—Retype the user password.
- Easy VPN Server To Be Added—Lets you specify the IP address of the remote VPN concentrator or headend.
- IP Address—The IP address of the remote VPN concentrator or headend.
- Add—Click to add the IP address to the IP Addresses box.
- Delete—Deletes the selected server IP address from the IP Addresses box.
- Easy VPN Server Addresses—Displays the configured IP addresses of the remote VPN concentrator or headend. The up and down arrows next to the IP Addresses Box let you reorder the servers.
- Advanced—Lets you configure MAC Exemption and Tunneled Management features in the Advanced Options dialog box.

Note PDM's main menu includes a VPN Wizard that leads you through creation of a site-to-site or remote access VPN. The VPN Wizard can only be used to add a VPN rule. For editing, you must use the items under the VPN tab.

Summary

This topic summarizes the information you learned in this lesson.



Summary

- PDM is a browser-based tool used to configure your PIX Firewall.
- Minimal setup on the PIX Firewall is required to run PDM.
- PDM contains several tools in addition to the GUI to help you configure your PIX Firewall.
- PDM can be used to create site-to-site and remote access VPNs.

Lab Exercise—Configuring the PIX Firewall with PDM

Complete the following lab exercise to practice what you learned in this lesson.

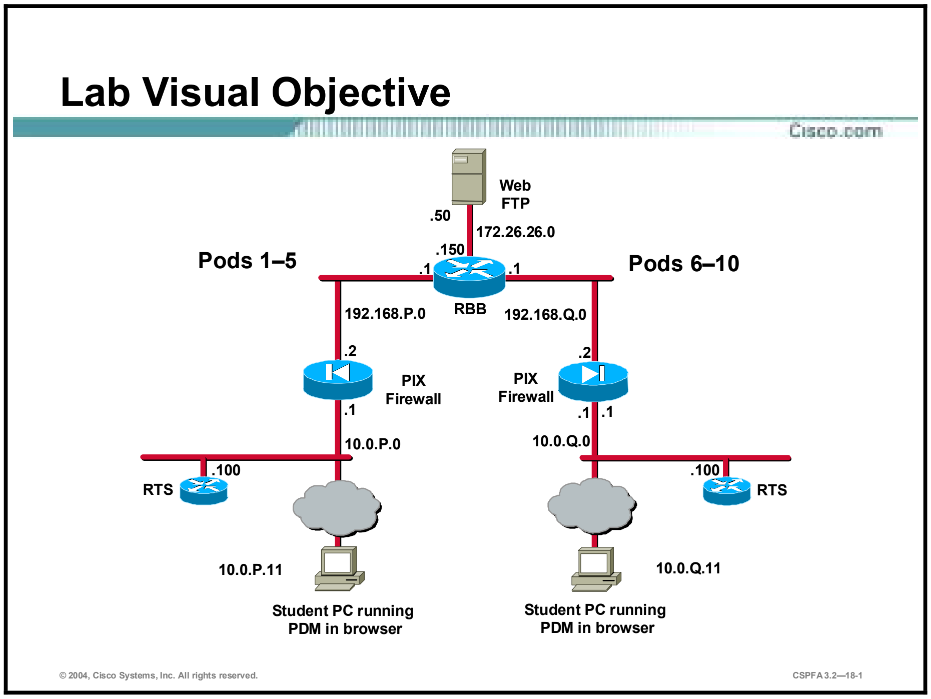
Objectives

In this lab exercise, you will complete the following tasks:

- Install PDM and access it from the browser.
- Test PDM’s warning for unsupported commands.
- Clear the PIX Firewall’s configuration and access the PDM Startup Wizard.
- Use the PDM Startup Wizard to configure a privileged mode password.
- Configure outbound access with NAT.
- Test connectivity through the PIX Firewall.
- Configure and test inbound access.
- Configure and test logging to a Syslog server.
- Configure intrusion detection.
- Configure the PIX Firewall to monitor intrusion detection.
- Create a site-to-site VPN.
- Test and verify the VPN.

Visual Objectives

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—18-1

Note: In the following lab exercise, you will bypass the initial security alert regarding the site security certificate. However, remember that when you remotely configure the PIX Firewall with PDM, you can use the security certificate for secure encrypted communication between PDM and the PIX Firewall. To do this, install the certificate by clicking **View Certificate** in the initial Security Alert window and following the prompts. Because the certificate is assigned to the PIX Firewall by name rather than by IP address, you will need to establish the connection with the PIX Firewall by entering its fully qualified domain name, rather than the IP address, in the browser. Using the name rather than an IP address requires that name resolution is enabled through DNS or a hosts file.

Task 1—Install PDM and Access It from the Browser

Complete the following steps to install PDM and access it from the browser:

Step 1 Load the PDM file into the PIX Firewall:

```
pixP(config)# copy tftp://10.0.P.11/pdm-301.bin flash:pdm
```

(where P = pod number)

Step 2 Enable the HTTP server in the PIX Firewall:

```
pixP(config)# http server enable
```

(where P = pod number)

Step 3 Grant permission for the inside host to initiate an HTTP connection to the PIX Firewall:

```
pixP(config)# http 10.0.P.11 255.255.255.0 inside
```

(where P = pod number)

Step 4 Access the PDM console by completing the following substeps:

1. Open the browser and enter **https://10.0.P.1**.
(where P = pod number)

2. In the Security Alert window, click **Yes**.

3. When prompted for the username and password, do not enter a username or password. Click **OK** to continue.

4. Click **Yes** in the Security Warning window. If the Update Config window opens, click **Proceed**.

Step 5 Notice that the current PIX Firewall configuration has been imported. Examine the configuration by clicking the **Configuration** button and then completing the following substeps:

1. Click the **Access Rules** tab. Notice that an access policy has been created to correspond to the ACLs you configured earlier in the course.

2. Click the **Translation Rules** tab. Notice that the static mappings, NAT, and global pools appear here.

3. Click the **Hosts/Networks** tab and observe the network topology.

4. Click the **System Properties** tab. Notice that the configuration of the PIX Firewall interfaces is displayed.

Step 6 Close the browser.

Step 7 The Are you sure window opens.

Step 8 Click **Yes**. PDM application closes.

Task 2—Test PDM's Warning for Unsupported Commands

Complete the following steps to verify that PDM warns you of unsupported commands:

Step 1 Configure the PIX Firewall with a **conduit** command that permits IP traffic from a peer pod's internal network to the internal network:

```
pixP(config)# conduit permit ip 192.168.P.0 255.255.255.0 192.168.Q.0
255.255.255.0
```

(where P = pod number, and Q = peer pod number)

Step 2 Display the conduit:

```
pixP(config)# show cond
conduit permit ip host 192.168.P.10 192.168.Q.0 255.255.255.0
```

(where P = pod number, and Q = peer pod number)

Step 3 Verify that conduits exist in the configuration along with ACLs by displaying the ACLs:

```
pixP(config)# show access-list

access-list ACLIN; 10 elements

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 object-
group FTPSERVERS object-group MYSERVICES

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq www (hitcnt=4)

access-list ACLIN line 1 permit tcp 192.168.Q.0 255.255.255.0 host
192.168.P.11 eq ftp (hitcnt=2)

access-list ACLIN line 2 permit tcp any object-group ALLSERVERS eq www

access-list ACLIN line 2 permit tcp any host 192.168.P.11 eq www
(hitcnt=0)

access-list ACLIN line 2 permit tcp any host 192.168.P.10 eq www
(hitcnt=2)

access-list ACLIN line 2 permit tcp any host 192.168.P.6 eq www
(hitcnt=0)

access-list ACLIN line 2 permit tcp any host 192.168.P.7 eq www
(hitcnt=0)

access-list ACLIN line 3 permit icmp any any object-group PING

access-list ACLIN line 3 permit icmp any any echo (hitcnt=12)

access-list ACLIN line 3 permit icmp any any echo-reply (hitcnt=4)

access-list ACLIN line 3 permit icmp any any unreachable (hitcnt=0)

access-list ACLIN line 4 deny ip any any (hitcnt=3)

access-list ACLDMZ; 3 elements

access-list ACLDMZ line 1 permit icmp any any object-group PING

access-list ACLDMZ line 1 permit icmp any any echo (hitcnt=0)

access-list ACLDMZ line 1 permit icmp any any echo-reply (hitcnt=8)

access-list ACLDMZ line 1 permit icmp any any unreachable (hitcnt=0)
```

(where P = pod number, and Q = peer pod number)

Step 4 Access the PDM console by completing the following substeps:

1. In the browser, enter **https://10.0.P.1**.
   (where P = pod number)

2. In the Security Alert window, click **Yes**.

3. When prompted for the username and password, do not enter a username or password. Click **OK** to continue. The Security Warning window opens.

4. Click **Yes**. The Unsupported Command Found window opens. This is because PDM does not support configurations that use both conduits and ACLs.

5. Read the statement in the window.

6. Click **OK** in the Unsupported Command Found window. The PDM main window opens.

7. Click the **Configuration** button to verify that the Configuration button is disabled. After you click the Configuration button, the Unsupported Command Found window opens. Click **OK**.

8. Minimize, but do *not* close, the PDM console.

Step 5 In the Telnet window, remove the conduit to restore full PDM capability:

```
pixP(config)# clear conduit
```

Step 6 Verify that the conduit has been removed:

```
pixP(config)# show conduit
```

Step 7 Maximize the PDM console.

Step 8 From the PDM main toolbar, click the **Refresh** button to refresh PDM with the current running configuration.

Step 9 Verify that the conduit has been removed by completing the following substeps:

1. Choose **Tools>Command Line Interface** from the main menu. The Command Line Interface window opens.
2. In the Command field, enter **show conduit**.
3. Click **Send**.
4. Close the Command Line Interface window.

Step 10 Close the browser PDM windows.

Task 3—Clear the PIX Firewall's Configuration and Access the PDM Startup Wizard

Complete the following steps to erase the current PIX Firewall configuration and access the PDM wizard:

Step 1 In the Telnet window, erase the current PIX Firewall configuration. When prompted to confirm, press **Enter**.

```
pixP(config)# write erase
Erase PIX configuration in flash memory? [confirm] <Enter>
```

Step 2 In the Telnet window, reload the PIX Firewall. When prompted to confirm, press **Enter**.

```
pixP(config)# reload
Proceed with reload? [confirm] <Enter>
```

Step 3 When prompted to pre-configure the PIX Firewall through interactive prompts, press **Enter**.

Step 4 Agree to use the current password by pressing **Enter**:

```
Enable password [<use current password>]: <Enter>
```

Step 5 Accept the default year by pressing **Enter**:

```
Clock (UTC):
  Year [2003]: <Enter>
```

Step 6 Accept the default month by pressing **Enter**:

```
Month [Nov]: <Enter>
```

Step 7 Accept the default day by pressing **Enter**:

```
Day [14]: <Enter>
```

Step 8 Accept the default time stored in the host computer by pressing **Enter**:

```
Time [11:21:25]: <Enter>
```

Step 9 Enter the IP address of the PIX Firewall's inside interface:

```
Inside IP address: 10.0.P.1
```

(where P = pod number)

Step 10 Enter the network mask that applies to inside IP address:

```
Inside network mask: 255.255.255.0
```

Step 11 Enter the hostname you want to display in the PIX Firewall command line prompt:

```
Host name: pixP
```

(where P = pod number)

Step 12 Enter the DNS domain name of the network on which the PIX Firewall runs:

```
Domain name: cisco.com
```

Step 13 Enter the IP address of the host running PDM:

```
10.0.P.11
```

(where P = pod number)

Step 14 Enter **y** at the prompt to save the information to the PIX Firewall's Flash memory.

Step 15 Access the PDM console by completing the following substeps:

1. In the browser, enter **https://10.0.P.1**.
   (where P = pod number)

2. In the Security Alert window, click **Yes**.

3. When prompted for the username and password, do not enter a username or password. Click **OK** to continue. The Security Warning window opens.

4. Click **Yes**. The Update Config window opens.

5. Click **Proceed**. If the Preview CLI Commands window opens, click **Send**. The PIX Device Manager main window opens.

Task 4—Use the PDM Startup Wizard to Configure a Privileged Mode Password

Complete the following steps to configure a privileged mode password:

Step 1 In the PIX Device Manager Startup Wizard window, click **Next**. The Startup Wizard's Basic Configuration group box appears.

Step 2 Verify that pixP appears in the PIX Host Name field.
(where P = pod number)

Step 3 Verify that cisco.com appears in the Domain Name field.

Step 4 Select **Change Enable Password** within the Enable Password group box.

Step 5 Enter **cisco** in the New Enable Password text box.

Step 6 Enter **cisco** in the Confirm New Enable Password text box.

Step 7 Click **Finish**. The Enter Network Password window opens.

Step 8 Leave the Username field blank, enter **cisco** in the password field, and click **OK**. The main Cisco PIX Device Manager window opens.

Task 5—Configure Outbound Access with NAT

Complete the following steps to configure the PIX Firewall's inside and outside interfaces, establish a default route, enable NAT for the internal network, and create a global pool of addresses for address translation:

Step 1 Click the **Configuration** button, then select the **System Properties** tab.

Step 2 Configure the inside interface by completing the following substeps:

1. Select **ethernet1** in the Interfaces table and click the **Edit** button. The Edit Interface window opens.
2. Verify that the Enable Interface check box is selected.
3. Verify that inside appears in the Interface Name field.
4. Verify that 10.0.P.1 appears in the IP Address field.
   (where P = pod number)
5. Verify that 255.255.255.0 appears in the Subnet Mask drop-down menu.
6. Verify that 100 appears in the Security Level field.
7. Click the **Properties** button. The Hardware Port window opens.
8. Choose **100full** from the Speed and duplex mode drop-down menu.
9. Click **OK**. You are returned to the Interface window.
10. Click **OK**. You are returned to the Systems Properties tab.

Step 3 Configure the outside interface by completing the following substeps:

1. Select **ethernet0** in the Interfaces table, and then click the **Edit** button. The Edit Interface window opens.
2. Select the **Enable Interface** check box.
3. Verify that outside appears in the Interface Name field.
4. Verify that the **Static IP Address** radio button is selected within the IP Address group box.
5. Enter **192.168.P.2** in the IP Address field.
   (where P = pod number)
6. Choose **255.255.255.0** from the Subnet Mask drop-down menu.
7. Verify that 0 appears in the Security Level field.
8. Click the **Properties** button. The Hardware Port window opens.

9. Choose **100full** from the Speed and duplex mode drop-down menu.
10. Click **OK**. You are returned to the Interface window.
11. Click **OK**. You are returned to the System Properties tab.
12. Click **Apply**.

Step 4 To establish a default route, complete the following substeps:

1. Verify that the System Properties tab is still active.
2. Expand the **Routing** branch in the Categories tree.
3. Choose **Static Route** from the Routing list.
4. Select **Add** from the Static Route group box. The Add Static Route window opens.
5. Choose **outside** from the Interface Name drop-down menu.
6. Enter **0.0.0.0** in the IP Address field.
7. Enter **192.168.P.1** in the Gateway IP field.
   (where P = pod number)
8. Enter **0.0.0.0** in the Mask drop-down menu.
9. Verify that 1 appears in the Metric field.
10. Click **OK**. The static route appears in the Static Route table.
11. Click **Apply**.

Step 5 Configure a global pool of addresses to be used for address translation by completing the following substeps:

1. Select the Translation Rules tab.
2. Click the **Manage Pools** button. The Manage Global Address Pools window opens.
3. Click **Add**. The Add Global Pool Item window opens.
4. Choose **outside** from the Interface drop-down menu.
5. Enter **1** in the Pool ID field.
6. Verify that the Range radio button is selected.
7. Enter **192.168.P.20** in the first IP address field.
   (where P = pod number)
8. Enter **192.168.P.254** in the second IP address field.
   (where P = pod number)
9. Enter **255.255.255.0** in the Network Mask field.
10. Click **OK**. You are returned to the Manage Global Address Pools window.
11. Click **OK**. You are returned to the Translation Rules tab.
12. Click **Apply**.

Step 6 Configure NAT by completing the following substeps:

1. Verify that the Translation Rules tab is still active.
2. Verify that the Translation Rules radio button is selected.
3. Choose **Rules>Add** from the main menu. The Add Address Translation Rule window opens.
4. Verify that the inside interface is chosen in the Interface drop-down menu.
5. Click **Browse**. The Select host/network window opens.
6. Verify that the inside interface is chosen in the drop-down menu.
7. Select the inside network by clicking **10.0.P.0**.
   (where P = pod number)
8. Click **OK**. You are returned to the Add Address Translation Rule window.
9. Verify that outside is chosen in the Translate address on interface drop-down menu.
10. Verify that Dynamic is selected in the Translate Address to group box.
11. Choose **1** from the Address Pool drop-down menu.
12. Verify that the global pool you configured earlier (192.168.P.20–192.168.P.254) appears under Address.
    (where P = pod number)
13. Click **OK** in the Add Address Translation Rule window. The new rule appears on the Translation Rules tab.
14. Click **Apply.**

Task 6—Test Connectivity Through the PIX Firewall

Complete the following steps to test interface connectivity and NAT:

Step 1 Test interface connectivity by completing the following substeps:

1. Choose **Tools>Ping**.
2. In the IP Address field, enter **10.0.P.1**.
   (where P = pod number)
3. Click **Ping**.
4. Observe the following output in the Ping Output window. The output should appear similar to the following:

```
10.0.P.1 response received -- 0ms
10.0.P.1 response received -- 0ms
10.0.P.1 response received -- 0ms
```

(where P = pod number)

5. Click **Clear Screen**.

Step 2 Repeat Step 1 for the following IP addresses. You should receive responses for all pings:

- The inside host: 10.0.P.11
  (where P = pod number)
- The outside interface: 192.168.P.2
  (where P = pod number)
- The backbone router: 192.168.P.1
  (where P = pod number)

Step 3 Exit the Ping window by clicking **Close**.

Step 4 Test the operation of the global and NAT you configured by originating connections through the PIX Firewall. To do this, complete the following substeps:

1. Open a web browser on the student PC.
2. Use the web browser to access the Super Server at IP address 172.26.26.50 by entering **http://172.26.26.50**.

Step 5 Observe the translation table by completing the following substeps:

1. Choose **Tools>Command Line Interface**. The Command Line Interface window opens.
2. In the Command field, enter **show xlate**.
3. Click **Send**.
4. Observe the output in the Response field. It should appear similar to the following:

```
Result of the PIX command: "show xlate"


1 in use, 1 most used
Global 192.168.P.20 Local 10.0.P.11
```

(where P = pod number)

Note that a global address chosen from the low end of the global range has been mapped to the student PC.

Step 6 Exit the Command Line Interface window by clicking **Close**.

Task 7—Configure and Test Inbound Access

Complete the following steps to configure the PIX Firewall to permit inbound access to hosts on the inside interface:

Step 1 Enable command preview by completing the following substeps:

1. Choose **Options>Preferences** from the main menu. The Preferences window opens.
2. Select **Preview commands before sending to firewall**.
3. Click **OK**.

Step 2 Create a static translation for the inside host by completing the following substeps:

1. Select the **Translation Rules** tab.

2. Select the **Add New Rule** icon in the toolbar. The Add Address Translation Rule window opens.
3. Verify that the inside interface is chosen in the Interface drop-down menu.
4. Click **Browse**. The Select host/network window opens.
5. Verify that the inside interface is chosen in the drop-down menu.
6. Select the inside host: **10.0.P.11**.
   (where P = pod number)
7. Click **OK**. You are returned to the Add Address Translation Rule window.
8. Verify that outside is chosen in the Translate Address on interface drop-down menu.
9. Select **Static** in the Translate address to group box.
10. Enter **192.168.P.10** in the IP Address field.
    (where P = pod number)
11. Click **OK**. The new rule appears on the Translation Rules tab.
12. Click **Apply**. The Preview CLI Commands window opens.
13. Click **Send**.

Step 3 Ping a peer pod's inside host from the internal host. The ping should fail because the peer pod's policy presently prevents pinging.

```
C:\> ping 192.168.Q.10
Pinging 192.168.Q.10 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
```

(where Q = peer pod number)

Close the Ping window.

Step 4 Configure an ACL to allow pinging through the PIX Firewall by completing the following substeps:

1. Select the **Access Rules** tab.
2. Choose **Rules** from the main menu.
3. Click **Add**. The Add Rule window opens.
4. Verify that **permit** is chosen in the Select an action drop-down menu.
5. Choose **outside** from the Interface drop-down menu in the Source Host/Network group box.
6. Choose **inside** from the Interface drop-down menu in the Destination Host/Network group box.
7. Select **ICMP** in the Protocol or Service group box.

8. Verify that **any** is selected in the ICMP type group box.

9. Click **OK**. The new rule appears on the Access Rules tab.

10. Click **Apply**. The Preview CLI Commands window opens.

11. Observe the ACLs to be sent to the PIX Firewall.

12. Click **Send**.

Step 5 Ping a peer pod's inside host from the internal host. Be sure to coordinate with the peer pod:

```
C:\> ping 192.168.Q.10
Pinging 192.168.Q.10 with 32 bytes of data:
Reply from 192.168.Q.10:  bytes=32 time<10ms TTL=125>
Reply from 192.168.Q.10:  bytes=32 time<10ms TTL=125>
Reply from 192.168.Q.10:  bytes=32 time<10ms TTL=125>
Reply from 192.168.Q.10:  bytes=32 time<10ms TTL=125>
```

(where Q = peer pod number)

Step 6 Close the Ping window.

Step 7 Configure an ACL to allow Web access to the inside host from the outside by completing the following substeps:

1. Select the **Access Rules** tab.

2. Choose **Rules>Add**. The Add Rule window opens.

3. Verify that **permit** is chosen in the Select an action drop-down menu.

4. Choose **outside** from the Interface drop-down menu within the Source Host/Network group box.

5. Choose **inside** from the Interface drop-down menu within the Destination Host/Network group box.

6. Click **Browse** in the Destination Host/Network group box. The Select host/network window opens.

7. Verify that **inside** is chosen in the interface drop-down menu.

8. Select the IP address of the inside host: **10.0.P.11**.
(where P = pod number)

9. Click **OK**. The Add Rule window becomes active.

10. Select **TCP** in the Protocol and Service group box.

11. Verify that = is chosen in the Service drop-down menu within the Source Port group box.

12. Verify that **any** appears in the Service field within the Source Port group box.

13. Verify that = is chosen in the Service drop-down menu within the Destination Port group box.

14. Click the **ellipsis** button within the Destination Port group box. The Service window opens.

15. Choose **http** from the Service list.

16. Click **OK**. You are returned to the Add Rule window.

17. Click **OK**.

18. Click **Apply**. The Preview CLI Commands window opens.

19. Observe the ACLs to be sent to the PIX Firewall.

20. Click **Send**.

Step 8 Clear current translations by completing the following substeps:

1. Choose **Tools>Command Line Interface**. The Command Line Interface window opens.

2. Enter **clear xlate** in the Command field.

3. Click **Send**.

4. Verify that the output in the Response field is similar to the following:

```
Result of firewall command: "clear xlate"
The command has been sent to the firewall.
```

Step 9 View current translations by completing the following substeps:

1. Click **Clear Response** in the Command Line Interface window.

2. Enter **show xlate** in the Command field.

3. Click **Send**.

4. Verify that the output in the Response field is similar to the following:

```
Result of firewall command: "show xlate"
0 in use, 3 most used
```

5. Click **Close** in the Command Line Interface window.

Step 10 Test Web access to the inside hosts of opposite pod groups by completing the following substeps:

1. Open a web browser on the student PC.

2. Use the web browser to access the inside host of the peer pod group **http://192.168.Q.10** (where Q = peer pod number). You should be able to establish a Web connection to the peer's inside host.

Step 11 Test FTP access to the inside hosts of other pod groups by completing the following substeps:

1. On the client PC, use FTP to get into the inside host of another pod group by choosing **Start>Run>ftp 192.168.Q.10** (where Q = peer pod number). You should be unable to access the peer's inside host via FTP.

2. Have an opposite pod group use FTP to attempt to get into the inside host.

Step 12 Observe the transactions by completing the following substeps:

1. Choose **Tools>Command Line Interface**. The Command Line Interface window opens.

2. Enter **show arp** in the Command field.

3. Click **Send**.

4. Verify that the output in the Response box is similar to the following:

```
result of firewall command: "show arp"


outside 192.168.P.1 0003.6ba4.ca60
inside 10.0.P.102 0050.da31.6130
```

(where P = pod number)

5. Click **Clear Response**.

6. Enter **show conn** in the Command field.

7. Click **Send**.

8. Verify that the output in the Response field is similar to the following:

```
result of firewall command: "show conn"
0 in use, 6 most used
TCP out 192.168.Q.10:80 in 10.1.P.11: 3893 idle 0:00:07 Bytes 463
flags UIO
TCP out 192.168.Q.10:80 in 10.1.P.11: 3893 idle 0:00:07 Bytes 463
flags UIO
```

9. Click **Clear Response**.

10. Enter **show xlate** in the Command field.

11. Click **Send**.

12. Verify that the output in the Response field is similar to the following:

```
result of firewall command: "show xlate"


2 in use, 3 most used
Global 192.168.P.10 Local 10.0.P.11
```

(where P = pod number)

13. Click **Close**.

Task 8—Configure and Test Logging to a Syslog Server

Complete the following steps to configure the PIX Firewall to send messages to a Syslog server:

Step 1 Select the **System Properties** tab.

Step 2 Expand Logging from the Categories tree on the left of the panel. Logging Setup appears under Logging.

Step 3 Select **Logging Setup**.

Step 4 Select **Enable logging** in the Logging Setup group menu.

Step 5 Click **Apply**. The Preview CLI Commands window opens.

Step 6 Click **Send**. You are returned to the System Properties tab.

Step 7 Select **Syslog** under Logging from the Categories tree on the left of the panel. The Syslog group box appears.

Step 8 Choose **Debugging** from the Level drop-down menu.

Step 9 Select **Include Timestamp**.

Step 10 Click **Add**. The Add Syslog Server window opens.

Step 11 Verify that inside is chosen in the Interface drop-down menu.

Step 12 Enter the IP address of the Syslog server in the IP Address field: **10.0.P.11**.
(where P = pod number)

Step 13 Click **OK**. You are returned to the System Properties tab.

Step 14 Click **Apply**. The Preview CLI Commands window opens.

Step 15 Click **Send**.

Step 16 Open the Kiwi Syslog Daemon on the desktop.

Step 17 From the Windows command line, telnet to the backbone router to generate traffic to be logged:

```
C:\> telnet 192.168.P.1
```

(where P = pod number)

Step 18 Close the Telnet window that opens.

Step 19 The Kiwi Syslog Daemon display should be similar to the following:

```
11-21-2002    10:58:59 Local4.Info 10.0.P.1 Nov 21 2002 10:49:25:
%PIX-6-302014: Teardown TCP connection 18 for outside:192.168.P.1/23
to inside:10.1.P.11/3978 duration 0:00:05 bytes 94 TCP FINs

11-21-2002    10:58:53 Local4.Info 10.0.P.1 Nov 21 2002 10:49:19:
%PIX-6-302013: Built outbound TCP connection 18 for
outside:192.168.P.1/23 (192.168.P.1/23) to inside:10.1.P.11/3978
(192.168.P.10/3978)
```

(where P = pod number)

Step 20 Close the Telnet window.

Step 21 Close Kiwi Syslog window.

Task 9—Configure Intrusion Detection

Complete the following steps to configure the PIX Firewall to detect ICMP packet attacks, drop the packets, and send an alarm to a Syslog server:

Step 1 Verify that the System Properties tab is still active.

Step 2 Expand **Intrusion Detection** from the Categories tree on the left of the panel. IDS Policy appears under Intrusion Detection.

Step 3 Select **IDS Policy**. The IDS Policy group box opens on the right.

Step 4 Click **Add**. The Add IDS Policy window opens.

Step 5 Enter **ATTACKPOLICY** in the Policy Name field.

Step 6 Verify that **Attack** is selected in the Policy Type group box.

Step 7 Select **Drop** and **Alarm** in the Action group box.

Step 8 Click **OK**. You are returned to the System Properties tab.

Step 9 In the Policy-to-Interface Mapping group box, choose **ATTACKPOLICY** from the drop-down menu for the inside interface under Attack Policy column.

Step 10 Click **Apply**. The Preview CLI Commands window opens.

Step 11 Click **Send**.

Task 10—Configure the PIX Firewall to Monitor Intrusion Detection

Complete the following steps to configure monitoring of intrusion detection:

Step 1 Select the **Monitoring** button.

Step 2 Expand **Interface Graphs** from the Categories tree on the left of the panel.

Step 3 Select **Outside**.

Step 4 Choose **Byte Counts** from the Available Graphs for: list.

Step 5 Click **Add**.

Step 6 Click **Graph It**. The New Graph window opens.

Step 7 Verify that Real-time, data every 10 sec is chosen in the View drop-down menu.

Step 8 From the Windows command line, ping the Super Server at IP address 172.26.26.50 continuously with an ICMP packet size of 100:

```
C:\> ping -l 100  172.26.26.50 -t
Pinging 172.26.26.50 with 100 bytes of data:
Reply from 172.26.26.50:  bytes=100 time<10ms TTL=125>
Reply from 172.26.26.50:  bytes=100 time<10ms TTL=125>
Reply from 172.26.26.50:  bytes=100 time<10ms TTL=125>
Reply from 172.26.26.50:  bytes=100 time<10ms TTL=125>
```

Step 9 Observe the graph in the Graph tab.

Step 10 Select the **Table** tab and observe the statistics in the table view.

Step 11 From the Command Line window, press **Ctrl-C** to end the continuous Ping.

Step 12 Close the Command Prompt window.

Step 13 Close the New Graph window.

Step 14 Save the PIX Firewall configuration to Flash memory by clicking the Save icon in the PDM toolbar. The Save Running Configuration to Flash window opens.

Step 15 Click **Apply**. The Preview CLI Commands window opens.

Step 16 Click **Send**.

Task 11—Configure a Site-to-Site VPN

Complete the following steps to create a secure site-to-site VPN between the PIX Firewall and the peer pod's PIX Firewall:

Step 1 Choose **Wizards>VPN Wizard** from the main menu. The VPN Wizard window opens.

Step 2 Verify that the Site-to-Site VPN radio button is selected.

Step 3 Verify that the outside interface is chosen from the drop-down box.

Step 4 Click **Next**. The Remote Site Peer window opens.

Step 5 Enter **192.168.Q.2** in the Peer IP Address field.
(where Q = peer pod number)

Step 6 Verify that the Pre-shared Key radio button is selected from the Authentication group box.

Step 7 Enter **cisco123** in the Pre-shared Key field.

Step 8 Enter **cisco123** in the Reenter Key field.

Step 9 Click **Next**. The IKE Policy window opens.

Step 10 Choose **DES** from the Encryption drop-down menu.

Step 11 Choose **SHA** from the Authentication drop-down menu.

Step 12 Choose **Group 1 (768-bit)** from the DH Group drop-down menu.

Step 13 Click **Next**. The Transform Set window opens.

Step 14 Choose **DES** from the Encryption drop-down menu.

Step 15 Choose **SHA** from the Authentication drop-down menu.

Step 16 Click **Next**. The IPSec Traffic Selector window opens.

Step 17 Verify that the IP Address radio button is selected within the Host/Network group box.

Step 18 Verify that **inside** is chosen from the Interface drop-down menu.

Step 19 Enter **192.168.P.10** in the IP Address field.
(where P = pod number)

Step 20 Choose **255.255.255.255** from the Mask drop-down menu.

Step 21 Click the arrow to move the host address to the Selected list. The Add host/network widow opens.

Step 22 Click **OK**. The Create host/network window opens. The IP address and netmask for the inside host appear in the Basic Information group box.

Step 23 Verify that **inside** appears in the Interface drop-down menu.

Step 24 Click **Next**. The Static Route frame appears.

Step 25 Click **Next**. The NAT (Network Address Translation) frame appears.

Step 26 Click **Finish**. You are returned to the IPSec Traffic Selector window.

Step 27 Click the arrow button to move the IP address 192.168.P.10 to the Selected list.
(where P = pod number)

Step 28 Click **Next**. The IPSec Traffic Selector (Continue) window opens.

Step 29 Verify that the IP Address radio button is selected within the Host/Network group box.

Step 30 Verify that **outside** is chosen in the Interface drop-down menu.

Step 31 Enter the statically mapped IP address of the peer's inside host, **192.168.Q.10,** in the IP Address field.
(where Q = peer pod number)

Step 32 Choose **255.255.255.255** from the Mask drop-down menu.

Step 33 Click the arrow button to move the IP address **192.168.Q.10** to the Selected list. The Add host/network? window opens.
(where Q = peer pod number)

Step 34 Click **OK**. The Create host/network window opens. The IP address and netmask for the peer's inside host appears in the Basic Information group box.

Step 35 Verify that **outside** appears in the Interface drop-down menu.

Step 36 Click **Next**. A reminder appears in the Create host/network window.

Step 37 Click **Finish**. You are returned to the IPSec Traffic Selector (Continue) window.

Step 38 Click the arrow button to move the IP address of the peer's inside host to the Selected list.

Step 39 Click **Finish**. The Preview CLI Commands window opens.

Step 40 Click **Send**. You are returned to the Cisco PIX Device Manager main window.

Step 41 Click the **Save** icon. The Save Running Configuration to Flash window opens.

Step 42 Click **Apply**. The Preview CLI Commands window opens.

Step 43 Click **Send**.

Task 12—Test and Verify the VPN

Complete the following steps to test the site-to-site VPN:

Step 1 Test the web access to your peer's inside host from your student PC by completing the following substeps:

1. Open a web browser on the student PC.
2. Use the web browser to access the peer's inside host by entering **http://192.168.Q.10**.
(where Q = peer pod number)

The home page of the peer's inside host should open in the web browser.

Step 2 Observe the IKE Security Association (SA):

```
pixP(config)# show crypto isakmp sa
```

Step 3 Display the IPSec SA, and observe the number of encrypted and decrypted packets:

```
pixP(config)# show crypto ipsec sa
```

Step 4 Refresh the browser.

Step 5 Display the IPSec SA again and observe the number of encrypted and decrypted packets. The number should have incremented.

```
pixP(config)# show crypto ipsec sa
```

Step 6 Clear the IPSec SAs with the **clear crypto sa** command:

```
pixP(config)# clear crypto sa
```

Step 7 Remove all **isakmp** command statements from the configuration with the **clear isakmp** command:

```
pixP(config)# clear isakmp
```

Step 8 Remove all parameters entered through the crypto map command with the clear crypto map command:

```
pixP(config)# clear crypto map
```

Step 9 Remove the **sysopt** command statements from the configuration with the **clear sysopt** command:

```
pixP(config)# clear sysopt
```

Step 10 From PDM, click the **Save** icon. The Save Running Configuration to Flash window opens.

Step 11 Click **Apply**. The Preview CLI Commands window opens.

Step 12 Click **Send**.

Step 13 Close the Cisco PIX Device Manager 3.0 window. The Unapplied Changes window opens.

Step 14 Click **Discard Changes**. The Are you sure window opens.

Step 15 Click **Yes**.

Introduction to Enterprise PIX Firewall Management

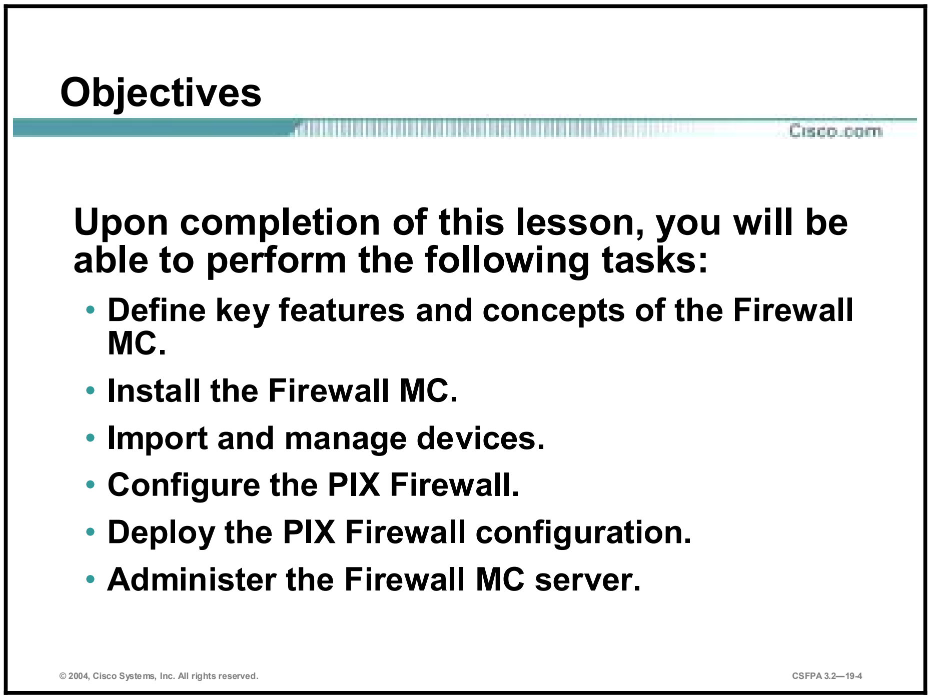
Overview

This lesson introduces enterprise PIX Firewall management using the CiscoWorks Management Center for Firewalls (Firewall MC). The following topics are covered in this lesson:

- Objectives
- Introduction
- Firewall MC hardware requirements
- Preparing for the Firewall MC
- Understanding the Firewall MC
- Importing and managing devices
- Configuring settings
- Configuring building blocks
- Configuring access and translation rules
- Managing workflow
- Reporting
- Summary
- Lab exercise

Objectives

This topic lists the lesson's objectives.



Objectives

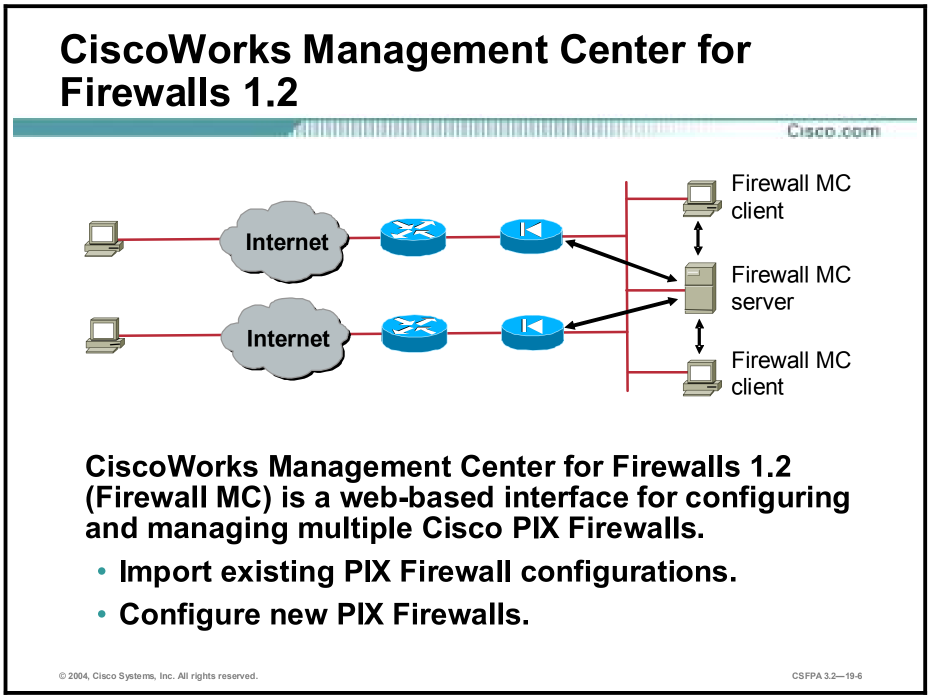
Upon completion of this lesson, you will be able to perform the following tasks:

- **Define key features and concepts of the Firewall MC.**
- **Install the Firewall MC.**
- **Import and manage devices.**
- **Configure the PIX Firewall.**
- **Deploy the PIX Firewall configuration.**
- **Administer the Firewall MC server.**

CSFPA 3.2—19-4

Introduction

This topic introduces the CiscoWorks Management Center for Firewalls (Firewall MC).



CiscoWorks Management Center for Firewalls 1.2

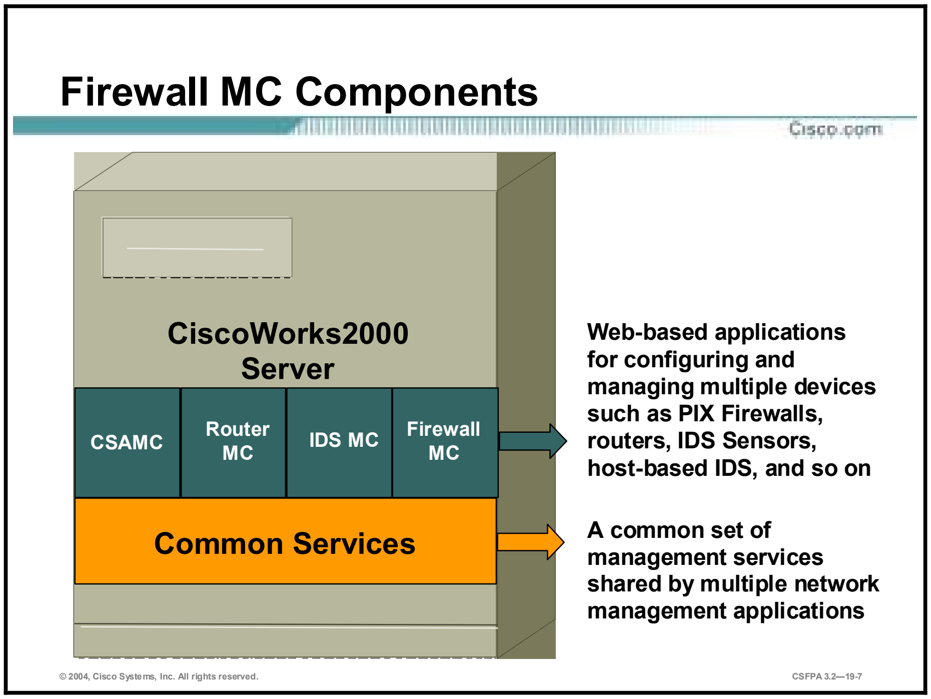
CiscoWorks Management Center for Firewalls 1.2 (Firewall MC) is a web-based interface for configuring and managing multiple Cisco PIX Firewalls.

- **Import existing PIX Firewall configurations.**
- **Configure new PIX Firewalls.**

CSFPA 3.2—19-6

The Firewall MC enables you to configure new PIX Firewalls or import existing firewalls to be managed by the Firewall MC. The Firewall MC provides the following features:

- Web-based interface for configuring and managing multiple PIX Firewalls
- Configuration hierarchy and user interface to facilitate configuration of settings, rules, and building blocks applied to groups, subgroups, and devices
- Support for PIX Firewall Software Versions 6.0 and above
- Ability to import configurations from existing PIX Firewalls
- Ability to support dynamically addressed PIX Firewalls
- Support for up to 1,000 PIX Firewalls
- Secure Sockets Layer (SSL) protocol support to ensure secure remote connectivity between browser and server communication, and between server and device communication
- Workflow and audit trail



Firewall MC Components

Web-based applications for configuring and managing multiple devices such as PIX Firewalls, routers, IDS Sensors, host-based IDS, and so on

A common set of management services shared by multiple network management applications

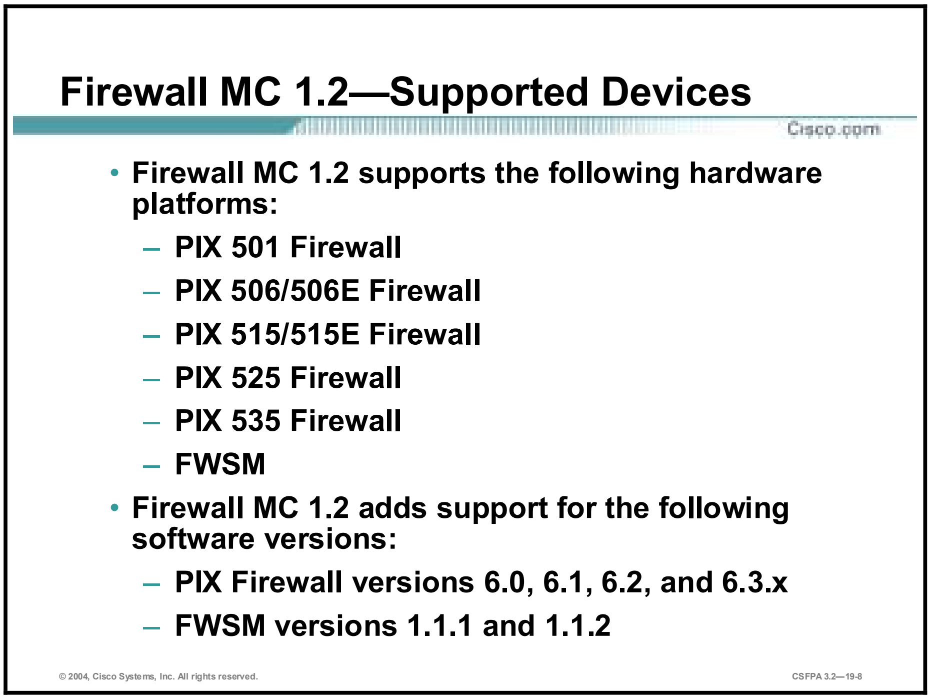
CSFPA 3.2—19-7

CiscoWorks Management Center for Firewalls 1.2 (Firewall MC) is a web-based interface for configuring and managing multiple Cisco PIX Firewalls. Firewall MC looks and feels similar to Cisco PIX Device Manager (PDM); however, with Firewall MC, you can configure multiple firewalls instead of only one. Firewall MC centralizes and accelerates the deployment and management of multiple PIX Firewalls.

CiscoWorks Management Center for Firewalls 1.2 consists of two components, CiscoWorks Common Services and Firewall MC. Common Services represents a common set of management services that are shared by multiple network management applications, such as Firewall MC, Router MC, Intrusion Detection System (IDS) MC, and so on. Common Services is required for the Firewall MC. Common Services provides the server-based components, software libraries, and software packages. Common Services provides services for:

- Interacting with the CiscoWorks desktop
- Setting up the CiscoWorks Server
- Administering the CiscoWorks Server
- Adding external connections to the CiscoWorks Server
- Database administration for MC applications
- System administration
- Logging
- Diagnosing problems with the CiscoWorks Server

The other component of the CiscoWorks Management Center for Firewalls is the Firewall MC. Firewall MC enables you to configure new PIX Firewalls and Firewall Services Modules (FWSMs) and import configurations from existing firewalls. You can configure firewall device settings, access rules, and translation rules and deploy these configurations to your network. Firewall MC is a powerful tool for controlling changes made to your network, showing configuration and status changes.



Firewall MC 1.2—Supported Devices

- Firewall MC 1.2 supports the following hardware platforms:
  - PIX 501 Firewall
  - PIX 506/506E Firewall
  - PIX 515/515E Firewall
  - PIX 525 Firewall
  - PIX 535 Firewall
  - FWSM
- Firewall MC 1.2 adds support for the following software versions:
  - PIX Firewall versions 6.0, 6.1, 6.2, and 6.3.x
  - FWSM versions 1.1.1 and 1.1.2

CSFPA 3.2—19-8

The Firewall MC 1.2 supports certain hardware platforms with specific software versions. The Firewall MC 1.2 supports the following hardware platforms:

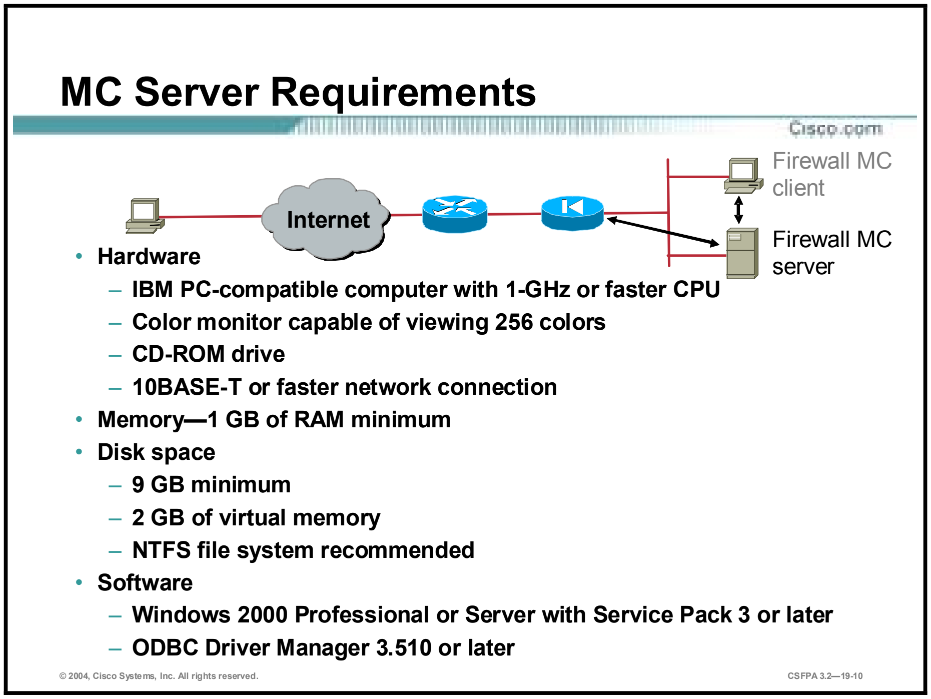
- PIX 501 Firewall
- PIX 506E Firewall
- PIX 515E Firewall
- PIX 525 Firewall
- PIX 535 Firewall
- FWSM

In addition, it supports the following operating system versions:

- PIX Firewall Software Versions 6.0, 6.1, 6.2, and 6.3.2
- FWSM Version 1.1.1 and 1.1.2

Firewall MC Hardware Requirements

This topic introduces and explains the server and client hardware requirements of the Firewall MC.



MC Server Requirements

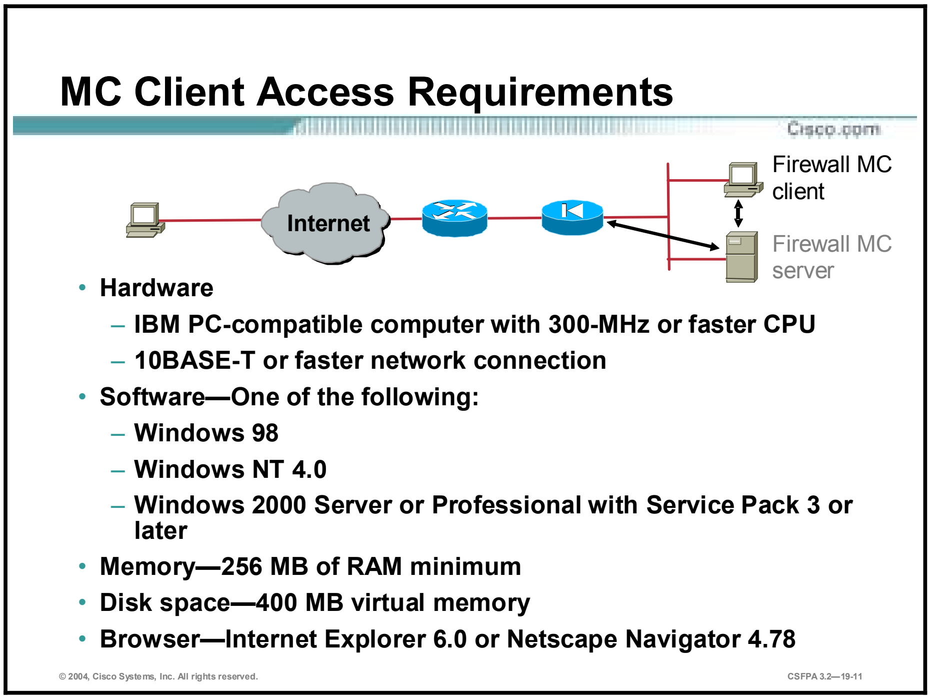
Cisco.com
Firewall MC client
Internet
Firewall MC server

- Hardware
  - IBM PC-compatible computer with 1-GHz or faster CPU
  - Color monitor capable of viewing 256 colors
  - CD-ROM drive
  - 10BASE-T or faster network connection
- Memory—1 GB of RAM minimum
- Disk space
  - 9 GB minimum
  - 2 GB of virtual memory
  - NTFS file system recommended
- Software
  - Windows 2000 Professional or Server with Service Pack 3 or later
  - ODBC Driver Manager 3.510 or later

CSFPA 3.2—19-10

Before you begin, verify that the server on which you plan to install the Firewall MC meets the following requirements:

- Hardware
  - IBM PC-compatible computer with 1-GHz or faster CPU
  - Color monitor capable of viewing 256 colors
  - A CD-ROM drive
  - A 10BASE-T or faster network connection
- Memory—1 GB of RAM minimum
- Disk space
  - 9 GB minimum
  - File Allocation Table 32 (FAT32) or New Technology File System (NTFS) recommended for security reasons
  - 2 GB of virtual memory
- Software
  - Microsoft Windows 2000 Server with Service Pack 3 or later.
  - Open Database Connectivity (ODBC) Driver Manager 3.510 or later



MC Client Access Requirements

- Hardware
  - IBM PC-compatible computer with 300-MHz or faster CPU
  - 10BASE-T or faster network connection
- Software—One of the following:
  - Windows 98
  - Windows NT 4.0
  - Windows 2000 Server or Professional with Service Pack 3 or later
- Memory—256 MB of RAM minimum
- Disk space—400 MB virtual memory
- Browser—Internet Explorer 6.0 or Netscape Navigator 4.78

CSFPA 3.2—19-11

Before you log in to the Firewall MC from a client, verify that the client machine meets the following requirements:

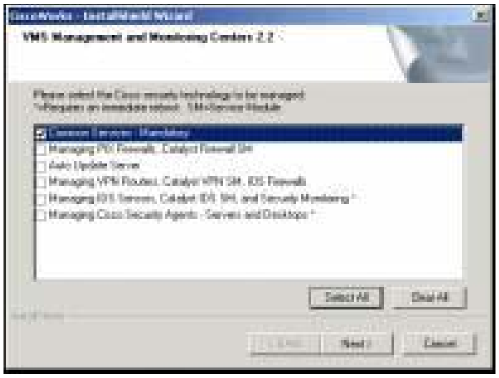
- Hardware—IBM PC-compatible computer with 300-MHz or faster CPU
- Software
  - Windows 98
  - Windows 2000 Professional with Service Pack 3 or later
  - Windows 2000 Server/Advanced Server with Service Pack 3 or later
- Memory—256 MB of RAM minimum
- Disk space—400 MB of virtual memory
- Browser—Internet Explorer 6.0 or Netscape Navigator 4.78

Installation Process

Cisco.com

Installation Process

- Step 1—Install Common Services
- Step 2—Install Firewall MC
  - Auto Update Server
  - Other MCs



CiscoWorks - InstallShield Wizard

VMS Management and Monitoring Centers 2.2

Please select the Cisco security technology to be managed:
*Requires an immediate reboot. 1 McService Module

☑ Common Services (Mandatory)
□ Managing PIX Firewalls, Catalyst Firewall SM
□ Auto Update Server
□ Managing VPN Routers, Catalyst VPN SM, IOS Firewalls
□ Managing IDS Sensors, Catalyst IDS SM, and Security Monitoring *
□ Managing Cisco Security Agents - Servers and Desktops *

CSFPA 3.2—19-12

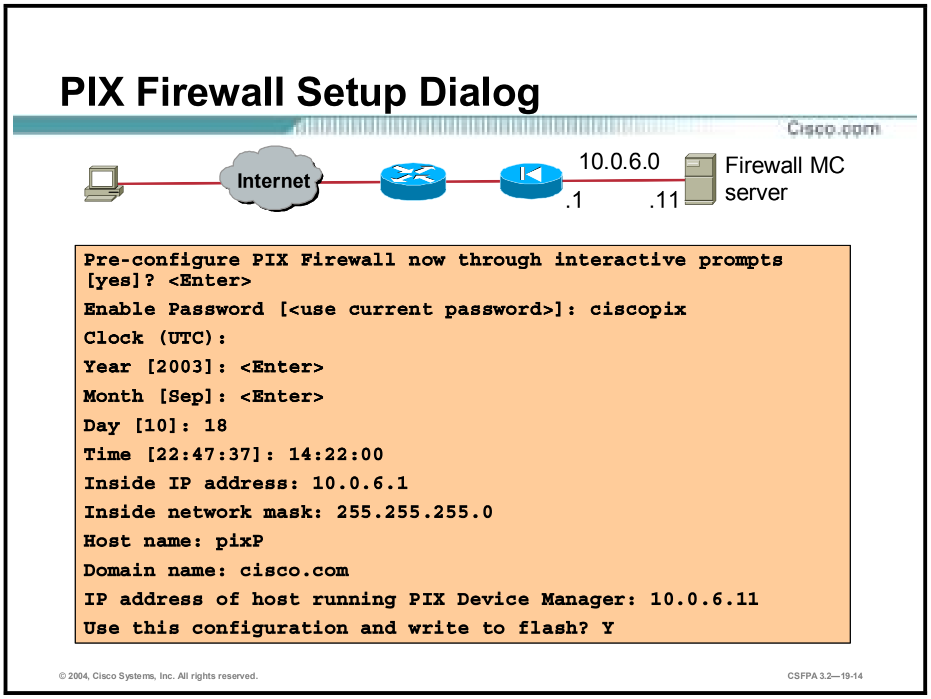
Installation of Firewall MC is a two-step process:

Step 1 The administrator installs CiscoWorks Common Services. Common Services is mandatory and must be installed first.

Step 2 After Common Services is installed, install the remaining Firewall MC applications, such as those for managing PIX Firewalls, managing Virtual Private Network (VPN) routers, managing Intrusion Detection System (IDS) Sensors, and so on.

Preparing for Firewall MC

This topic explains the prerequisites for managing the PIX Firewall via the Firewall MC. The PIX must be preconfigured so the Firewall MC can communicate with the PIX Firewall.



PIX Firewall Setup Dialog

```
Pre-configure PIX Firewall now through interactive prompts
[yes]? <Enter>
Enable Password [<use current password>]: ciscopix
Clock (UTC):
Year [2003]: <Enter>
Month [Sep]: <Enter>
Day [10]: 18
Time [22:47:37]: 14:22:00
Inside IP address: 10.0.6.1
Inside network mask: 255.255.255.0
Host name: pixP
Domain name: cisco.com
IP address of host running PIX Device Manager: 10.0.6.11
Use this configuration and write to flash? Y
```

CSFPA 3.2—19-14

An unconfigured PIX Firewall starts in an interactive setup dialog to enable the administrator to perform the initial configuration required to use the Firewall MC. The administrator can also access the setup dialog by entering **setup** at the configuration mode prompt.

The dialog asks for several responses, including the inside IP address, network mask, hostname, domain name, and PIX Device Manager (or Firewall MC host). The hostname and domain name are used to generate the default certificate for the SSL connection.

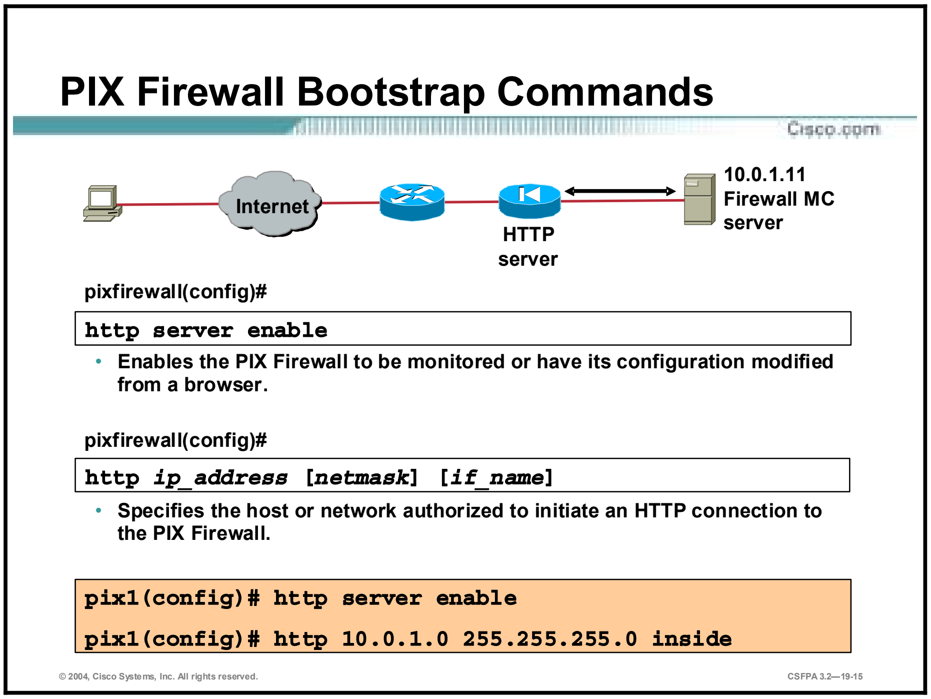
The example in the figure shows how to respond to the **setup** command prompts. Pressing the **Enter** key instead of entering a value at the prompt accepts the default value within the brackets. The administrator must fill in any fields that show no default values, and change default values as necessary. Type **y** to write the configuration to Flash memory.

Caution The clock must be set for the PIX Firewall *to* generate a valid certification. Set the PIX Firewall clock to Universal Coordinated Time (UTC, also known as Greenwich Mean Time).

The following explains each prompt in the setup dialog:

- Enable password—Enables you to specify an enable password for this PIX Firewall.
- Clock (UTC)—Enables you to set the PIX Firewall clock to UTC.
- Year [system year]—Enables you to specify the current year or return to the default year stored in the host computer.

- Month [system month]—Enables you to specify the current month or return to the default month stored in the host computer.
- Day [system day]—Enables you to specify the current day or return to the default day stored in the host computer.
- Time [system time]—Enables you to specify the current time in hh:mm:ss format or return to the default time stored in the host computer.
- Inside IP address—The network interface IP address of the PIX Firewall.
- Inside network mask—A network mask that applies to the inside IP address. Use 0.0.0.0 to specify a default route. The 0.0.0.0 netmask can be abbreviated as 0.
- Host name—The hostname you want to display in the PIX Firewall command line prompt.
- Domain name—The Domain Name System (DNS) domain name of the network on which the PIX Firewall runs (for example, cisco.com).
- IP address of host running PIX Device Manager—Host IP address of the PIX Firewall device manager, PDM or Firewall MC.
- Use this configuration and write to flash?—Enables you to store the new configuration to Flash memory. This prompt is the same as the **write memory** command. If you answer "yes," the inside interface is enabled and the requested configuration is written to Flash memory. If you answer anything else, the setup dialog repeats, using the values already entered as the defaults for the questions.



PIX Firewall Bootstrap Commands

pixfirewall(config)#

```
http server enable
```

- Enables the PIX Firewall to be monitored or have its configuration modified from a browser.

pixfirewall(config)#

```
http ip_address [netmask] [if_name]
```

- Specifies the host or network authorized to initiate an HTTP connection to the PIX Firewall.

```
pix1(config)# http server enable
pix1(config)# http 10.0.1.0 255.255.255.0 inside
```

CSFPA 3.2—19-15

Before you can manage a PIX Firewall with the Firewall MC, the administrator must configure the firewall to be modified with a browser and specify which host or network is allowed to initiate an HTTP connection to the PIX Firewall. The figure shows the commands necessary to complete this operation.

Complete the following steps to configure a PIX Firewall for Firewall MC HTTP access:

Step 1 Enable the PIX Firewall to have its configuration modified from a browser:

```
pixfirewall(config)# http server enable
```

Step 2 Specify the host or network authorized to initiate an HTTP connection to the PIX Firewall:

```
pixfirewall(config)# http ip_address [netmask] [if_name]
```

Step 3 Store the current configuration in Flash memory:

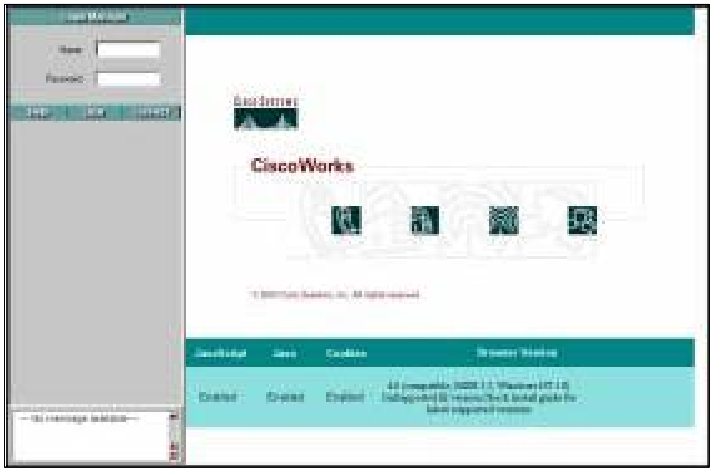
```
pixfirewall(config)# write memory
```

Step 4 Exit the configuration mode:

```
pixfirewall(config)# exit
```

CiscoWorks Login

Cisco.com



© 2004, Cisco Systems, Inc. All rights reserved. CSFPA 3.2—19-17

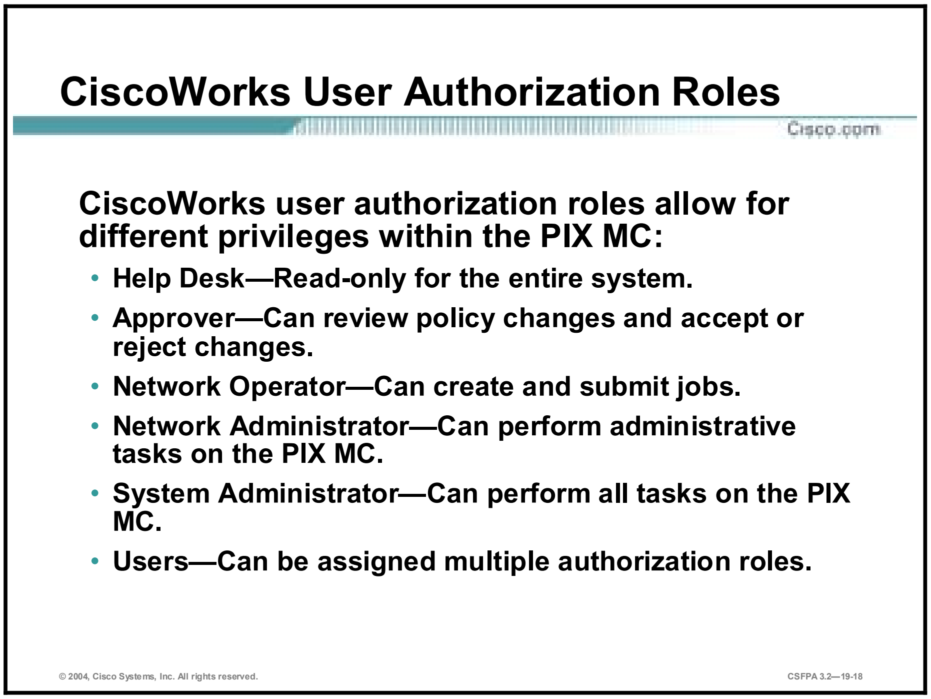
To launch the Firewall MC, you must first log in to CiscoWorks. Complete the following steps to launch the Firewall MC:

Step 1 Open a browser and point your browser to the IP address of the CiscoWorks server with a port number of 1741. If the CiscoWorks server is local, you would type the following address in the browser:

```
http://127.0.0.1:1741 <enter>
```

Step 2 Use the default user name and password of **admin** and **admin** to log in to CiscoWorks for the first time.

Step 3 Click **Connect**.



CiscoWorks User Authorization Roles

CiscoWorks user authorization roles allow for different privileges within the PIX MC:

- **Help Desk—Read-only for the entire system.**
- **Approver—Can review policy changes and accept or reject changes.**
- **Network Operator—Can create and submit jobs.**
- **Network Administrator—Can perform administrative tasks on the PIX MC.**
- **System Administrator—Can perform all tasks on the PIX MC.**
- **Users—Can be assigned multiple authorization roles.**

CSFPA 3.2—19-18

After your username and password pair has been authenticated, your authorization is based on the privileges that you have. Your role is a collection of privileges that dictate the type of system access you have.

Authorization roles can be used to delegate responsibilities to users who log in to the Firewall MC. For example, you can specify who can generate configurations or who can approve configurations. The following types of user authorization roles are available if you use CiscoWorks as your authentication, authorization, and accounting (AAA) provider:

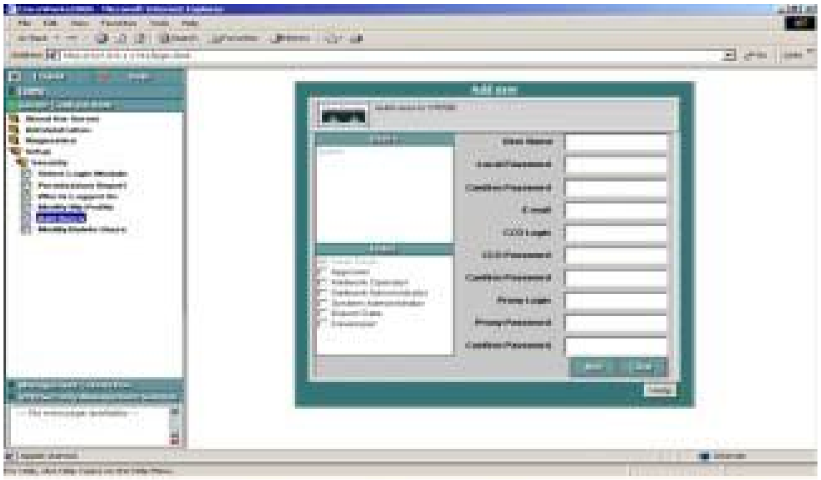
- Help desk—Read only for the entire system
- Approver—Can review policy change and accept or reject changes
- Network operator—Can create and submit jobs
- Network administrator—Can perform administrative tasks on the Firewall MC
- System administrator—Can perform all tasks on the Firewall MC

Users can be assigned multiple authorization roles.

CiscoWorks Add User

Cisco.com

Choose Server Configuration>Setup>Security>Add Users.



CSFPA 3.2—19-19

CiscoWorks provides two predefined login IDs:

- Guest—The login named “guest” is the equivalent of the help desk.
- Admin—The login named “admin” is the equivalent of the superuser (in UNIX) or administrator (in Windows) for CiscoWorks. This login provides access to all CiscoWorks tasks.

Prior to logging in to the Firewall MC, the administrator might want to add some other users based upon how the organization’s security policy is set up. From the Security level, the administrator can manage user accounts by:

- Adding, modifying, and deleting users
- Viewing roles and privileges
- Determining who is logged into the CiscoWorks2000 Server
- Viewing or modifying the CiscoWorks2000 Server Login Module

Administrator determines a user’s CiscoWorks access privileges. Users are assigned one or more roles. The user role, or combination of roles, dictates which CiscoWorks applications are presented to the users on the navigation tree. The following roles are available:

- Help desk
- Approver
- Network operator
- Network administrator
- System administrator
- Export data
- Developer

- Partition administrator

Complete the following steps to add users based upon how CiscoWorks user authorization roles work:

Step 1 Log into the CiscoWorks2000 Server Desktop. The CiscoWorks2000 Server Desktop is displayed.

Step 2 Select **Server Configuration>Setup>Security>Add Users**. The Add User page is displayed.

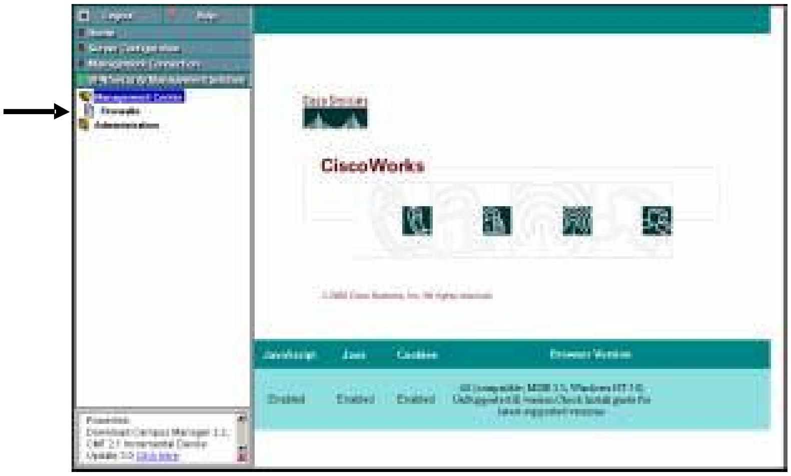
Step 3 Complete the following substeps to add a user to the CiscoWorks database:

1. Enter the user's name in the User Name field.
2. Enter the password for the user in the Local Password field.
3. Re-enter the password in the Confirm Password field.
4. (Optional.) Enter the user's e-mail address.
5. (Optional.) Enter the user's Cisco.com login name in the Cisco.com Login field, if the user has one.
6. (Optional.) Enter the password that is associated with the Cisco.com login in the Cisco.com Password field. Confirm this password by entering it in the Confirm Password field.
7. (Optional.) Enter the user name for the proxy server login, if a proxy server exists on the network.
8. (Optional.) Enter the password associated with the proxy server login. Confirm this password by entering it in the Confirm Password field.
9. Locate the Roles section on the lower left hand side of the Add User page. Use the check boxes to select the roles the user will fulfill.

Step 4 Click **Add** to complete the addition of the user to the CiscoWorks database. The Add User page refreshes to indicate that the change was received.

Launch Firewall MC

Cisco.com



© 2004, Cisco Systems, Inc. All rights reserved.

CSFPA 3.2—19-20

The navigation tree consists of several drawers. Each drawer contains a group of folders, which in turn contain groups of associated or similar tasks, tools, reports, or other options. The contents of the navigation tree vary based on which applications or suites you have installed, and your user role. However, certain drawers and features are common to all applications and suites. The four drawers listed are as follows:

- Home—Includes links to additional resources on Cisco.com and a folder for storing frequently accessed tasks
- Server Configuration—Includes applications and tools for setting up, administering, and diagnosing the CiscoWorks2000 Server
- Management Connection—Includes applications for adding external links to CiscoWorks and a collection of links to commonly used tools on Cisco.com
- VPN/Security Management Solution—Includes applications and tools for configuration management and administration services for MCs. This drawer is available only if Management Center (MC) applications are installed on the CiscoWorks2000 Server, such as the Firewall MC.

Complete the following steps to launch the Firewall MC:

Step 1 Log in to the CiscoWorks server where the Firewall MC has been installed.

Step 2 Select **VPN/Security Management Solution>Management Center**. After the Management Center folder has expanded, you will see the icon for Firewalls.

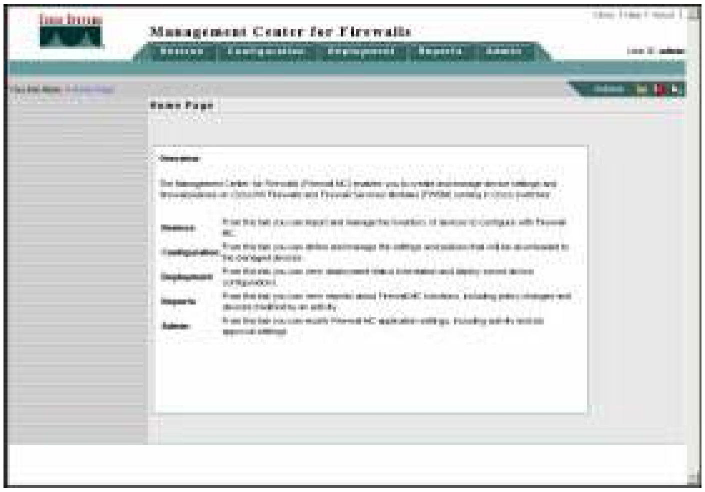
Step 3 Click the **Firewalls** icon to launch the Firewall MC. A Security Alert window is displayed.

Step 4 Click **Yes** to accept the security certificate. The Firewall MC home page opens in another window.

Step 5 Minimize CiscoWorks in the background to avoid confusion when working with the Firewall MC.

Firewall MC Home Page

Cisco.com



CSFPA 3.2—19-21

When an administrator first logs in to the Firewall MC, the administrator is presented with five tabs. The Firewall MC enables the administrator to create and manage device settings and firewall policies for PIX Firewalls and FWSMs running in Cisco switches. The five tabs are as follows:

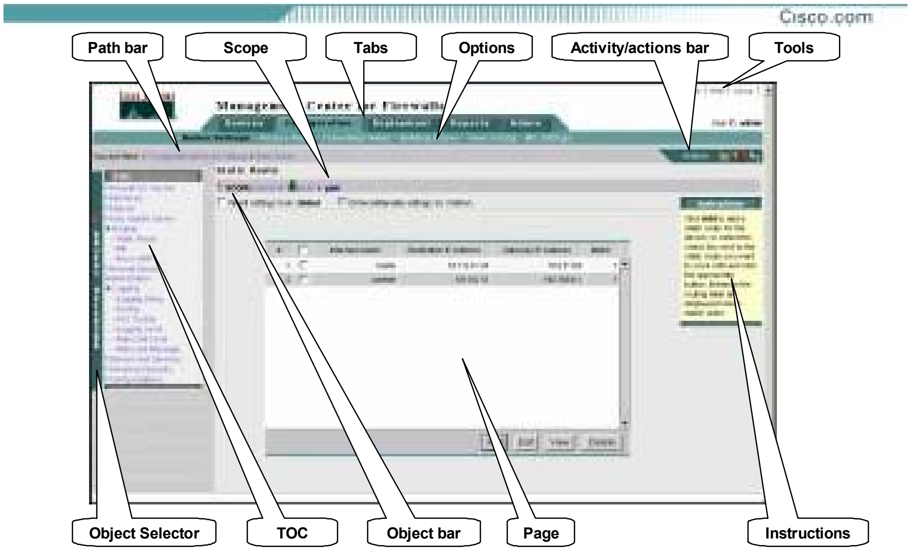
- Devices—Import and manage the inventory of devices to configure with the Firewall MC.
- Configuration—Define and manage the settings and policies that are downloaded to the managed devices.
- Deployment—View deployment status information and deployed saved configurations.
- Reports—View reports about the Firewall MC.
- Admin—Modify Firewall MC application settings, including activity and job approval settings.

A sixth tab, Workflow, is displayed only if workflow is enabled. By default, workflow is disabled. The Workflow tab is used to manage activities and jobs.

Understanding the Firewall MC

This topic introduces and explains the Firewall MC GUI and the components of the basic user task flow.

Firewall MC Interface

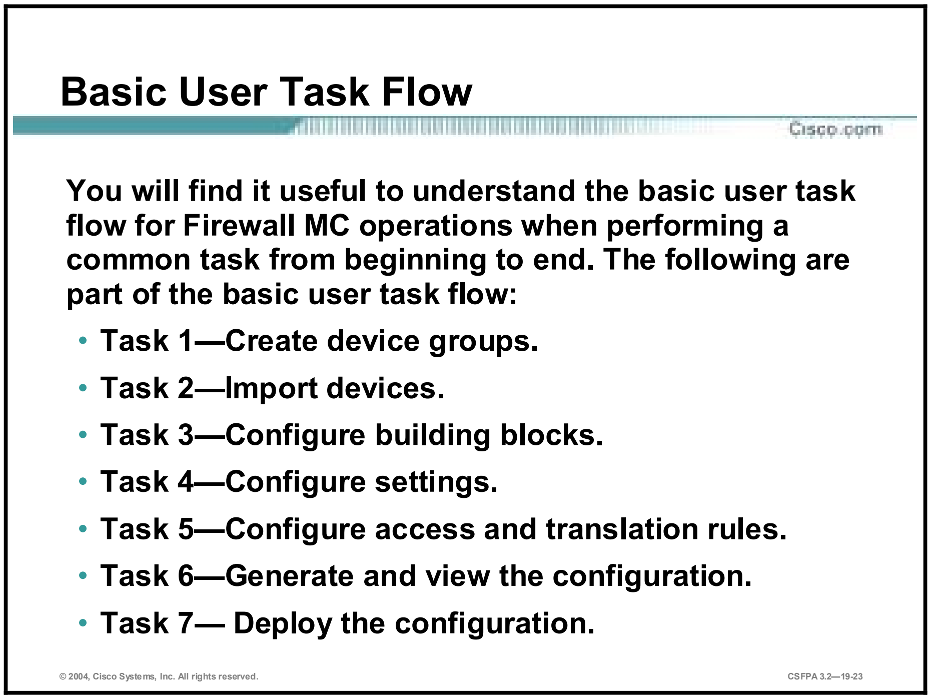


CSFPA 3.2—19-22

By learning the elements contained in the Firewall MC interface, you will be able to carry out the basic user task flow and navigate the Firewall MC with ease. The figure illustrates the Firewall MC GUI. The elements of the Firewall MC GUI are:

- Path bar—Provides a context for the displayed page. This bar shows tab, option, and then current page.
- TOC—Table of contents. The TOC displays available suboptions, if any are available.
- Options bar—This bar displays the options available for the tab.
- Tabs—Tabs provide access to product functionality. Clicking a tab gives you access to its options.
  - Devices—Set up the Firewall MC for operation; import devices and arrange them into groups.
  - Configuration—Enter PIX Firewall configuration information by identifying settings, rules, and building blocks.
  - Deployment—Deploy configurations to devices, a file, or a CiscoWorks Auto Update Server (AUS); displayed when the workflow feature is disabled (default).
  - Workflow—Manage activities and jobs (disabled by default).
  - Reports—Display a report listing activity information.
  - Admin—Perform administrative tasks.

- Activity and actions bar—This bar displays activity and action icons that change, depending upon what state the activity is in; viewed from the Devices and Configuration tabs only. Options are:
  - Add—Add a new activity.
  - Open—Open a new or existing activity. A popup window opens from which you make your selection.
  - Close—Close the activity shown by the activity bar.
  - Save and Deploy—Save and generate a device configuration file. This option enables the administrator to deploy the configuration or postpone the deployment.
  - Submit—Submit an activity.
  - Reject—Reject an activity.
  - Approve—Approve the activity shown by the activity bar.
  - Undo—Discard the activity shown by the activity bar.
  - View Details—Show the details of the current changes.
- Instructions—This area provides a brief overview of how to use the page.
- Page—This pane displays the area in which you perform application tasks.
- Object bar—This area displays the object selected in the Object Selector.
- Object Selector—The Object Selector enables you to select groups or devices.



Basic User Task Flow

You will find it useful to understand the basic user task flow for Firewall MC operations when performing a common task from beginning to end. The following are part of the basic user task flow:

- Task 1—Create device groups.
- Task 2—Import devices.
- Task 3—Configure building blocks.
- Task 4—Configure settings.
- Task 5—Configure access and translation rules.
- Task 6—Generate and view the configuration.
- Task 7— Deploy the configuration.

CSFPA 3.2—19-23

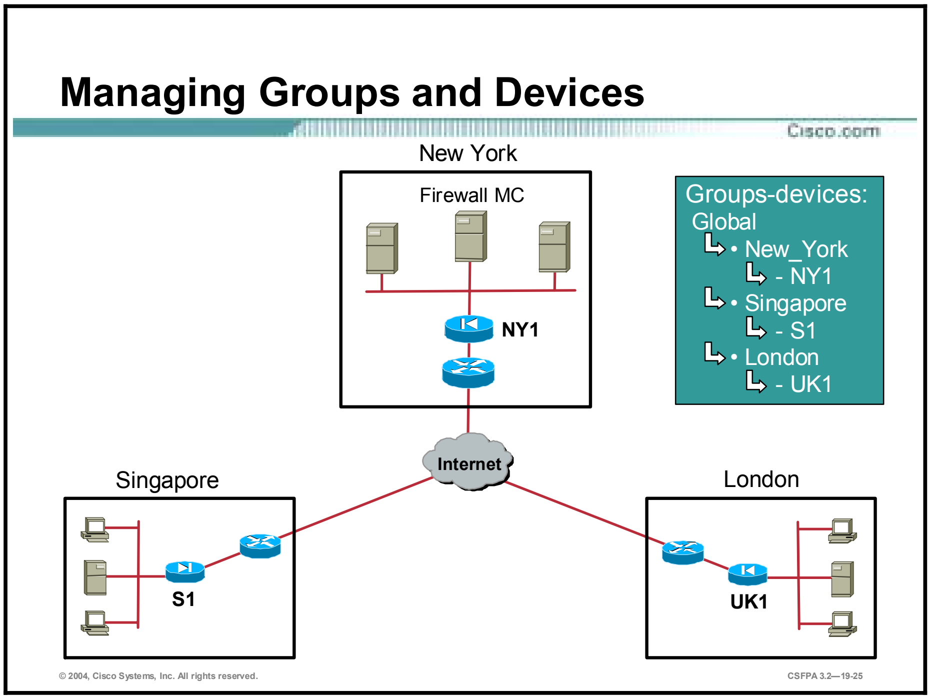
The Firewall MC enables you to navigate to various areas of the application as appropriate to your needs and the task at hand. For first-time users, it is useful to understand the basic flow for performing a common task from beginning to end. The basic user task flow is as follows:

- Task 1—(Optional.) Create device groups. This optional task allows the administrator to define groups so that PIX Firewalls can be grouped during the importation process or after the importation process.
- Task 2—Import devices. In the Firewall MC, “importing a device” can be defined as either creating a device or importing the PIX Firewall configuration.
- Task 3—Configure building blocks. With this task, the administrator can optimize the configuration through the logical grouping of hosts, protocols, or services.
- Task 4—Configure settings. The administrator can configure various options for PIX Firewalls, such as routing and advanced security.
- Task 5—Configure access and translation rules. With this task, the administrator can configure rules for access control and the translation of addresses.
- Task 6—Generate and view the configuration. When generating and viewing the configuration prior to submission, the administrator can check for errors in the configuration. The administrator can then go back and correct any errors as needed.
- Task 7—Deploy the configuration. With this task, the administrator deploys configuration files to devices.

Note The approval process for activities or a job is disabled by default. Select **Admin>Workflow Setup** to enable the approval process. If the approval process is enabled, all activities must be approved before you can propose configuration changes for deployment. If the approval process is *disabled*, the submit and approve process is a single step—the activity is approved automatically when it is submitted.

Importing and Managing Devices

This topic explains importing and managing devices.



Managing Groups and Devices

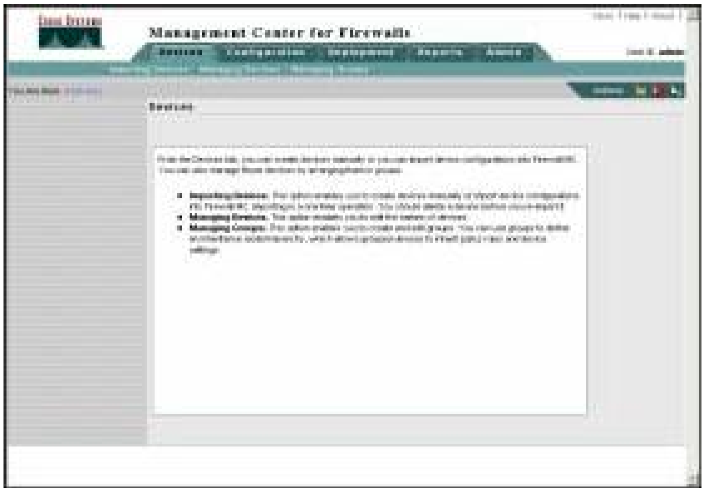
Groups-devices:
Global
↳ • New_York
↳ - NY1
↳ • Singapore
↳ - S1
↳ • London
↳ - UK1

CSFPA 3.2—19-25

In the Firewall MC, there is a default group named "Global." The administrator can add or import devices directly into the global group or define subgroups in which devices can reside. It is recommended that the administrator set up easily identifiable groups and add, or import, PIX Firewalls into these groups. In the example in the figure, the administrator has three sites, with one PIX Firewall located at each site. The administrator defined a group for each site, New_York, Singapore, and London. The administrator added, or imported, the local PIX Firewall into its subgroup for example, PIX Firewall NY1 is in the New_York folder. Groups and devices can be added, deleted, and modified from the Devices tab.

Devices Tab

Cisco.com



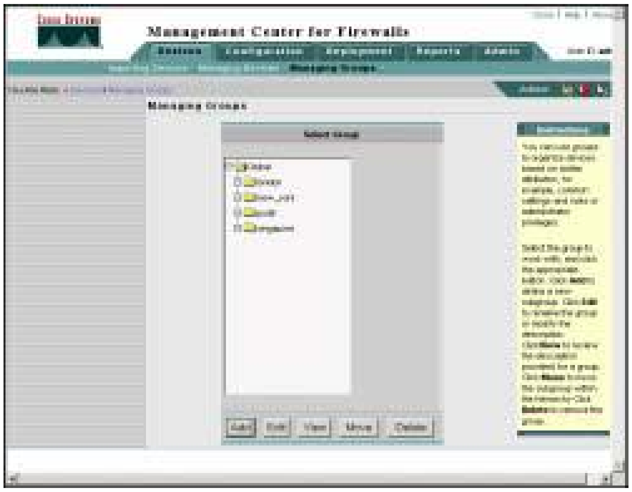
CSFPA 3.2—19-26

From the Devices tab, the administrator can import device configurations into the Firewall MC. The administrator can also manage those devices by arranging them into groups. The Devices tab offers three options:

- Importing Devices—Create devices manually or import device configurations into the Firewall MC. Importing is a one-time operation.
- Managing Devices—Edit names of devices.
- Managing Groups—Create and edit groups. Use groups to define an inheritance model hierarchy, which allows grouped devices to inherit policy rules and devices settings.

Managing Groups

Cisco.com



CSFPA 3.2—19-27

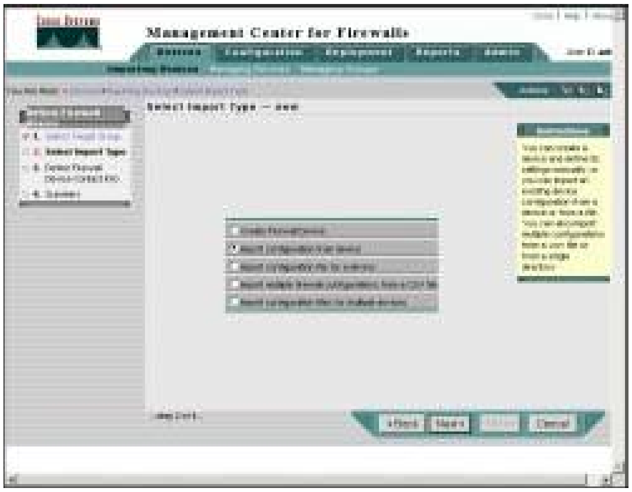
The Managing Groups feature allows you to add new groups to the system, modify existing groups, or delete existing groups. You access this feature by selecting **Devices>Managing Groups**.

When you set up a new group, it is recommended that you use a name and description that allows easy identification. For example, you can define and identify groups by region, department within your company, or any other grouping based on a commonality. A subgroup cannot have the same name as the enclosing group, and no two subgroups within an enclosing group can have the same name.

In the example in the figure, the administrator created four groups, london, new_york, pod6, and singapore.

Import Configuration from Device

Cisco.com



© 2004, Cisco Systems, Inc. All rights reserved.

CSFPA 3.2—19-28

Once groups are defined, you are ready to import a device. The first step for importing a device is to select the group in which you want the device to reside. To select a group, select **Devices>Importing Devices>Select Target Group**. Once the target group is selected, the administrator is prompted by the Firewall MC to choose an import type.

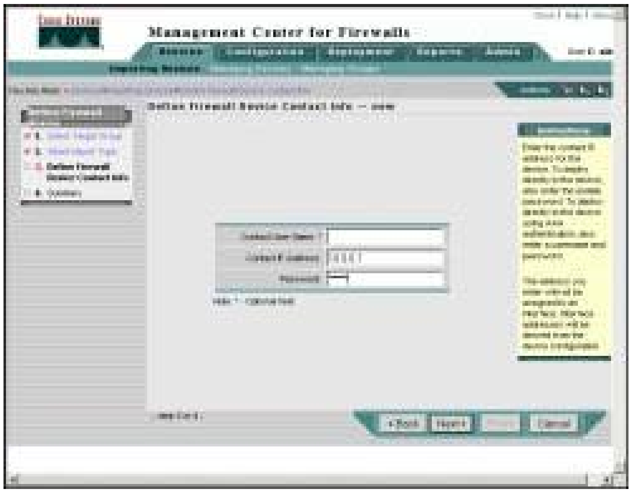
In the Firewall MC, importing a device can be defined as any of the import types shown in the figure:

- Create Firewall Device—Allows you add a single device manually.
- Import configuration from device—Allows you to manually provide device credentials that allow the Firewall MC server to “talk” directly to a device to retrieve configuration information. This option specifies that you want the Firewall MC to connect to and discover current settings on a PIX Firewall.
- Import configuration file for a device—Allows you to import configuration information for a single device from a configuration file.
- Import multiple firewall configurations from a CSV file—Allows the Firewall MC server to “talk” directly to multiple PIX Firewalls specified in a CSV file to retrieve configuration information.
- Import configuration files for multiple devices—Allows you to import multiple configuration files from a single directory. Each file contains configuration information for a single device.

Note You can import from a device only once. If you need to reimport a PIX Firewall’s configuration information, you must delete the information and then reimport it.

PIX Firewall Contact Information

Cisco.com


Management Center for Firewalls
Define Firewall Device Contact Info — new
Define Firewall Device Contact Info
< Back
Next
Cancel

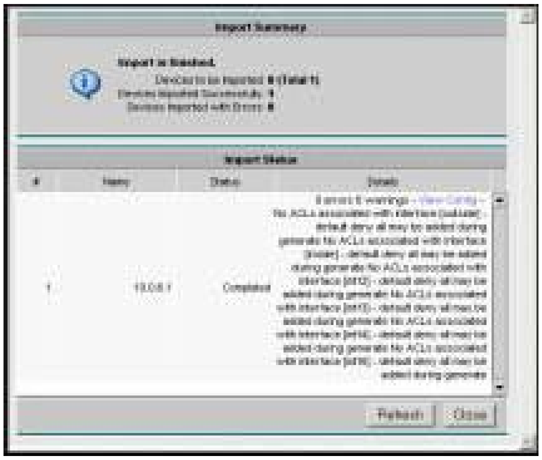
CSFPA 3.2—19-29

If you select **Import configuration from device** from the Select Import Type—new page and click **Next**, the Define Firewall Device Contact Info—new page is displayed. Enter the following information in this page:

- Contact IP Address—The IP address that the Firewall MC uses to contact the PIX Firewall.
- Password—The PIX Firewall enable password.

This information enables the Firewall MC to communicate with the target PIX Firewall.

Import Summary



Import Summary

Import is finished.

Devices to be Imported: 0 (Total 1)
Devices Imported Successfully: 1
Devices Imported with Errors: 0

Import Status

#	Name	Status	Details
1	10.0.0.1	Completed	0 errors 6 warnings - View Config - No ACLs associated with interface [outside] - default deny all may be added during generate No ACLs associated with interface [inside] - default deny all may be added during generate No ACLs associated with interface [intf2] - default deny all may be added during generate No ACLs associated with interface [intf3] - default deny all may be added during generate No ACLs associated with interface [intf4] - default deny all may be added during generate No ACLs associated with interface [intf5] - default deny all may be added during generate

CSFPA 3.2—19-30

When you have supplied the required information, click **Next**, verify that the information displayed on the Summary page is correct, and click **Finish**. The Import Status popup window is displayed.

The Import Status window refreshes automatically every 60 seconds; however, you can click the **Refresh** button to update the import status manually. If the import is successful, the message "Completed" is displayed in the Status column. If the import is unsuccessful, an error message is displayed.

After the import status is displayed in the Status column, you can select a device in the table and click the **View Config** link. A new window opens with the configuration file displayed.

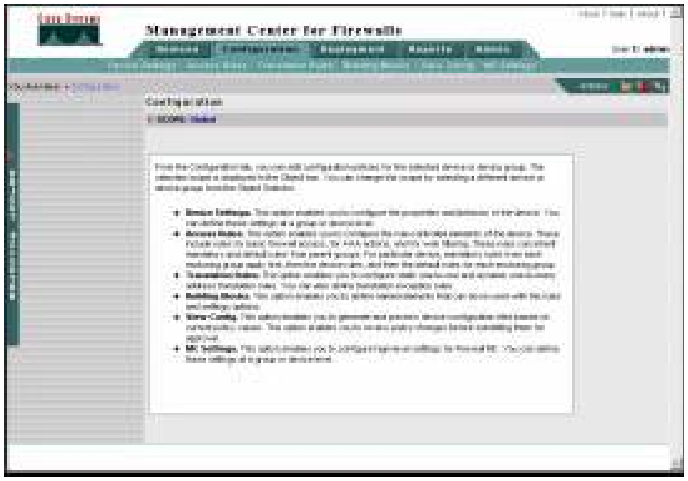
Close the window after you view the contents, and then close the Import Status popup window. You are returned to the Importing Devices page, which shows the imported device information.

Configuring Settings

This topic introduces and explains configuring settings.

Configuration Tab

Cisco.com



© 2004, Cisco Systems, Inc. All rights reserved.

CSFPA 3.2—19-32

From the Configuration tab, the administrator can edit configuration policies for the selected device or device group. Under the Configuration tab are six options:

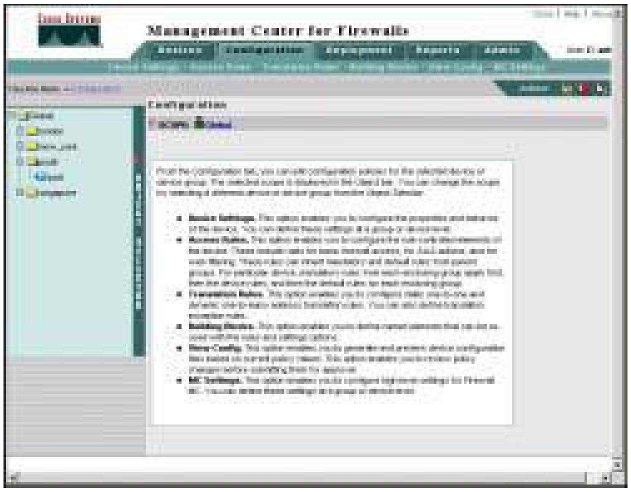
- Device settings—Use to change the properties and behavior of a device.
- Access Rules—Configure the ruled-controlled elements of a device. These rules include basic firewall access, AAA access, and web filtering.
- Translation Rules—Configure static and dynamic translation rules.
- Building Blocks—Define named elements that can be reused with rules and settings options.
- View Config—Generate and preview device configuration files based on current policy values.
- MC Settings—Define group- or device-level settings for the Firewall MC.

Object Selector

Cisco.com

Configuration:

- Global
  - pod6
    - pix6



From the Configuration bar, you can edit configuration policies for the selected device or device group. The selected scope is displayed in the Object bar. You can change the scope by selecting a different device or device group from the Object Selector.

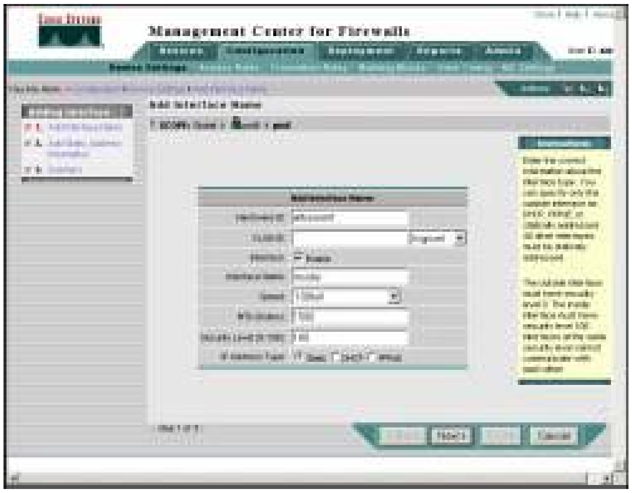
- **Device Settings.** [illegible]
- **Access Rules.** [illegible]
- **Translation Rules.** [illegible]
- **Building Blocks.** [illegible]
- **View Config.** [illegible]
- **MC Settings.** [illegible]

Configuration settings are those settings that control individual features of a firewall device. Configuration settings define characteristics for the device, such as interface definitions and failover settings, that are required for the device to operate on the network. When defining settings under Configuration, you specify whether to define them for a group or device. Selecting a group allows you to define settings for all the members of that group—inheritable attributes. Selecting a device allows the administrator to define device specific attributes.

The selected scope is displayed in the object bar. The administrator can change the scope by selecting a different device or device group from the Object Selector. In the example in the figure, the administrator can configure settings at the Global, pod6 group, or pix6 device level. The scope defaults to Global.

Add Interface

Cisco.com



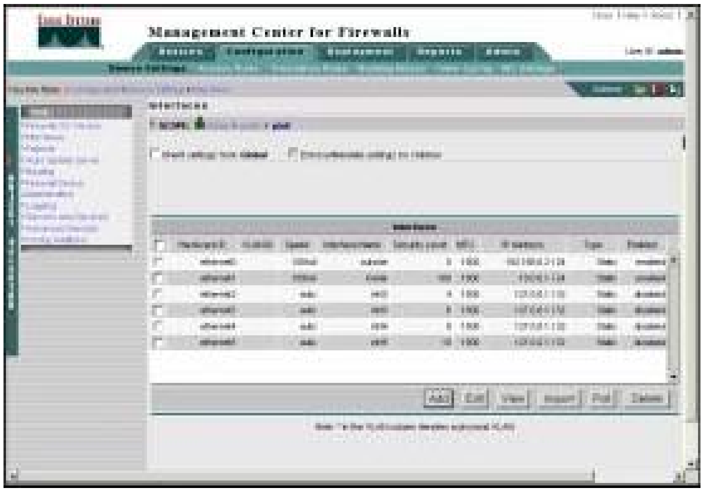
CSFPA 3.2—19-34

The Interfaces feature allows you to enable, disable, and edit network interface configurations. Each PIX Firewall must be configured, and each active interface must be enabled. Inactive interfaces can be disabled. When disabled, the interface does not transmit or receive data, but the configuration information is retained.

Use the Object Selector to select a group or device. In the example in the figure, the scope is set for pix6. The displayed interface settings are configured at the device level.

Configuration—Interfaces

Cisco.com



If the administrator bootstrapped a new firewall device, the setup feature configures only the addresses and names associated with the inside interface. You must define the remaining interfaces on that device before you can specify access and translation rules for traffic traversing that firewall.

In the figure in the example, for pix6, the Hardware ID column shows that ethernet0 and ethernet1 are configured and enabled. Ethernet2 through ethernet5 display their default settings and are currently disabled.

Configuring Building Blocks

This topic introduces and explains configuring building blocks.

Building Blocks

Cisco.com

CSFPA 3.2—19-37

Building blocks allow you to optimize your configuration. Objects such as hosts, protocols, or services can be grouped, allowing you to issue a single command to every item in the group by using the name of the group. The building blocks feature is used to associate names that can be used in place of corresponding data values in settings and rules. This feature facilitates maintenance.

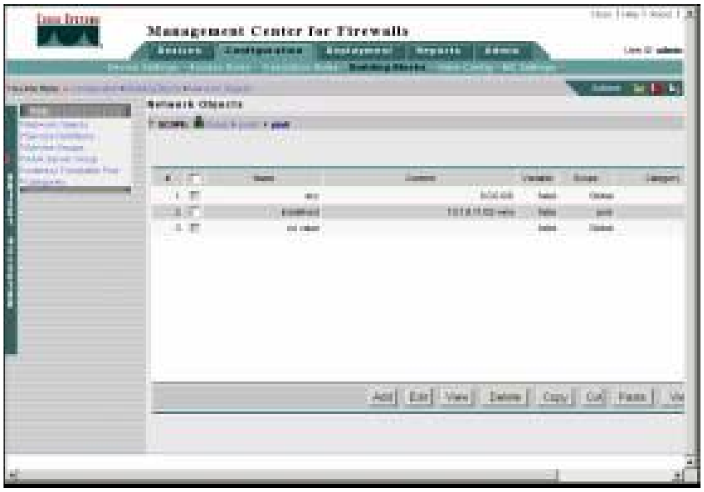
Building blocks consist of the following items:

- Network objects
- Service definitions
- Service groups
- AAA server groups
- Address translation pools

Using building blocks, you identify objects that will be used on your network. For example, you can identify the servers used for AAA authentication by group name, interface, IP address, key, and timeout. You can apply the AAA server group name elsewhere in the configuration. This design facilitates network updates because building blocks are defined only once and in one location.

Network Objects—Added

Cisco.com



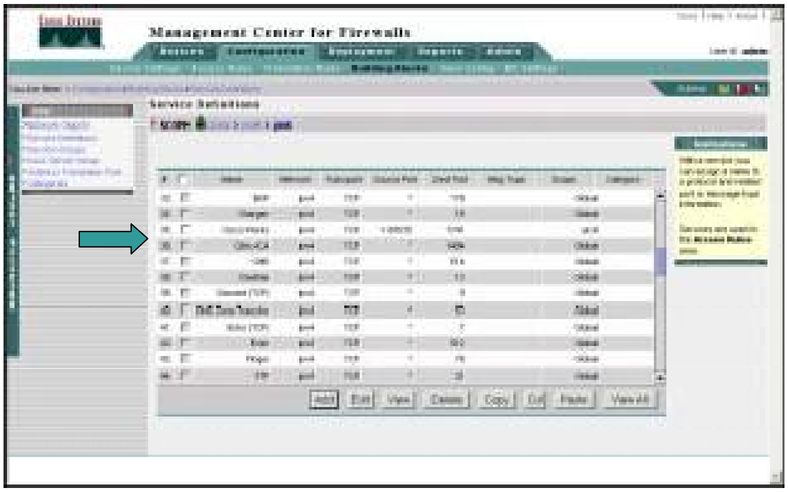
CSFPA 3.2—19-38

Network objects are associations of names with IP addresses or other network objects. This information provides the basic identification information for that network object. The Firewall MC uses the name and IP address-netmask pair to resolve references to the network object in the source and destination conditions of access rules and in translation rules.

In the example in the figure, insidehost is associated with the IP address of 10.1.6.11/32. The Firewall MC will associate the IP address of 10.1.6.11/32 with any reference to the insidehost in any access or translation rule.

Service Definition—Added

Cisco.com

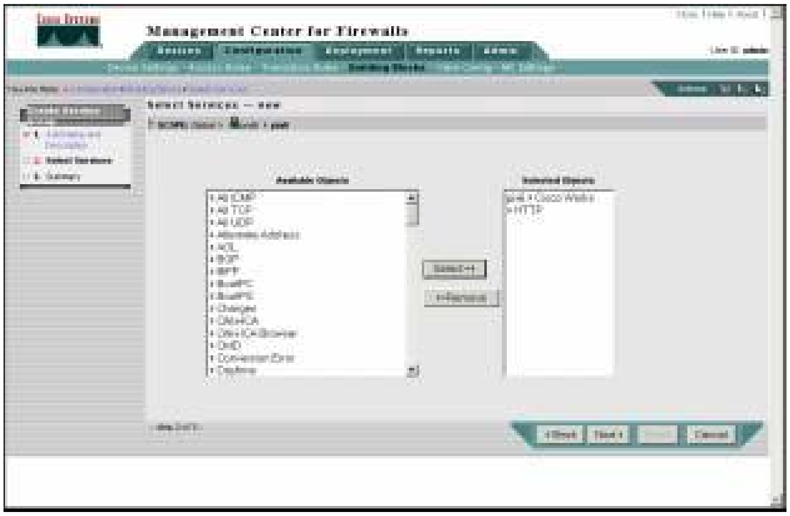


CSFPA 3.2—19-39

Firewall MC service definitions can contain IP protocols, TCP and UDP source and destination ports, and Internet Control Message Protocol (ICMP) message types. With a service definition, the administrator can assign a name to a protocol and related port and message type information. In the figure in the example, a service definition is given the name of CiscoWorks. CiscoWorks uses TCP transport protocol. The destination port is 1741. The source port range is from 1 to 65535.

Enter Service Group Objects

Cisco.com

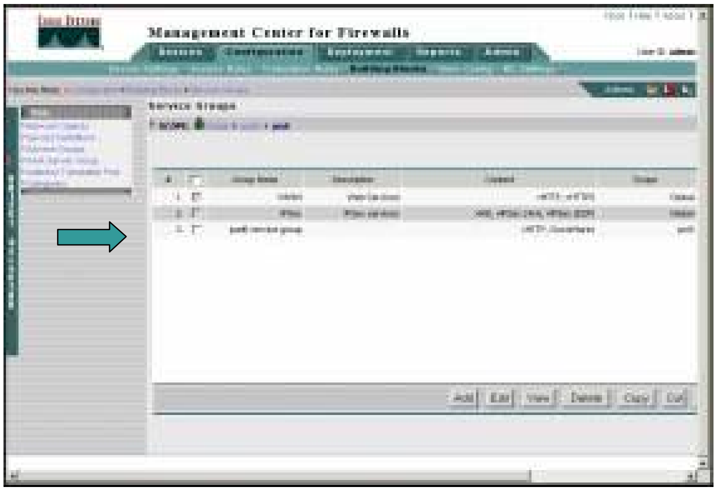


© 2004, Cisco Systems, Inc. All rights reserved.

CSFPA 3.2—19-40

Service Groups enables the administrator to specify a name for a group of services. A single service group can support multiple service objects, for example, HTTP, FTP, Telnet, and so on. In the example in the figure, the administrator added HTTP and CiscoWorks, destination TCP port 1741, to this service definition from the available services list.

Service Groups—Added



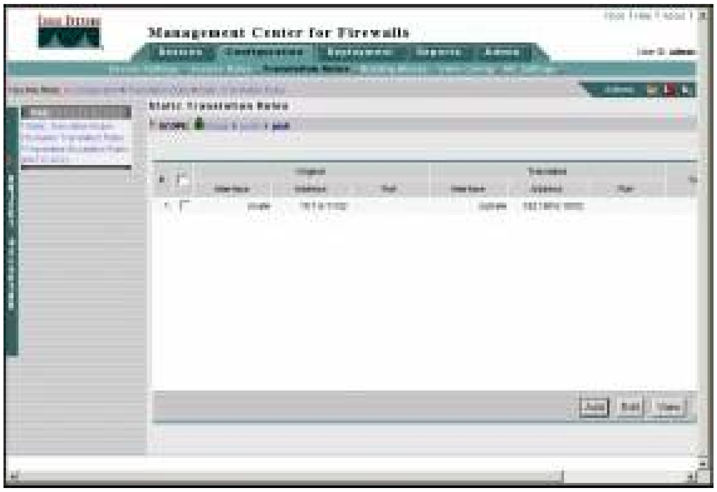
In the figure in the example, HTTP and CiscoWorks are available in the pod6 service group. The service group can replace multiple single service-item access rules.

Configuring Access and Translation Rules

This topic introduces and explains configuring access and translation rules and deploying configuration changes using the Firewall MC.

Static Translation Rules—Added

Cisco.com



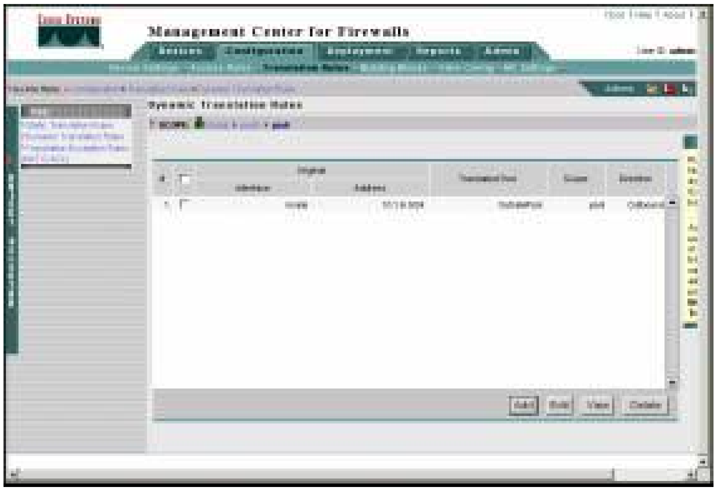
CSFPA 3.2—19-43

Translation rules allow you to view all address translation rules applied to your network. Both Port Address Translation (PAT) and Network Address Translation (NAT) are supported by the Firewall MC.

Static translation rules have internal IP addresses that are assigned permanently to a global IP address. These rules assign a host address on a higher security-level interface to a global address on a lower security-level interface. In the figure in the example, a static rule assigns the local address of a web server, 10.1.6.11/32 (on a perimeter network), to a global address, 192.168.6.10/32, that hosts (on the outside interface) use to access the Web server.

Dynamic Translation Rules—Added

Cisco.com



#	☐	Original Interface	Original Address	Translated Pool	Scope	Direction
1	☐	inside	10.1.6.0/24	OutsidePool	pix6	Outbound

CSFPA 3.2—19-44

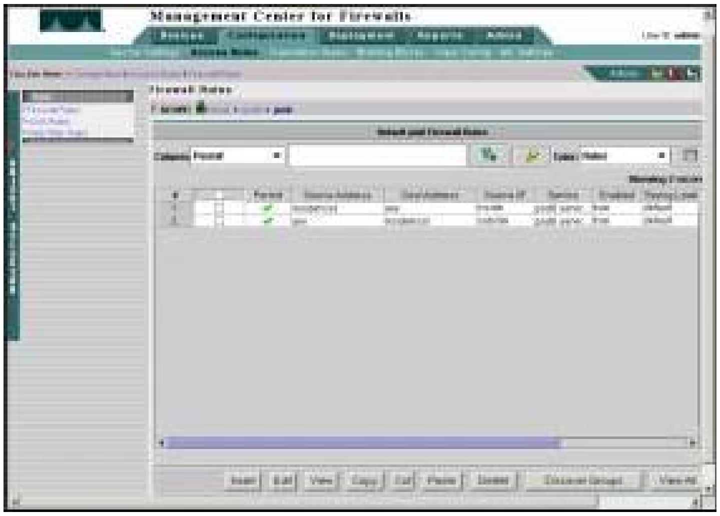
Dynamic translation rules use internal IP addresses that are dynamically translated using IP addresses from a pool of global addresses or, in the case of PAT, a single address. These rules translate host addresses on a higher security-level interface to addresses selected from a pool of addresses for traffic sent to a lower security-level interface. Dynamic translations are often used to assign local, RFC 1918 IP addresses to addresses that can be routed through the Internet.

Prior to configuring a dynamic translation rule, you will need to configure an address translation pool: select **Configure>Building Blocks>Address Translation Pool**.

In the example in the figure, any host on the inside network, 10.1.6.0/24, transiting the PIX Firewall in the outbound direction is assigned an address from the translated pool, OutsidePool.

Access Rules—Added

Cisco.com



CSFPA 3.2—19-45

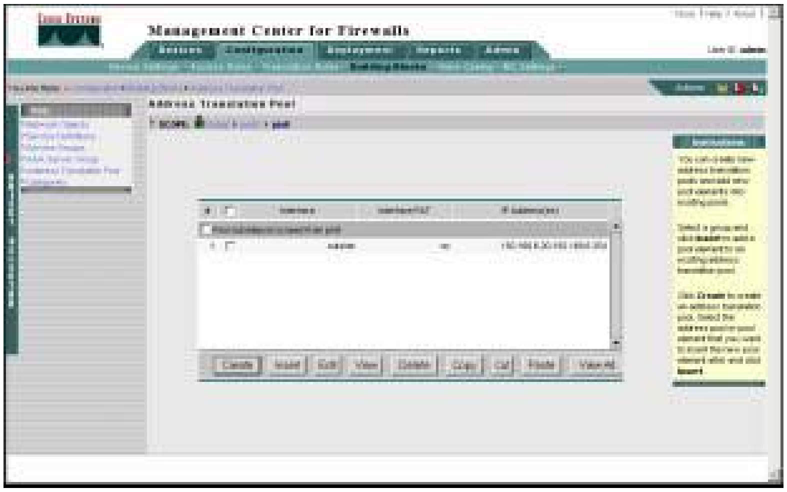
Access rules are used to define your network security policy; they control the traffic that flows through a firewall device. Access rules are recognized in the form of an ordered list, which is represented in the Firewall MC as a table. Rules are processed by a firewall device from first to last. When a rule matches the network traffic that a firewall device is processing, the firewall device uses that rule's action to decide whether traffic is permitted.

Each access rule defined in the Firewall MC will eventually correspond to a single entry in the access control list (ACL) for an interface on a particular firewall device. Access rules can be applied to the Global group, its subgroups, or individual devices. Access rules are grouped by the interface on which they are configured and enforced. The Firewall MC sorts the rules by interface and uses the remaining information in the rule to create the access control element (ACE) that will be included in the ACL for that interface.

In the example in the figure, any traffic from the inside host to any destination using a service defined by the pod6 service group, HTTP and TCP port 1741, is permitted. In the second access rule, any traffic on the outside interface destined for the inside host using the protocols defined in the pod6 service group, HTTP and TCP port 1741, is permitted.

Address Translation Pool—Added

Cisco.com

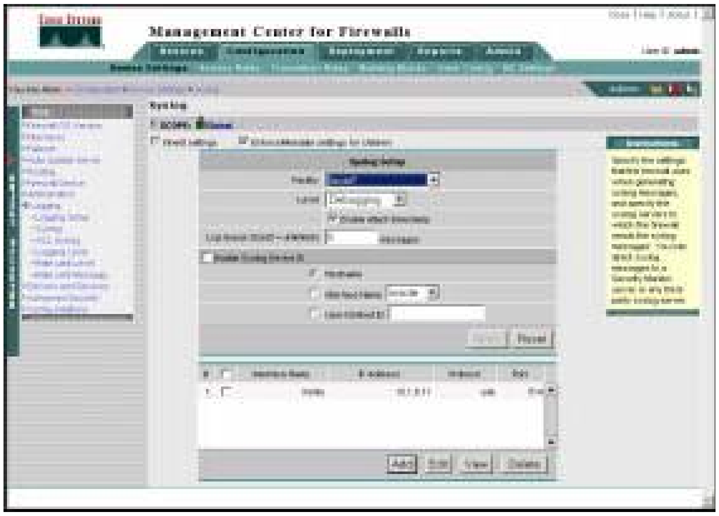


CSFPA 3.2—19-46

The address translation pool feature allows you to create global address pools used in dynamic NAT rules. In the example in the figure, on the outside interface, a pool of IP addresses, from 192.168.6.20 to 192.168.6.254, is available for dynamic address translations.

Enter Syslog Setup

Cisco.com

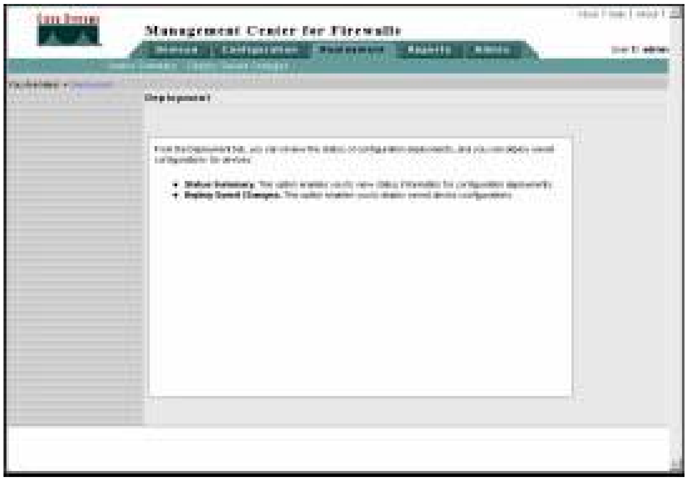


CSFPA 3.2—19-47

Syslog messages can be sent to the Security Monitor tool as well as to third-party Syslog servers. To use third-party Syslog servers, the administrator must configure the logging settings associated with each PIX Firewall device on the network. In the example in the figure, the PIX Firewall is configured to send Syslog messages out the inside interface to the Syslog host at IP address 10.1.6.11.

Deployment Tab

Cisco.com



Management Center for Firewalls
Devices
Configuration
Deployment
Reports
Admin
Status Summary | Deploy Saved Changes

Deployment

From the Deployment tab, you can review the status of configuration deployments, and you can deploy saved configurations to devices.

- **Status Summary.** This option enables you to view status information for configuration deployments.
- **Deploy Saved Changes.** This option enables you to deploy saved device configurations.

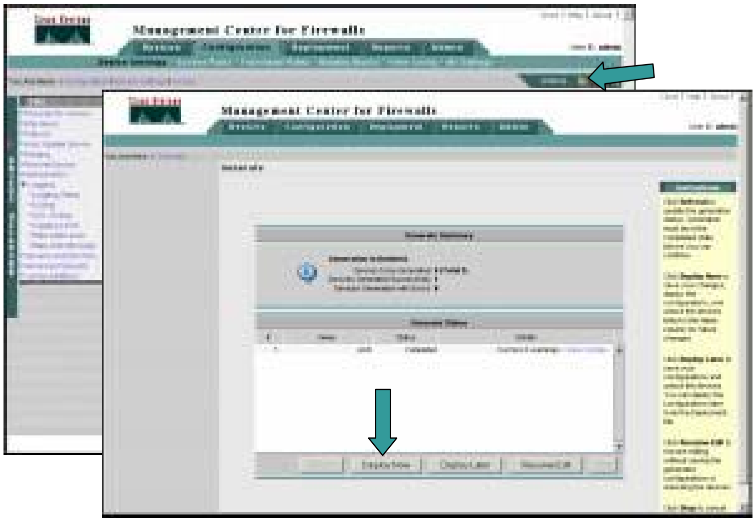
CSFPA 3.2—19-48

From the Deployment tab, the administrator can review the status of configuration deployments and deploy saved configurations. The Deployment tab contains the following options:

- Status Summary—View status information for configuration deployments.
- Deploy Saved Changes—Deploy saved device configurations.

Generate Summary and Deploy Now

Cisco.com


Management Center for Firewalls
Management Center for Firewalls
Deploy Now
Deploy Later

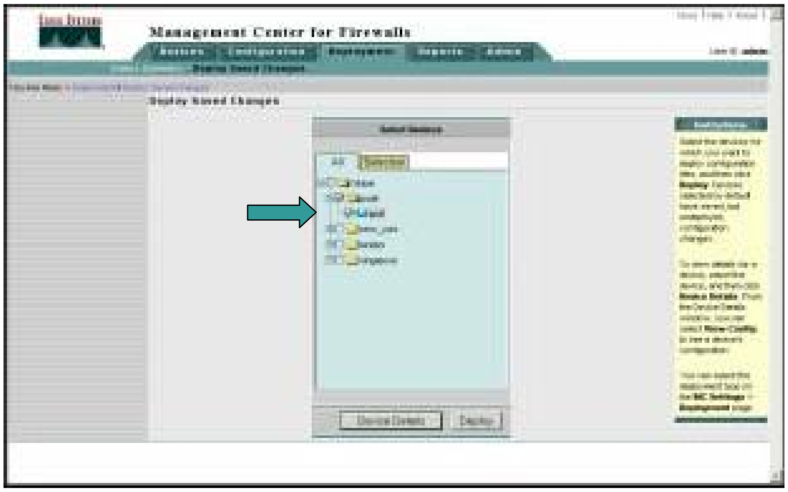
CSFPA 3.2—19-49

When workflow is disabled (default), the administrator does not need to define activities for tracking changes in the Firewall MC; instead, the Firewall MC defines activities automatically when a device’s configuration is modified. You also do not need to define jobs to deploy configurations. After configuration changes are made, deploy those changes by clicking the **Save and Deploy** icon in the actions bar of the Devices and Configuration tabs, or you can discard the changes by clicking the **Undo** icon. When you click the **Save and Deploy** icon, new configurations are generated for any modified devices. You can then deploy the new configurations by clicking **Deploy Now**, or you can save them and deploy them later by clicking **Deploy Later**.

In the example in the figure, the administrator has finished configuring the PIX Firewall and wants to deploy the changes. Clicking the **Save and Deploy** icon in the actions bar of the Configuration tab (rear GUI screen) opens the Generate Summary window and generates the configuration. Clicking **Deploy Now** (front GUI screen) deploys the saved configuration.

Deploy Later

Cisco.com



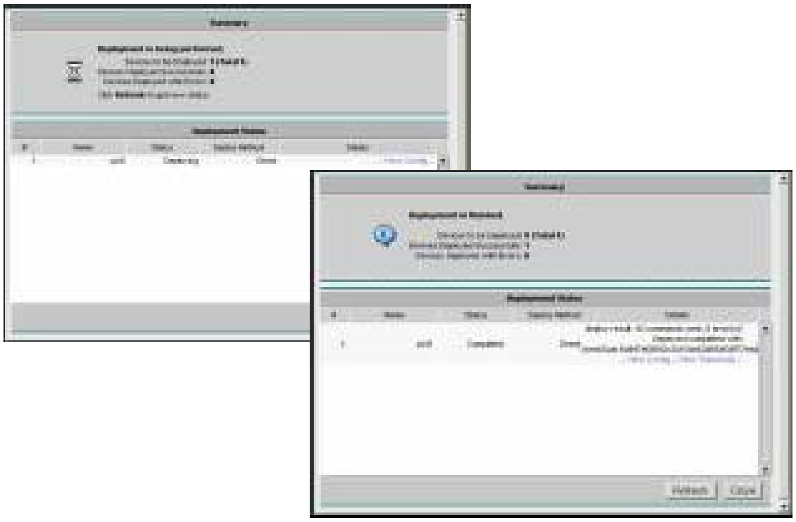
CSFPA 3.2—19-50

Instead of deploying a configuration right away, the administrator may choose to deploy the configuration changes at a later time, for instance, Sunday night at 12 p.m.. To deploy saved configurations, go to **Deployment>Deploy Saved**. The Deploy Saved Changes window opens. From the Deploy Saved Changes window, select the devices for which you want to deploy the configuration files, and then click **Deploy**. Devices selected by default have saved, but not deployed, configuration changes. To view configuration details for a device, select the device and then click the **Device Details** button. From the Device Details window, you can select the **View Config** link to see a device's configuration.

In the example in the figure, pix6 is selected. To view the configuration, click the **Device Details** button and then click **View Config** link. The pix6 Config window opens. To deploy the configuration, click **Deploy**. The pix6 Deployment Summary window opens.

Deployment Summary

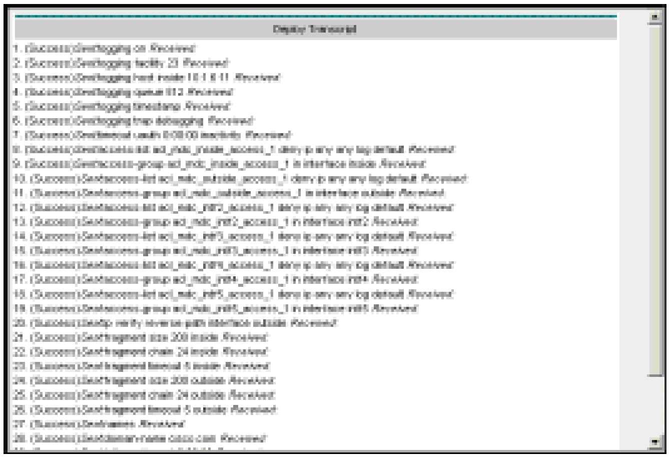
Cisco.com



CSFPA 3.2—19-51

When you click the **Deploy** button, the Deployment Status Summary popup window opens. To update the status information in the popup window, click the **Refresh** button. From the Deployment Status popup window, you can click the **View Config** link to view a device’s configuration, or, if the configuration is deployed directly to a device, you can select the **View Transcript** link to see the deployment transcript for a device.

Deployment Transcript



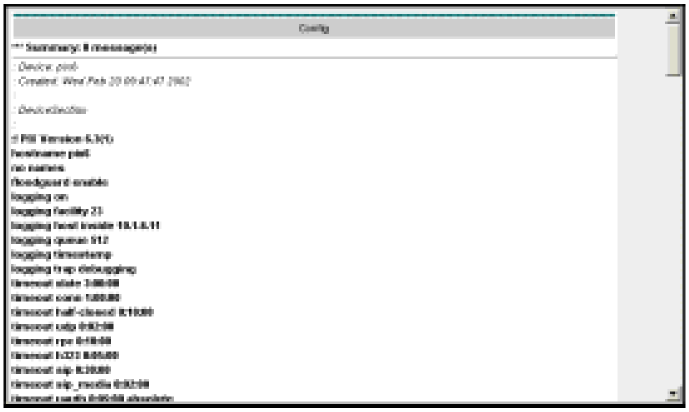
Deploy Transcript

```
1. (Success)Sent:logging on Received
2. (Success)Sent:logging facility 23 Received
3. (Success)Sent:logging host inside 10.1.6.11 Received
4. (Success)Sent:logging queue 512 Received
5. (Success)Sent:logging timestamp Received
6. (Success)Sent:logging trap debugging Received
7. (Success)Sent:timeout uauth 0:05:00 inactivity Received
8. (Success)Sent:access-list acl_mdc_inside_access_1 deny ip any any log default Received
9. (Success)Sent:access-group acl_mdc_inside_access_1 in interface inside Received
10. (Success)Sent:access-list acl_mdc_outside_access_1 deny ip any any log default Received
11. (Success)Sent:access-group acl_mdc_outside_access_1 in interface outside Received
12. (Success)Sent:access-list acl_mdc_intf2_access_1 deny ip any any log default Received
13. (Success)Sent:access-group acl_mdc_intf2_access_1 in interface intf2 Received
14. (Success)Sent:access-list acl_mdc_intf3_access_1 deny ip any any log default Received
15. (Success)Sent:access-group acl_mdc_intf3_access_1 in interface intf3 Received
16. (Success)Sent:access-list acl_mdc_intf4_access_1 deny ip any any log default Received
17. (Success)Sent:access-group acl_mdc_intf4_access_1 in interface intf4 Received
18. (Success)Sent:access-list acl_mdc_intf5_access_1 deny ip any any log default Received
19. (Success)Sent:access-group acl_mdc_intf5_access_1 in interface intf5 Received
20. (Success)Sent:ip verify reverse-path interface outside Received
21. (Success)Sent:fragment size 200 inside Received
22. (Success)Sent:fragment chain 24 inside Received
23. (Success)Sent:fragment timeout 5 inside Received
24. (Success)Sent:fragment size 200 outside Received
25. (Success)Sent:fragment chain 24 outside Received
26. (Success)Sent:fragment timeout 5 outside Received
27. (Success)Sent:names Received
28. (Success)Sent:domain-name cisco.com Received
```

CSFPA 3.2—19-52

In the figure is an example of a deployment transcript. Scroll down to view a record of the commands sent to the device.

Deployment Config



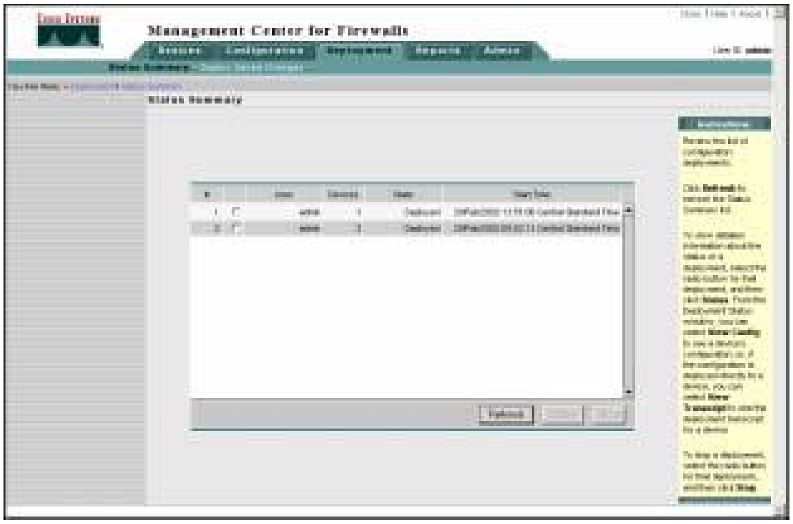
```
*** Summary: 0 message(s)
: Device: pix6
: Created: Wed Feb 20 09:47:47 2002
:
: DeviceSection
:
: PIX Version 6.3(1)
hostname pix6
no names
floodguard enable
logging on
logging facility 23
logging host inside 10.1.0.11
logging queue 512
logging timestamp
logging trap debugging
timeout xlate 3:00:00
timeout conn 1:00:00
timeout half-closed 0:10:00
timeout udp 0:02:00
timeout rpc 0:10:00
timeout h323 0:05:00
timeout sip 0:30:00
timeout sip_media 0:02:00
timeout uauth 0:05:00 absolute
```

CSFPA 3.2—19-53

In the figure is an example of the saved configuration file. Scroll down to view the device’s configuration file saved on the Firewall MC.

Deployed

Cisco.com



CSFPA 3.2—19-54

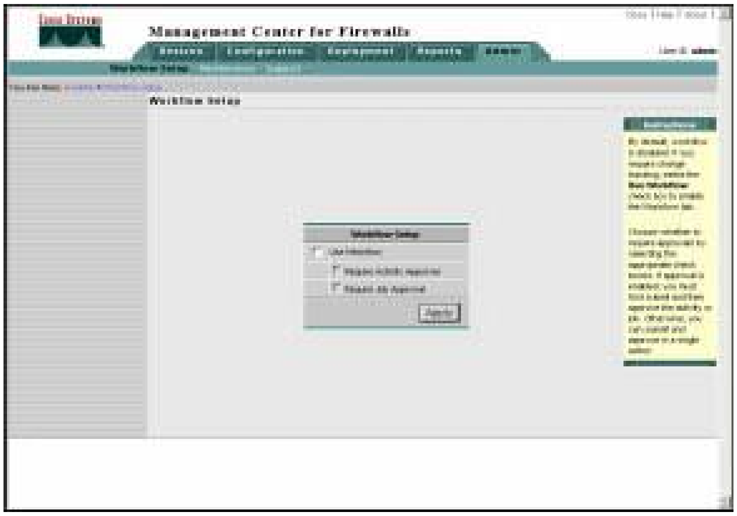
The Status Summary window provides a list of configuration deployments. Click the **Refresh** button to refresh the status summary list. To view information concerning a specific deployment, select the radio button for that deployment, and then click **Status**. From the Deployment Status window, click the **View Config** link to see a device's configuration. Click the **View Transcript** link to see the deployment transcript for the selected device.

Managing Workflow

By default, workflow is disabled. This topic explains how to enable workflow. It describes how to define and approve activities and how to define, approve, and deploy jobs.

Workflow Setup

Cisco.com



© 2004, Cisco Systems, Inc. All rights reserved. CSFPA 3.2—19-56

If a company is using the workflow setup option (Admin>Workflow Setup), the administrator can specify whether to use jobs and activities to track changes made in the Firewall MC and whether those jobs or activities should be approved formally before the next phase of the workflow is performed. Before selecting workflow options, the administrator should understand how the various options affect the operation of the Firewall MC.

Many organizations benefit from having a separation of responsibility for defining, implementing, and deploying corporate firewall policies. For example, a security administrator might be responsible for defining a device configuration file, another administrator for approving the configuration file, and a network operator for deploying the resulting configuration to a device. This separation of responsibility helps maintain the integrity of deployed device configurations.

The Firewall MC supports this separation of responsibility by using *activities* and *jobs,* also referred to as *workflo*w. Activities control policy changes, and jobs define deployment of configuration files. You can set up the Firewall MC to require formal approval for activities, jobs, or both. Workflow and the formal approval process are disabled by default, but you can enable either by selecting **Admin>Workflow Setup**. When workflow is enabled, you use activities to track changes to configurations and use jobs to track the deployment of these configurations to the firewall devices that you manage.

Create Activity

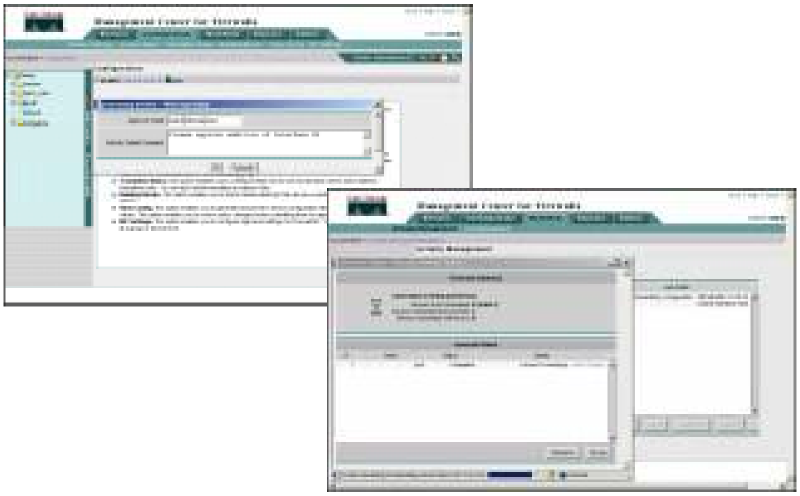
© 2004, Cisco Systems, Inc. All rights reserved.

CSFPA 3.2—19-57

An activity is a collection of policy changes typically made for a single purpose. For example, an activity might be created to collect the changes needed to address a problem identified in a trouble ticket filed with your help desk. Alternatively, an activity might be created to implement a new policy. Before you can begin importing devices and adding configuration settings and rules, you must define an activity. To access activity management, select **Workflow>Activity Managemen**t. Click the **Add** button from the Activity Management window. The Creating New Activity popup window opens. By default, the Firewall MC applies a name to the activity, user login level, and current date and time. You may choose to name the activity something else, such as “add interface E2.” Click the **OK** button in the Creating New Activity popup window to add the activity is added and make it available.

In the example in the figure, the activity Add interface E2 is added. An administrator can now make the requisite changes to the configuration. Once the configuration changes are complete, the activity must be approved and a new configuration generated.

Submit Activity and Generate Configuration



CSFPA 3.2—19-58

After an activity is created, it can be submitted for approval. If the activity is approved, a configuration is generated, ready to be deployed through the use of a job. If the activity is rejected, the submitter has the option of modifying the activity and again submitting it for approval or deleting the activity from the system and beginning again.

Job Requested and Approved

Cisco.com

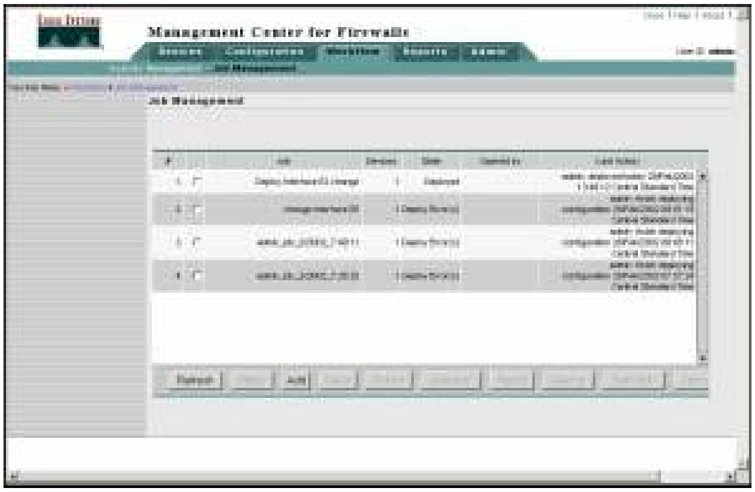
CSFPA 3.2—19-59

The Job Management interface is similar to the Activity Management interface. After an activity has been submitted and approved (optional), it is ready to be defined in a job. A job identifies a set of devices and configuration files for deployment and defines a method for deployment.

Select **Workflow>Job Management.** The Job Management window opens. To create a job, click the **Add** button**,** and the wizard takes you through the job definition process, activities deployed, and devices selected. Once created, the job must be approved. To approve a job, select the radio button next to the job and click the **Approve** button.

Job Deployed

Cisco.com



CSFPA 3.2—19-60

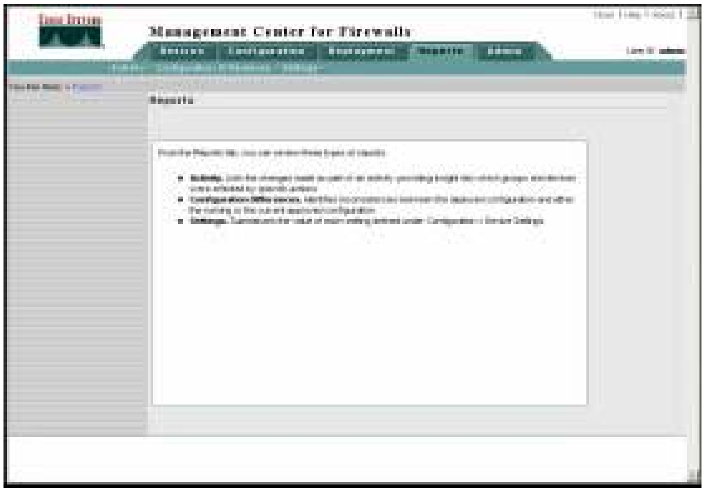
After a job has been defined and approved, it is ready for deployment. Select the radio button next to the job and click the **Deploy** button. The Firewall MC will deploy the configuration.

Reporting

This topic introduces and explains the report tab of the Firewall MC.

Reports

Cisco.com

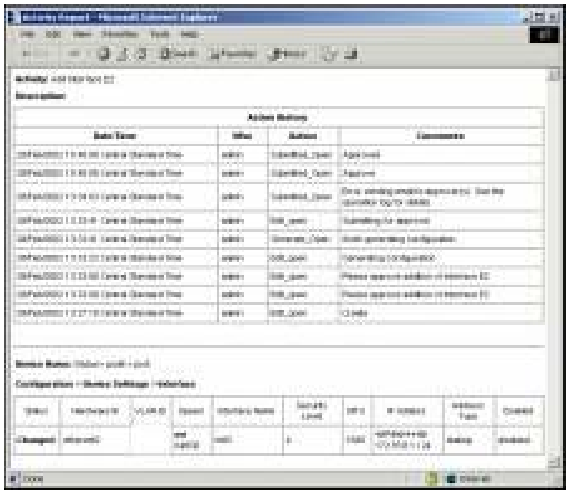


CSFPA 3.2—19-62

From the Reports tab, you can view three types of reports:

- Activity—Lists the changes made as part of an activity, providing insight into which groups and devices were affected by specific activities
- Configuration Differences—Identifies inconsistencies between the deployed configuration and the running or the current approved configuration
- Settings—Summarizes the value of each setting defined under Configuration>Device Setting

Activity Report



CSFPA 3.2—19-63

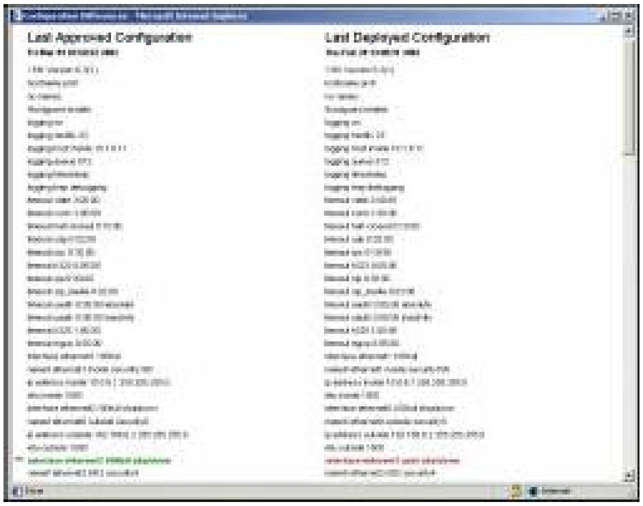
The Report feature allows you to display information about the activities in the Firewall MC. The Activity report provides two types of information about an activity. The first type shows which user has performed which action of an activity, at the time the action occurred. For example, as shown in the figure, one user created an activity. It was submitted and approved on the same day. The second type of information shows policy changes that were made as part of the activity. For example, an interface was added when an activity was created.

The Activity report is useful when you want to review:

- Changes that were made to an activity before the activity was submitted for approval
- Changes that were made before the activity was submitted for job deployment

Configuration Differences Report

Cisco.com



Last Approved Configuration

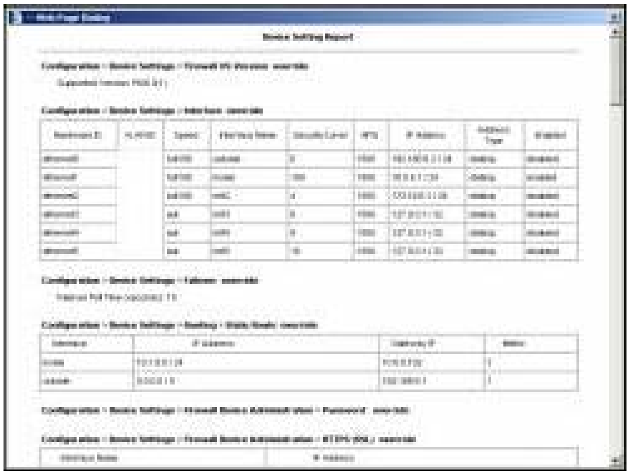
Last Deployed Configuration

CSFPA 3.2—19-64

The Firewall MC provides a report that identifies whether a device’s running configuration matches the latest configuration deployed to that device. It also identifies whether a device is using a configuration that can be replaced with a more recent approved configuration.

In the example in the figure, the last approved configuration is different from the last deployed configuration. The last approved configuration has a different configuration for interface E2, as denoted by the asterisks.

Settings Report



Device Setting Report

Configuration > Device Settings > Firewall OS Version: override

Supported Version: PIX 6.3(1)

Configuration > Device Settings > Interface: override

Hardware ID	VLAN ID	Speed	Interface Name	Security Level	MTU	IP Address	Address Type	Enabled
ethernet0		full100	outside	0	1500	[illegible]	static	[illegible]
ethernet1		full100	inside	100	1500	[illegible]	static	[illegible]
ethernet2		full100	dmz	4	1500	[illegible]	static	[illegible]
ethernet3		auto	intf3	0	1500	127.0.0.1/32	static	[illegible]
ethernet4		auto	intf4	0	1500	127.0.0.1/32	static	[illegible]
ethernet5		auto	intf5	10	1500	127.0.0.1/32	static	[illegible]

Configuration > Device Settings > Failover: override

Failover Poll Time (seconds): 15

Configuration > Device Settings > Routing > Static Route: override

Interface	IP Address	Gateway IP	Metric
inside	[illegible]	[illegible]	1
outside	0.0.0.0/0	[illegible]	1

Configuration > Device Settings > Firewall Device Administration > Password: override

Configuration > Device Settings > Firewall Device Administration > HTTPS (SSL): override

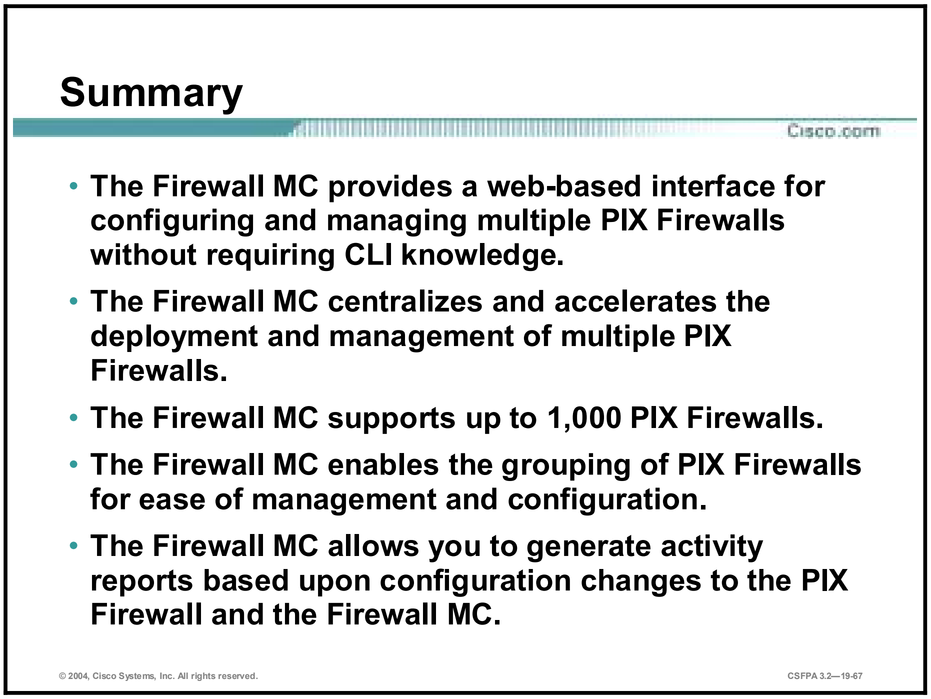
Interface Name	IP Address

CSFPA 3.2—19-65

The Firewall MC allows you to view a report that identifies device settings for a device or device group. This report also identifies whether a particular setting is inherited, mandatory, or overridden.

Summary

This topic summarizes the information you learned in this lesson.



Summary

Cisco.com

- The Firewall MC provides a web-based interface for configuring and managing multiple PIX Firewalls without requiring CLI knowledge.
- The Firewall MC centralizes and accelerates the deployment and management of multiple PIX Firewalls.
- The Firewall MC supports up to 1,000 PIX Firewalls.
- The Firewall MC enables the grouping of PIX Firewalls for ease of management and configuration.
- The Firewall MC allows you to generate activity reports based upon configuration changes to the PIX Firewall and the Firewall MC.

CSFPA 3.2—19-67

Lab Exercise—Introduction to Enterprise PIX Firewall Management

Complete the following lab exercise to practice what you learned in this lesson.

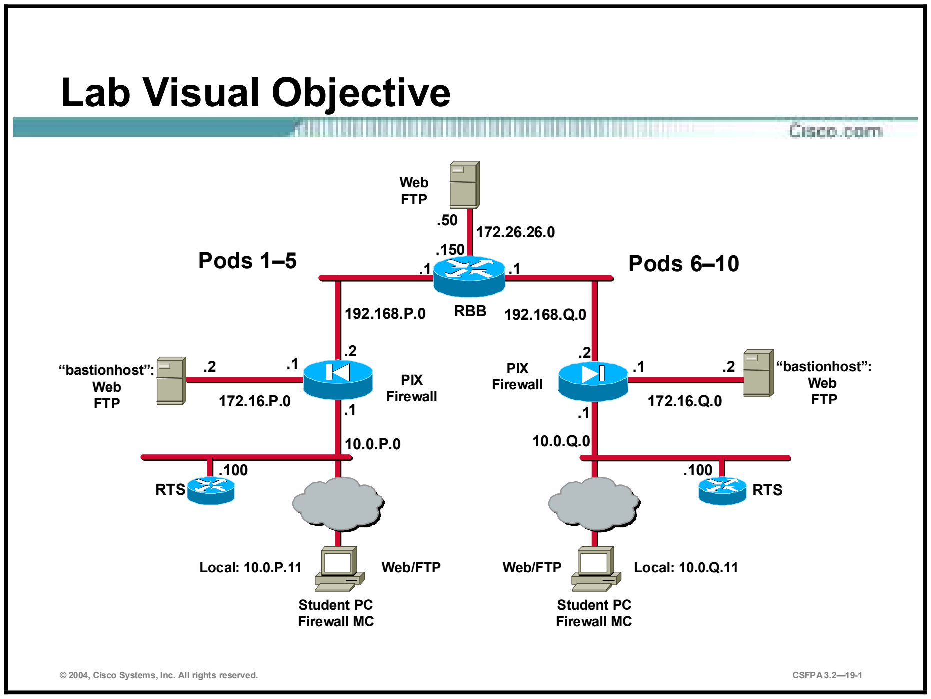
Objectives

In this lab exercise you will complete the following tasks:

- Install the Firewall MC.
- Bootstrap the PIX Firewalls.
- Launch the Firewall MC.
- Create a group.
- Import PIX Firewalls.
- Configure the inside and outside interfaces.
- Configure service definitions, service groups, and address translation pool building blocks.
- Create translation rules.
- Configure the PIX Firewall to allow HTTP and CiscoWorks traffic to the inside host.
- Configure a global security policy.
- Deploy the configuration.
- Test the PIX Firewall configuration.

Visual Objectives

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

© 2004, Cisco Systems, Inc. All rights reserved.

CSFPA 3.2—19-1

Setup

Before starting this lab exercise, ensure that the CiscoWorks2000 VPN/Security Management Solution (VMS) Common Services has been installed on the student PC. Notify your instructor if the VMS Common Services is not installed.

Task 1—Install the Firewall MC

This task involves the student installing the Firewall MC onto the student PC. Complete the following steps to install the Firewall MC:

Step 1 Log in to your student PC as the local administrator.

Step 2 Locate and run the Firewall MC installation files as directed by your instructor.

Step 3 Once the installation process starts, the Welcome Window is displayed.

Step 4 Click **Next**. The Software License Agreement window is displayed.

Step 5 Click **Yes** to agree to the Software License Agreement. The System Requirements window is displayed.

Step 6 Click **Next**. The Ports Configuration window opens.

Step 7 Click **Next**. The Summary window is displayed.

Step 8 Click **Next** to start the installation of the Firewall MC. Once setup has been completed, the Setup Complete window is displayed. Setup will take several minutes to complete.

Step 9 Click **Finish** to complete the installation.

Task 2—Bootstrap the PIX Firewalls

A PIX Firewall must allow SSH connections before a Firewall MC can use it. Complete the following steps to bootstrap the PIX Firewalls:

Step 1 Access the PIX Firewall console as directed by your instructor.

Step 2 Erase your current PIX Firewall configuration. When prompted to confirm, press **Enter**.

```
pixP(config)# write erase
Erase PIX configuration in flash memory? [confirm]
```

Step 3 Reload the PIX Firewall. When prompted to confirm, press **Enter**.

```
pixP(config)# reload
Proceed with reload? [confirm]
```

Step 4 When prompted to pre-configure the PIX Firewall through interactive prompts, press **Enter**.

Step 5 Enter **cisco** as the enable password:

```
Enable password [<use current password>]: cisco
```

Step 6 Accept the default year by pressing **Enter**:

```
Clock (UTC):
  Year [2003]: <Enter>
```

Step 7 Accept the default month by pressing **Enter**:

```
Month [Nov]: <Enter>
```

Step 8 Accept the default day by pressing **Enter**:

```
Day [14]: <Enter>
```

Step 9 Accept the default time stored in the host computer by pressing **Enter**:

```
Time [11:21:25]: <Enter>
```

Step 10 Enter the IP address of your PIX Firewall’s inside interface:

```
Inside IP address: 10.0.P.1
```

(where P = pod number)

Step 11 Enter the network mask that applies to inside IP address:

```
Inside network mask: 255.255.255.0
```

Step 12 Enter the hostname you want to display in the PIX Firewall command line prompt:

```
Host name: pixP
```

(where P = pod number)

Step 13 Enter the DNS domain name of the network on which the PIX Firewall runs:

```
Domain name: cisco.com
```

Step 14 Enter the IP address of the host running PDM:

Note: Even though you are configuring the PIX Firewall to use PIX Device Manager with the setup dialog, you will be using the Firewall MC. This is because the PIX Firewall needs to enable the HTTP server and HTTP access from the inside interface, the same commands that the Firewall MC and PIX Device Manager require to control the PIX Firewall.

```
IP address of host running PIX Device Manager: 10.0.P.11
```

(where P = pod number)

Step 15 Enter **y** at the prompt to save the information to the PIX Firewall's Flash memory.

Step 16 Create a default route:

```
pixP(config)# route outside 0 0 192.168.P.1
```

(where P = pod number)

Task 3—Launch the Firewall MC

Complete the following tasks to launch the Firewall MC:

Step 1 Access the CiscoWorks Server from your web browser by entering the following in the URL field:

```
http://127.0.0.1:1741
```

Step 2 Click **Yes** in the Security Warning window.

Note: Be patient when connecting to the CiscoWorks2000 server. The server requires additional software to be installed. The software is installed during the initial access. Subsequent attempts should not require you to install additional software. Select Grant Always when you see the Java Plug-in security window.

Step 3 Log in by entering the default username and password of **admin** and **admin**, respectively.

Step 4 Click **Connect**. You are now logged in to the CiscoWorks Desktop.

Step 5 Select the **VPN/Security Management Solution** drawer located in the left panel.

Step 6 Select the **Management Center** folder located in the VPN/Security Management Solution drawer.

Step 7 Select **Firewalls** from the Management Center folder. The Security Alert window opens prompting you to accept a digital certificate.

Step 8 Click **Yes**. You are now logged in to the Firewall MC.

Step 9 Minimize the CiscoWorks2000 browser window before proceeding to the next task.

Task 4—Create a Group

Complete the following steps to create a group:

Step 1 Select **Devices>Managing Groups**. The Managing Groups page is displayed.

Step 2 Select the **Global** group and click **Add**. The Define Group Information (new) page is displayed.

Step 3 Enter **podP** in the Group Name field. Enter a description in the Group Description field.

(where P = pod number)

Step 4 Click **Next**. The Summary page is displayed.

Step 5 Verify that information displayed in the Summary page is correct and click **Finish**. The Managing Groups page is displayed with the newly created group, podP.

Task 5—Import PIX Firewalls

Complete the following steps to open an activity and import PIX Firewalls:

Step 1 Select **Devices>Importing Devices**. The Importing Devices page is displayed.

Step 2 Click **Import**. The Select Target Group (new) page is displayed.

Step 3 Select the **podP** group and click **Next**. The Select Import Type (new) page is displayed.

(where P = pod number)

Step 4 Select the **Import configuration from device** radio button and click **Next**. The Define PIX Firewall Contact Info (new) page is displayed.

Step 5 Enter **10.0.P.1** in the Contact IP address field. Enter **cisco** in the Enable Password field.

(where P = pod number)

Step 6 Click **Next**. The Summary page is displayed.

Step 7 Verify that the information displayed is correct and click **Finish**. The Import Summary page is displayed. Wait until the status reads "Import is finished" before proceeding. You may have to click **Refresh** to refresh the Import status screen.

Step 8 From the Import Status window, click **View Config**. View the current PIX configuration. Close the Config window.

Step 9 Click **Close** to close the Import Summary window.

Task 6—Configure the Inside and Outside Interfaces

Complete the following steps to enable the inside and outside interfaces for the PIX Firewalls:

Step 1 Complete the following substeps to configure pixP's inside interface:

1. Select **Configuration>Device Settings**.
2. Notice the scope is set to Global. Use the Object Selector to select **pixP**. The Object Bar refreshes to indicate that pixP is being configured. The scope resets to Global>pod6>pix6. (where P = pod number)
3. Select **Interfaces** from the TOC. The Interfaces page is displayed.
4. Select **ethernet1** using the check box located next to the interface and click **Edit**. The Add Interface Name page is displayed.

5. Verify that the **Enable** check box for the interface is selected.

6. Select **100full** from the Speed drop-down menu.

7. Select **Static** for the IP Address Type button.

8. Click **Next**. The Add Static Address Information page is displayed.

9. Verify that **10.0.P.1** appears in the IP Address field.
(where P = pod number)

10. Verify that **24** appears in the Mask field.

11. Click **Next**. The Summary page is displayed. Verify that the settings displayed on the Summary page are correct.

12. Click **Finish**. The Interfaces page is displayed with the changes to the interface, ethernet1.

Step 2 Complete the following substeps to configure the pixP firewall's outside interface:

1. Select **Configure>Device Settings**. The Device Settings page is displayed.

2. Select **Interfaces** from the TOC. The Interfaces page is displayed.

3. Select **ethernet0** using the check box located next to the interface and click **Edit**. The Add Interface Name page is displayed.

4. Select the **Enable** check box for the interface.

5. Select **100full** from the Speed drop-down menu.

6. Select **Static** for the IP Address Type button.

7. Click **Next**. The Add Static Address Information page is displayed.

8. Enter **192.168.P.2** in the IP Address field. Enter **24** in the Mask field.
(where P = pod number)

9. Click **Next**. The Summary page is displayed. Verify that the settings displayed on the Summary page are correct.

10. Click **Finish**. The Interfaces page is displayed with the changes to the interface, ethernet0.

Task 7—Configure Service Definitions, Service Groups, and Address Translation Pool Building Blocks

Complete the following steps to configure the building blocks that will be used later to enable traffic to the student PC:

Step 1 Select **Configure>Building Blocks**. The Building Blocks page is displayed.

Step 2 Use the Object Selector to select **podP**. The Object Bar refreshes to indicate that podP is being configured.

(where P = pod number)

Step 3 Complete the following substeps to add your student PC as a Network Object:

1. Select **Network Objects** from the TOC. The Network Objects page is displayed.

2. Click **Add**. The Enter Definition (new) page is displayed.

3. Enter **insidehost** in the Network Entity Name. Enter **podP inside host** in the Description field.

4. Click **Next**. The Enter IP(s) (new) page is displayed.

5. Enter the IP address of your student PC and its netmask in the Network IP Address/Mask field: **10.0.P.11/32**.
(where P = pod number)

6. Click **Next**. The Select Networks (new) page is displayed.

7. Highlight **any** in the Available Objects column and click **Select=>** to move any to the Selected Objects column.

8. Click **Next**. The Summary page is displayed. Verify that the settings listed are correct.

9. Click **Finish**. The Network Objects page is displayed with the newly created network object.

Step 4 Complete the following substeps to configure a service definition that defines a service with a range of ports:

1. Select **Service Definitions** from the TOC. The Service Definitions page is displayed.

2. Click **Add**. The Specify Name and Select Transport (new) page is displayed.

3. Enter **CiscoWorks** in the Service Name field.

4. Select **TCP** from the Transport Protocol drop-down menu.

5. Click **Next**. The Select TCP/UDP Values (new) page is displayed.

6. Enter **1741** in the Destination Port Range field.

7. Enter **1–65535** in the Source Port Range field.

8. Click **Next**. The Summary page is displayed. Verify that the settings listed are correct.

9. Click **Finish**. The Service Definitions page is displayed with the newly created service definition. To locate the new service definition, use the scroll bar to find the definition among the list.

Step 5 Complete the following substeps to configure a service group that defines traffic to be referenced in ACLs:

1. Select **Service Groups** from the TOC. The Service Groups page is displayed.

2. Click **Add**. The Add Name and Description (new) page is displayed.

3. Enter **podP service group** in the Service Group Name field.

4. Click **Next**. The Select Services (new) page is displayed.

5. Scroll to the bottom of the list of Available Services and highlight PIXP>**CiscoWorks**.

6. Click the **Select=>** button to move CiscoWorks to the list of Selected Services.

7. Scroll through the list of Available Services and highlight **HTTP**.

8. Click the **Select=>** button to move HTTP to the list of Selected Services.

9. Click **Next**. The Summary page is displayed.

10. Verify that the information listed is correct and click **Finish**. The Service Group's page is displayed with the newly created Service Group.

Step 6 Complete the following substeps to create an IP address pool for NAT:

1. Use the Object Selector to select **pixP**. The Object Bar refreshes to indicate that pixP is being configured.
   (where P = pod number)

2. Select **Address Translation Pool** from the TOC. The Address Translation Pool window opens.

3. Click **Create**. The Enter Pool Name (new) page is displayed.

4. Enter **OutsidePool** in the Pool Name field.

5. Click **Next**. The Enter Pool Elements (new) page is displayed.

6. Select **outside** from the Interface drop-down menu.

7. Enter **192.168.P.20-192.168.P.254** in the Address Ranges/Mask (optional) field.
   (where P= pod number)

8. Click **Next**. The Summary page is displayed.

9. Verify that the information list is correct and click **Finish**. The Address Translation Pool page is displayed with the new address translation pool.

Task 8—Create Translation Rules

Complete the following steps to create the following translation rules:

- A static translation for the inside host
- A dynamic translation rule for the inside network

Step 1 Select **Configuration>Translation Rules**. The Translation Rules page is displayed.

Step 2 Select **Static Translatio**n **Rules** from the TOC. The Static Translation Rules page is displayed.

Step 3 Click **Add**. The Enter Static Translation Rule (new) page is displayed.

Step 4 Select **inside** from the Original Interface drop-down menu.

Step 5 Enter the address of the inside host in the Original Address field: **10.0.P.11**.

(where P = pod number)

Step 6 Select **outside** from the Translated Interface drop-down menu.

Step 7 Enter **192.168.P.10** in the Translated Address field.

(where P = pod number)

Step 8 Click **Next**. The Summary page is displayed.

Step 9 Verify that the information displayed is correct and click **Finish**. The Static Translation Rules page is displayed with the new translated address.

Step 10 Select **Dynamic Translation Rules** from the TOC. The Dynamic Translation Rules page is displayed.

Step 11 Click **Add**. The Enter Dynamic Translation Rule (new) page is displayed.

Step 12 Select **inside** from the Original Interface drop-down menu.

Step 13 Enter the address of the network to be translated in the Original Address field: **10.0.P.0/24**.

(where P = pod number)

Step 14 Verify that **OutsidePool** appears in the Address Pool drop-down menu.

Step 15 Click **Next**. The Summary page is displayed.

Step 16 Verify that the information displayed is correct and click **Finish**. The Dynamic Translation Rules page is displayed with the new address translation.

Task 9—Configure the PIX Firewall to Allow HTTP and CiscoWorks Traffic to the Inside Host

Complete the following steps to configure the PIX Firewall to allow HTTP and CiscoWorks traffic to your student PC:

Step 1 Select **Configuration>Access Rules**. The Access Rules page is displayed.

Step 2 Verify **Default pixP Firewall Rules** window is open. The Access Rules page indicates that you are configuring access rules at the pixP level.

(where P = pod number)

Step 3 Click **Insert**. The Enter Rule Data (new) page is displayed.

Step 4 Select the **Permit** radio button.

Step 5 Click the **Select** button below the Source Addresses field. The Selecting Network Objects page is displayed.

Step 6 Select **Any** from the list of Available Objects and click **Select=>** to move it to the list of Selected Objects.

Step 7 Click **OK**. The Enter Rule Data (new) page is refreshed with the Source Addresses field containing a new address.

Step 8 Click the **Select** button below the Destination Addresses field. The Selecting Network Objects page is displayed.

Step 9 Select **insidehost** from the list of Available Objects and click **Select=>** to move it to the list of Selected Objects.

(where P=pod number)

Step 10 Click **OK.** The Enter Rule Data (new) page is refreshed with the Destination Addresses field containing a new address.

Step 11 Click the **Select** button below the Services field. The Selecting Services page is displayed.

Step 12 Select **podP service group** from the list of Available Objects and click **Select=>** to move it to the list of Selected Objects.

(where P=pod number)

Step 13 Click **OK**. The Enter Rule Data (new) page is refreshed with the Services field containing the new service.

Step 14 Select **outside** from the Source Interface drop-down menu.

Step 15 Click **OK**. The Summary page is displayed.

Step 16 The Access Rules page is refreshed with the newly created Access Rule.

Step 17 Click **Insert**. The Enter Rule Data (new) page is displayed.

Step 18 Select the **Permit** radio button.

Step 19 Click the **Select** button below the Source Addresses field. The Selecting Network Objects page is displayed.

Step 20 Select **insidehost** from the list of Available Objects and click **Select=>** to move it to the list of Selected Objects.

Step 21 Click **OK**. The Enter Rule Data (new) page is refreshed with the Source Addresses field containing a new address.

Step 22 Click the **Select** button below the Destination Addresses field. The Selecting Network Objects page is displayed.

Step 23 Select **any** from the list of Available Objects and click **Select=>** to move it to the list of Selected Objects.

(where P=pod number)

Step 24 Click **OK.** The Enter Rule Data (new) page is refreshed with the Destination Addresses field containing a new address.

Step 25 Click the **Select** button below the Services field. The Selecting Services page is displayed.

Step 26 Select **podP service group** from the list of Available Objects and click **Select=>** to move it to the list of Selected Objects.

(where P=pod number)

Step 27 Click **OK**. The Enter Rule Data (new) page is refreshed with the Services field containing the new service.

Step 28 Select **inside** from the Source Interface drop-down menu.

Step 29 Click **OK**. The Access Rules page is refreshed with the newly created Access Rule.

Task 10—Configure a Global Security Policy

Complete the following steps to configure a security policy for the global group:

Step 1 Complete the following substeps to enable Syslog generation for the Global group and all children objects:

1. Select **Configuration>Device Settings**. The Device Settings page is displayed.

2. Click the Object Selector to select the **Global** group. The Object Bar refreshes to indicate that Global group is being configured.
3. Click **Logging>Logging Setup** from the TOC. The Logging Setup page is displayed.
4. Select the **Enforce/Mandate settings for children** check box. A window is displayed which reads, "The members of the current group will lose their values. Are you sure you want to mandate these group values?"
5. Click **OK**.
6. Select the **Enable Logging Setup** check box to enable Syslog generation on all output interfaces.
7. Click **Apply**. The Logging Setup page refreshes to indicate that the Firewall MC received the changes.

Step 2 Complete the following substeps to configure Syslog settings for the Global group and all children objects:

1. Click **Logging>Syslog** from the TOC. The Syslog page is displayed.
2. Select the **Enforce/Mandate settings for children** check box. A window is displayed which reads, "The members of the current group will lose their values. Are you sure you want to mandate these group values?"
3. Click **OK**.
4. The Syslog page refreshes to indicate that the Firewall MC received the changes.
5. Select **local7** from the Facility drop-down menu.
6. Select **Debugging** from the Level drop-down menu.
7. Select the **Enable attach timestamp** check box.
8. Enter **0** in the Log Queue Size field.
9. Click **Apply**. The Syslog page refreshes to indicate that the Firewall MC received the changes.
10. Click **Add**. The Enter Syslog Server (new) page is displayed.
11. Select **Inside** from the Interface drop-down menu.
12. Enter the IP address of the inside host in the IP Address field: use IP address **10.0.P.11**. (where P = pod number)
13. Verify that the **UDP** radio button is selected for Protocol.
14. Click **Next**. The Summary page is displayed.
15. Verify that the information listed is correct and click **Finish**. The Syslog page refreshes to display the newly created interface.

Step 3 Use the Object Selector to select **pixP**. The Object Bar refreshes to indicate that pixP is being configured.

(where P = pod number)

Step 4 Select **Logging>Logging Setup** from the TOC. The Logging Setup page is displayed.

Step 5 Try to change the Logging Setup settings. Noticed that the options to configure here are grayed out. This is because we enforced the settings for children objects at the Global group level.

Step 6 Select **Logging>Syslog** from the TOC. The Syslog page is displayed.

Step 7 Try to change the Syslog settings. Noticed that the options to configure here are grayed out. This is due to the fact we enforced the settings for children objects on the Global group level.

Task 11—Deploy the Configuration

You have been configuring the PIX Firewall.. Now it is time to deploy the configuration to the PIX Firewall. Complete the following tasks to deploy the configuration to the PIX Firewall:

Step 1 Complete the following substeps to save the configuration:

1. Click the **Save and Deploy** icon on the Activity bar. A Generate Summary window opens. After a brief moment, a "Generation is finished" message should appear.
2. Click **View Config** to view the generated configuration.
3. Close the Config window.

Step 2 Complete the following substeps to deploy the configuration:

1. Click **Deploy Now**. The Summary window opens. It takes several moments for the Firewall MC to deploy the configuration. A "Deployment finished" message is displayed when deployment has finished.
2. Click **View Config** to view the configuration. Close the Config window.
3. Click **View Transcript** to view the transcript of the deployment. Close the Deploy Transcript window.
4. Close the Summary window.
5. From the Status Summary window, click Refresh. The state parameter changes to deployed.

Task 12—Test the PIX Firewall Configuration

Complete the following steps to test the configuration:

Step 1 Test logging by completing the following substeps:

1. Launch the Kiwi Syslog daemon.
2. Access the **pixP** firewalls.
   (where P = pod number)
3. Execute the **show interface** command on each firewall to generate Syslog messages.
4. Verify that the Kiwi Syslog server has received Syslog messages from both firewalls in reference to the commands that were issued.

Step 2 Test web access to your peer's inside host by completing the following substeps. You should be able to access your peer's inside host:

1. Open a web browser on your student PC.

2. Use the web browser to access your peer's inside host by entering: **http://192.168.Q.10**.
(where Q = peer pod number)

3. Have a peer pod group attempt to access your inside host in the same way.

Step 3 Test web access to the Firewall MC running on your peer's inside host by completing the following substeps. You should be able to access your peer's Firewall MC:

1. Open a web browser on your student PC.

2. Use the web browser to access your peer's inside host by entering:
http://192.168.Q.10:1741.
(where Q = peer pod number)

3. Have a peer pod group attempt to access your inside host in the same way.

20

Enterprise PIX Firewall Maintenance

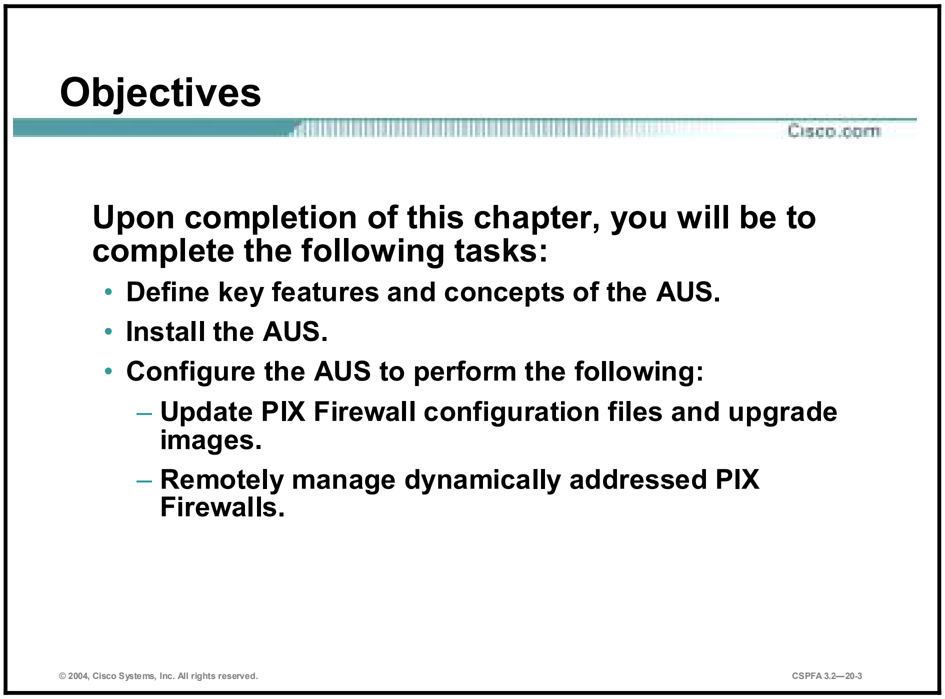
Overview

This chapter introduces Enterprise PIX Firewall Maintenance using the Auto Update Server (AUS). The following topics are covered in this chapter:

- Objectives
- Introduction to the Auto Update Server
- PIX Firewall preparation and AUS communication settings
- Getting started
- Devices, images, and assignments
- Reports and administration
- Summary
- Lab exercise

Objectives

This section lists that chapter's objectives.



Objectives

Cisco.com

Upon completion of this chapter, you will be to complete the following tasks:

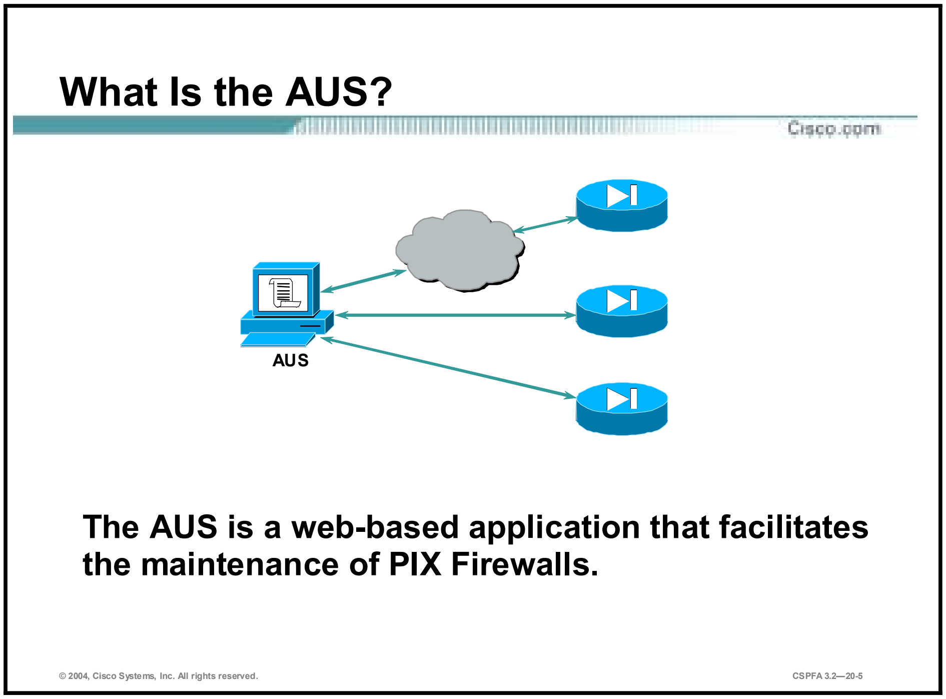
- **Define key features and concepts of the AUS.**
- **Install the AUS.**
- **Configure the AUS to perform the following:**
  - **Update PIX Firewall configuration files and upgrade images.**
  - **Remotely manage dynamically addressed PIX Firewalls.**

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—20-3

Introduction to the Auto Update Server

This section introduces the AUS, its installation requirements, access requirements, supported devices, and installation process.



What Is the AUS?

Cisco.com

The AUS is a web-based application that facilitates the maintenance of PIX Firewalls.

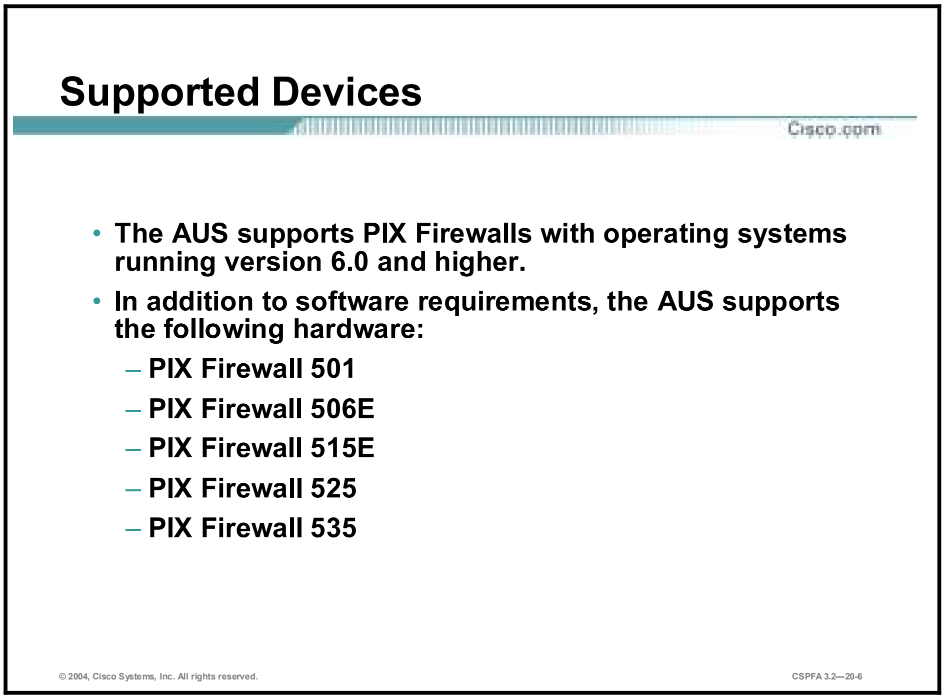
CSPFA 3.2—20-5

AUS facilitates the managing of up to one thousand firewalls. Firewalls operating in auto-update mode periodically contact AUS to upgrade software images, configurations, and versions of PDM, and to pass device information and status to AUS. Using AUS also facilitates the managing of devices that obtain their addresses through Dynamic Host Configuration Protocol (DHCP) or that sit behind Network Access Translation (NAT) boundaries.

If you want to deploy AUS behind a NAT boundary in either the enterprise network or the enterprise DMZ, then the PIX Firewalls being managed by AUS must all be on the same side of the NAT boundary. For example, you can deploy AUS in the DMZ behind a NAT boundary and manage devices that were deployed only on the Internet; however you cannot deploy AUS in the DMZ behind a NAT boundary with some devices using private addresses on the inside of the boundary and some outside on the Internet.

The AUS provides the following features:

- Web-based interface for maintaining multiple PIX Firewalls.
- Support for PIX Firewall operating systems 6.0 and above.
- Ability to support dynamically addressed PIX Firewalls.
- Support for up to 1,000 PIX Firewalls.



Supported Devices

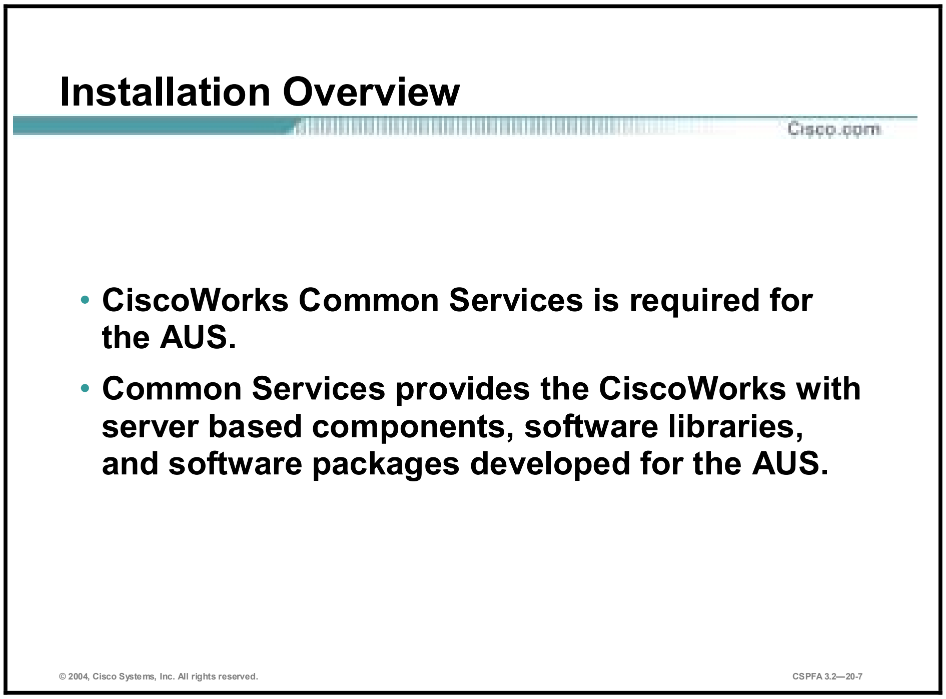
Cisco.com

- The AUS supports PIX Firewalls with operating systems running version 6.0 and higher.
- In addition to software requirements, the AUS supports the following hardware:
  - PIX Firewall 501
  - PIX Firewall 506E
  - PIX Firewall 515E
  - PIX Firewall 525
  - PIX Firewall 535

© 2004, Cisco Systems, Inc. All rights reserved. CSPFA 3.2—20-6

The AUS supports PIX Firewalls with operating systems running version 6.0 and higher. In addition to software requirements, the AUS supports the following hardware:

- PIX 501
- PIX 506E
- PIX 515E
- PIX 525
- PIX 535



Installation Overview

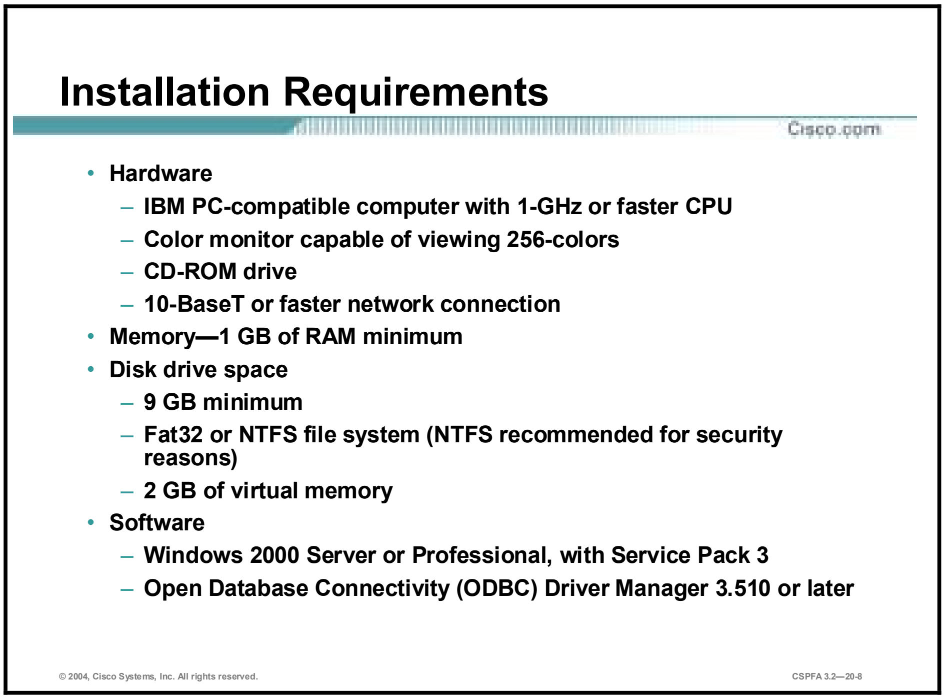
Cisco.com

- CiscoWorks Common Services is required for the AUS.
- Common Services provides the CiscoWorks with server based components, software libraries, and software packages developed for the AUS.

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—20-7

CiscoWorks Common Services is required for the AUS. Common Services provides the CiscoWorks 2000 Server base components, software libraries, and software packages developed for the AUS.



Installation Requirements

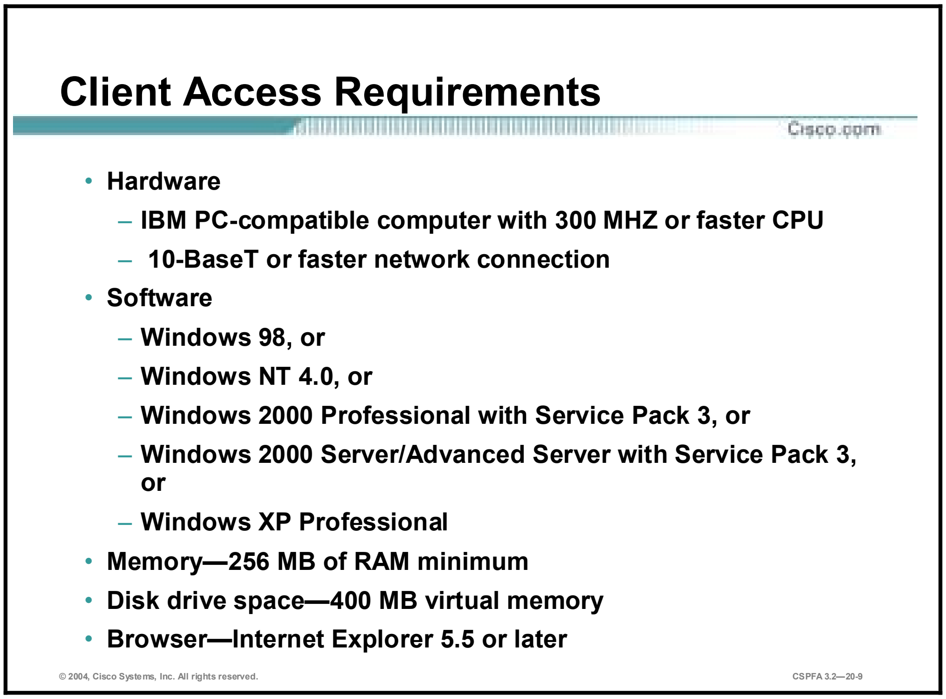
- Hardware
  - IBM PC-compatible computer with 1-GHz or faster CPU
  - Color monitor capable of viewing 256-colors
  - CD-ROM drive
  - 10-BaseT or faster network connection
- Memory—1 GB of RAM minimum
- Disk drive space
  - 9 GB minimum
  - Fat32 or NTFS file system (NTFS recommended for security reasons)
  - 2 GB of virtual memory
- Software
  - Windows 2000 Server or Professional, with Service Pack 3
  - Open Database Connectivity (ODBC) Driver Manager 3.510 or later

CSPFA 3.2—20-8

Before you begin, verify that the server on which you plan to install the AUS meets the following requirements:

- Hardware
  - IBM PC-compatible computer with 1-GHz or faster CPU
  - Color monitor capable of viewing 256-colors
  - A CD-ROM drive
  - A 10-BaseT or faster network connection
- Memory—1 GB of RAM minimum
- Disk Drive Space
  - 9 GB minimum
  - File Allocation Table 32 (FAT-32) or New Technology File System (NTFS recommended for security reasons)
  - 2 GB of virtual memory
- Software
  - Windows 2000 Server or Professional, with Service Pack 2
  - Open Database Connectivity (ODBC) Driver Manager 3.510 or later

Note: CiscoWorks Common Services is required for the AUS to work. CiscoWorks Common Services provides the CiscoWorks2000 Server base components and software developed specifically for the AUS, including the necessary software libraries and packages. For more information see the *Quick Start Guide for the VPN/Security Management Solution.*



Client Access Requirements

Cisco.com

- Hardware
  - IBM PC-compatible computer with 300 MHZ or faster CPU
  - 10-BaseT or faster network connection
- Software
  - Windows 98, or
  - Windows NT 4.0, or
  - Windows 2000 Professional with Service Pack 3, or
  - Windows 2000 Server/Advanced Server with Service Pack 3, or
  - Windows XP Professional
- Memory—256 MB of RAM minimum
- Disk drive space—400 MB virtual memory
- Browser—Internet Explorer 5.5 or later

© 2004, Cisco Systems, Inc. All rights reserved.

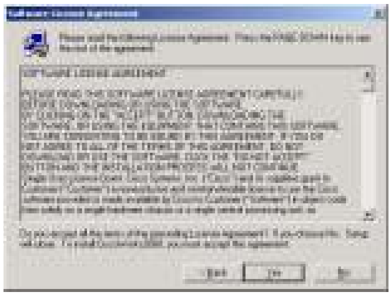
CSPFA 3.2—20-9

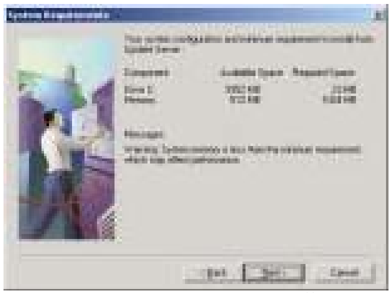
Before you log in to the AUS, verify that the client machine used to log in to the AUS meets the following requirements:

- Hardware—IBM PC-compatible computer with 300 MHZ or faster CPU
- Software
  - Windows 98, or
  - Windows 2000 Professional with Service Pack 3, or
  - Windows 2000 Server/Advanced Server with Service Pack 2, or
  - Windows XP Professional
- Memory—256 MB of RAM minimum
- Disk drive space—400 MB virtual memory
- Browser—Internet Explorer 5.5 or later

Installation Process

Cisco.com





© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—20-10

Complete the following steps to install the AUS, assuming the CiscoWorks Common Services has been installed prior to beginning:

Step 1 Insert the Cisco AUS CD into the CD-ROM drive. If autorun is enabled, the CD should start the installation process automatically. If not, then locate the setup.exe file on the CD-ROM execute it. Once the installation process starts, the Welcome Window is displayed.

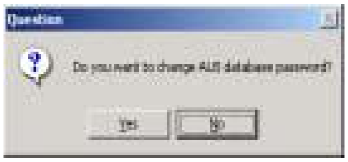
Step 2 Click **Next**. The Software License Agreement window is displayed.

Step 3 If you agree to the Software License Agreement, click **Yes** to process. If you, click No and the installation process stops. The System Requirements window is displayed.

Step 4 Click **Next**. The Verification window is displayed.

Step 5 Click **Next**. A Question window is displayed.

Installation Process (Cont.)



Do you want to change AUS database password?

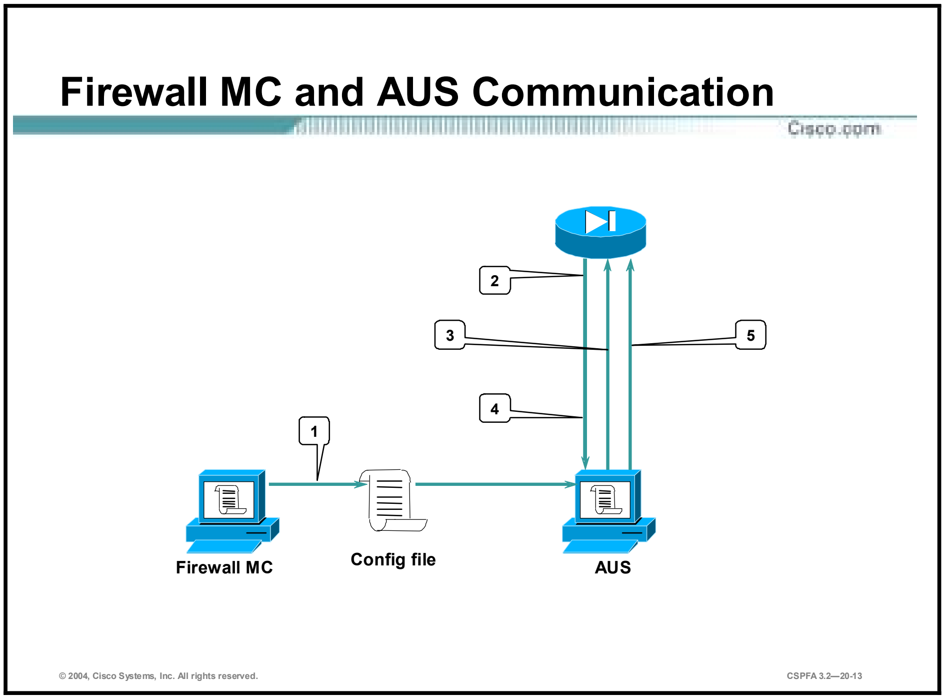
CSPFA 3.2—20-11

Step 6 If you wish to change the AUS database password, click **Yes**. If you do not wish to change the AUS database password, click **No**. After making your selection, the AUS installation process will begin. Upon completion, the Setup Complete window is displayed.

Step 7 Click **Finish**. The AUS installation is now complete.

PIX Firewall and AUS Communication Settings

This section introduces and explains the PIX MC and AUS communication interactions along with using the PIX MC to prepare the PIX Firewall to use the AUS.



Firewall MC and AUS Communication

CSPFA 3.2—20-13

Prior to configuring the PIX MC to operate with the AUS, it is necessary to understand the interaction between the two components. The following steps describe the interaction between the PIX Firewall, PIX MC, and AUS:

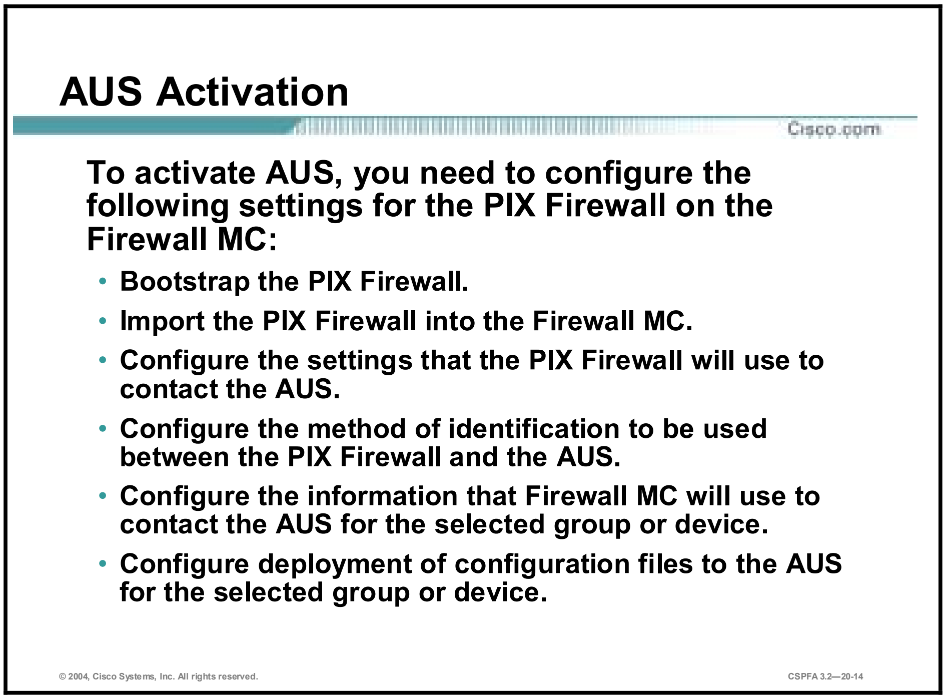
Step 1 The PIX MC deploys the configuration file to the AUS.

Step 2 At the preset polling interval, the PIX Firewall contacts the AUS for updates.

Step 3 The AUS sends a list of the files that the PIX Firewall should be running. The list can include image files and/or configuration files.

Step 4 The PIX Firewall verifies whether it is running the correct file. If not, it requests the file from the AUS.

Step 5 The file is downloaded to the PIX Firewall.



AUS Activation

To activate AUS, you need to configure the following settings for the PIX Firewall on the Firewall MC:

- **Bootstrap the PIX Firewall.**
- **Import the PIX Firewall into the Firewall MC.**
- **Configure the settings that the PIX Firewall will use to contact the AUS.**
- **Configure the method of identification to be used between the PIX Firewall and the AUS.**
- **Configure the information that Firewall MC will use to contact the AUS for the selected group or device.**
- **Configure deployment of configuration files to the AUS for the selected group or device.**

CSPFA 3.2—20-14

Complete the following steps to activate the AUS:

Note: These steps assume that the PIX Firewall has already been configured with a minimal configuration that allows it to communicate with the PIX MC and that the PIX MC is installed and operational.

Step 1 From the PIX Firewall console, use the **http** command(s) to enable the PIX Firewall to accept HTTP connections from the AUS.

Step 2 In the PIX MC, complete the following sub-steps:

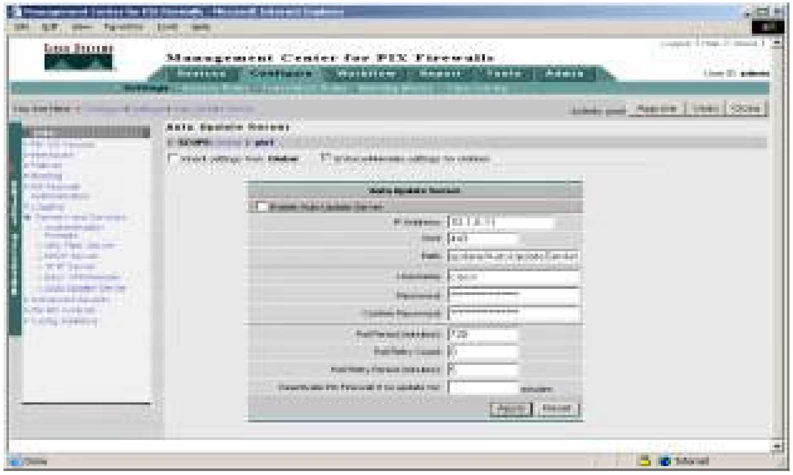
1. Enable the AUS and configure the settings that the PIX Firewall will use to communicate with the AUS.
2. Configure the method of identification that will be used in these communications.
3. Configure settings to enable the PIX MC to communicate with the AUS.
4. Configure the deployment of configuration files to the AUS.

Step 3 Configure the PIX Firewall to use the AUS by copying and pasting the auto-update configuration from the PIX MC to the PIX Firewall.

AUS and PIX Firewall Communications

Cisco.com

Choose Configure>Settings>Servers and Services>Auto Update Server.



CSPFA 3.2—20-15

After you have configured the PIX Firewall to accept HTTP connections from the AUS, you are ready to configure the AUS settings in the PIX MC. Complete the following steps to enable the AUS and PIX Firewall communication settings:

Step 1 Launch and log into the PIX MC.

Step 2 Choose **Configure>Settings**. The Settings page is displayed.

Step 3 Use the Activity Bar to select an existing activity or create a new activity.

Step 4 Use the Object Selector to select a group or device.

Step 5 Click **Servers and Services>Auto Update Server** in the TOC. The Auto Update Server page is displayed.

Step 6 Select the **Enable Auto Update Server Settings** check box.

Step 7 Enter the IP address of the AUS in the IP Address field.

Step 8 Enter the port number on which AUS is listing in the Port field, typically port 443.

Step 9 Enter the directory path where the update if stored on the AUS, the default path is /autoupdate/AutoUpdateServelet in the Path field.

Step 10 Enter the Unique ID the PIX Firewall will use to contact the AUS in the Username field.

Step 11 Enter a password that corresponds to the username in the Password field.

Step 12 Confirm the password by entering it into the Confirm Password field.

Note: The password must be the enable password.

Step 13 Enter the number of minutes between tries the PIX Firewall checks the AUS for updates in the Poll Period (minutes) field. The default is 720 minutes.

Step 14 Enter the number of times the PIX Firewall will try to contact the AUS, if the initial contact fails, in the Poll Retry Count field. The default is 0.

Step 15 Enter the number of minutes between poll retries in the Poll Retry Period (minutes) field; the default is 5 minutes.

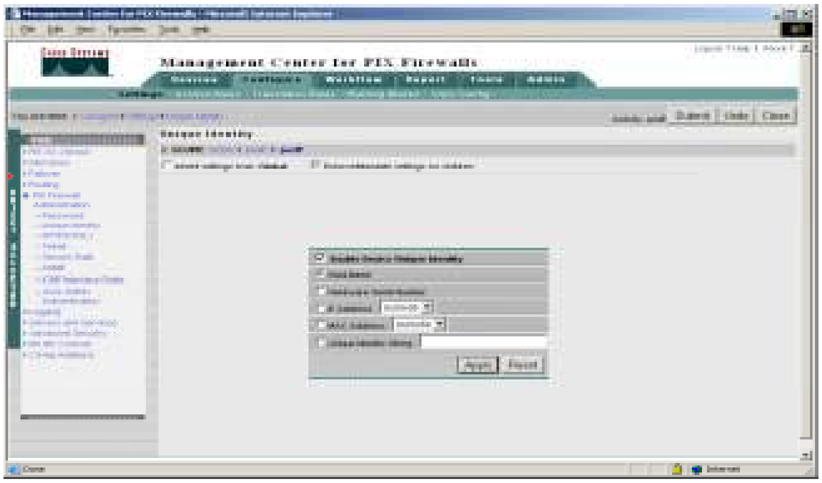
Step 16 Enter a number of count down minutes the PIX Firewall will count off until it deactivates itself, the default is 0.

Step 17 Click **Apply**. The Auto Update Server page refreshes to indicate that the PIX MC received the changes.

PIX Firewall Unique Identity

Cisco.com

Choose Configure>Settings>PIX Firewall Administration>Unique Identity.



CSPFA 3.2—20-16

The Unique Identity feature enables you to assign an identifier to each PIX Firewall. When the PIX Firewall "calls home" to check for updates, it uses the unique identifier to identify itself to the AUS. The AUS then uses the unique identifier to match the current version of software or configuration of the PIX Firewall to its database of current assignments. Complete the following steps to configure the method of identification for communications between the PIX Firewall and the AUS:

Step 1 Launch and log into the PIX MC.

Step 2 Choose **Configure>Settings**. The Settings page is displayed.

Step 3 Use the Activity Bar to select an existing activity or create a new activity.

Step 4 Use the Object Selector to select a group or device.

Step 5 Click **PIX Firewall Administration>Unique Identity** in the TOC. The Unique Identity page is displayed.

Step 6 Enable the PIX Firewall to use a unique identity by selecting the Enable Device Unique Identity check box.

Step 7 Set the unique identifier by selecting the radio button next to one of the following:

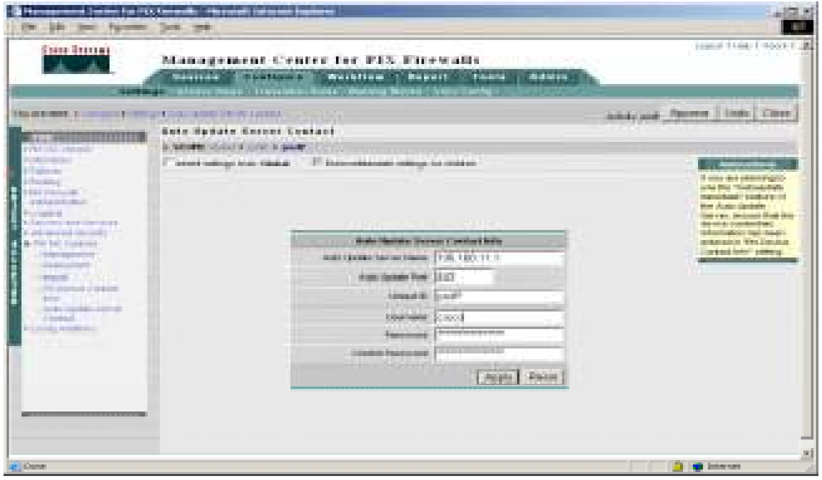
- Host Name
- Hardware Serial Number
- IP Address—When you choose this option, you must use the corresponding drop-down menu to choose from a list of interfaces.
- MAC Address—When you choose this option, you must use the corresponding drop-down menu to choose from a list of interfaces.
- Unique Identity String—When you choose this option, you must manually enter a unique identification string.

Step 8 Click **Apply**. The Unique Identity page refreshes to indicate that the PIX MC received the change. Configure the settings that the PIX Firewall will use to contact the AUS.

AUS Contact Information

Cisco.com

Choose Configure>Settings>PIX MC Controls>Auto Update Server Contact.


Management Centre for PIX Firewalls
Auto Update Server Contact
Apply
Reset

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—20-17

The PIX Firewall needs to know how to contact the AUS. The AUS contact information includes an address, port number, username, and password. Complete the following steps to configure the AUS contact information:

Step 1 Launch and log into the PIX MC.

Step 2 Choose **Configure>Settings**. The Settings page is displayed.

Step 3 Use the Activity Bar to select an existing activity or create a new activity.

Step 4 Use the Object Selector to select a group or device.

Step 5 Click **PIX MC Controls>Auto Update Server Contact** in the TOC. The Auto Update Server Contact page is displayed.

Step 6 Complete the following sub-steps to configure the Auto Update Server Contact settings:

1. Enter the IP address of the AUS in the Auto Update Server Name field.
2. Enter the port number to be used to contact the AUS in the Auto Update Port field, default of 443.
3. Enter the identity that the AUS uses to identify the PIX Firewall in the Unique ID field. This is the identifier that you configured in **Configure>Settings>PIX Firewall Administration>Unique Identity**.
4. Enter the CiscoWorks username that is used to log in to the AUS in the Username field.

Note: You will need to use a CiscoWorks account that has administrator privileges in the PIX MC.

5. Enter the username's password in the Password field.

Note: You will need to use the password that is associated with the CiscoWorks account that is going to be used.

6. Enter the same password again to confirm the original entry in the Confirm Password field.

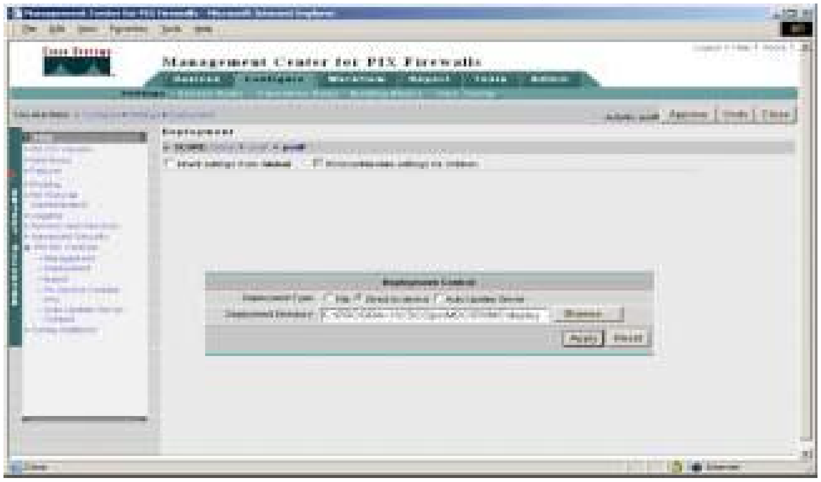
Note: You will need to use the password that is associated with the CiscoWorks account that is going to be used.

Step 7 Click **Apply**. The Auto Update Server Contact page will refresh to indicate that the PIX MC received the changes.

PIX Firewall Configuration Deployment

Cisco.com

Choose Configure>Settings>PIX MC Controls>Deployment**.**



The Deployment feature allows you to specify the method by which configuration information is deployed by the PIX MC. Deployment options include saving the information to a file, downloading directly to devices, or using the Auto Update Server to act as a repository. Complete the following steps to configure the deployment of configuration files to the AUS:

Step 1 Choose **Configure>Settings**. The Settings page is displayed.

Step 2 Click **PIX MC Controls>Deployment** in the TOC. The Deployment page is displayed.

Step 3 Select the **Auto Update Server** radio button to change the Deployment Type.

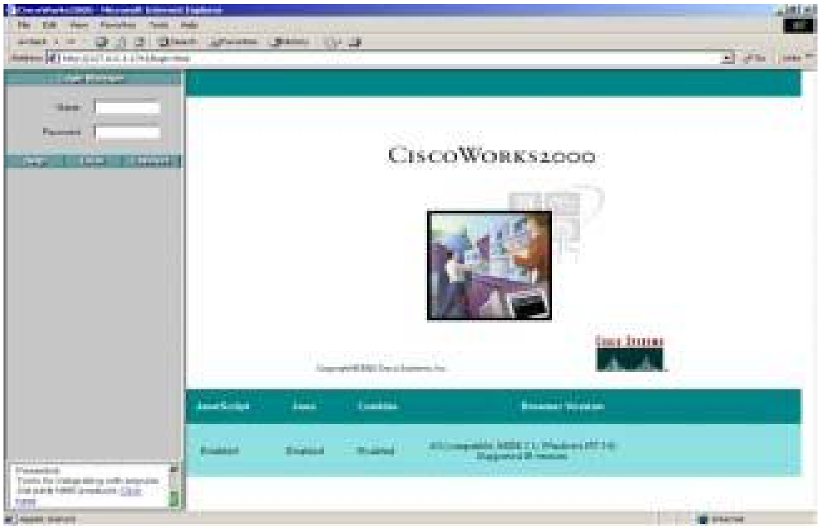
Step 4 Click **Apply**. The Deployment page refreshes to indicate that the PIX MC received the changes.

Getting Started

This section introduces launching the AUS and explains the AUS interface.

CiscoWorks Login

Cisco.com



© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—20-20

Complete the following steps to launch the AUS:

Step 1 Open a browser and point your browser to the IP address of the CiscoWorks machine with a port number of 1741. If the CiscoWorks server is local, you would type the following address in the browser:

```
http://127.0.0.1:1741 <enter>
```

Step 2 Use the default user name and password of **admin** and **admin** to log in to CiscoWorks for the first time.

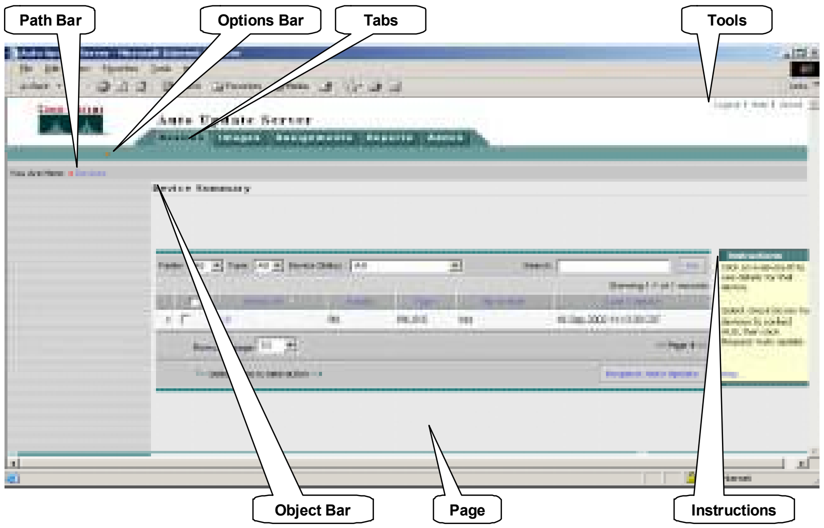
Step 3 Select the **VPN/Security Management Solution** drawer located on the lower left-hand side of the CiscoWorks page. The VPN/Security Management Solution drawer will display to reveal a set of folders and the Auto Update Server icon.

Step 4 Click the **Auto Update Server** icon to launch the AUS. A Security Alert window is displayed.

Step 5 Click **Yes** to accept the security certificate. The AUS opens in another window.

Step 6 Minimize CiscoWorks in the background to avoid confusion when working with the AUS.

AUS Interface



© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—20-21

By learning the elements contained in the AUS interface, you will be able to carry out the basic user task flow and navigate the AUS with ease. The figure illustrates the AUS graphic user interface (GUI). The elements of the AUS are as follow:

- Path bar—Provides a context for the displayed page. Shows the tab, option, and the current page.
- Options bar—Displays the options available for the tab.
- Tabs—Provide access to product functionality, by clicking a tab you are able to access its options.
  - Devices—Displays summary information about devices.
  - Images—Provides information about PIX Firewall software images, PDM images, and configuration files. Images also allow you to add and delete PIX Firewall software images and PDM images.
  - Assignments—Provides assignment information and allows you to change device to image assignments and image to device assignments.
  - Reports—Displays reports.
  - Admin—Allows you to perform administrative tasks.
- Tools—Contains the following buttons:
  - Logout—Logs you out of CiscoWorks2000.
  - Help—Opens a new window that displays context-sensitive help for the displayed page. The window also contains buttons that you use to go to the overall help contents, index, and search tool.
  - About—Displays the version of the application.
- Instructions—Provides a brief overview of how to use the page.
- Page—Displays the area in which you perform application tasks.

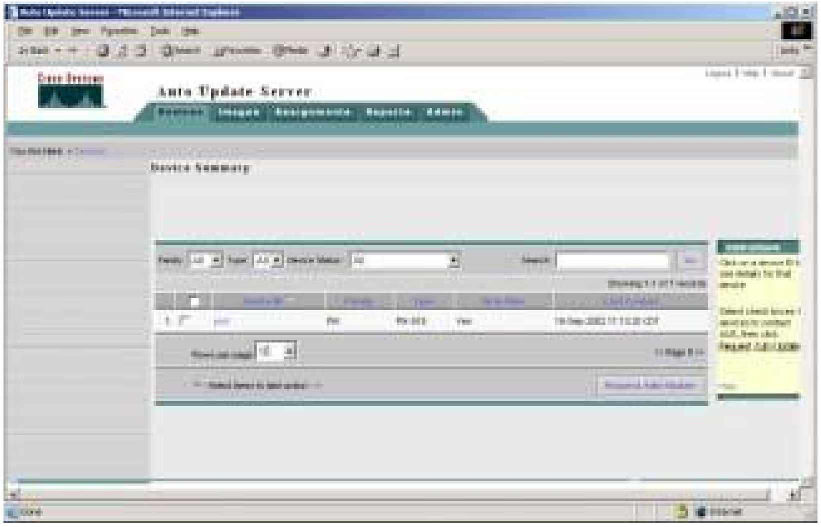
Note: If you are having issues with no devices, images, or reports appearing as normally expected, it may be necessary run a filter or clear your browser cache.

Devices, Images, and Assignments

This section explains the devices, images, and assignment tabs:

AUS—Devices

Cisco.com



CSPFA 3.2—20-23

Clicking on the Device tab displays the Device Summary table. The table shows all managed PIX Firewalls, PIX Firewalls that have not yet contacted AUS, or PIX Firewalls whose image files are not up to date. The table contains information about the PIX Firewalls in AUS, such as the device ID, platform family, platform type, and the last time that the PIX Firewall device contacted AUS.

You can click on a column name to sort the table by column information; or, you can filter and search the table. Use the drop-down menus for Family, Type, or Device Status to filter the Device Summary table. Use the Search field to enter a search string and then the Go button to execute the search.

The Device Summary table columns are as follow, from left to right:

- Device ID—Displays the name the PIX Firewall device uses when identifying itself to AUS. It is typically the Unique ID that is displayed, which can be a hostname or serial number, among others. Click on an entry in the Device ID column to display a table that shows details and associated files for that particular Device ID. The table appears in a new window.
- Family—Series to which the PIX Firewall belongs. Select the family from the list to filter the table according to family.
- Type—Select the type from the list to filter the table according to type. The options that are available in the Type list correspond to the family specified in the Family list.
- Up to Date—Displays if the device is running the newest files.
- Last Contact—Displays the last time that the PIX Firewall contacted the AUS.

The Request Auto Update button is used to have the PIX Firewall contact the AUS immediately. This is to ensure that the PIX Firewall is running the newest files instead of waiting for the device to contact AUS at the specified interval. For example, you may want to request that a device contact AUS if the security of your network has been affected.

Complete the following steps to force the PIX Firewall to contact the AUS:

Step 1 Choose **Devices**. The Devices page is displayed.

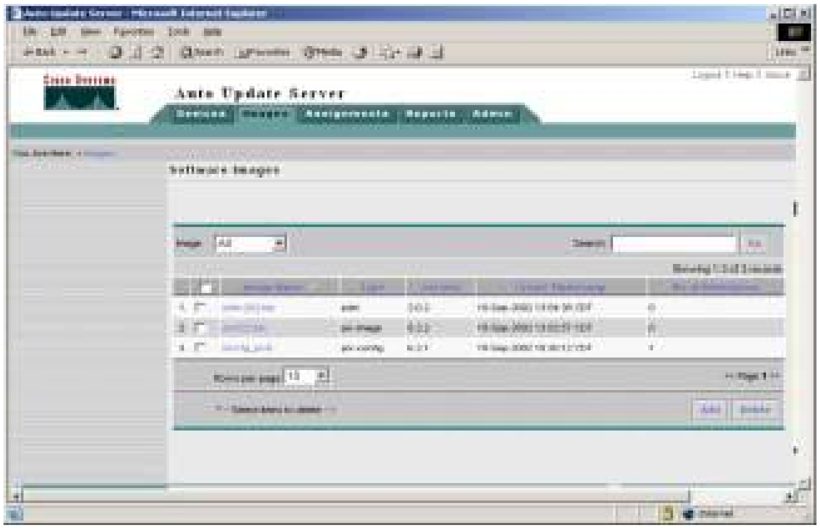
Step 2 Select the check box next to the PIX Firewall that you wish to contact the AUS immediately.

Step 3 Click the **Request Auto Update** button below the Device Summary table. The Request Auto Update Confirmation page is displayed. The Request Auto Update Confirmation page displays a message that reads, “Requests are successfully queued. Please check Event Report for details.”

Step 4 Click **Event Report** to view if the AUS was contacted. The Event Report page is displayed with the event report for the PIX Firewall.

AUS—Images

Cisco.com



CSPFA 3.2—20-24

The AUS allows you to manage PIX Firewall software images, PDM images, and PIX Firewall configuration files. From the Images tab, you can add or delete PIX Firewall software images or PDM images and delete PIX Firewall configuration files. PIX Firewall configuration files can be added to the AUS only by using the PIX MC deployment feature.

The Software Images table columns are as follow, from left to right:

- Image Name—Name of the image that is stored on the AUS. You can click on an entry in the Image Name column to display a table of information about the image.
- Type—Type of image, such as PDM, PIX Firewall software image or configuration file.
- Version—Version of the image.
- Create Timestamp—Time that the image was added to the AUS.
- No. of References—Number of devices that have been assigned to the file.

Complete the following steps to add an image to the AUS:

Step 1 Choose **Images**. The Images page is displayed.

Step 2 Click the **Add**. The Add Image page is displayed.

Step 3 Manually enter the location and name of the image file or use the Browse button to locate the software image.

Step 4 Select either PDM or pix-image from the Image Type drop-down menu.

Step 5 Click **OK**. The Software Images page refreshes to display the newly added PDM or PIX Firewall image file.

Complete the following steps to delete an image from the AUS:

Step 1 Select the **Images** tab. The Images page is displayed.

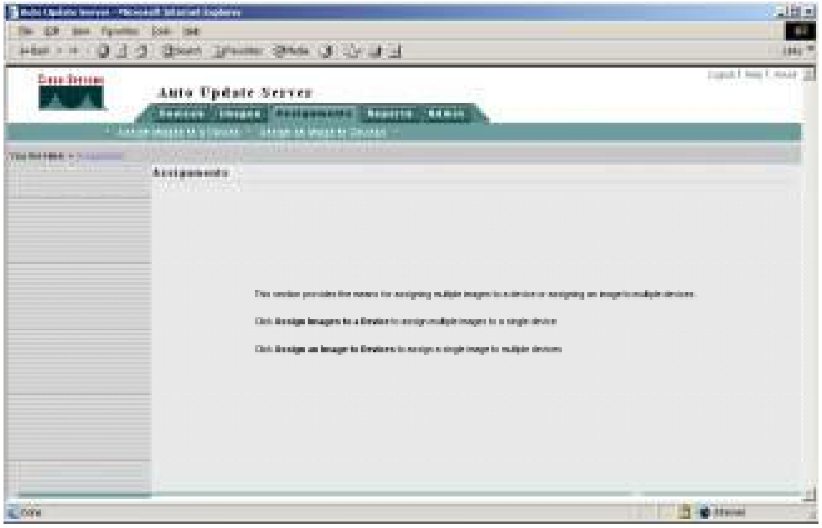
Step 2 Select the check box next to the image that you wish to delete.

Step 3 Click **Delete**. The Delete Image page is displayed.

Step 4 Click **OK**. The Images page is displayed with the image removed from the list of images.

Note: Deleting an image file that is assigned to multiple devices deletes *all* references between the image file and the devices. To delete a PIX configuration file, you must use the PIX MC.

AUS—Assignments



This section provides the means for assigning multiple images to a device or assigning an image to multiple devices.

Click **Assign Images to a Device** to assign multiple images to a single device.

Click **Assign an Image to Devices** to assign a single image to multiple devices.

When a new image file is released, you can download the file, add it to AUS, and assign the image to a device. This is beneficial, for example, when a new file release fixes a security problem that an older version of the file did not address or you want to upgrade the configuration files after deployment using the AUS. From the Assignments tab, you can do the following:

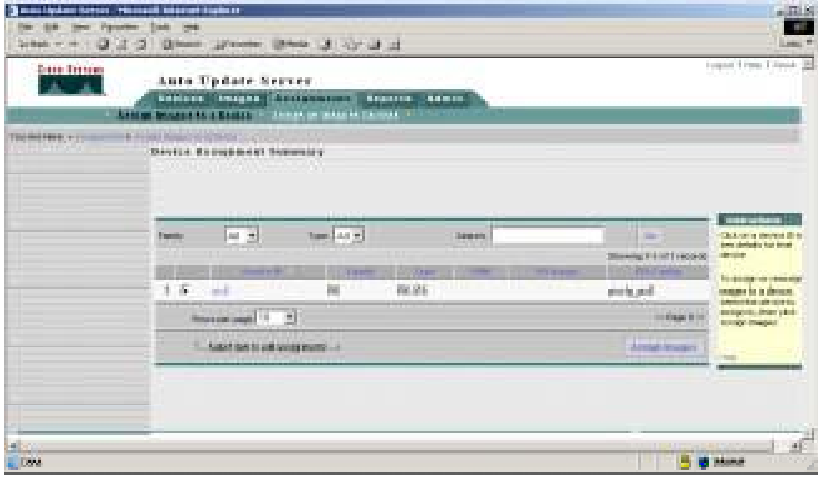
- Assign one or more images to one device.
- Assign one or more images to multiple devices.

Note: Depending on the speed of your network and the sizes of the images that you are deploying, the amount of time will vary greatly. Be patient.

Assignments—Images to a Device

Cisco.com

Choose Assignments>Images to a Device.



CSPFA 3.2—20-26

Complete the following steps to assign one or more images to a single PIX Firewall:

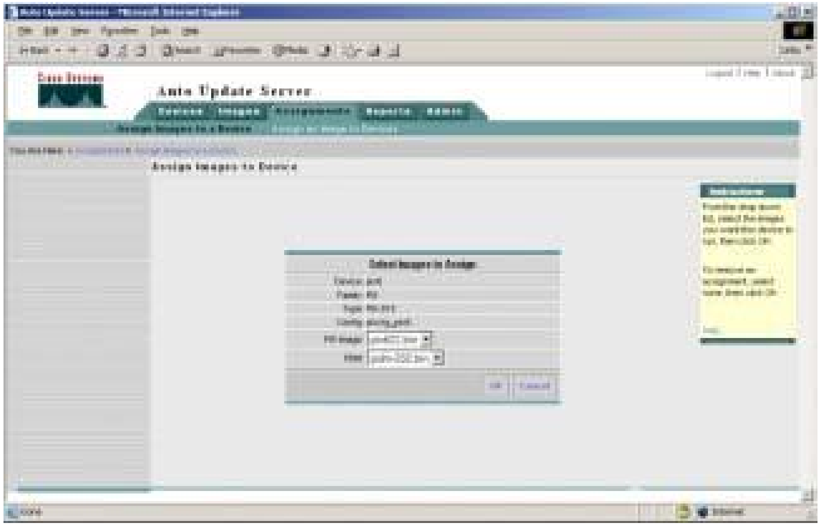
Step 1 Choose **Assignments**. The Assignments page is displayed.

Step 2 Select **Assign Images to a Device**. The Device Assignment Summary page is displayed.

Step 3 Select the radio button next to the PIX Firewall to which you wish to assign images and click **Assign Images**. The Assign Images to Device page is displayed.

Assignments—Images to a Device (Cont.)

Cisco.com



CSPFA 3.2—20-27

Step 4 Select the PIX Firewall software image you wish to assign from the PIX Image drop-down menu.

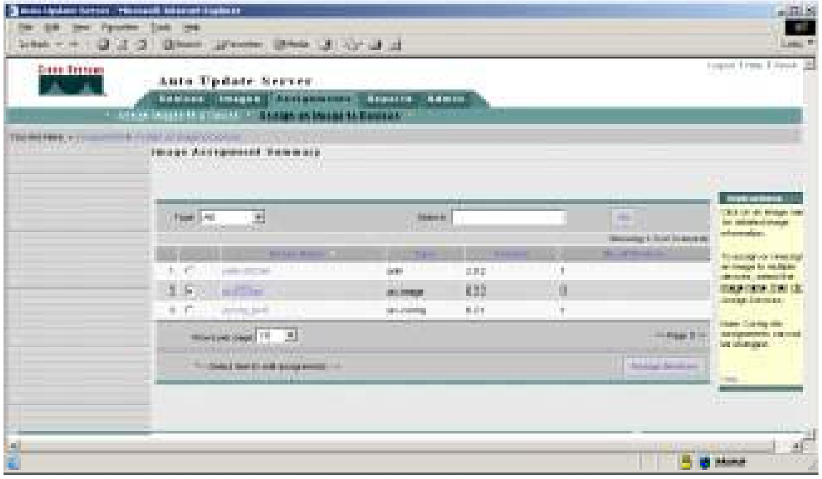
Step 5 Select the PDM image you wish to assign from the PDM drop-down menu.

Step 6 Click **OK**. The Device Assignment Summary page is displayed with the appropriate image files updated in the respective columns.

Assignments—An Image to Devices

Cisco.com

Choose Assignments>Assign an Image to Devices.



CSPFA 3.2—20-28

Complete the following steps to assign one or more images to multiple PIX Firewalls:

Step 1 Choose **Assignments**. The Assignments page is displayed.

Step 2 Select **Assign an Image to Devices**. The Image Assignment Summary page is displayed.

Step 3 Select the radio button by the image that you wish to assign to Devices and click Assign Devices. The Assign Image to Devices page is displayed.

Step 4 Select the check box next to the PIX Firewall to which you wish to have the image assigned.

Step 5 Click **OK**. The Image Assignment Summary page is displayed with the No. of Devices column incremented with the number of PIX Firewalls chosen earlier.

Click on the image name itself to view what PIX Firewalls are assigned to certain images. In addition to PIX Firewall software images and PDM images, you can also assign PIX Firewall configuration files.

Note: Depending on the speed of your network and the sizes of the images that you are deploying, the amount of time will vary greatly. Be patient.

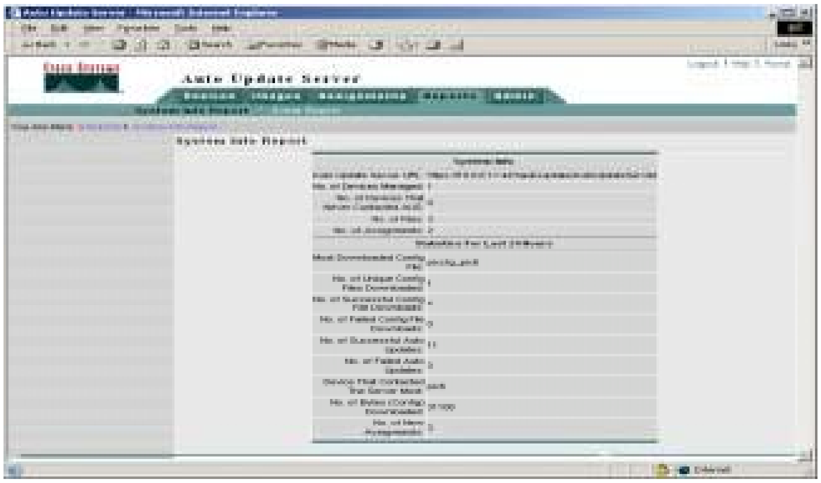
Reports and Administration

This section introduces and explains the reporting and administration feature of the AUS.

Reports—System Information

Cisco.com

Choose Reports>System Info Report.



CSPFA 3.2—20-30

The System Info Report table shows general information about AUS and also shows how busy the server is. The report contains information such as how many devices have contacted AUS and how many configuration files have been downloaded within the last 24 hours. Choose **Reports>System Info** to view the System Info Report.

The System Info Report Table contains the following information:

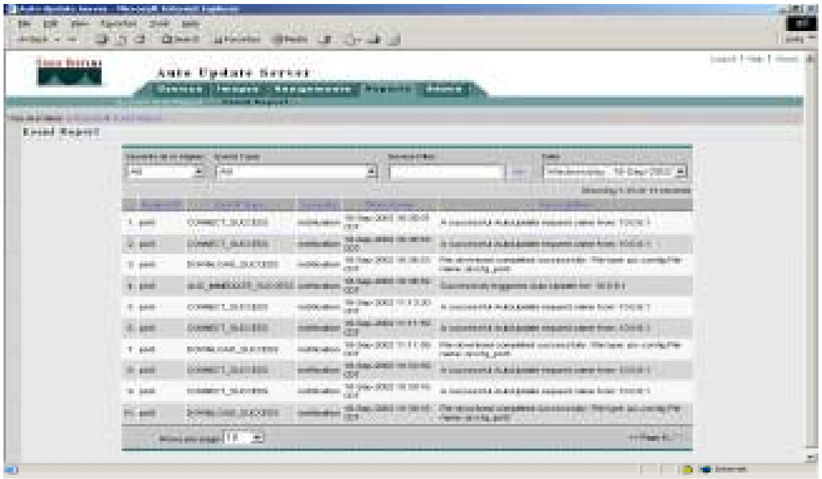
- Auto Update Server URL—URL that the PIX Firewalls use to contact the AUS.
- No. of Devices Managed—The number of PIX Firewalls that are being managed by the AUS.
- No. of Files—The number of files that the AUS contains. These files include PIX Firewall images, PDM images, and PIX Configuration files.
- No. of Assignments—The number of image to device and device to image assignments.
- Most Downloaded Config File—The PIX Firewall configuration file that has been downloaded the most in a 24-hour period.
- No. of Unique Config Files Downloaded—The number of unique configuration files that the AUS downloaded during the last 24 hours.
- No. of Successful Config File Downloads—The number of configuration file downloads that completed successfully during last 24 hours.
- No. of Failed Config File Downloads—The number of times an error occurred while a device was performing an Auto Update during the last 24 hours.
- No. of Successful Auto Updates—The number of times that devices successfully contacted AUS during the last 24 hours.

- No. of Failed Auto Updates—The number of times that devices did not contact AUS during the last 24 hours.
- Device That Contacted Server Most—The device that contacted AUS most often during the last 24 hours.
- No Bytes (Config) Downloaded—The number of bytes that have been downloaded for configuration files during the last 24 hours.
- No. of New Assignments—The number of new image to device and device to image assignments during the last 24 hours.

Reports—Event Report

Cisco.com

Choose Reports>Event Report.



CSPFA 3.2—20-31

The Event Report shows information about devices that have contacted the AUS. The report describes information such as the event type and result of the event. It also shows information about notifications sent from PIX Firewall devices to AUS. For example, if a PIX Firewall device downloads a configuration file and discovers errors, it sends an alert to the AUS, which the table displays. Entries are added each time a device contacts the AUS or a file is downloaded.

Click on a column name to sort the table by column information. When you click on the Device ID column, the table is sorted first by device ID, then by timestamp. You can also filter the table and search the table for a specified device ID. Another way to filter the event report data is to use the drop-down menus located at the top of the Event Report.

The possible Event Types are as follow:

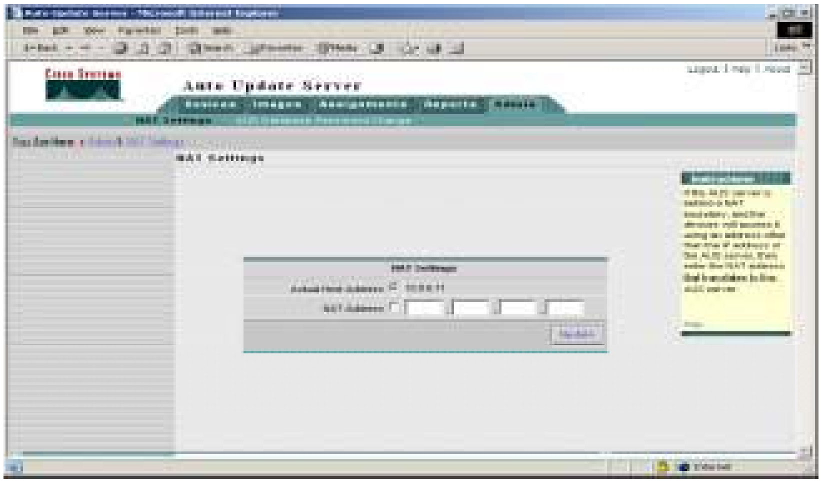
- CONNECT_SUCCESS—The PIX Firewall contacted the AUS successfully and reported its inventory details.
- CONNECT_FAILURE—A problem occurred during an auto update attempt. Possible causes are:
  - — Error while parsing XML.
  - — Invalid credentials.
  - — Device has not been added to AUS.
  - — Connectivity problems.
  - — Database down while trying to add record.
- DEVICE_CONFIG_ERROR—The PIX Firewall reported errors to the AUS or errors occurred while loading configuration file assigned to the device.
- GENERAL_DEVICE_ERROR—The PIX Firewall reported a non-configuration file error to AUS. Possible causes are:

  - Problems connecting to the Auto Update servlet.
  - Problems with the downloaded image (invalid checksum).
- DOWNLOAD SUCCESS—The file was successfully sent to the remote device without error. This does not mean that the device is running the image successfully; this message could be followed by either DEVICE_CONFIG_ERROR or GENERAL_DEVICE_ERROR.
- DOWNLOAD_FAILURE—An error occurred while an image or configuration file was being downloaded. Possible causes are:
  - Invalid credentials.
  - Communication problems.
  - Database problem.
- AUS_IMMEDIATE_SUCCESS—The AUS successfully contacted and updated the device.
- AUS_IMMEDIATE_FAILURE—An error occurred while the device was being updated. Possible causes are:
  - The server does not have direct connectivity to the device (for example, it is behind a NAT boundary).
  - The Enable or TACACS username and password that the device uses to authenticate AUS are incorrect.
  - An internal error occurred.
- SYSTEM_ERROR—An internal error occurred.

Admin—NAT Settings

Cisco.com

Choose Admin>NAT Settings.



CSPFA 3.2—20-32

The Admin tab allows you to change NAT settings. NAT Settings allows you to specify the correct address for devices to use when they try to download images or report errors. If AUS is behind a NAT boundary, the address that the device uses to contact AUS is most likely different from the actual IP address. Therefore, you must specify the IP address that devices on the public side of the NAT boundary must use to access AUS.

Complete the following steps to configure NAT settings:

Step 1 Choose **Admin>NAT Settings**. The NAT Settings page is displayed.

Step 2 Select the **NAT Address** radio button.

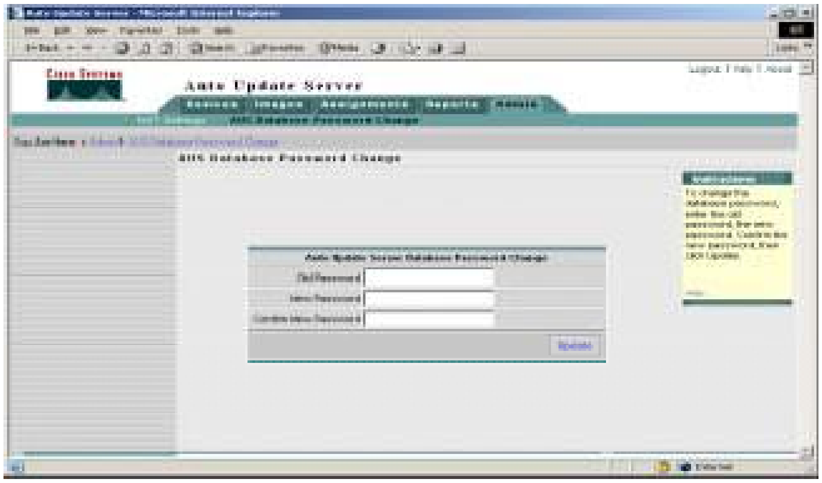
Step 3 Enter the IP address that corresponds to the public IP address.

Step 4 Click **Update**. The NAT Settings page will refresh to indicate that the AUS received the changes.

Admin—AUS Database Password Change

Cisco.com

Choose Admin>AUS Database Password Change.



AUS Database Password Change

Instructions

To change the database password, enter the old password, the new password. Confirm the new password, then click Update.

Auto Update Server Database Password Change

Old Password

New Password

Confirm New Password

Update

CSPFA 3.2—20-33

The AUS Database Password Change page allows you to change the password of the AUS database. It is recommended that you change this password for security purposes.

Complete the following steps to change the AUS Database Password:

Step 1 Choose **Admin>AUS Database Password Change**. The AUS Database Password Change page is displayed.

Step 2 Enter the current password in the Old Password field.

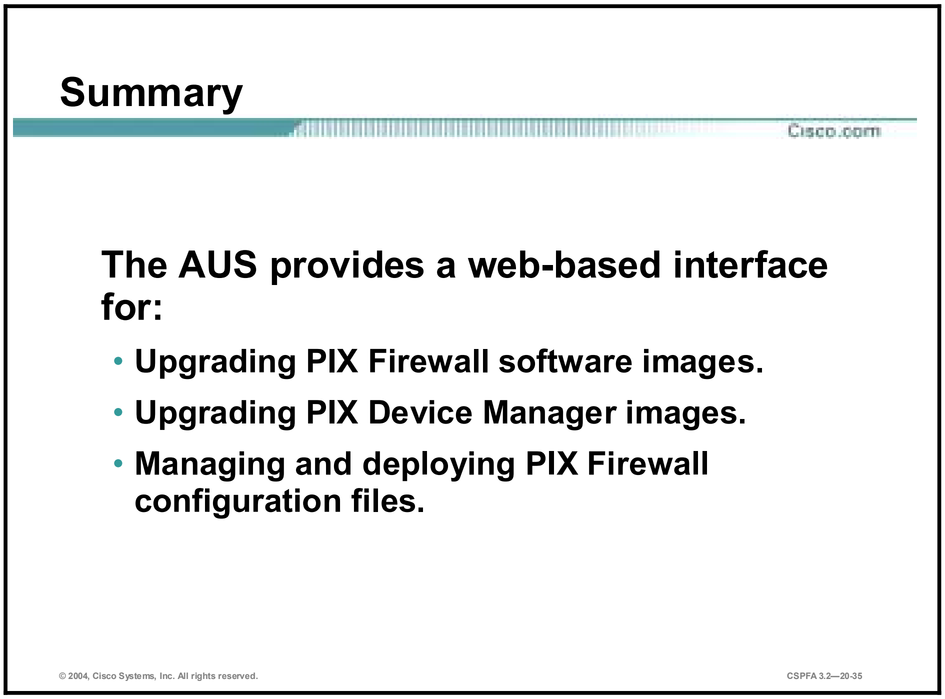
Step 3 Enter the new password in the New Password field.

Step 4 Enter the new password again to confirm it, in the Confirm New Password field.

Step 5 Click **Update**. The AUS Database Password Change page will refresh to indicate that the AUS received the changes.

Summary

This section summarizes the information you learned in this chapter.



Summary

Cisco.com

The AUS provides a web-based interface for:

- Upgrading PIX Firewall software images.
- Upgrading PIX Device Manager images.
- Managing and deploying PIX Firewall configuration files.

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—20-35

21

Firewall Services Module

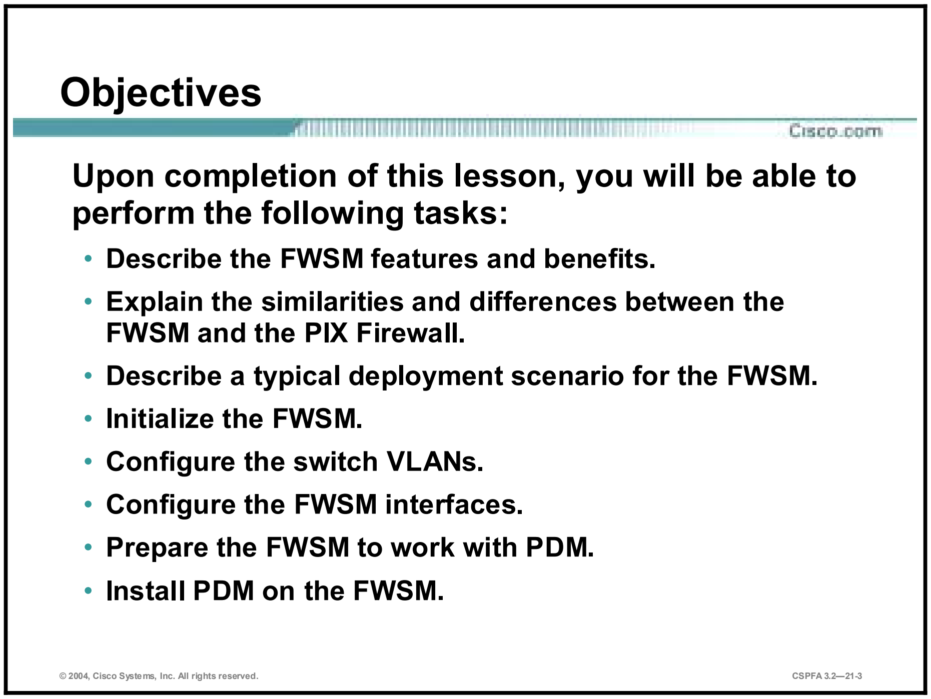
Overview

This lesson includes the following topics:

- Objectives
- FWSM overview
- Network model
- Getting started
- Using PDM with the FWSM
- Troubleshooting the FWSM
- Summary

Objectives

This topic lists the lesson's objectives.



Objectives

Cisco.com

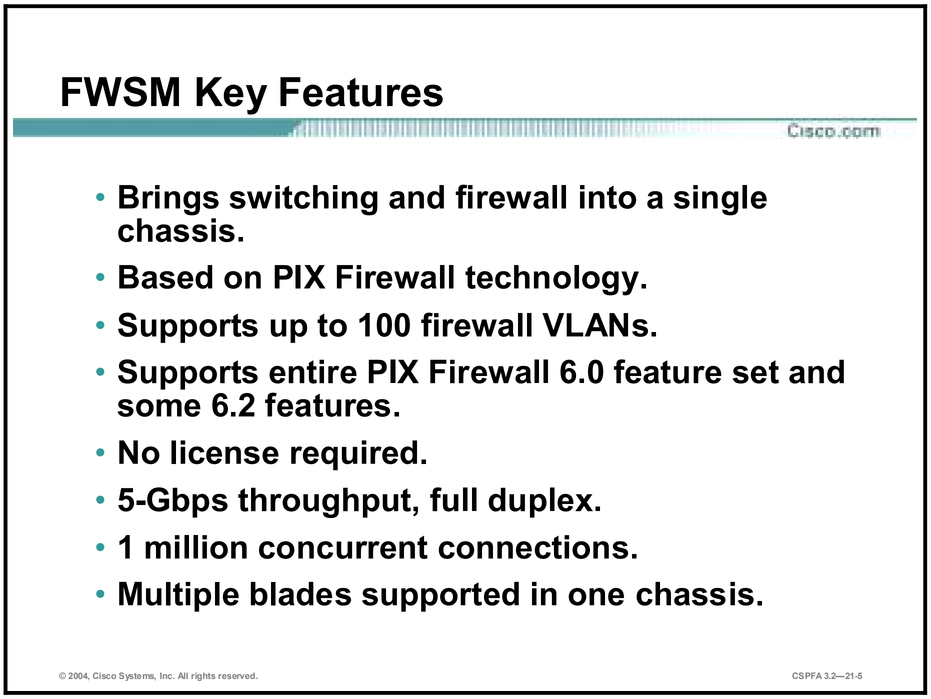
Upon completion of this lesson, you will be able to perform the following tasks:

- Describe the FWSM features and benefits.
- Explain the similarities and differences between the FWSM and the PIX Firewall.
- Describe a typical deployment scenario for the FWSM.
- Initialize the FWSM.
- Configure the switch VLANs.
- Configure the FWSM interfaces.
- Prepare the FWSM to work with PDM.
- Install PDM on the FWSM.

CSPFA 3.2—21-3

FWSM Overview

This topic introduces the FWSM.



FWSM Key Features

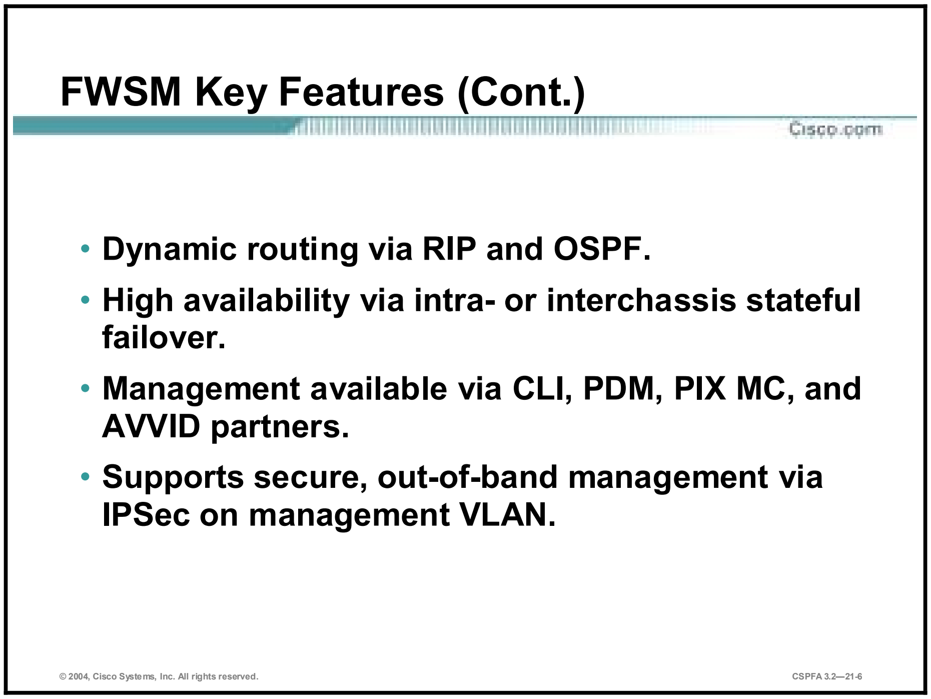
- Brings switching and firewall into a single chassis.
- Based on PIX Firewall technology.
- Supports up to 100 firewall VLANs.
- Supports entire PIX Firewall 6.0 feature set and some 6.2 features.
- No license required.
- 5-Gbps throughput, full duplex.
- 1 million concurrent connections.
- Multiple blades supported in one chassis.

CSPFA 3.2—21-5

The Cisco Firewall Services Module (FWSM) is an integrated module for the Cisco Catalyst 6500 Series Switch and the Cisco 7600 Series Internet Router. The Cisco Catalyst 6500 provides intelligent services such as firewall capability, intrusion detection, and virtual private networking, along with multilayer LAN, WAN, and MAN switching capabilities. The Cisco 7600 Series Internet Router offers optical WAN and metropolitan-area network (MAN) networking with line-rate IP services at the network edge.

The Cisco FWSM is a high-performance firewall solution, providing 5 Gbps of throughput per module and scaling to 20 GB of bandwidth with multiple modules in one chassis. The FWSM is completely VLAN aware, offers dynamic routing, and is fully integrated within the Cisco Catalyst 6500 Series switches. The FWSM is based on Cisco PIX Firewall technology, and therefore offers the same security and reliability as the PIX Firewall. It includes the entire PIX Firewall 6.0 software feature set and some of the features of PIX Firewall Software Version 6.2.

Note: The 5 Gbps throughput is based on UDP traffic.



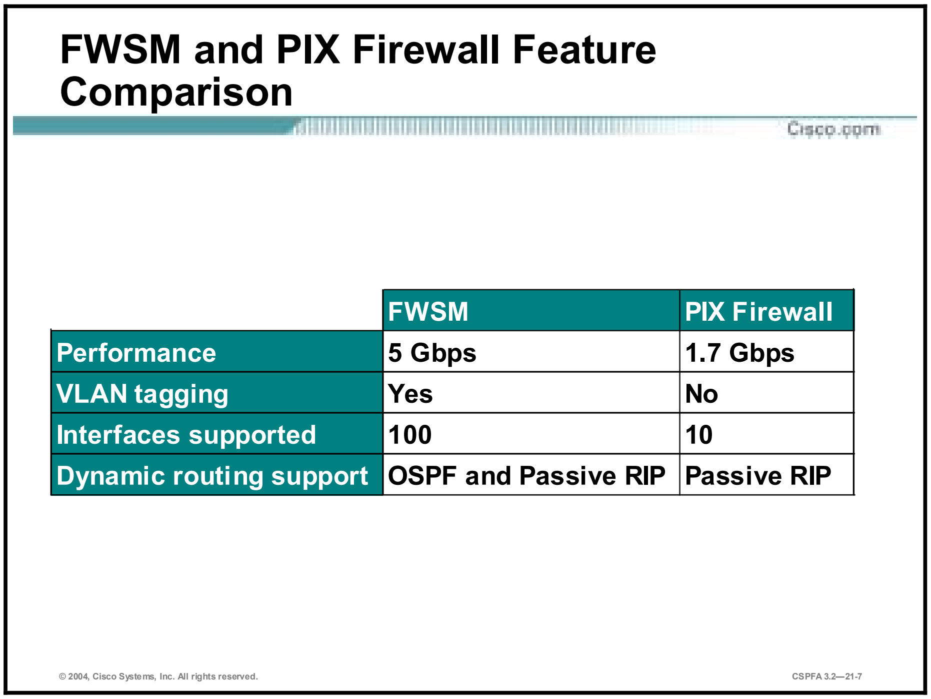
FWSM Key Features (Cont.)

- Dynamic routing via RIP and OSPF.
- High availability via intra- or interchassis stateful failover.
- Management available via CLI, PDM, PIX MC, and AVVID partners.
- Supports secure, out-of-band management via IPSec on management VLAN.

CSPFA 3.2—21-6

In addition to the entire PIX Firewall 6.0 feature set, the FWSM offers dynamic routing via RIP and Open Shortest Path First (OSPF), intra- or interchassis failover, a variety of management options, and secure out-of-band management via IPSec. It supports the following features of PIX Firewall Software Version 6.2:

- Command authorization
- Object grouping
- Internet Locator Service (ILS)/NetMeeting fixup
- URL filtering enhancement



FWSM and PIX Firewall Feature Comparison

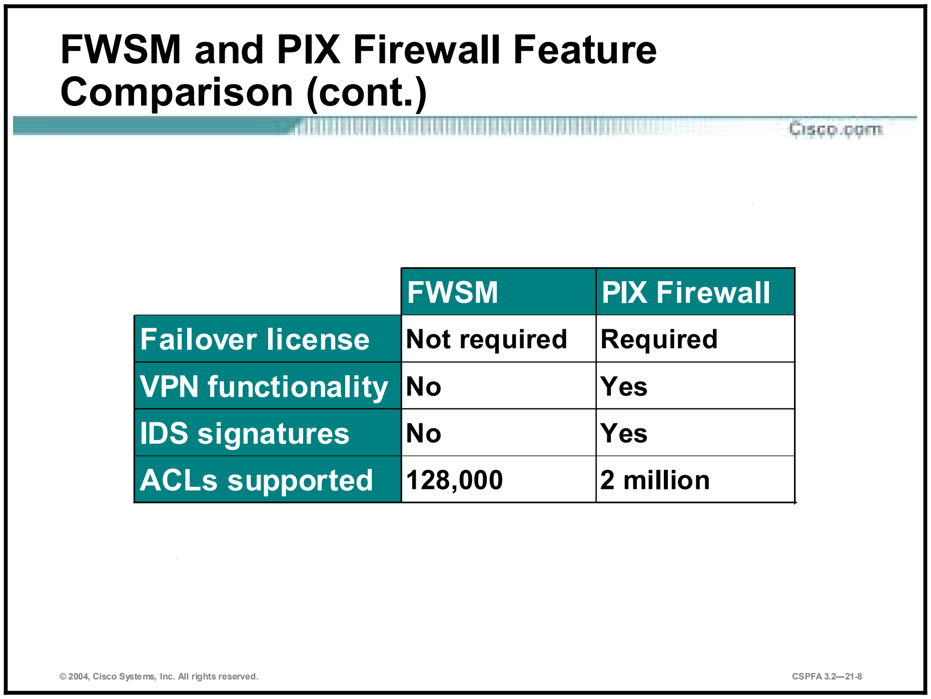
	FWSM	PIX Firewall
Performance	5 Gbps	1.7 Gbps
VLAN tagging	Yes	No
Interfaces supported	100	10
Dynamic routing support	OSPF and Passive RIP	Passive RIP

CSPFA 3.2—21-7

The FWSM understands 802.1q and Inter-Switch Link (ISL) protocols and supports up to 100 firewall VLANs. Other ways in which the FWSM differs from the PIX Firewall appliance are as follow:

- By default, both inbound and outbound connections are denied.
- The **conduit** command is not supported.
- ActiveX and Java filtering fixups are not supported.
- By default, the HTTP fixup is disabled.
- Bidirectional Network Address Translation (NAT) is not supported.
- The OSPF routing protocol is supported.

Note: All FWSM interfaces, including the console, are mapped via VLANs.



FWSM and PIX Firewall Feature Comparison (cont.)

Cisco.com

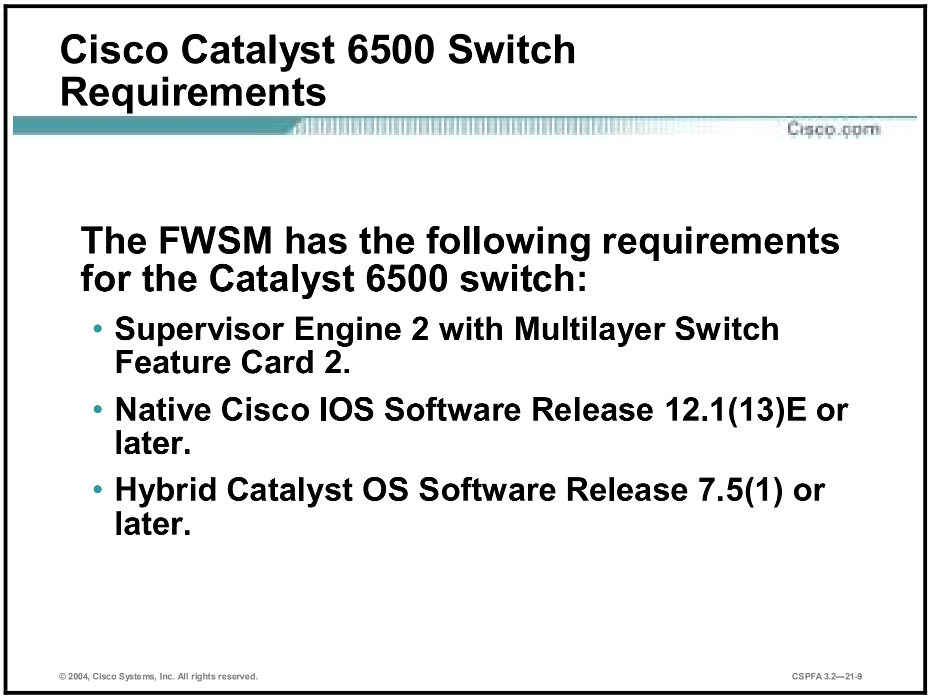
	FWSM	PIX Firewall
Failover license	Not required	Required
VPN functionality	No	Yes
IDS signatures	No	Yes
ACLs supported	128,000	2 million

CSPFA 3.2—21-8

Other differences between the FWSM and the PIX Firewall appliance include the following:

- No licensing is required.
- Virtual private network (VPN) functionality (IPSec, Point-to-Point Tunneling Protocol [PPTP] and Layer 2 Tunneling Protocol [L2TP]) for packets flowing across the firewall is not supported.
- Intrusion detection system (IDS) Syslog messages are not generated.
- The maximum number of access control lists (ACLs) supported is 128,000.

Note: The 128,000 number refers to the number of access list entry (ACE) nodes available on the network processor. The actual practical number of access control lists (ACLs) supported by the FWSM is approximately 85,000.



Cisco Catalyst 6500 Switch Requirements

The FWSM has the following requirements for the Catalyst 6500 switch:

- **Supervisor Engine 2 with Multilayer Switch Feature Card 2.**
- **Native Cisco IOS Software Release 12.1(13)E or later.**
- **Hybrid Catalyst OS Software Release 7.5(1) or later.**

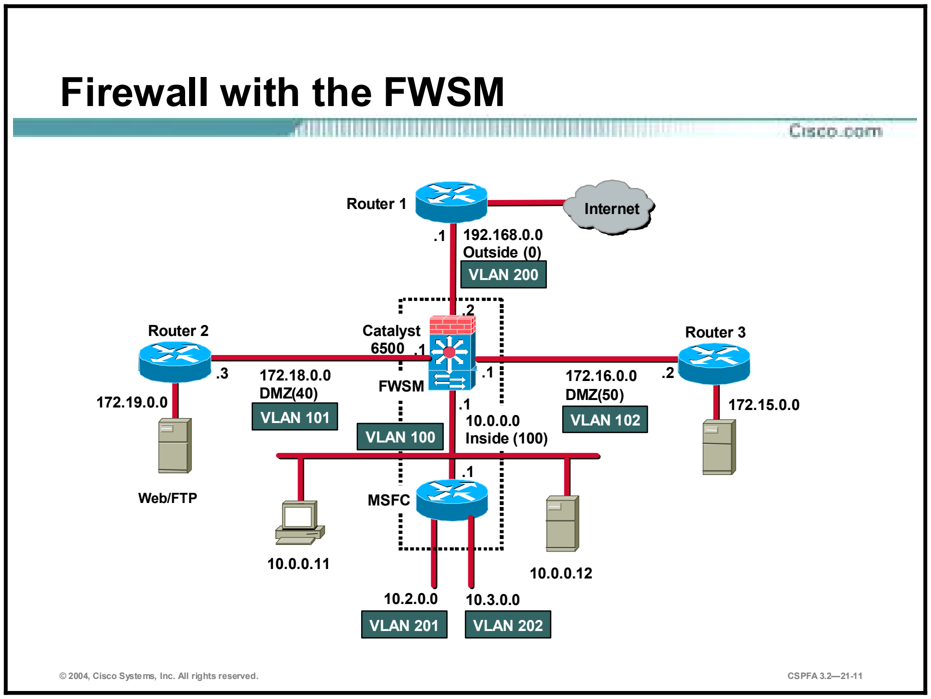
CSPFA 3.2—21-9

The FWSM occupies one slot in a Cisco Catalyst 6500 switch. Up to four modules can be installed in the same switch chassis. The FWSM has the following requirements for the Catalyst 6500 switch:

- Supervisor Engine 2 with Multilayer Switch Feature Card 2 (MSFC2)
- Native Cisco IOS Software Release 12.1(13)E or higher
- Hybrid Catalyst Operating System minimum Software Release 7.5(1)

Network Model

This topic explains typical FWSM deployments and traffic flow.



Firewall with the FWSM

Cisco.com

CSPFA 3.2—21-11

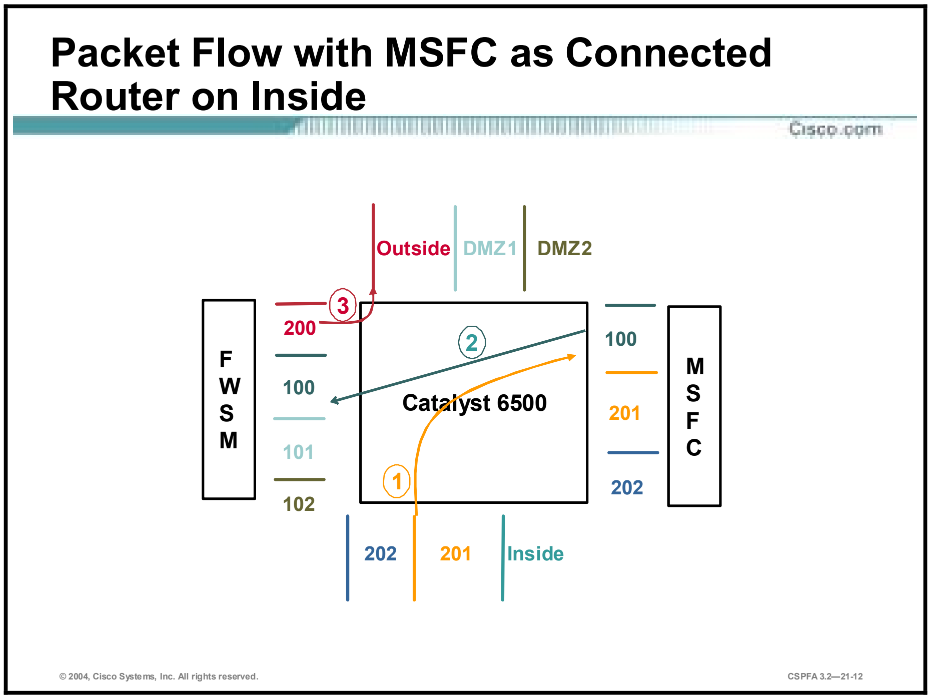
The Firewall Services Module can be used in a variety of topologies depending on the needs of your network. For example, in a data center you may want to provide access control or segregate your security domains. The security domain can be a collection of servers with the same security level. Within that domain, multiple subnets or server farms can exist. When you configure the FWSM to function on the perimeter of the network, the module can provide access control to the inside network as a whole, or segregate multiple security zones through VLAN interfaces of different security levels. The security zones can be either in the same network or can define the boundaries of multiple networks.

The FWSM configuration has the following characteristics:

- Each firewall interface is a Layer 3 interface. It is uniquely associated with a VLAN, a Security Level and an IP address.
- An interface is protected by a firewall depending on where the interface is used. The module interfaces are firewall protected, while all other interfaces in the system are considered to be outside the firewall. Each firewall interface has a fixed VLAN.
- The MSFC may be configured as a connected router on any one and only one firewall interface, but it is not necessary to configure the MSFC as a connected router. The FWSM views all networks or subnetworks beyond an interface as belonging to the same security level.
- Traffic from all of the nonfirewall VLANs in the switch (those not recognized by the module) is routed through the MSFC without being stopped by the firewall.

The figure shows a firewall configuration with the FWSM. The switch and the router beneath it represent a FWSM and a Multilayer Switch Feature Card (MSFC), respectively, within same switch. The MSFC, which provides multiprotocol routing with multilayer switching for the Catalyst 6000 family switch Ethernet interfaces, is used in this example as a router on the network inside the firewall. VLANs 100, 101 and 102 are configured as firewall VLANs. The MSFC is connected to only one of the controlled firewall interfaces. All router interfaces configured on the MSFC are considered to be the same security level as the firewall interface to which the MSFC is connected. In this example, VLANs 201 and 202, which are not configured as controlled, are considered inside the firewall, but traffic between them is routed by the MSFC without being protected by the firewall.

Note: The behavior described in this topic applies only to version 1 of the FWSM. Subsequent versions will behave differently.



Packet Flow with MSFC as Connected Router on Inside

CSPFA 3.2—21-12

In the figure, some of the VLANs carrying traffic are assigned to the FWSM. Only the traffic on those VLANs is protected by firewalls. The arrows trace a connection originating from VLAN 201 (effectively inside) and destined for the outside interface. The following sequence of events occurs:

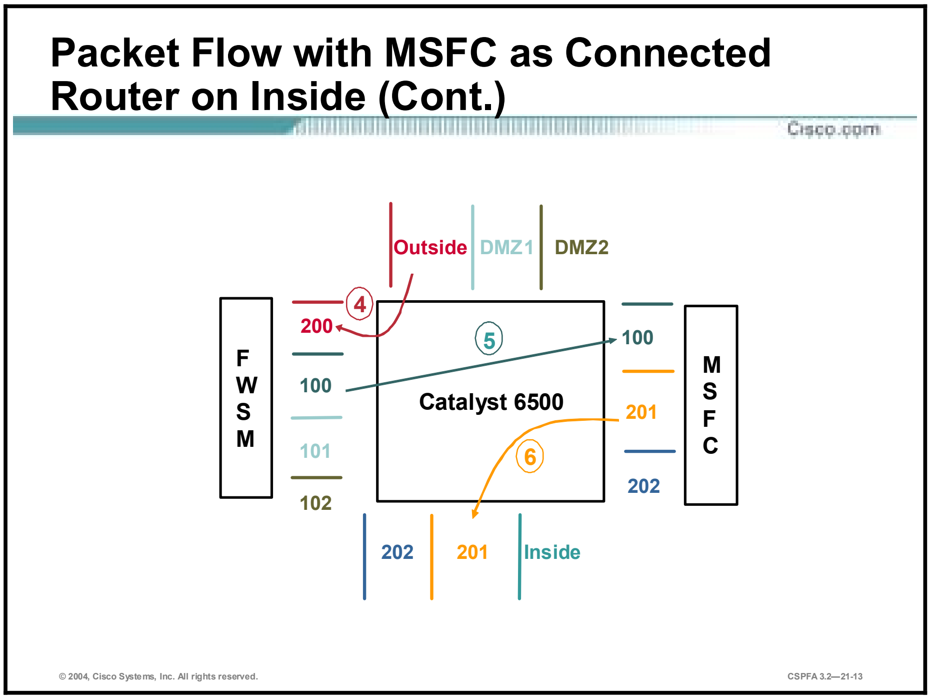
1. The packet from the inside interface (VLAN 201) is bridged to the MSFC interface.

2. The MSFC routes the packet to the firewall interface (VLAN 100).

Note: The MSFC should have the firewall interface as its next hop.

3. The firewall module rewrites the packet with the destination VLAN as 200. The packet from the firewall module is bridged to the outside interface.

Note: The MSFC can only be attached to one interface on the FWSM. This prevents you from accidentally configuring a bypass route around the FWSM.



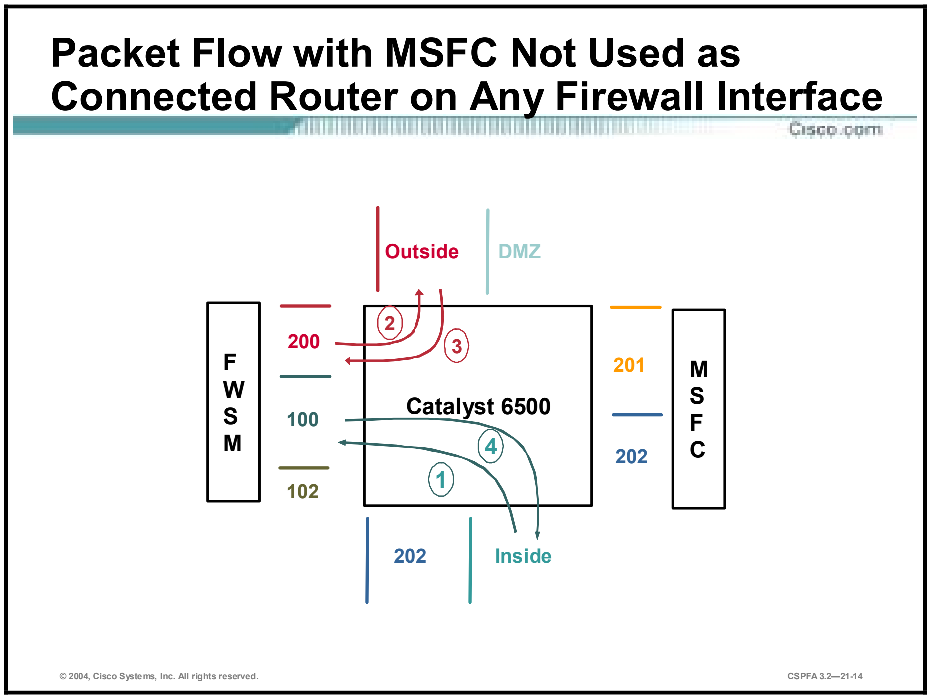
Packet Flow with MSFC as Connected Router on Inside (Cont.)

CSPFA 3.2—21-13

4. The return packet from the outside interface on VLAN 200 is bridged to the firewall interface.

5. The firewall rewrites the packet with the destination VLAN as 100. The packet from the FWSM is bridged to the MSFC interface.

6. The MSFC routes the packet back to the inside interface (VLAN 201).



Packet Flow with MSFC Not Used as Connected Router on Any Firewall Interface

Cisco.com

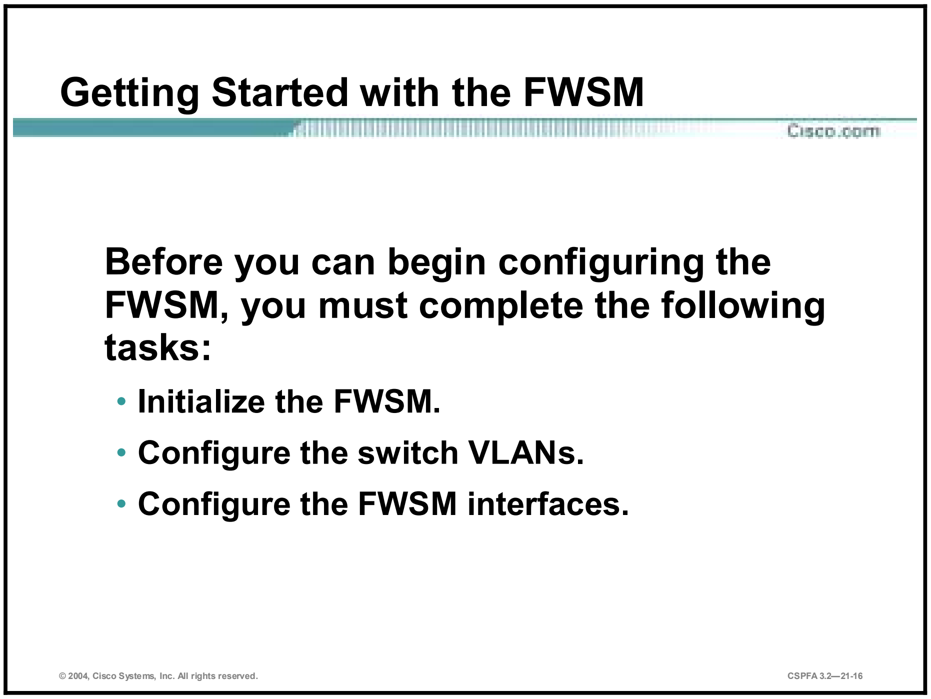
CSPFA 3.2—21-14

The figure explains the case in which the MSFC is not used as a connected router on any firewall interface. The arrows trace a connection originating from VLAN 100 (inside) and destined for the outside interface. The following sequence of events occurs:

1. The packet from the inside interface on VLAN 100 is bridged to the FWSM interface.
2. Depending on the firewall configuration, the FWSM rewrites the packet with the destination VLAN as 200. The packet from the firewall module is bridged to the outside interface.
3. The return packet from the outside interface on VLAN 200 is bridged to the FWSM interface.
4. Depending on the firewall configuration and earlier state maintained as a result of packet 1, the FWSM rewrites the packet with the destination VLAN as 100. The packet from the firewall module is bridged to the inside interface.

Getting Started

This topic explains how to prepare to configure the FWSM.



Getting Started with the FWSM

Cisco.com

Before you can begin configuring the FWSM, you must complete the following tasks:

- **Initialize the FWSM.**
- **Configure the switch VLANs.**
- **Configure the FWSM interfaces.**

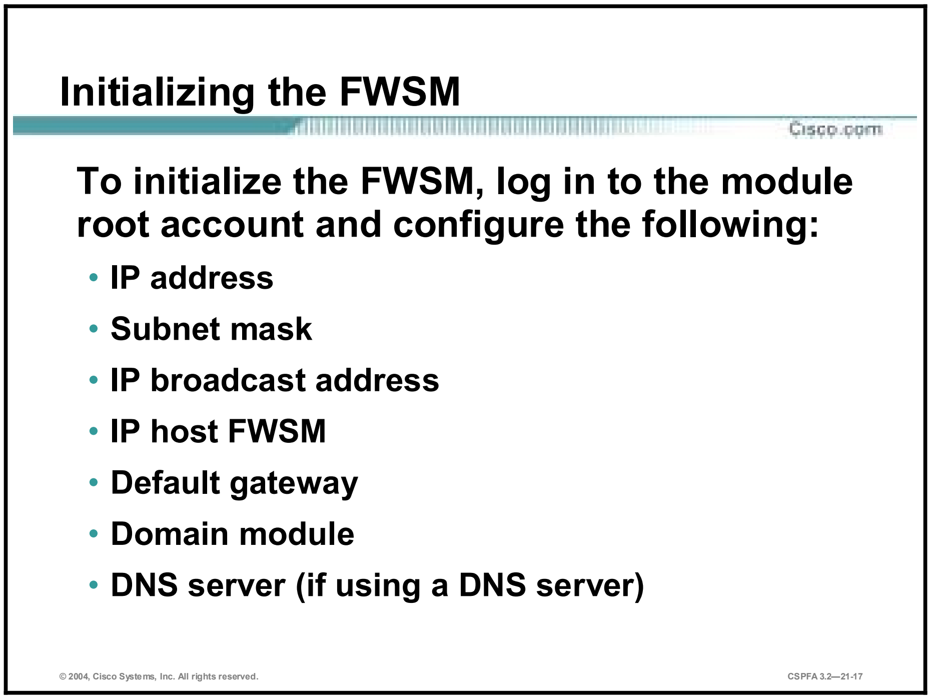
CSPFA 3.2—21-16

You can access the switch CLI through a Telnet connection to the switch or through the switch console interface. From the switch console, you can session into the FWSM to configure it.

Note: Session is a Telnet interface through the Ethernet out-of-band channel of the switch backplane.

Before you can begin configuring the FWSM, you must complete the following tasks:

- Initialize the FWSM.
- Configure the switch VLANs.
- Configure the FWSM interfaces.



Initializing the FWSM

To initialize the FWSM, log in to the module root account and configure the following:

- **IP address**
- **Subnet mask**
- **IP broadcast address**
- **IP host FWSM**
- **Default gateway**
- **Domain module**
- **DNS server (if using a DNS server)**

© 2004, Cisco Systems, Inc. All rights reserved.

CSPFA 3.2—21-17

Before you can use the FWSM, you must complete the following steps to initialize it:

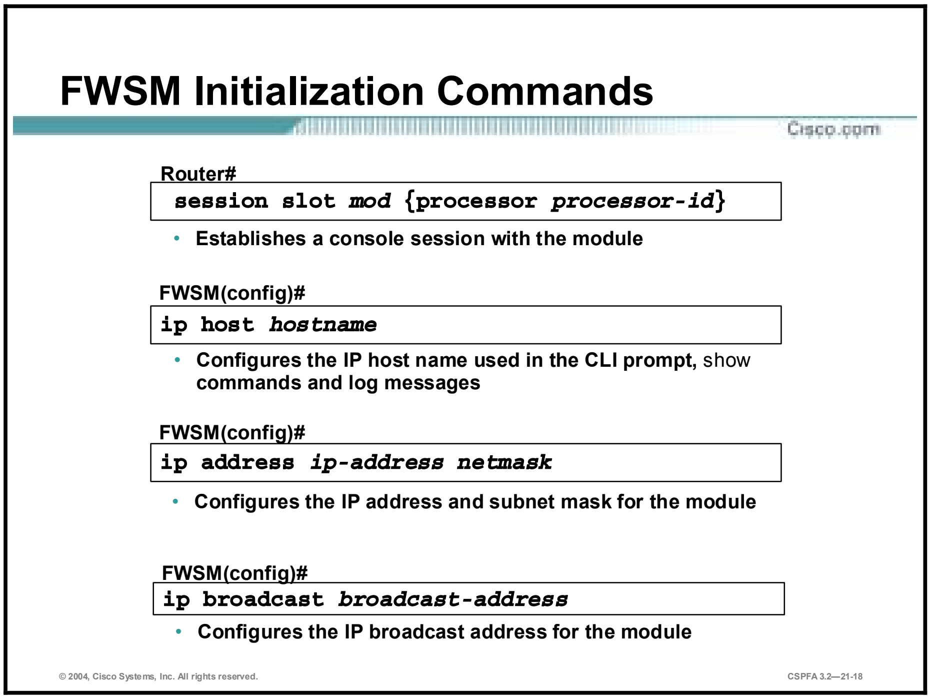
Step 1 Enter the **show module** command to verify that the system acknowledges the new module and has brought it online. The syntax for the **show module** command is as follows:

show module [*mod-num* | all]

mod-num	Number of the module and the port on the module.
all	Displays the information for all modules.

The following is an example of the output of the **show module** command:

```
Router# show module

Mod Slot Ports  Module-Type              Model              Sub
Status
                           --- ---- ----- ------------------------- --
----------------- --- --------
 1   1    2     1000BaseX Supervisor      WS-X6K-S2U-MSFC2   yes  ok
 15  1    1     Multilayer Switch Feature WS-F6K-MSFC2       no   ok
 2   2    6     Firewall Service Module   WS-SVC-FWM-1       no   ok
```



FWSM Initialization Commands

Router#

```
session slot mod {processor processor-id}
```

- Establishes a console session with the module

FWSM(config)#

```
ip host hostname
```

- Configures the IP host name used in the CLI prompt, show commands and log messages

FWSM(config)#

```
ip address ip-address netmask
```

- Configures the IP address and subnet mask for the module

FWSM(config)#

```
ip broadcast broadcast-address
```

- Configures the IP broadcast address for the module

CSPFA 3.2—21-18

Step 2 Use the **session slot** command to establish a console session with the module. The syntax for the **session slot** command is as follows:

session slot *mod* {**processor** *processor-id*}

mod	Slot number
processor *processor-id*	Processor ID

Step 3 At the login prompt, type **root** to log in to the root account.

Step 4 At the password prompt, type **root** as the root password.

Step 5 Use the **ip host** command to configure the IP host module used in the command line interface (CLI) prompt, **show** commands, and log messages. The syntax for the **ip host** command is as follows:

ip host ***hostname***

hostname	Module name to be used in CLI prompt, **show** commands, and log messages

Step 6 Use the **ip address** command to configure the IP address and subnet mask. The syntax for the **ip address** command is as follows:

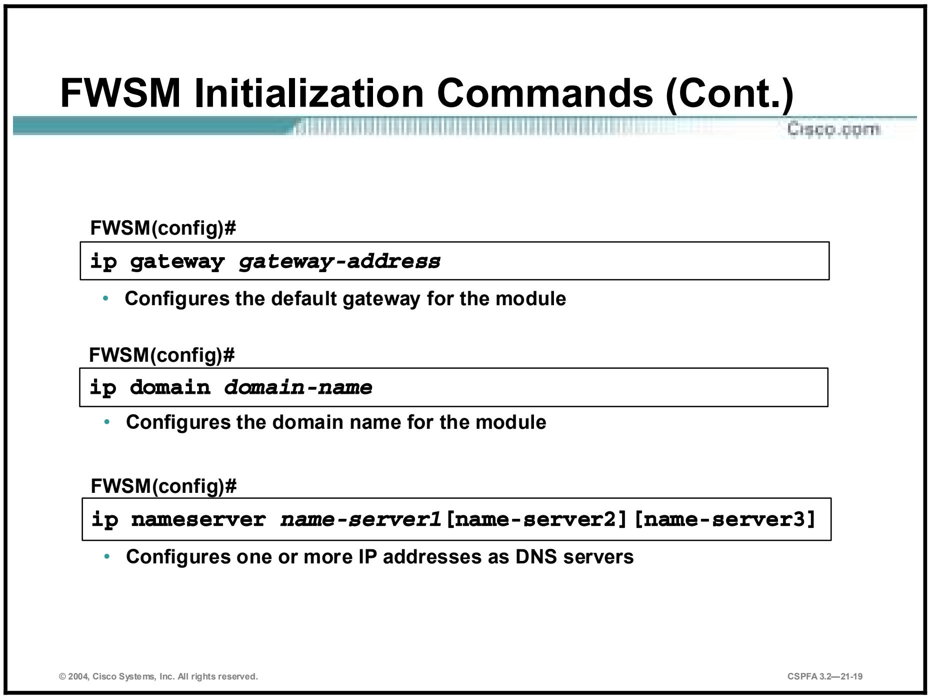
ip address ip-address netmask

Ip-address	IP address of the module
netmask	Netmask for ip-address

Step 7 Use the **ip broadcast** command to configure the IP broadcast address. The syntax for the **ip broadcast** command is as follows:

```
ip broadcast broadcast-address
```

broadcast-address	Broadcast address for network of ip-address



FWSM Initialization Commands (Cont.)

FWSM(config)#

```
ip gateway gateway-address
```

- Configures the default gateway for the module

FWSM(config)#

```
ip domain domain-name
```

- Configures the domain name for the module

FWSM(config)#

```
ip nameserver name-server1[name-server2][name-server3]
```

- Configures one or more IP addresses as DNS servers

CSPFA 3.2—21-19

Step 8 Use the **ip gateway** command to configure the default gateway. The syntax for the **ip gateway** command is as follows:

ip gateway ***gateway-address***

gateway-address	Gateway of last resort to be used by the module

Step 9 Use the **ip domain** command to configure the domain name for the module. The syntax for the **ip domain** command is as follows:

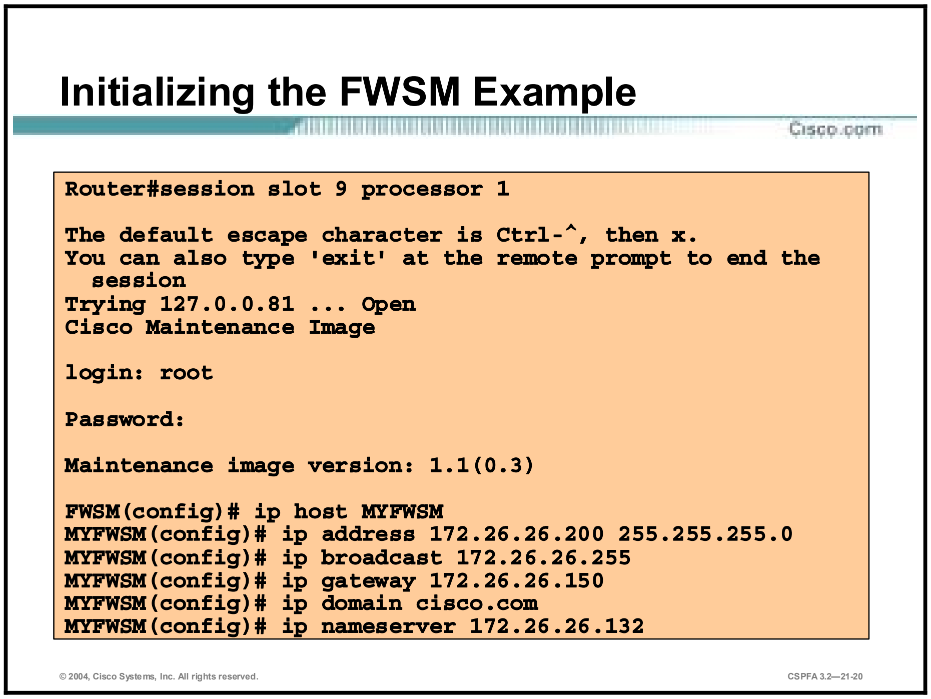
ip domain ***domain-name***

domain-name	Domain name for the module

Step 10 Use the **ip nameserver** command to configure one or more IP addresses as DNS name servers. The syntax for the **ip nameserver** command is as follows:

ip nameserver ***name-server1 [name-server2][name-server3]]***

name-server1	IP address of DNS server(s)
name-server2	IP address of second DNS server if using a second DNS server
name-server3	IP address of third DNS server, if using a third DNS server

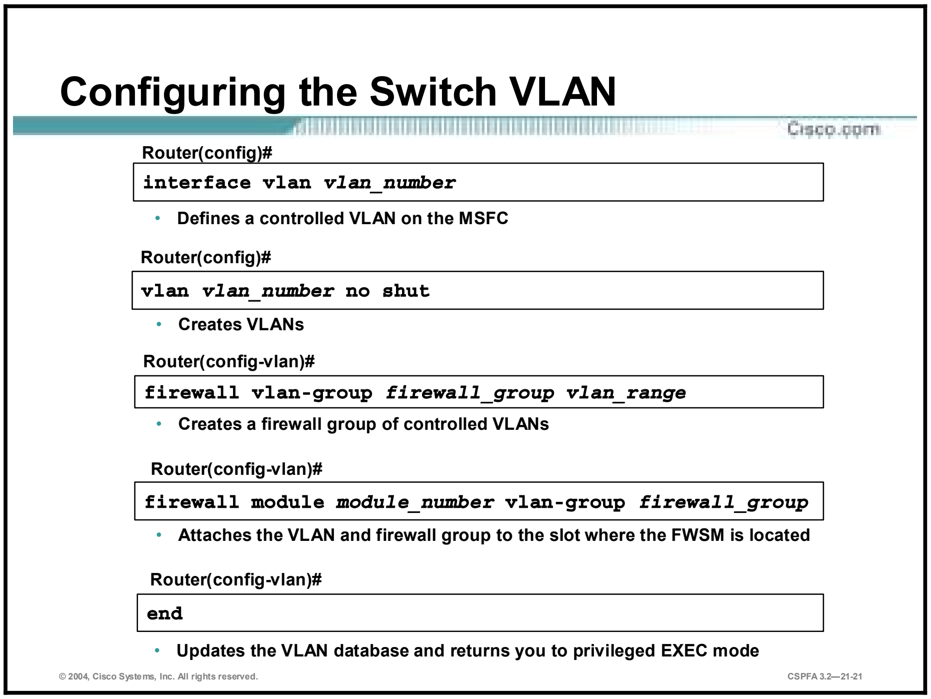


Initializing the FWSM Example

```
Router#session slot 9 processor 1

The default escape character is Ctrl-^, then x.
You can also type 'exit' at the remote prompt to end the
  session
Trying 127.0.0.81 ... Open
Cisco Maintenance Image

login: root

Password:

Maintenance image version: 1.1(0.3)

FWSM(config)# ip host MYFWSM
MYFWSM(config)# ip address 172.26.26.200 255.255.255.0
MYFWSM(config)# ip broadcast 172.26.26.255
MYFWSM(config)# ip gateway 172.26.26.150
MYFWSM(config)# ip domain cisco.com
MYFWSM(config)# ip nameserver 172.26.26.132
```

CSPFA 3.2—21-20

The figure shows an example of initializing a firewall module in slot 9 of a Catalyst 6500 switch.



Configuring the Switch VLAN

Router(config)#

```
interface vlan vlan_number
```

- Defines a controlled VLAN on the MSFC

Router(config)#

```
vlan vlan_number no shut
```

- Creates VLANs

Router(config-vlan)#

```
firewall vlan-group firewall_group vlan_range
```

- Creates a firewall group of controlled VLANs

Router(config-vlan)#

```
firewall module module_number vlan-group firewall_group
```

- Attaches the VLAN and firewall group to the slot where the FWSM is located

Router(config-vlan)#

```
end
```

- Updates the VLAN database and returns you to privileged EXEC mode

CSPFA 3.2—21-21

To prevent losing the switch configuration or the definition of a firewall interface becoming out of synchronization between the module and the route processor, you should first configure a VLAN on the route processor MSFC and then configure the VLANs for the module. VLAN IDs must be the same for the switch and the module. The route processor configuration for the module requires only a single VLAN for the firewall and a non-firewall VLAN. After the route processor VLAN is configured, the controlled VLAN is sent to the module. You can then configure the module firewall functions.

Note: Refer to the Catalyst 6000 Family IOS Software Configuration Guide for details.

To enable a single controlled VLAN as the router interface on the route processor, complete the following steps:

Step 1 Use the **interface vlan** command to define a controlled VLAN on the MSFC (route processor). The syntax of the **interface vlan** command is as follows:

interface vlan ***vlan_number***

vlan_number	Number of the VLAN

Step 2 Use the **vlan** command to create VLANs. The syntax for the **vlan** command is as follows:

vlan *vlan_number* **no shut**

vlan_number	Number of the VLAN

Step 3 Use the **firewall vlan-group** command to create a firewall group of controlled VLANs. The syntax for the **firewall vlan-group** command is as follows:

firewall vlan-group ***firewall_group vlan_range***

firewall_group	Name of firewall vlan group
vlan_range	Numerical range of VLAN numbers to be included in the group

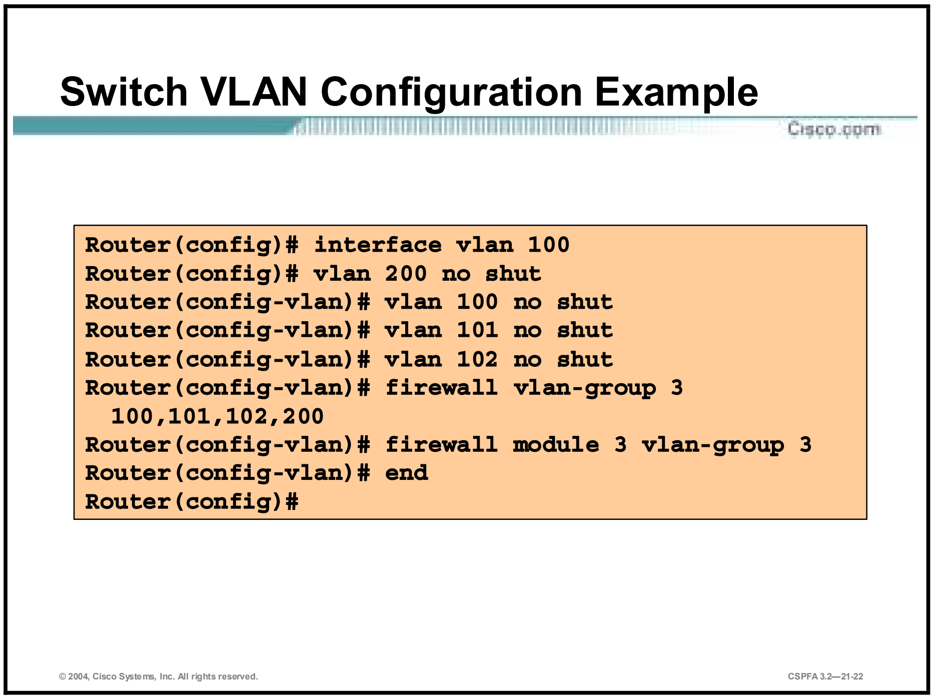
Step 4 Use the **firewall module** command to attach the VLAN and firewall group to the slot where the module is located. The syntax for the **firewall module** command is as follows:

firewall module ***module_number*** **vlan-group** ***firewall_group***

module_number	Number of the module
firewall_group	Name of firewall vlan group

Step 5 Use the **end** command to update the VLAN database and return to privileged EXEC mode. The syntax for the **end** command is as follows:

end

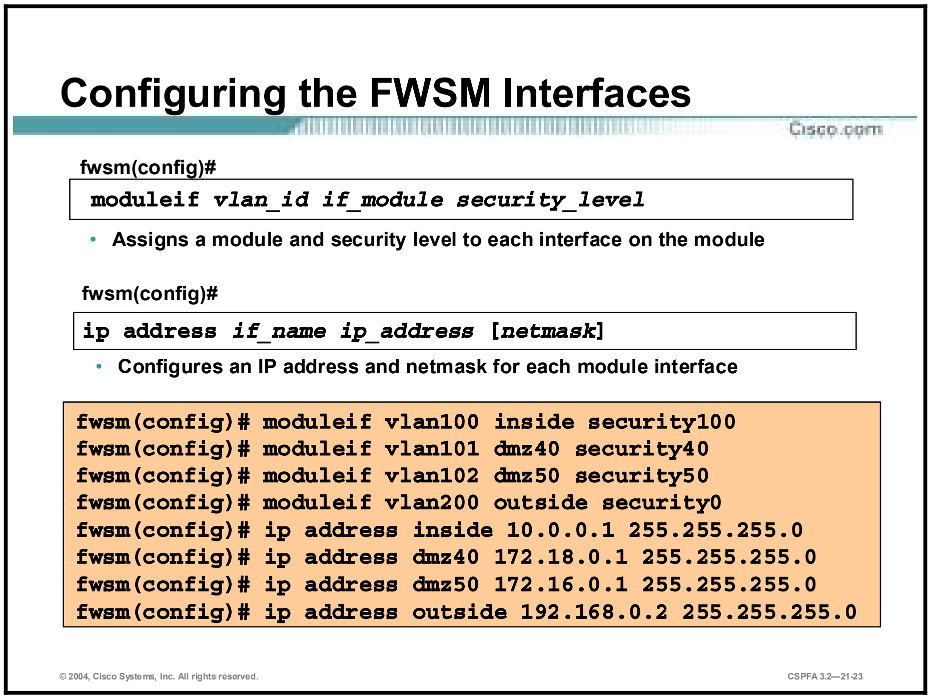


Switch VLAN Configuration Example

```
Router(config)# interface vlan 100
Router(config)# vlan 200 no shut
Router(config-vlan)# vlan 100 no shut
Router(config-vlan)# vlan 101 no shut
Router(config-vlan)# vlan 102 no shut
Router(config-vlan)# firewall vlan-group 3
  100,101,102,200
Router(config-vlan)# firewall module 3 vlan-group 3
Router(config-vlan)# end
Router(config)#
```

CSPFA 3.2—21-22

The figure shows how to enable a controlled VLAN in global configuration mode. VLAN 100 is defined as the single controlled VLAN on the MSFC. VLANs 200, 100, 101, and 102 are then created in the switch and assigned to firewall vlan group 3. Group 3 is attached to slot 3, the slot in which the FWSM is installed.



Configuring the FWSM Interfaces

fwsm(config)#

```
moduleif vlan_id if_module security_level
```

- Assigns a module and security level to each interface on the module

fwsm(config)#

```
ip address if_name ip_address [netmask]
```

- Configures an IP address and netmask for each module interface

```
fwsm(config)# moduleif vlan100 inside security100
fwsm(config)# moduleif vlan101 dmz40 security40
fwsm(config)# moduleif vlan102 dmz50 security50
fwsm(config)# moduleif vlan200 outside security0
fwsm(config)# ip address inside 10.0.0.1 255.255.255.0
fwsm(config)# ip address dmz40 172.18.0.1 255.255.255.0
fwsm(config)# ip address dmz50 172.16.0.1 255.255.255.0
fwsm(config)# ip address outside 192.168.0.2 255.255.255.0
```

CSPFA 3.2—21-23

Complete the following steps to configure the module interfaces:

Step 1 Use the **moduleif** command to assign a module and security level to each interface on the module. The syntax of the **moduleif** command is as follows:

moduleif *vlan_id if_module security_level*

Step 2 Use the **ip address** command to configure an IP address and netmask for each module interface. The syntax for the **ip address** command is as follows:

ip address *if_module ip_address* [*netmask*]

Once you have initialized the FWSM, configured the switch vlans, and configured the FWSM interfaces, you are ready to configure the FWSM to allow the desired traffic to protected networks. You will need to create ACLs to allow outbound as well as inbound traffic because the FWSM, unlike the PIX Firewall, denies all inbound and outbound connections that are not explicitly permitted by ACLs. Configuring your firewall policy in the FWSM is much like doing so in the PIX Firewall because the FWSM application software is similar to that of the PIX Firewall software.

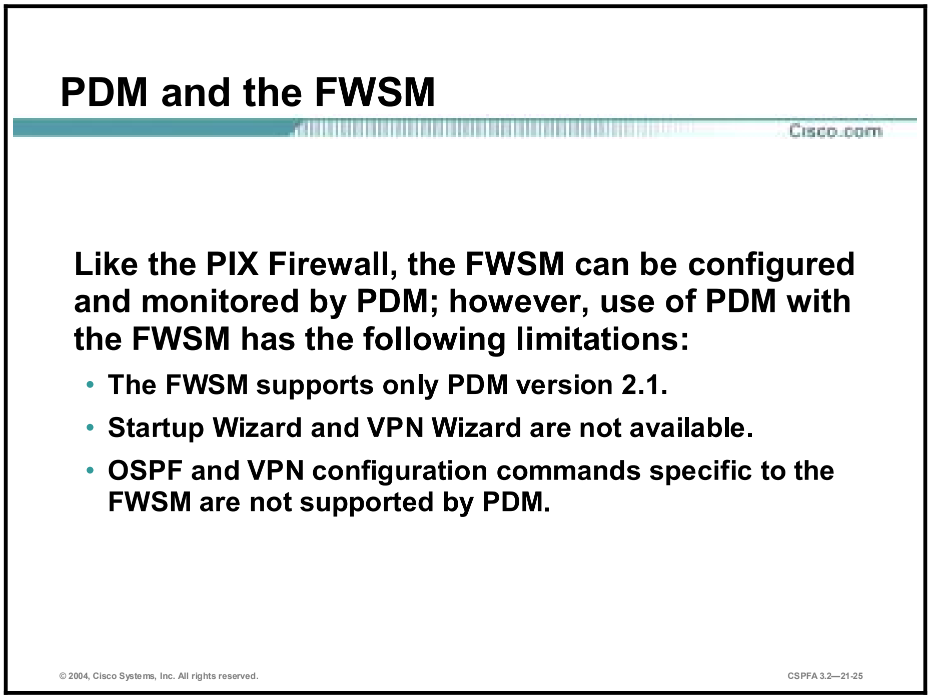
For a description of the PIX Firewall commands supported by the FWSM, go to the following URL: http://www.cisco.com/en/US/products/hw/switches/ps708/products_installation_and_configuration_guide_chapter09186a00800e3be0.html

This page contains the following:

- Commands that support the maintenance software
- Cisco IOS commands that support the FWSM
- New commands specific to the FWSM
- PIX Firewall commands that were changed for the FWSM
- PIX Firewall commands that are not used by the FWSM
- PIX Firewall commands used by the FWSM and their corresponding PIX Firewall software versions

Using PDM with the FWSM

This topic explains how to use PDM to configure and monitor the FWSM.



PDM and the FWSM

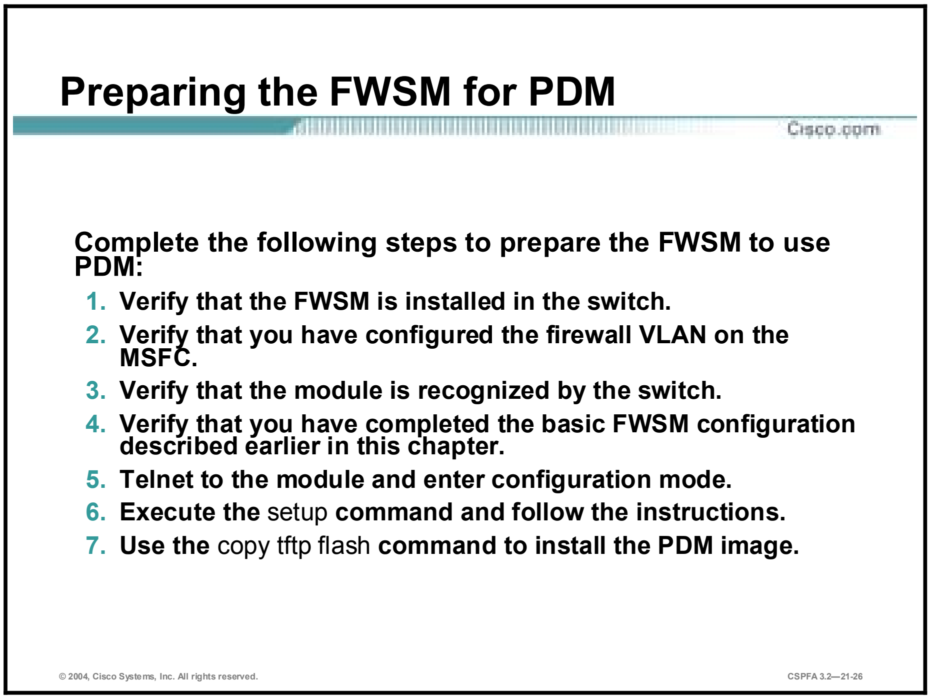
Cisco.com

Like the PIX Firewall, the FWSM can be configured and monitored by PDM; however, use of PDM with the FWSM has the following limitations:

- **The FWSM supports only PDM version 2.1.**
- **Startup Wizard and VPN Wizard are not available.**
- **OSPF and VPN configuration commands specific to the FWSM are not supported by PDM.**

CSPFA 3.2—21-25

PDM can be used to configure and monitor your FWSM. When running on the FWSM, PDM looks somewhat different because it does not have the Wizards menu or the VPN tab. Furthermore, the System Properties Interfaces table looks different. Running on the FWSM, the interfaces in the table are a combination of interfaces configured on the FWSM and the output from the **show vlan** command. PDM supports VLANs and Syslog rate limiting but does not support OSPF.



Preparing the FWSM for PDM

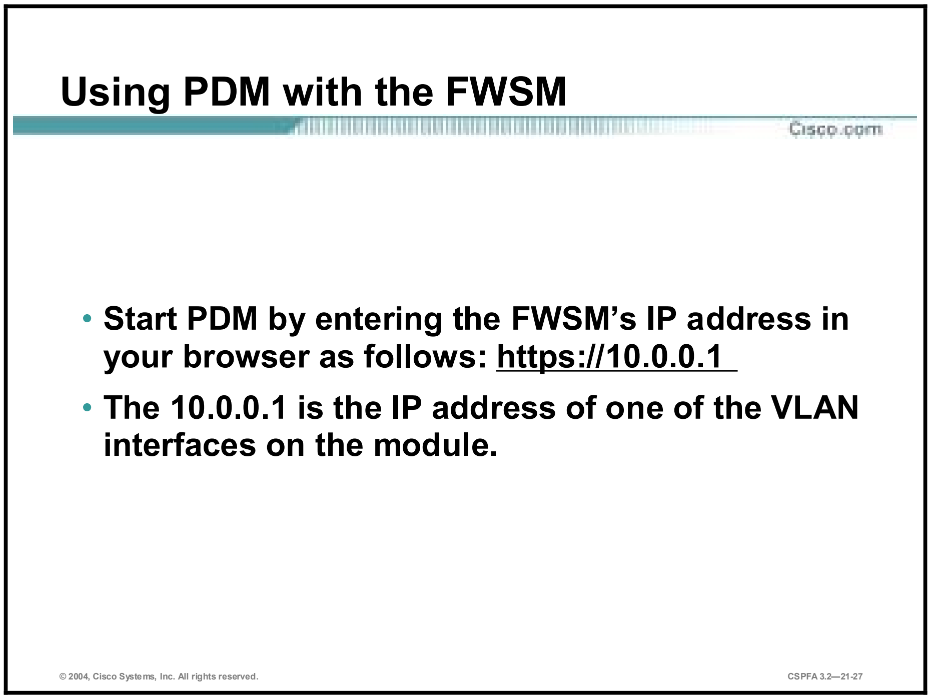
Cisco.com

Complete the following steps to prepare the FWSM to use PDM:

1. **Verify that the FWSM is installed in the switch.**
2. **Verify that you have configured the firewall VLAN on the MSFC.**
3. **Verify that the module is recognized by the switch.**
4. **Verify that you have completed the basic FWSM configuration described earlier in this chapter.**
5. **Telnet to the module and enter configuration mode.**
6. **Execute the** setup **command and follow the instructions.**
7. **Use the** copy tftp flash **command to install the PDM image.**

CSPFA 3.2—21-26

The figure shows the steps needed to prepare the FWSM to use PDM. Be sure you have initialized the FWSM before attempting to install PDM.



Using PDM with the FWSM

Cisco.com

- Start PDM by entering the FWSM’s IP address in your browser as follows: https://10.0.0.1
- The 10.0.0.1 is the IP address of one of the VLAN interfaces on the module.

CSPFA 3.2—21-27

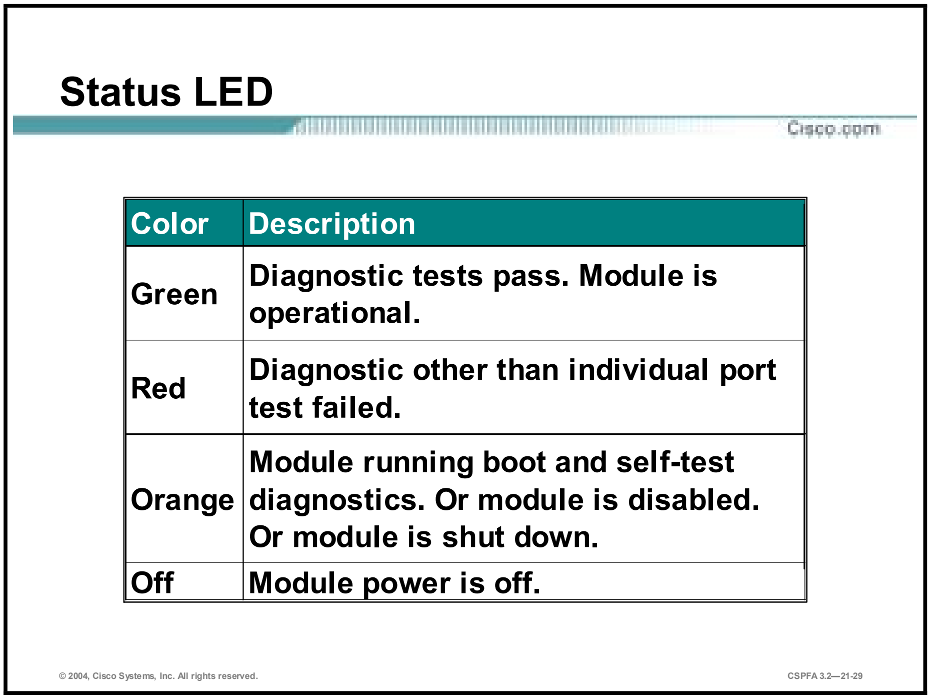
To start using PDM to configure the FWSM, use the HTTP secure (https) command and enter the following address:

https://IP address of FWSM

The IP address is the address of one of the VLAN interfaces on the FWSM.

Troubleshooting the FWSM

This topic provides troubleshooting information that can be used to determine the possible causes for the Catalyst 6000 FWSM not functioning properly.



Status LED

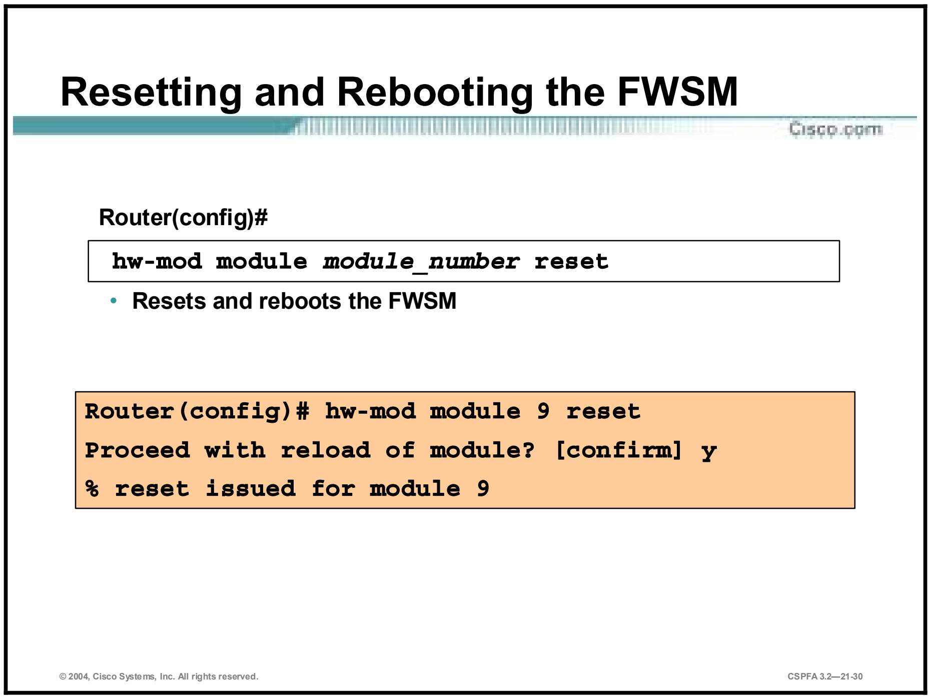
Cisco.com

Color	Description
Green	Diagnostic tests pass. Module is operational.
Red	Diagnostic other than individual port test failed.
Orange	Module running boot and self-test diagnostics. Or module is disabled. Or module is shut down.
Off	Module power is off.

CSPFA 3.2—21-29

The status LED is a quick method to determine the state of the FWSM. The status LED is located in the left corner of the module. LED status colors are described in the following table:

Status color	Description
Green	FWSM is operational.
Red	Diagnostic other than an individual port test failed.
Orange	One of the following conditions exists: ■ FWSM is running boot and self-test diagnostics. ■ FWSM is disabled. ■ FWSM is shut down.
Off	FWSM power is off.



Resetting and Rebooting the FWSM

Router(config)#

```
hw-mod module module_number reset
```

- Resets and reboots the FWSM

```
Router(config)# hw-mod module 9 reset
Proceed with reload of module? [confirm] y
% reset issued for module 9
```

© 2004, Cisco Systems, Inc. All rights reserved.

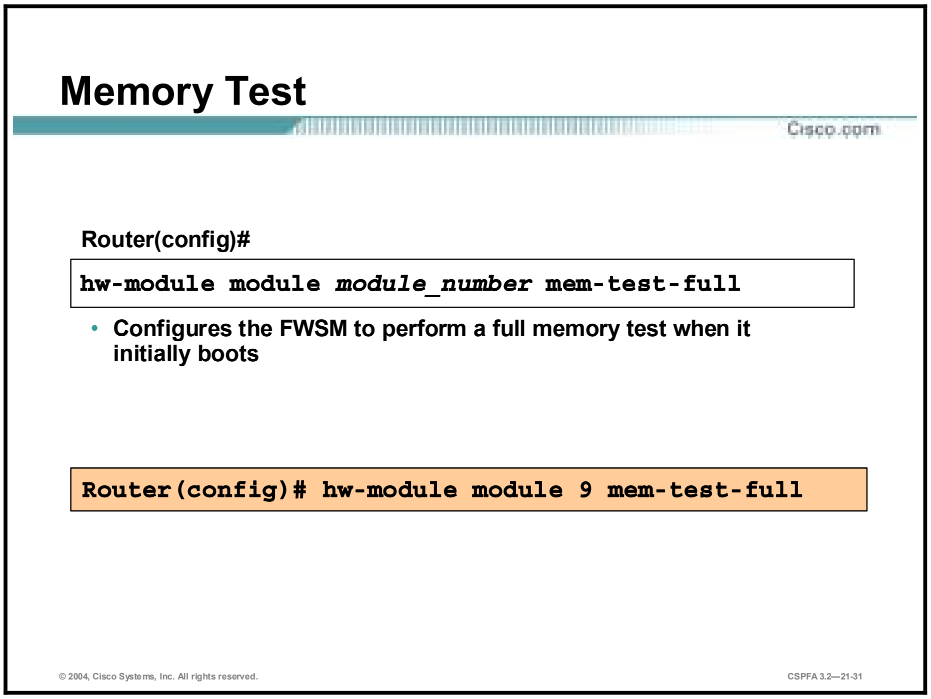
CSPFA 3.2—21-30

If you cannot reach the module through the CLI or an external Telnet session, enter the **hw-mod module module_number reset** command to reset and reboot the module. The reset process requires several minutes. The syntax for the command is as follows:

hw-module module *module_number* reset

module_number	Number of module you wish to reset

The figure shows how to reset the module, installed in slot 9, from the CLI.



Memory Test

Cisco.com

Router(config)#

```
hw-module module module_number mem-test-full
```

- Configures the FWSM to perform a full memory test when it initially boots

```
Router(config)# hw-module module 9 mem-test-full
```

CSPFA 3.2—21-31

When the FWSM initially boots, by default it runs a partial memory test. To perform a full memory test, use the **hw-module module *module_number* mem-test-full** command. The syntax of the command is as follows:

hw-module module *module_number* mem-test-full

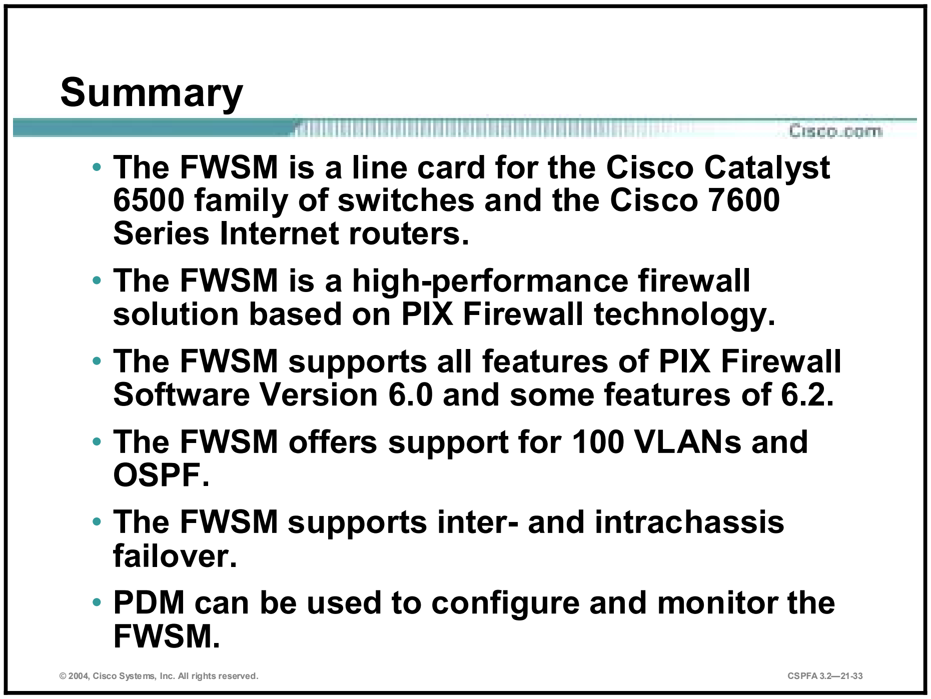
module_number	Number of module

A full memory test takes more time to complete than a partial memory test depending on the memory size. The table lists the memory and approximate boot time for a long memory test.

Memory size	Boot time
512MB	3 minutes
1GB	6 minutes

Summary

This topic summarizes what you have learned in this lesson.



Summary

Cisco.com

- The FWSM is a line card for the Cisco Catalyst 6500 family of switches and the Cisco 7600 Series Internet routers.
- The FWSM is a high-performance firewall solution based on PIX Firewall technology.
- The FWSM supports all features of PIX Firewall Software Version 6.0 and some features of 6.2.
- The FWSM offers support for 100 VLANs and OSPF.
- The FWSM supports inter- and intrachassis failover.
- PDM can be used to configure and monitor the FWSM.

CSPFA 3.2—21-33