Designing Network Security

March 1999

Welcome to Cisco Press

Welcome to the **employee only** Cisco Press web site. The above "Welcome" page link presents a FAQ sheet for Cisco Press, including information about how you can buy Cisco Press books!.

New information on the Cisco Press Marketing Incentive Plan is also now available.

As source material becomes available from the publisher, the complete text of each Cisco Press publication will be presented here for use by Cisco employees. Sample chapters are presented at the public site hosted by Cisco.

Design and Implementation

Publications focusing on network design and implementation strategies.

Internet Routing Architectures
ISBN: 1-56205-652-2
By Bassam Halabi
Explores the ins and outs of interdomain routing network designs.

Designing Campus Networks
ISBN: 1-57870-030-2
By Terri Quinn-Andry and Kitty Haller
Focuses on designing scalable networks supporting campus LAN traffic.

OSPF Network Design Solutions
ISBN: 1-57870-046-9
By Thomas M. Thomas II
Presents detailed, applied coverage of Open Shortest Path First protocol.

Internetworking SNA with Cisco Routers
ISBN: 1-57870-083-3
By George Sackett and Nancy Sackett
Provides comprehesive coverage of terms, architectures, protocols, and implementations for internetworking SNA. **Content not available.**

Residential Broadband
ISBN: 1-57870-020-5
By George Abe
Presents emerging high-bandwidth access network issues.

Cisco Router Configuration
ISBN: 1-57870-022-1
By Allan Leinwand and Bruce Pinsky
Presents router deployment tips from long-time Cisco experts.

Top-Down Network Design
ISBN: 1-57870-069-8
By Priscilla Oppenheimer
Learn a network design methodology based on standard techniques for structured systems analysis.

Cisco Career Certification and Training

Publications developed in cooperation with Cisco Worldwide Training that support Cisco's Career Certification and customer training initiatives.

Introduction to Cisco Router Configuration (ICRC)
ISBN: 1-57870-076-0
Edited by Laura Chappell
Based on the Cisco course, presents readers with the concepts and commands required to configure Cisco routers. **Content not available.**

Advanced Cisco Router Configuration (ACRC)
ISBN: 1-57870-074-4
Edited by Laura Chappell
Advanced guide focuses on scalable operation in large and/or growing multiprotocol internetworks.

Cisco CCNA Preparation Library
ISBN: 1-57870-125-2
By Cisco Systems, Inc.
Bundle includes two publications: *Introduction to Cisco Router Configuration* and *Internetworking Technologies Handbook, Second Edition* (plus *High-Performance Solutions for Desktop Connectivity* in CD-ROM format). **Content not available.**

Cisco Certified Internetwork Expert (CCIE) Professional Development Series

Publications supporting Cisco's CCIE program.

Cisco CCIE Fundamentals: Network Design and Case Studies
ISBN: 1-57870-066-3
By Cisco Staff
Network design fundamentals and case examples assembled to help prepare CCIE candidates.

CCIE Professional Development: Routing TCP/IP
ISBN: 1-57870-041-8
By Jeff Doyle
Covers basics through details of each IP routing protocol. Essential reading! **Content not available.**

Networking Fundamentals

Support publications providing technology and configuration basics.

Internetworking Technologies Handbook (2nd Edition)
ISBN: 1-56205-102-8
By Cisco Staff and Kevin Downes
Survey of technologies and protocols.

IP Routing Primer
ISBN: 1-57870-108-2
By Robert Wright
Technical tips and hints focusing on how Cisco routers implement IP functions.

Internetworking Troubleshooting Handbook
ISBN: 1-56205-024-8
By Cisco Staff and Kevin Downes
Summarizes connectivity and performance problems, helps develop a strategy for isolating problems. **Content not available.**

IP Routing Fundamentals
ISBN: 1-57870-071-X
By Mark Sportack
Provides a detailed examination of routers and the common IP routing protocols.

Cisco Documentation from Cisco Press

A number of Cisco IOS cross-platform software publications have been ported to a retail format by Cisco Press. Cisco Press is selling these documents via retail channels as a courtesy to simplify access for Cisco customers. All these documents, whether sold as Cisco product documents or as the Cisco Press publications, are available in electronic form via Cisco's free web-based,documentation site.

To find publications offered by Cisco Press, please refer to the catalog of publications presented at the Cisco Press page hosted by Macmillan:

- **Complete Cisco Press Publication Catalog**

The links below direct you to the documents presented within the official Cisco documentation environment (and out of the Cisco Press web area).

- **Cisco IOS Software Release 11.3 Documentation**
- **Cisco IOS Software Release 12.0 Documentation**

Cisco Press Internal

- Designing Network Security
  Cisco Press title
- Developing IP Multicast Networks

HOME CONTENTS PREVIOUS NEXT GLOSSARY FEEDBACK SEARCH HELP

Cisco Press Publications

Developing IP Multicast Networks

Internetworking Terms and Acronyms

- Z
- ITA New Terms October 2000

Cisco Press Search

Enter your query here:

Search Reset

Search Cisco Connection Online

Search Help

HOME CONTENTS PREVIOUS NEXT GLOSSARY SEARCH HELP

Cisco Press Help

- User Interface Overview

  Basic notes about the Cisco Press site user interface.
- Searching Cisco Press

  Instructions regarding use of the multi-document search feature provided with this product.

HOME CONTENTS PREVIOUS NEXT GLOSSARY SEARCH HELP

Table of Contents

Port Numbers

This appendix lists the assigned port numbers from the Internet Assigned Numbers Authority (IANA). For a more complete list, go to http://www.isi.edu/in-notes/iana/assignments/port-numbers.

The port numbers are divided into three ranges, which are described in Table C-1:

- The *well-known ports* are those in the range 0 through 1023.
- The *registered ports* are those in the range 1024 through 49151.
- The *dynamic* or *private ports* are those in the range 49152 through 65535.

Table C-1: Port Assignments

Keyword	Decimal	Description
ssh	22/tcp	SSH Remote Login Protocol
ssh	22/udp	SSH Remote Login Protocol
tacacs	49/tcp	Login Host Protocol (TACACS)
tacacs	49/udp	Login Host Protocol (TACACS)
domain	53/tcp	Domain Name Server

domain	53/udp	Domain Name Server
tacacs-ds	65/tcp	TACACS-Database Service
tacacs-ds	65/udp	TACACS-Database Service
kerberos	88/tcp	Kerberos
kerberos	88/udp	Kerberos
https	443/tcp	HTTP protocol over TLS/SSL
https	443/udp	HTTP protocol over TLS/SSL
smtps	465/tcp	SMTP protocol over TLS/SSL (was ssmtp)
smtps	465/udp	SMTP protocol over TLS/SSL (was ssmtp)
isakmp	500/tcp	ISAKMP protocol
isakmp	500/udp	ISAKMP protocol
nntps	563/tcp	NNTP protocol over TLS/SSL (was snntp)
nntps	563/udp	NNTP protocol over TLS/SSL (was snntp)
sshell	614/tcp	SSL shell
sshell	614/udp	SSL shell
kerberos-adm	749/tcp	Kerberos administration
kerberos-adm	749/udp	Kerberos administration
kerberos-iv	750/udp	Kerberos Version 4
ftps-data	989/tcp	FTP protocol, data, over TLS/SSL
ftps-data	989/udp	FTP protocol, data, over TLS/SSL

ftps	990/tcp	FTP protocol, control, over TLS/SSL
ftps	990/udp	FTP protocol, control, over TLS/SSL
telnets	992/tcp	Telnet protocol over TLS/SSL
telnets	992/udp	Telnet protocol over TLS/SSL
imaps	993/tcp	IMAP4 protocol over TLS/SSL
imaps	993/udp	IMAP4 protocol over TLS/SSL
ircs	994/tcp	IRC protocol over TLS/SSL
ircs	994/udp	IRC protocol over TLS/SSL
pop3s	995/tcp	POP3 protocol over TLS/SSL (was spop3)
pop3s	995/udp	POP3 protocol over TLS/SSL (was spop3)
socks	1080/tcp	SOCKS
socks	1080/udp	SOCKS
pptp	1723/tcp	PPTP
pptp	1723/udp	PPTP
radius	1812/tcp	RADIUS
radius	1812/udp	RADIUS
radius-acct	1813/tcp	RADIUS Accounting
radius-acct	1813/udp	RADIUS Accounting
http-alt	8080/tcp	HTTP Alternate (see port 80)

http-alt	8080/udp	HTTP Alternate (see port 80)

continues

HOME CONTENTS PREVIOUS NEXT GLOSSARY FEEDBACK SEARCH HELP

Posted: Wed Jun 14 11:28:58 PDT 2000

Table of Contents

Security Technologies

2

Security Technologies

A wide range of security technologies exists that provide solutions for securing network access and data transport mechanisms within the corporate network infrastructure. Many of the technologies overlap in solving problems that relate to ensuring user or device identity, data integrity, and data confidentiality.

Note Throughout this book, *authentication*, *authorization*, and *access control* are incorporated into the concept of *identity*. Although these concepts are distinct, they all pertain to each individual user of the network---be it a person or device. Each person or device is a distinct entity that has separate abilities within the network and is allowed access to resources
based on who they are. Although in the purest sense, identity really pertains only to authentication, in many cases, it makes sense to discuss the entities' authorization and access control at the same time.

Authentication is the process of validating the claimed identity of an end user or a device (such as clients, servers, switches, routers, firewalls, and so on). *Authorization* is the process of granting access rights to a user, groups of users, or specified system; *access control* is limiting the flow of information from the resources of a system to only the authorized persons or systems in the network. In most of the cases we will study, authorization and access control are subsequent to successful authentication.

This chapter describes security technologies commonly used for establishing identity (authentication, authorization, and access control) as well as for ensuring some degree of data integrity and confidentiality in a network. Data *integrity* ensures that the data has not been altered or destroyed except by people who are explicitly intended to modify it; data *confidentiality* ensures that only the entities allowed to see the data see it in a usable format.

The intent is to develop a basic understanding of how these technologies can be implemented in corporate networks and to identify their strengths and weaknesses. The following categories have been selected in an attempt to group the protocols according to shared attributes:

- Identity technologies
- Security in TCP/IP structured layers
- Virtual private dial-up security technologies
- Public Key Infrastructure and distribution models

Note Many of the technologies discussed here either have been, or are in the process of being standardized by the IETF. For information on more technical details and latest developments, refer to Appendix A, "Sources of Technical Information." This appendix contains pointers to the IETF working groups that produce the RFCs and drafts relating to the technologies discussed here.

Identity Technologies

This section describes the primary technologies used to establish identity for a host, an end-user, or both. Authentication is an extremely critical element because everything is based on who you are. In many corporate networks, you would not grant authorized access to specific parts of the network before establishing who is trying to gain access to restricted resources. How foolproof the authentication method is depends on the technology used.

We can loosely categorize authentication methods as those where there is *local control* and those where you provide authentication verification through a *trusted third party*.

One of the potential weaknesses in some authentication methods is who you trust. Many authentication methods rely on a third party to verify someone's identity. The strength of this verification is the limiting factor in the strength of the authentication. When using a third party to authenticate an end user or device, ask yourself, "What is the likelihood that the third party I'm counting on to provide the authentication verification has been compromised?"

The technologies discussed in this section include variants of secure passwords, which provide varying degrees of security and are offered by most vendors today. Many protocols will authorize some form of connection setup after authentication is successfully verified. In dial-up environments, a peer-to-peer link

level connection is established; sometimes, additional access control mechanisms can be employed at higher levels of the protocol stack, such as permitting access to hosts with certain IP addresses accessing specific applications. We will look at different protocols that often use an initial authentication process to then grant authorization and access control.

Note Digital certificates can be used as an authentication method, as discussed in detail in "Public Key Infrastructure and Distribution Models," later in this chapter.

Secure Passwords

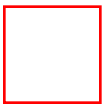
Although passwords are often used as proof for authenticating a user or device, passwords can easily be compromised if they are easy to guess, if they are not changed often enough, and if they are transmitted in cleartext across a network. To make passwords more secure, more robust methods are offered by encrypting the password or by modifying the encryption so that the encrypted value changes each time. This is the case with most one-time password schemes; the most common being the S/Key protocol and the token password authentication schemes.

S/Key Password Protocol

The *S/Key One-Time Password System*, released by Bellcore and defined in RFC 1760, is a one-time password generation scheme based on MD4 and MD5. The S/Key protocol is designed to counter a replay attack when a user is attempting to log in to a system. A replay attack in the context of login is when someone eavesdrops on a network connection to get the login ID and password of a legitimate user and later uses it to gain access to the network.

The operation of the S/Key protocol is client/server based: the client is typically a PC, and the server is some flavor of UNIX. Initially, both the client and the server must be configured with the same pass phrase and an iteration count. The *iteration count* specifies how many times a given input will be applied to the hash function. The client initiates the S/Key exchange by sending an initialization packet; the server responds with a sequence number and seed, as shown in Figure 2-1.

Figure 2-1: The Initial S/Key Exchange



The client then computes the one-time password, a process that involves three distinct steps: a preparatory step, a generation step, and an output function (see Figure 2-2).

Figure 2-2: Computing the S/Key One-Time Password

1. In the **preparatory step**, the client enters a secret pass phrase. This pass phrase is concatenated with the seed that was transmitted from the server in cleartext.

2. The **generation step** applies the secure hash function multiple times, producing a 64-bit final output.

3. The **output function** takes the 64-bit one-time password and displays it in readable form.

The last phase is for the client to pass the one-time password to the server, where it can be verified (see Figure 2-3).

Figure 2-3: Verifying the S/Key Password

The server has a file (on the UNIX reference implementation, it is /etc/skeykeys) containing, for each user, the one-time password from the last successful login. To verify an authentication attempt, the authentication server passes the received one-time password through the secure hash function once. If the result of this operation matches the stored previous one-time password, the authentication is successful and the accepted one-time password is stored for future use.

Because the number of hash function applications executed by the client decreases by one each time, this ensures a unique sequence of generated passwords. However, at some point, the user must reinitialize the system to avoid being unable to log in again. The system is reinitialized using the keyinit command, which allows the changing of the secret pass phrase, the iteration count, and the seed.

When computing the S/Key password on the client side, the client pass phrase can be of any length---more than eight characters is recommended. The use of the non-secret seed allows a client to use the same secret pass phrase on multiple machines (using different seeds) and to safely recycle secret pass phrases by changing the seed.

Note Many implementations require the generated one-time password to be entered either using a cut-and-paste approach, or manually. In manual entry scenarios, the one-time password is converted to, and accepted, as a sequence of six short (one- to four-letter) English words. Each word is chosen from a dictionary of 2,048 words; at 11 bits per word, all one-time passwords may be encoded. Interoperability requires that all S/Key system hosts and calculators use the same dictionary.

S/Key is an alternative to simple passwords. Free as well as commercial implementations are widely available.

Token Password Authentication Schemes

Token authentication systems generally require the use of a special card (called a *smart card* or *token card*), although some implementations are done using software to alleviate the problem of losing the smart card or token card. These types of authentication mechanisms are based on one of two alternative schemes: *challenge-response* and *time-synchronous authentication*.

The challenge-response approach is shown in Figure 2-4. The following steps carry out the authentication exchange:

Step 1 The user dials into an authentication server, which then issues a prompt for a user ID.

Step 2 The user provides the ID to the server, which then issues a *challenge*---a random number that appears on the user's screen.

Step 3 The user enters that challenge number into the token or smart card, a credit-card-like device, which then encrypts the challenge with the user's encryption key and displays a response.

Step 4 The user types this response and sends it to the authentication server. While the user is obtaining a response from the token, the authentication server calculates what the appropriate response should be based on its database of user keys.

Step 5 When the server receives the user's response, it compares that response with the one it has calculated.

If the two responses match, the user is granted access to the network. If they don't match, access is denied.

Figure 2-4: Challenge-Response Token Authentication

The time-synchronous authentication scheme is shown in Figure 2-5. In this scheme, a proprietary algorithm executes in the token and on the server to generate identical numbers that change over time. The user dials into the authentication server, which issues a prompt for an access code. The user enters a personal identification number (PIN) on the token card, resulting in digits displayed at that moment on the token. These digits represent the one-time password and are sent to the server. The server compares this entry with the sequence it generated; if they match, it grants the user access to the network.

Figure 2-5: Time-Synchronous Token Authentication

Use of either the challenge-response or time-synchronous token password authentication scheme generally requires the user to carry a credit-card-like device to provide authentication credentials. This can be a burden to some users because they have to remember to carry the device, but it has the flexibility to allow fairly secure authenticated access from anywhere in the world. It is extremely useful for mobile users who frequently log in from remote sites. If the mobile users have their own laptop, the token can be installed as software, which relieves the burden of remembering to carry an additional device. These schemes are very robust and scalable from a centralized database point of view.

Note Using the one-time password scheme only protects you from replay attacks when initially logging in to the site. If you then continue to log in to other machines at the campus site, the password will be sent in the clear. It is best to combine one-time password use with some form of confidentiality (encryption) technique if protection is required for more than just the initial login sequence.

PPP Authentication Protocols

Passwords are incorporated into many protocols that provide authentication services. For dial-in connections, the Point-to-Point Protocol (PPP) is most often used to establish a dial-in connection over serial lines or ISDN. PPP authentication mechanisms include the Password Authentication Protocol (PAP), the Challenge Handshake Protocol (CHAP), and the Extensible Authentication Protocol (EAP). In all these cases, the peer device is being authenticated rather than the user of the device.

The PPP Protocol

PPP is a standardized Internet encapsulation of IP over point-to-point links. PPP addresses issues including assignment and management of IP addresses, asynchronous (start/stop) and bit-oriented synchronous encapsulation, network protocol multiplexing, link configuration, link quality testing, error detection, and option negotiation for such capabilities as network-layer address negotiation and data compression negotiation. PPP addresses these issues by providing an extensible Link Control Protocol (LCP) and a family of Network Control Protocols (NCPs) to negotiate optional configuration parameters and facilities. After the link has been established, PPP provides for an optional authentication phase before proceeding to the network-layer protocol phase.

PPP Link Layer

The PPP PDU uses the HDLC frame as stipulated in ISO 3309-1979 (and amended by ISO 3309-1984/PDAD1).

The PPP frame format is shown in Figure 2-6. The fields of a PPP frame are as follows:

Field	Description
Flag	A single byte that indicates the beginning or end of a frame. The flag field consists of the binary sequence 01111110.
Address	A single byte that contains the binary sequence 11111111, the standard broadcast address. PPP does not assign individual station addresses.
Control	A single byte that contains the binary sequence 00000011, which calls for transmission of user data in an unsequenced frame.
Protocol	Two bytes that identify the protocol encapsulated in the information field of the frame. The most up-to-date values of the protocol field are specified in the most recent Assigned Numbers Request for Comments (RFC).
Data	Zero or more bytes that contain the datagram for the protocol specified in the protocol field. The end of the information field is found by locating the closing flag sequence and allowing two bytes for the FCS field. The default maximum length of the information field is 1,500 bytes. By prior agreement, consenting PPP implementations can use other values for the maximum information field length.
Frame Check Sequence (FCS)	Normally two bytes. By prior agreement, consenting PPP implementations can use a 4-byte FCS for improved error detection.

The LCP can negotiate modifications to the standard PPP frame structure. However, modified frames will always be clearly distinguishable from standard frames.

Figure 2-6: The PPP Frame Format

PPP Negotiations

PPP negotiation consists of LCP and NCP negotiation. *LCP* is responsible for establishing the connection with certain negotiated options, maintaining the connection, and providing procedures to terminate the connection. To perform these functions, LCP is organized into the following four phases:

1. Link establishment and configuration negotiation
2. Link quality determination
3. Network layer protocol configuration negotiation
4. Link termination

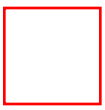
To establish communications over a point-to-point link, each end of the PPP link must first send LCP packets to configure the data link during the link establishment phase. After the link has been established, PPP provides for an optional authentication phase before proceeding to the network layer protocol phase. The NCP phase then establishes and configures different network layer protocols such as IP.

By default, authentication before the NCP phase is not mandatory. If authentication of the link is desired, an implementation will specify the authentication protocol configuration option during the link establishment phase. These authentication protocols are intended for use primarily by hosts and routers that connect to a PPP network server through switched circuits or dial-up lines, but can be applied to dedicated links as well. The server can use the identification of the connecting host or router in the selection of options for network layer negotiations.

PPP Password Authentication Protocol

The *Password Authentication Protocol (PAP)* provides a simple way for a peer to establish its identity to the authenticator using a two-way handshake. This is done only at initial link establishment. There exist three PAP frame types, as shown in Figure 2-7.

Figure 2-7: The Three PPP PAP Frame Types



After the link establishment phase is completed, the authenticate-request packet is used to initiate the PAP authentication. This packet contains the peer name and password, as shown in Figure 2-8.

Figure 2-8: PPP PAP Authentication Request

This request packet is sent repeatedly until a valid reply packet is received or an optional retry counter expires. If the authenticator receives a Peer-ID/Password pair that is both recognizable and acceptable, it should reply with an Authenticate-Ack (where Ack is short for *acknowledge*). If the Peer-ID/Password pair is not recognizable or acceptable, the authenticator should reply with an Authenticate-Nak (where Nak is short for *negative acknowledge*).

Figure 2-9 shows the sequence of PPP negotiations between a branch router (the peer) trying to authenticate to the NAS, the network access server (the authenticator).

Figure 2-9: PPP PAP Authentication

PAP is not a strong authentication method. PAP authenticates only the peer, and passwords are sent over the circuit "in the clear." There is no protection from replay attacks or repeated trial-and-error attacks. The peer is in control of the frequency and timing of the attempts.

PPP Challenge-Handshake Authentication Protocol

The *Challenge-Handshake Authentication Protocol (CHAP)* is used to periodically verify the identity of a host or end user using a three-way handshake. CHAP is performed at initial link establishment and can be repeated any time after the link has been established. Four CHAP frame types exist, as shown in Figure 2-10.

Figure 2-10: PPP CHAP Frame Types

Figure 2-11 shows a scenario in which a branch router (the peer) is trying to authenticate to the NAS (the authenticator).

CHAP imposes network security by requiring that the peers share a plaintext secret. This secret is never sent over the link. The following sequence of steps is carried out:

Step 1 After the link establishment phase is complete, the authenticator sends a challenge message to the peer. The challenge consists of an identifier (ID), a random number, and either the host name of the local device or the name of the user on the remote device.

Step 2 The receiving peer calculates a value using a one-way hash function; the secret is the input to the one-way hash function.

Step 3 The peer sends the challenge response, which consists of:

- An encrypted version of the ID
- A secret password (the calculated hash value)
- The random number
- Either the host name of the remote device, or the name of the user on the remote device

Step 4 When the authenticator receives the challenge response, it verifies the secret by looking up the name given in the response and performing the same encryption operation. The authenticator checks the response against its own calculation of the expected hash value.

Step 5 If the values match, the authenticator acknowledges the authentication and sends a success message, and the LCP establishes the link.

Figure 2-11: PPP CHAP Authentication

The secret passwords must be identical on the remote and local devices. These secrets should be agreed on, generated, and exchanged out-of-band in a secure manner. Because the secret is never transmitted, other devices are prevented from stealing it and gaining illegal access to the system. Without the proper response, the remote device cannot connect to the local device.

CHAP provides protection against playback attack through the use of an incrementally changing identifier and a variable challenge value. The use of repeated challenges is intended to limit the time of exposure to any single attack. The authenticator is in control of the frequency and timing of the challenges.

Note Typically, MD5 is used as the CHAP one-way hash function; the shared secrets are required to be stored in plaintext form. Microsoft has a variation of CHAP (MS-CHAP), in which the password is stored encrypted in both the peer and the authenticator. Therefore, MS-CHAP can take advantage of irreversibly encrypted password databases commonly available, whereas the standards-based CHAP cannot.

PPP Extensible Authentication Protocol

The PPP *Extensible Authentication Protocol (EAP)* is a general protocol for PPP authentication that supports multiple authentication mechanisms. EAP does not select a specific authentication mechanism at the link control phase; rather, it postpones this until the authentication phase so that the authenticator can request more information before determining the specific authen-tication mechanism. This arrangement also permits the use of a "back-end" server, which actually implements the various authentication mechanisms while the PPP authenticator merely passes through the authentication exchange.

Figure 2-12 shows how PPP EAP works. In the figure, the branch router (the peer) is trying to authenticate to the NAS (the authenticator). The sequence of steps is as follows:

Step 1 When the link establishment phase is complete, the authenticator sends one or more requests to authenticate the peer. The request has a type field to indicate what is being requested. Examples of request types include identity, MD5-challenge, S/Key, generic token card, and so on. The MD5-challenge type corresponds closely to the CHAP authentication protocol.

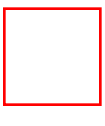
Note Typically, the authenticator sends an initial identity request followed by one or more requests for authentication information. However, an initial identity request is not required and may be bypassed in cases where the identity is presumed (for example, with leased lines, dedicated dial-ups, and so on).

Step 2 The peer sends a response packet in reply to each request. As with the request packet, the response packet contains a type field that corresponds to the type field of the request.

Step 3 The authenticator ends the authentication phase with a success or failure packet.

EAP adds more flexibility to PPP authentication and provides the capability to use new technologies---such as digital certificates---when they become widely available.

Figure 2-12: PPP EAP Authentication



PPP Authentication Summary

PPP authentication is required for dial-in connectivity. Any of the three standard mechanisms---PAP, CHAP, and EAP---can be used. Table 2-1 gives a summary of the strengths and weak-nesses of these mechanisms.

Table 2-1: PPP Authentication Summary

Protocol	Strength	Weakness
PAP	Easy to implement	Does not have strong authentication; password is sent in the clear between client and server; no playback protection
CHAP	Password encrypted	Password must be between client and stored in cleartext on server; both client and server playback protection
EAP	Flexible, more robust authentication support	New; may not yet be widely deployed

Protocols Using Authentication Mechanisms

Many protocols require authentication verification before providing authorization and access rights to the user or device. TACACS+, RADIUS, Kerberos, DCE, and FORTEZZA are examples of such protocols. TACACS+ and RADIUS are often used in dial-in environments to provide a scalable authentication database and can incorporate a variety of authentication methods. Kerberos is a protocol used in some campus environments to first verify that users and the network services they use are really who and what they claim to be before granting access privileges. For completeness, the Distributed Computing Environment (DCE) and FORTEZZA authentication mechanisms are included in this section, although their use is not widespread.

The TACACS+ Protocol

The *TACACS+* protocol is the latest generation of TACACS. TACACS is a simple UDP-based access control protocol originally developed by BBN for the MILNET. Cisco has enhanced (extended) TACACS several times, and Cisco's implementation, based on the original TACACS, is referred to as *XTACACS*. The fundamental differences between TACACS, XTACACS, and TACACS+ are given here:

- TACACS: Combined authentication and authorization process
- XTACACS: Separated authentication, authorization, and accounting
- TACACS+: XTACACS with extended attribute control and accounting

TACACS+ uses TCP for its transport. The server daemon usually listens at port 49, the LOGIN port assigned for the TACACS protocol. This port is reserved in the assigned number's RFC for both UDP and TCP. Current TACACS and extended TACACS implementations also use port 49.

TACACS+ is a client/server protocol; the TACACS+ client is typically a NAS and the TACACS+ server is usually a daemon process running on some UNIX or NT machine. A fundamental design component of TACACS+ is the separation of authentication, authorization, and accounting.

TACACS+ Authentication

TACACS+ allows for arbitrary length and content authentication exchanges, which allows any authentication mechanism to be used with TACACS+ clients (including PPP PAP, PPP CHAP, PPP EAP, token cards, and Kerberos). Authentication is not mandatory; it is a site-configured option. Some sites do not require it at all; others require it only for certain services.

TACACS+ authentication has three packet types:

- START, which is always sent by the client
- CONTINUE, which is always sent by the client
- REPLY, which is always sent by the server

Authentication begins with the client sending a START message to the server. The START message describes the type of authentication to be performed (for example, simple cleartext password, PAP, or CHAP), and may contain the username and some authentication data. The START packet is sent only as the first message in a TACACS+ authentication session, or as the packet immediately following a restart. (A restart may be requested by the server in a REPLY packet.) A START packet always has s sequence number equal to 1.

In response to a START packet, the server sends a REPLY. The REPLY message indicates whether the authentication is finished, or whether it should continue. If the REPLY indicates that authentication should continue, the message also indicates what new information is requested. The client gets that information and returns it in a CONTINUE message. This process repeats until all authentication information is gathered, and the authentication process concludes.

TACACS+ Authorization

Authorization is the action of determining what a user is allowed to do. Generally, authen-tication precedes authorization, but, this is not required. An authorization request may indicate that the user is not authenticated (that is, we don't know who they are). In this case, it is up to the authorization agent to determine whether an unauthenticated user is allowed the services in question.

When authentication is completed (if authentication is used), the client can start the authorization process, if authorization is required. An *authorization session* is defined as a single pair of messages: a REQUEST followed by a RESPONSE. The authorization REQUEST message contains a fixed set of fields that describe the authenticity of the user or process, and a variable set of arguments that describes the services and options for which authorization is requested.

Note In TACACS+, authorization does not merely provide yes or no answers---it may also customize the service for the particular user. Here are some examples of when authorization would be performed: When a user first logs in and wants to start a shell; when a user starts PPP and wants to use IP over PPP with a particular IP address. The TACACS+ server daemon might respond to these requests by allowing the service, by placing a time restriction on the login shell, or by requiring IP access lists on the PPP connection.

TACACS+ Accounting

Accounting is typically the third action after authentication and authorization. Accounting is the action of recording what a user is doing or has done. Accounting in TACACS+ can serve two purposes:

- It may be used to account for services used, such as in a billing environment.
- It may be used as an auditing tool for security services.

To this end, TACACS+ supports three types of accounting records:

- *Start records* indicate that a service is about to begin.
- *Stop records* indicate that a service has just terminated.
- *Update records* are intermediate notices that indicate that a service is still being performed.

TACACS+ accounting records contain all the information used in the authorization records and also contain accounting-specific information such as start and stop times (when appropriate) and resource usage information.

TACACS+ Transactions

Transactions between the TACACS+ client and TACACS+ server are authenticated through the use of a shared secret, which is never sent over the network. Typically, the secret is manually configured in both entities. TACACS+ encrypts all traffic between the TACACS+ client and the TACACS+ server daemon.

Figure 2-13 shows the interaction between a dial-in user and the TACACS+ client and server.

Figure 2-13: A TACACS+ Exchange

The RADIUS Protocol

The *Remote Address Dial-In User Service (RADIUS)* protocol was developed by Livingston Enterprises, Inc., as an access server authentication and accounting protocol. In June 1996, the RADIUS protocol specification was submitted to the IETF. The RADIUS specification (RFC 2058) and RADIUS accounting standard (RFC 2059) are now proposed standard protocols.

RADIUS uses UDP as its transport. Generally, the RADIUS protocol is considered to be a connectionless service. Issues related to server availability, retransmission, and timeouts are handled by the RADIUS-enabled devices rather than by the transmission protocol.

RADIUS is a client/server protocol. The RADIUS client is typically a NAS; the RADIUS server is usually a daemon process running on some UNIX or NT machine. The client is responsible for passing user information to designated RADIUS servers and then acting on the response that is returned. RADIUS servers are responsible for receiving user connection requests, authenticating the user, and then returning all configuration information necessary for the client to deliver the service to the user. A

RADIUS server can act as a proxy client to other RADIUS servers or to other kinds of authentication servers.

RADIUS Authentication

The RADIUS server can support a variety of methods to authenticate a user. When the server is provided with the user name and original password given by the user, the server can support PPP PAP or CHAP, UNIX login, and other authentication mechanisms. What is supported depends on what a vendor has implemented.

Typically, a user login consists of a query (Access-Request) from the NAS to the RADIUS server and a corresponding response (Access-Accept or Access-Reject) from the server. The Access-Request packet contains the username, encrypted password, NAS IP address, and port. The format of the request also provides information about the type of session the user wants to initiate.

When the RADIUS server receives the Access-Request packet from the NAS, it searches a database for the username listed. If the username does not exist in the database, either a default profile is loaded or the RADIUS server immediately sends an Access-Reject message. This Access-Reject message can be accompanied by an optional text message, which can indicate the reason for the refusal.

RADIUS Authorization

In RADIUS, the authentication and authorization functionalities are coupled together. If the username is found and the password is correct, the RADIUS server returns an Access-Accept response, including a list of attribute-value pairs that describe the parameters to be used for this session. Typical parameters include service type (shell or framed), protocol type, IP address to assign the user (static or dynamic), access list to apply, or a static route to install in the NAS routing table. The configuration information in the RADIUS server defines what will be installed on the NAS.

RADIUS Accounting

The accounting features of the RADIUS protocol can be used independently of RADIUS authentication or authorization. The RADIUS accounting functions allow data to be sent at the start and end of sessions, indicating the amount of resources (such as time, packets, bytes, and so on) used during the session. An Internet service provider (ISP) might use RADIUS access control and accounting software to meet special security and billing needs.

RADIUS Transactions

Transactions between the client and RADIUS server are authenticated through the use of a shared secret, which is never sent over the network. In addition, any user passwords are sent encrypted between the client and RADIUS server to eliminate the possibility that someone snooping on an unsecure network could determine a user's password.

Figure 2-14 shows the RADIUS login and authentication process.

Figure 2-14: RADIUS Login and Authentication

NOTE With both TACACS+ and RADIUS, it is important to remember that encryption is performed between the TACACS+/RADIUS client and the TACACS+/RADIUS server. If the TACACS+/RADIUS client is a NAS and not the client PC, any communication between the PC and the NAS is not encrypted (see Figure 2-15).

Figure 2-15: TACACS+/RADIUS Encryption

The Kerberos Protocol

Kerberos is a secret-key network authentication protocol, developed at Massachusetts Institute of Technology (MIT), that uses the Data Encryption Standard (DES) cryptographic algorithm for encryption and authentication. The Kerberos Version 5 protocol is an Internet standard specified by RFC 1510.

Kerberos was designed to authenticate user requests for network resources. Kerberos is based on the concept of a trusted third party that performs secure verification of users and services.
In the Kerberos protocol, this trusted third party is called the *key distribution center (KDC)*, sometimes also called the *authentication server*. The primary use of Kerberos is to verify that users and the network services they use are really who and what they claim to be. To accomplish this, a trusted Kerberos server issues "tickets" to users. These tickets have a limited lifespan and are stored in the user's credential cache. They can later be used in place of the standard username-and-password authentication mechanism.

Kerberos Authentication Request and Reply

Initially, the Kerberos client has knowledge of an encryption key known only to the user and the KDC, K_{client}. Similarly, each application server shares an encryption key with the KDC, K_{server} (see Figure 2-16).

Figure 2-16: Kerberos Keys

When the client wants to create an association with a particular application server, the client uses the authentication request and response to first obtain a ticket and a session key from the KDC (see Figure 2-17).

Figure 2-17: Kerberos Authentication Request and Reply

The steps are as follows:

Step 1 The client sends an authentication request to the KDC. This request contains the following information:

- Its claimed identity
- The name of the application server
- A requested expiration time for the ticket
- A random number that will be used to match the authentication response with the request

Step 2 The KDC verifies the client access rights and creates an authentication response.

Step 3 The KDC returns the response to the client. The authentication response contains the following information:

- The session key, $K_{session}$
- The assigned expiration time
- The random number from the request
- The name of the application server
- Other information from the ticket

This information is all encrypted with the user's password, which was registered with the authentication server, K_{client}. The KDC also returns a Kerberos ticket containing the random session key, $K_{session}$, that will be used for authentication of the client to the application server; the name of the client to whom the session key was issued; and an expiration time after which the session key is no longer valid. The Kerberos ticket is encrypted using K_{server}.

Step 4 When the client receives the authentication reply, it prompts the user for the password. This

password, K_{client}, is used to decrypt the session key, $K_{session}$.

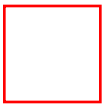
Now the client is ready to communicate with the application server.

Note K_{client} is used as the bootstrap mechanism, but in subsequent communication between the KDC and the client, a short-term client key, $K_{client-session}$, is used. $K_{client-session}$ is created by having the KDC convert the user's password to the short-term client key. The KDC sends the short-term client key, $K_{client-session}$, encrypted with the user's password, to the client. The user decrypts the short-term client key and subsequent KDC to client communication use $K_{client-session}$.

Kerberos Application Request and Response

The application request and response is the exchange in which a client proves to an application server that it knows the session key embedded in a Kerberos ticket. The exchange is shown in Figure 2-18.

Figure 2-18: Kerberos Application Request and Reply



The steps in the application request and response are as follows:

Step 1 The client sends two things to the application server as part of the application request:

- The Kerberos ticket (described earlier)
- An authenticator, which includes the following (among other fields):
- The current time
- A checksum
- An optional encryption key

  These elements are all encrypted with the session key, $K_{session}$, from the accompanying ticket.

Step 2 After receiving the application request, the application server decrypts the ticket with K_{server}, extracts the session key, $K_{session}$, and uses the session key to decrypt the authenticator.

If the same key was used to encrypt the authenticator as was used to decrypt it, the checksum will match, and the verifier can assume that the authenticator was generated by the client named in the ticket and to whom the session key was issued. By itself, this check is not sufficient for authentication because an attacker can intercept an authenticator and replay it later to impersonate the user. For this reason, the verifier also checks the timestamp. If the timestamp is within a specified window (typically 5 minutes), centered around the current time on the verifier, and if the timestamp has not been seen on other requests within that window, the verifier accepts the request as authentic.

At this point, the identity of the client has been verified by the server. For some applications, the client also wants to be sure of the server's identity. If such mutual authentication is required, a third step is required.

Step 3 The application server generates an application response by extracting the client's time from the authenticator and returns it to the client together with other information, all encrypted using the session key, $K_{session}$.

Reuse of Credentials

The basic Kerberos authentication protocol allows a client with knowledge of the user's password to obtain a ticket and session key and to prove its identity to any verifier (usually an application server) registered with the KDC. The user's password must be presented each time the user performs authentication with a new verifier. A system should support *single sign-on*, where the user logs in to the system once and provides the password at that time; subsequent authentication occurs automatically.

The obvious way to cache the user's password on the workstation is dangerous. Although a Kerberos ticket and the key associated with it are valid for only a short time, an intruder knowing the user's password can obtain valid tickets and impersonate the user until the password is changed. This is why the short-term client key, $K_{client\text{-}session}$, is used in place of the user's actual password in all but the initial bootstrap communication. The Kerberos approach is to cache only tickets and encryption keys (collectively called *credentials*) that will work for a limited time period.

The ticket-granting exchange of the Kerberos protocol allows a user to obtain tickets and encryption keys using such short-lived credentials, without reentering the user's password. When the user first logs in, an authentication request is issued, and a ticket and the client session key for the ticket-granting service is returned by the KDC. This ticket, called a *ticket-granting ticket (TGT)*, has a relatively short life (typically on the order of 8 hours). The response is decrypted, the ticket and session key are saved, and the user's password is forgotten. Subsequently, when the user wants to prove its identity to a new verifier, a new ticket is requested from the KDC using the ticket-granting exchange.

Note The ticket-granting exchange is identical to the authentication exchange except that the ticket-granting request has embedded within it an application request (authenticating the client to the authentication server), and the ticket-granting response is encrypted using the client session key from the ticket-granting ticket rather than from the user's password.

Practical Considerations

Multiple realms, or domains, are supported in Kerberos to allow for scalable implementations. Assume that a corporation has implemented a Kerberos system with two separate realms, Italy and Hungary. When a client in Italy's realm connects to a server in Hungary's realm, Italy's KDC authenticates the client to Hungary's KDC. Hungary's KDC authenticates the client to Hungary's server. Multi-KDC chaining is not allowed, and trust for KDC chaining should go back only one level.

Several utility programs must be installed on the workstation to allow users to obtain Kerberos credentials (kinit), destroy credentials (kdestroy), list credentials (klist), and change their Kerberos passwords (kpasswd). Some sites choose to integrate the Kerberos login tool kinit with the workstation

login program so that users do not have to type their password twice. This makes the use of Kerberos nearly transparent; users may not even be aware they are using Kerberos.

Note Client/server applications must be modified to use Kerberos for authentication; such Kerberos-aware applications are said to be *Kerberized*.

You should also consider using some method of accurate time in all systems because Kerberos has a time-dependency issue through the use of timestamps. A synchronized, dependable mechanism of obtaining time is needed; most likely, the use of NTP is warranted.

The Distributed Computing Environment

The *Distributed Computing Environment (DCE)* is a set of functional specifications from the Open Software Foundation (OSF, found at http://www.opengroup.org/). DCE is a set of distributed computing technologies that provide security services to protect and control access to data; name services that make it easy to find distributed resources; and a highly scalable model for organizing widely scattered users, services, and data.

DCE has a modular design and supports authentication and authorization. The implemented authentication part is Kerberos Version 5 (although, in theory, another mechanism can be substituted). The authorization part works in a manner similar to Kerberos but is implemented by privilege servers and registration servers. In practice, these are usually delivered with the KDC. The registration server ties the KDC with the user's privileges, which are found in the privilege server. The privilege server combines the universal unique ID (UUID) and the groups into a Kerberos ticket for secure transmission. Kerberos uses usernames (which may not always be consistent or unique across the enterprise). DCE uses the UUIDs, which are 128 bits long. On most systems, the user ID (UID) and group ID (GID) fields are 32 bits each.

In practice, a user can authenticate from any workstation with a username and password. The TGT is issued by the KDC. The workstation then uses that session key to form a session to the privilege server. The UUID and access control list (ACL) information is then passed to the workstation through a privilege ticket-granting ticket (PTGT) from the privilege server. The session key encrypted in the PTGT is used. The UUID and the group information are then used as the authorization information to allow or disallow access to services and resources.

Note The DCE effort has not produced the groundswell effect its supporters hoped for. Today, some organizations have embraced it, but it is manpower intensive to support (as is Kerberos) because it is fairly complex and relies on several other DCE services being implemented. Therefore, it is not found in use very often.

The FORTEZZA

Multilevel Information Systems Security Initiative (MISSI) is a network security initiative, under the leadership of the National Security Agency (NSA). MISSI provides a framework for the development and evolution of interoperable, complementary security products to provide flexible, modular security for networked information systems across the Defense Information Infrastructure (DII) and the National

Information Infrastructure (NII). These MISSI building blocks share a common network security infrastructure and are based on common security protocols and standards. Flexible solutions are tailored from these building blocks to meet a system's security requirements and may easily evolve, as future MISSI components provide additional backwardly compatible security services and assurance.

Although some MISSI components result from government-contracted developments, most components are offered by commercial vendors as off-the-shelf products. The MISSI building blocks include:

- FORTEZZA and FORTEZZA Plus
- Firewalls
- Guards
- Inline encryptors
- Trusted computing

FORTEZZA, combined with FORTEZZA-enabled applications, provides security services appropriate for protecting sensitive-but-unclassified (SBU) data. FORTEZZA provides the following features:

- Protection for SBU data when used on a commercial off-the-shelf (COTS) workstation in LAN or WAN environments
- Identification and authentication, confidentiality, data integrity, and nonrepudiation services
- Support for various workstation operating systems (DOS/Windows and UNIX at a minimum)

FORTEZZA Plus supports users of classified information with strong encryption methods. FORTEZZA Plus is an upgraded version of FORTEZZA that can be used to encrypt classified information up through Top Secret information. FORTEZZA Plus must be used in conjunction with a high assurance guard such as the secure network server (SNS), which ensures that the encryption of information is invoked. The use of FORTEZZA Plus to process classified information at different levels can be affected by the security limitations of other components in the system.

The FORTEZZA card is a cryptographic peripheral (a PC Card) that provides encryption/decryption and digital signature functions. The card also stores certificates that include individualized key material used by the cryptographic and signature functions. The software on the workstation (PC, UNIX, and so on) exchanges commands and data with the FORTEZZA card to encrypt and sign messages before it sends them. It likewise uses the card to decrypt and verify the signatures of received messages. Each time the card is inserted into a workstation, the owner must unlock the card by entering a PIN. FORTEZZA card PINs can range from 4 to 12 characters. PINs may be a combination of alpha and numeric characters.

To perform application functions for the user, FORTEZZA must interoperate with FORTEZZA-enabled applications. These applications are either government developed
or COTS applications (such as e-mail) that have been modified to interface with and use FORTEZZA security features. A large variety of such applications exist; more are being
added as they are developed and tested.

Major types of FORTEZZA-enabled applications include these:

- *Electronic messaging.* FORTEZZA can secure e-mail, electronic data interchange (EDI), electronic commerce, and facsimile to provide message encryption, authentication, and data integrity.

- *World Wide Web (WWW).* FORTEZZA can protect secure Web transactions using strong identification and authentication and secure-sockets-layer (SSL) interactions. Netscape has built a FORTEZZA-enabled version of its browser that links SSL with FORTEZZA.
- *File and media encryptors.* These encryptors are applications written to enable FORTEZZA to secure user files on storage media.
- *Identification and authentication.* After the FORTEZZA card has been installed in the workstation and the PIN has been correctly entered, the identity of the user is known and trusted. Access to other devices across a network can be authorized by exchanging this identification and authentication information in a trusted manner.

Security in TCP/IP Layers

This part of the chapter describes the primary technologies used to ensure data integrity and data confidentiality in varying layers of TCP/IP. Often, the collection of communication protocols are organized into seven distinct layers as specified by the Open Systems Interconnection (OSI) Reference Model. How the TCP/IP layers relate to the OSI protocol model is shown in Figure 2-19.

Figure 2-19: The TCP/IP Layered Model

The Application layer pertains to the details of a particular application such as Telnet, FTP, or HTTP and doesn't concern itself with the details of the movement of data across a network. The Transport layer provides the details of moving the flow of data between two hosts. Both the Application layer and the Transport layer use end-to-end protocols, in which end systems are responsible for providing security for the application or transport protocol. The Network layer provides hop-by-hop handling of data packets, where intermediary systems in a network, such as routers, would be involved.

Not many protocols are specific to only Application-level security. SHTTP is one of the few and provides Application-layer security for Web transactions. More protocols exist to secure the Transport layer. SSL/TLS and SSH are Transport-layer security protocols that provide secure client-to-server data transfer at the Transport layer. SOCKS also works at the Transport layer and provides a framework for client-server applications in both the TCP and UDP domains to conveniently and securely use the services of a network firewall.

IPsec is a set of standards that ensure data integrity and confidentiality at the Network (IP) layer.

Application Layer Security Protocols

There aren't many security protocols specifically designed for individual applications. There are too many applications to make such an approach scalable. However, because the World Wide Web has become one of the fastest growing applications in the Internet, a specific security protocol was designed to be used for secure Web transactions: Secure HyperText Transport Protocol (SHTTP).

SHTTP

SHTTP is a secure message-oriented communications protocol designed to be used for securing messages using the HTTP protocol. The protocol preserves the characteristics of HTTP while allowing request and reply messages to be signed, authenticated, encrypted, or any combination of these (including no protection). SHTTP clients can communicate with non-HTTP supported servers (although in these cases, the security features of SHTTP would not be applied).

Multiple key-management mechanisms are supported, including password-style manually shared secrets and public-key key exchange. If some hosts do not have a public key pair, it is possible to use prearranged symmetric session keys to send confidential messages. These would usually be provided out of band.

Secure HTTP can verify message integrity and sender authenticity for a message using the computation of a Message Authentication Code (MAC). The MAC is computed as a keyed hash over the document using a shared secret.

SHTTP uses *option negotiation* to allow clients and servers to agree on the following:

- *Transaction modes.* What should be signed or encrypted or both?
- *Cryptographic algorithms.* Which algorithm should be used for signing and encrypting?
- *Certificate selection.* Which certificate should be used (Verisign, Entrust, other)?

The main benefit of using an application-specific protocol such as SHTTP is that very specific security needs can be met. Consider these examples:

- The application could deal with a message containing digital signatures by several different agents and make decisions based on who signed what.
- Cryptographic security measures could be defined for individual Web pages such that individually encrypted Web pages could be published on any Web server but could only be read by those with authorized keys.

In practice, SHTTP has achieved limited use. Transport layer security implementations are more easily available and more often used for Web security.

Transport Layer Security Protocols

The following sections describe the security protocols that operate over TCP/IP or some other reliable but insecure transport. They are categorized as *Transport layer security protocols* because their intent is to secure the Transport layer as well as to provide methods for implementing privacy, authentication, and integrity above the Transport layer.

The Secure Socket Layer Protocol

The *Secure Socket Layer (SSL)* is an open protocol designed by Netscape; it specifies a mechanism for providing data security layered between application protocols (such as HTTP, Telnet, NNTP, or FTP) and TCP/IP. It provides data encryption, server authentication, message integrity, and optional client authentication for a TCP/IP connection.

The primary goal of SSL is to provide privacy and reliability between two communicating applications. This process is accomplished with the following three elements:

- *The handshake protocol.* This protocol negotiates the cryptographic parameters to be used between the client and the server session. When an SSL client and server first start communicating, they agree on a protocol version, select cryptographic algorithms, optionally authenticate each other, and use public-key encryption techniques to generate shared secrets.
- *The record protocol.* This protocol is used to exchange Application layer data. Application messages are fragmented into manageable blocks, optionally compressed, and a MAC (message authentication code) is applied; the result is encrypted and transmitted. The recipient takes the received data and decrypts it, verifies the MAC, decompresses and reassembles it, and delivers the result to the application protocol.
- *The alert protocol.* This protocol is used to indicate when errors have occurred or when a session between two hosts is being terminated.

Let's look at an example using a Web client and server. The Web client initiates an SSL session by connecting to an SSL-capable server. A typical SSL-capable Web server accepts SSL connection requests on a different port (port 443 by default) than standard HTTP requests (port 80 by default). When the client connects to this port, it initiates a handshake that establishes the SSL session. After the handshake finishes, communication is encrypted and message integrity checks are performed until the SSL session expires. SSL creates a session during which the handshake must happen only once.

The SSL handshake process is shown in Figure 2-20. (Refer to "Public Key Infrastructure and Distribution Models," later in this chapter, for more information about digital certificates.) The steps in the process are as follows:

Step 1 The SSL client connects to the SSL server and requests the server to authenticate itself.

Step 2 The server proves its identity by sending its digital certificate. This exchange may optionally include an entire certificate chain, up to some root certificate authority (CA). Certificates are verified by checking validity dates and verifying that the certificate bears the signature of a trusted CA.

Step 3 The server may then initiate a request for client-side certificate authen-tication. However, because of a lack of a public key infrastructure, most servers today do not do client-side authentication.

Step 4 The message encryption algorithm for encryption and the hash function for integrity are negotiated. Usually the client presents a list of all the algorithms it supports, and the server selects the strongest cipher available.

Step 5 The client and server generate the session keys by following these steps:

(a) The client generates a random number that it sends to the server, encrypted with the server's

public key (obtained from the server's certificate).

(b) The server responds with more random data (encrypted with the client's public key, if available; otherwise, it sends the data in cleartext).

(c) The encryption keys are generated from this random data using hash functions.

Figure 2-20: The SSL Handshake Process

The advantage of the SSL protocol is that it provides connection security that has three basic properties:

- The connection is private. Encryption is used after an initial handshake to define a secret key. Symmetric cryptography is used for data encryption (for example, DES and RC4).
- The peer's identity can be authenticated using asymmetric, or public key, cryptography (for example, RSA and DSS).
- The connection is reliable. Message transport includes a message integrity check using a keyed MAC. Secure hash functions (such as SHA and MD5) are used for MAC computations.

Acceptance of SSL has only been within HTTP. The other protocols have been demonstrated to work but are not yet widely deployed.

Note A new protocol is being defined by the IETF called *Transport Layer Security (TLS)*. It is based on the SSL 3.0 protocol specification as published by Netscape; it is likely that the industry will move towards TSL for a standardized protocol for Transport layer security. There are, however, significant differences between TLS and SSL 3.0---mostly in the cryptographic algorithms they support---such that TLS 1.0 and SSL 3.0 do not necessarily interoperate.

The Secure Shell Protocol

The *Secure Shell (SSH)* is a protocol for secure remote login and other secure network services over an insecure network. It provides support for secure remote login, secure file transfer, and the secure forwarding of TCP/IP and X Window system traffic. It can automatically encrypt, authenticate, and compress transmitted data. The work in progress to define the SSH protocol ensures that the SSH protocol can provide strong security against cryptanalysis and protocol attacks, can work reasonably well without a global key management or certificate infra-structure, and can use existing certificate infrastructures (such as DNSSEC and X.509) when available.

The SSH protocol consists of three major components:

- The Transport layer protocol, which provides server authentication, confidentiality, and integrity with perfect forward secrecy. Optionally, it may also provide compression.

- The user authentication protocol, which authenticates the client to the server.
- The connection protocol, which multiplexes the encrypted tunnel into several logical channels.

The SSH transport layer is a secure low-level transport protocol. It provides strong encryption, cryptographic host authentication, and integrity protection. Authentication in SSH is host-based; this protocol does not perform user authentication. A higher-level protocol for user authentication can be designed on top of SSH.

The protocol has been designed to be simple and flexible enough to allow parameter negotiation and to minimize the number of round trips. The key exchange method, the public key algorithm, the symmetric encryption algorithm, the message authentication algorithm, and the hash algorithm are all negotiated.

Data integrity is protected by including with each packet a message authentication code (MAC) computed from a shared secret, a packet sequence number, and the contents of the packet.

SSH implementations can be found for UNIX, Windows, and Macintosh systems. It is a widely accepted protocol that uses well-known and well-established encryption, integrity, and public key algorithms.

The SOCKS Protocol

Socket security (SOCKS) is a Transport layer-based secure networking proxy protocol. It is designed to provide a framework for client/server applications in both the TCP and UDP domains to conveniently and securely use the services of a network firewall.

SOCKS was originally developed by David and Michelle Koblas; the code was made freely available on the Internet. Several major revisions have occurred since then, but the software has remained freely available. SOCKS Version 4 provides for unsecured firewall traversal for TCP-based client/server applications (including Telnet, FTP, and the popular information discovery protocols such as HTTP, WAIS, and Gopher). SOCKS Version 5, defined in RFC 1928, extends the SOCKS Version 4 model to include UDP, extends the framework to include provisions for generalized strong authentication schemes, and extends the addressing scheme to encompass domain-name and IPv6 addresses.

A proposal currently exists to create a mechanism for managing the entrance or exit of IP multicast through a firewall. It does this by defining extensions to the existing SOCKS Version 5 protocol, which provides a framework for user-level, authenticated firewall traversal of unicast TCP and UDP traffic. However, because the current UDP support in SOCKS Version 5 has scalability problems as well as other deficiencies (which must be addressed before multicast support can be achieved), the extensions are defined in two parts:

- Base-level UDP extensions
- Multicast UDP extensions

SOCKS works by replacing the standard network system calls in an application with special versions. (This is why SOCKS is sometimes referred to as an application-level proxy.) These new system calls open connections to a SOCKS proxy server (configured in the application by the user, or by a system configuration file) on a well-known port (usually 1080/TCP). If the connection request succeeds, the client enters a negotiation for the authentication method to be used, authenticates with the chosen method, and then sends a relay request. The SOCKS server evaluates the request and either establishes the appropriate connection or denies it. After the connection is established with the SOCKS server, the

client application sends the server the name of the machine and the port number to which the user wants to connect. The SOCKS server actually makes the connection with the remote host and then transparently moves data back and forth between the application and the remote machine. The user has no idea that the SOCKS server is even in the loop (see Figure 2-21).

Figure 2-21: The SOCKS Security Model

The difficulty with using SOCKS is that somebody has to replace the network system calls with the SOCKS versions (this process is generally referred to as *SOCKS-ification* or *SOCKS-ifying an application*). Fortunately, most of the common network applications (such as Telnet, FTP, finger, and whois) have already been SOCKS-ified, and many vendors are now including SOCKS support in commercial applications.

Network Layer Security

Network layer security pertains to security services at the IP layer of the TCP/IP protocol stack. Many years of work have produced a set of standards from the IETF that, collectively, define how to secure services at the IP Network layer.

The IP Security Protocol Suite

The *IP Security (IPsec)* protocol suite comprises a set of standards used to provide privacy and authentication services at the IP layer. The current ratified IPsec standards include four algorithm-independent base specifications:

- RFC 2401, the IP Security Architecture, defines the overall architecture and specifies elements common to both the IP Authentication Header (AH) and the IP Encapsulating Security Payload (ESP).
- RFC 2402, the IP Authentication Header (AH), defines an algorithm-independent mechanism for providing exportable cryptographic authentication without encryption to IPv4 and IPv6 packets.
- RFC 2406, the IP Encapsulating Security Payload (ESP), defines an algorithm-independent mechanism for providing encryption to IPv4 and IPv6 packets.
- RFC 2408, the Internet Security Association and Key Management Protocol (ISAKMP) defines procedures and packet formats to establish, negotiate, modify, and delete Security Associations (SA).

The set of security services IPsec can provide includes access control, connectionless integrity, data origin authentication, rejection of replayed packets (a form of partial sequence integrity), confidentiality

(encryption), and limited traffic flow confidentiality. Because these services are provided at the IP layer, they can be used by any higher layer protocol (such as TCP, UDP, ICMP, BGP, and so on).

Security Services

IPsec uses two protocols to provide traffic security, each of which defines a new set of headers to be added to IP datagrams:

- *Authentication Header (AH).* This header, when added to an IP datagram, ensures the integrity and data origin authentication of the data, including the invariant fields in the outer IP header. It does not provide confidentiality protection. AH commonly uses a keyed hash function rather than digital signatures, because digital signature technology is too slow and greatly reduces network throughput. AH is an appropriate protocol to employ when confidentiality is not required (or is not permitted, as when government regulations restrict the use of encryption).
- *Encapsulating Security Payload (ESP).* This header, when added to an IP datagram, protects the confidentiality, integrity, and data origin authentication of the data. The scope of the authentication offered by ESP is narrower than it is for AH (the IP header "outside" the ESP header is not protected). If only the upper-layer protocols must be authenticated, ESP authentication is an appropriate choice and is more space efficient than using AH to encapsulate ESP.

AH and ESP can be used independently or in combination to provide a desired set of security services. For both of these protocols, IPsec does not define the specific security algorithms to use; rather, it provides an open framework for implementing industry-standard algorithms. Initially, most implementations of IPsec will support MD5 from RSA Data Security or the Secure Hash Algorithm (SHA) as defined by the U.S. government for integrity and authen-tication. The Data Encryption Standard (DES) is currently the most commonly offered bulk encryption algorithm, although specifications in various RFCs are available that define how to use many other encryption systems, including IDEA, Blowfish, and RC4.

Each protocol supports two modes of use: transport mode and tunnel mode.

In *transport mode*, two hosts provide protection primarily for upper-layer protocols; the cryptographic endpoints (where the encryption and decryption take place) are the source and destination of the data packet. In IPv4, a transport mode security protocol header appears immediately after the IP header and before any higher-layer protocols (such as TCP or UDP). This process is shown in Figure 2-22.

Figure 2-22: The IPsec IPv4 Transport Mode

In the case of AH in transport mode, all upper-layer information is protected, and all fields in the IPv4 header excluding the fields typically are modified in transit. The fields of the IPv4 header that are *not* included are, therefore, set to 0 before applying the authentication algorithm. These fields are as follows:

- TOS
- TTL
- header checksum
- hoffset
- flags

In the case of ESP in transport mode, security services are provided only for the higher-layer protocols, not for the IP header.

A *tunnel* is a vehicle for encapsulating packets inside a protocol that is understood at the entry and exit points of a given network. These entry and exit points are defined as *tunnel interfaces*. Tunnel mode can be supported by data packet endpoints as well as by intermediate security gateways. In *tunnel mode*, there is an "outer" IP header that specifies the IPsec processing destination, plus an "inner" IP header that specifies the ultimate destination for the packet. The source address in the outer IP header is the initiating cryptographic endpoint; the source address in the inner header is the true source address of the packet. The security protocol header appears after the outer IP header and before the inner IP header (see Figure 2-23).

Figure 2-23: IPsec IPv4 Tunnel Mode

If AH is employed in tunnel mode, portions of the outer IP header are given protection (those same fields as for transport mode, described earlier in this section), as well as all of the tunneled IP packet (that is, all of the inner IP header is protected as are the higher-layer protocols). If ESP is employed, the protection is afforded only to the tunneled packet, not to the outer header.

Security Associations

The concept of a *Security Association (SA)* is fundamental to IPsec. An SA is a relationship between two or more entities that describes how the entities will use security services to communicate securely. The SA includes the following:

- An encryption algorithm
- An authentication algorithm
- A shared session key

Because an SA is unidirectional, two SAs (one in each direction) are required to secure typical, bi-directional communication between two entities. The security services associated with an SA can be used for AH or ESP, but not for both. If both AH and ESP protection is applied to a traffic stream, two (or more) SAs are created for each direction to protect the traffic stream.

The SA is uniquely identified by a randomly chosen unique number called the *security parameter index (SPI)* and the destination IP address of the destination. When a system sends a packet that requires IPsec protection, it looks up the SA in its database and applies the specified processing and security protocol (AH/ESP), inserting the SPI from the SA into the IPsec header. When the IPsec peer receives the packet, it looks up the SA in its database by destination address, protocol, and SPI and then processes the packet as required.

Key Management

IPsec uses cryptographic keys for authentication/integrity and encryption services. Both manual and automatic distribution of keys is supported.

The lowest (but least desirable) level of management is *manual management*, in which a person manually configures each system by keying material and SA management data relevant to secure communication with other systems. Manual techniques are practical in small, static environments but they do not scale well. If the number of sites using IPsec security services is small, and if all the sites come under a single administrative domain, manual key management techniques may be appropriate. Manual key management may also be appropriate when only selected communications must be secured within an organization for a small number of hosts or gateways. Manual management techniques often employ statically configured, symmetric keys, although other options also exist.

The default *automated key management protocol* selected for use with IPsec is *Internet Key Management Protocol (IKMP)*, sometimes simply referred to as the *Internet Key Exchange (IKE)*. IKE authenticates each peer involved in IPsec, negotiates the security policy, and handles the exchange of session keys.

Note Although IKE is specified as the public-key-based approach for automatic key management, other automated key distribution techniques can be used. For example, KDC-based systems such as Kerberos and other public-key systems such as SKIP can be employed.

IKE is a hybrid protocol, combining parts of the following protocols to negotiate and derive keying material for SAs in a secure and authenticated manner. IKE is derived from the following three protocols, as stated in RFC 2409:

- ISAKMP (Internet Security Association and Key Management Protocol), which provides a framework for authentication and key exchange but does not define them. ISAKMP is designed to be key exchange independent; that is, it is designed to support many different key exchanges.
- Oakley, which describes a series of key exchanges, called *modes*, and details the services provided by each (for example, perfect forward secrecy for keys, identity protection, and authentication).
- SKEMI (Secure Key Exchange Mechanism for Internet), which describes a versatile key exchange technique that provides anonymity, repudiability, and quick key refreshment.

IKE creates an authenticated, secure tunnel between two entities and then negotiates the security association for IPsec. This is performed in two phases.

In Phase 1, the two ISAKMP peers establish a secure, authenticated channel with which to communicate. This channel is called the *ISAKMP SA*.

Note Main Mode for Phase 1 provides identity protection. When identity protection is not needed,

Aggressive Mode can be used to further reduce round trips.

The following attributes are used by IKE and are negotiated as part of the ISAKMP SA:

- Encryption algorithm
- Hash algorithm
- Authentication method (can be digital signature, public-key encryption, or pre-shared key)
- Information about a group on which to perform Diffie-Hellman

After the attributes are negotiated, both parties must be authenticated to each other. IKE supports multiple authentication methods. At this time, the following mechanisms are generally implemented:

- *Preshared keys.* The same key is pre-installed on each host. IKE peers authenticate each other by computing and sending a keyed hash of data that includes the preshared key. If the receiving peer can independently create the same hash using its preshared key, it knows that both parties must share the same secret, and thus the other party is authenticated.
- *Public key cryptography.* Each party generates a pseudo-random number (a *nonce*) and encrypts it and its ID using the other party's public key. The ability for each party to compute a keyed hash containing the other peer's nonce and ID, decrypted with the local private key, authenticates the parties to each other. This method does not provide nonrepudiation; either side of the exchange could plausibly deny that it took part in the exchange. Currently, only the RSA public key algorithm is supported.
- *Digital signature.* Each device digitally signs a set of data and sends it to the other party. This method is similar to the public-key cryptography approach except that it provides nonrepudiation. Currently, both the RSA public-key algorithm and the digital signature standard (DSS) are supported.

Note Both digital signature and public-key cryptography require the use of digital certificates to validate the public/private key mapping. IKE allows the certificate to be accessed independently or by having the two devices explicitly exchange certificates as part of IKE.

Both parties must have a shared session key to encrypt the IKE tunnel. The Diffie-Hellman protocol is used to agree on a common session key. The exchange is authenticated as just described to guard against "man-in-the-middle" attacks.

In Phase 2 of the IKE process, SAs are negotiated on behalf of services such as IPsec AH or ESP. IPsec uses a different shared key than does IKE. The IPsec shared key can be derived by using Diffie-Hellman again or by refreshing the shared secret derived from the original Diffie-Hellman exchange that generated the IKE SA by hashing it with nonces. The first method provides greater security but is slower. After this step is complete, the IPsec SAs are established. Now the data traffic can be exchanged with the negotiated IPsec parameters.

Figure 2-24 shows the creation of an IPsec protected datastream.

Figure 2-24: Establishing IPsec Protection

IPsec is designed to protect IP packets from modification or snooping. It is starting to become widely available in many vendor implementations.

Using Security in TCP/IP Layers

The security protocol you use in a given environment depends on the security services required and on the applications that need protection. Any application-level security protocol has the advantage that the security service can be specifically defined in terms of the application's activities. For example, for Web servers, varying security measures could be applied to individual Web pages. However, most application-level security protocols, such as HTTP, are being made obsolete by the use of Transport layer or Network layer protocols.

In Transport layer security, all application messages must be treated the same. However, you can still specify various security services for different applications as long as vendor imple-mentations support it. SSL has gained wide acceptance and is largely deployed in World Wide Web environments because it is often bundled with World Wide Web applications. SSH is a good all-around protocol for securing Transport layer protocols and is largely used for secure remote login (Telnet) and remote file transfers (FTP).

Network layer security through the use of IPsec can define security services at the IP layer. Depending on vendor implementations, security services can be defined based on IP addresses or can be as granular as providing different security services based on a combination of IP address, transport protocol, and application. IPsec has the advantage of hiding Transport layer information and can support Transport layer protocols other than TCP (such as UDP). However, because it hides Transport layer information, if the Transport layer header information is required to support other network requirements (such as for quality of service that may have to look at TCP/UDP port numbers), you may have problems.

Usually there is a requirement to combine security protocols; most environments use some combination of transport level security protocols and IPsec.

Virtual Private Dial-Up Security Technologies

Virtual Private Dial-Up Networks (VPDNs) enable large enterprises to extend their private networks across dial-up lines. Instead of incurring large costs to ensure security by dialing into a campus site from anywhere in the world or lessening security by dialing in locally and using the Internet as the transport to get to the main enterprise campus, new technologies enable remote sites and users to securely connect to the enterprise infrastructure using local dial-up access to the Internet.

Three similar protocols currently exist to accomplish this goal:

- The Layer 2 Forwarding (L2F) protocol

- The Point-to-Point Tunneling Protocol (PPTP)
- The Layer 2 Tunneling Protocol (L2TP)

The Layer 2 Forwarding Protocol

The *Layer 2 Forwarding (L2F) protocol* was created by Cisco Systems. It permits the tunneling of the link layer---that is, High-Level Data Link Control (HDLC), async HDLC, or Serial Line Internet Protocol (SLIP) frames---of higher-level protocols. Figure 2-25 shows the format of the tunneled packet.

Figure 2-25: The Format of a Tunneled Packet

Using such tunnels, it is possible to decouple the location of the initial dial-up server from the location at which the dial-up protocol connection is terminated and access to the network is provided. These tunnels also enable applications that require support for privately addressed IP, IPX, and AppleTalk dial-up using SLIP/PPP across the existing Internet infrastructure.

A Sample Scenario

Figure 2-26 shows a sample virtual dial-up scenario for L2F. The following steps are carried out:

Step 1 The remote user initiates a PPP connection to an ISP over the PSTN (or natively over ISDN).

Step 2 The NAS accepts the connection, and the PPP link is established.

Step 3 The ISP authenticates the end system or user using CHAP or PAP.

Note If permitted by the organization's security policy, the authorization of the dial-in user at the NAS can be performed only on a domain name within the username field and not on every individual username. This setup can substantially reduce the size of the authorization database. If a virtual dial-up service is not required, traditional access to the Internet may be provided by the NAS. All address assignment and authentication would be performed locally by the ISP in this situation.

Step 4 NAS initiates the L2F tunnel to the desired corporate gateway.

Step 5 The corporate gateway authenticates the remote user and either accepts or rejects the tunnel.

NOTE The initial setup notification may include the authentication information required to allow the corporate gateway to authenticate the user and decide to accept or decline the connection. In the case of CHAP, the setup packet includes the challenge, username, and raw password; for PAP, the setup packet includes the username and cleartext password. The corporate gateway can be configured to use this information to complete its authentication, avoiding an additional cycle of authentication.

Note also that the authentication takes place at the corporate customer, allowing the corporation to impose its own security and corporate policy on the remote users accessing its network. In this way, the organization does not have to fully trust the authentication performed by the ISP.

Step 6 The corporate gateway confirms acceptance of the call and L2F tunnel.

NOTE If the corporate gateway accepts the connection, it creates a virtual interface for PPP in a manner analogous to what it would use for a direct-dialed connection. With this virtual interface in place, Link layer frames can now pass over this tunnel in both directions. Frames from the remote user are received at the NAS, stripped of any link framing or transparency bytes, encapsulated in L2F, and forwarded over the appropriate tunnel.

The corporate gateway accepts these frames, strips L2F, and processes them as normal incoming frames for the appropriate interface and protocol. The virtual interface behaves very much like a hardware interface, except that the hardware in this case is physically located at the ISP NAS. The reverse traffic direction behaves analogously, with the corporate gateway encapsulating the packet in L2F, and the NAS stripping L2F encapsulation before transmitting it out the physical interface to the remote user.

Step 7 The corporate gateway exchanges PPP negotiations with the remote user. Because the remote user has become simply another dial-up client of the corporate gateway access server, client connectivity can now be managed using traditional mechanisms with respect to further authorization, address negotiation, protocol access, accounting, and filtering.

Step 8 End-to-end data is tunneled between the remote user and the corporate gateway.

Figure 2-26: A Sample Scenario for L2F

The Point-to-Point Tunneling Protocol

The *Point-to-Point Tunneling Protocol (PPTP)* was initiated by Microsoft. It is a client/server architecture that allows the Point-to-Point Protocol (PPP) to be tunneled through an IP network and decouples functions that exist in current NASs.

Decoupling Traditional NAS Functionality

Traditionally, the following functions are implemented by a NAS:

- Providing a physical native interface to PSTN or ISDN networks and controlling external modems or terminal adapters.
- Providing the logical termination of a Point-to-Point-Protocol (PPP) Link Control Protocol (LCP) session.

- Participating in PPP authentication protocols.
- Providing channel aggregation and bundle management for PPP Multilink Protocol.
- Performing the logical termination of various PPP Network Control Protocols (NCPs).
- Performing multiprotocol routing and bridging between NAS interfaces.

PPTP divides these functions between two entities:

- *PPTP Access Concentrator (PAC).* This device is attached to one or more PSTN or ISDN lines capable of PPP operation and of handling the PPTP protocol.
- *PPTP Network Server (PNS).* This device handles the server side of the PPTP protocol. Because PPTP relies completely on TCP/IP and is independent of the interface hardware, the PNS may use any combination of IP interface hardware, including LAN and WAN devices.

The PAC is responsible for providing the physical interface to the PSTN and ISDN networks and for providing the logical termination for PPP LCP sessions. Participation in PPP authen-tication protocols can be part of either the PAC or the PNS. The PNS is responsible for channel aggregation, logical termination of PPP NCPs, and multiprotocol routing and bridging between NAS interfaces. The protocol used to carry PPP protocol data units (PDUs) between the PAC and PNS; in addition, call control and management issues are addressed by PPTP.

Protocol Overview

PPTP is connection oriented. The PNS and PAC maintain connection information for each user attached to a PAC. A session is created when an end-to-end PPP connection is attempted between a dial-up user and the PNS. The datagrams related to a session are sent over the tunnel between the PAC and the PNS.

A tunnel is defined by a PNS-PAC pair. The tunnel carries PPP datagrams between the PAC and the PNS. Many sessions are multiplexed on a single tunnel. A control connection operating over TCP manages the establishment, release, and maintenance of sessions and of the tunnel itself.

There are two parallel components of PPTP, as described in the following sections:

- A *control connection* between each PAC-PNS pair operating over TCP.
- An *IP tunnel* operating between the same PAC-PNS pair, which is used to transport GRE-encapsulated PPP packets for user sessions between the pair.

The Control Connection

A *control connection* must be established between the PNS-PAC pair before PPP tunneling can occur between them. The control connection is a standard TCP session over which PPTP call control and management information is passed. The TCP session for the control connection is established by initiating a TCP connection to port 1723. The control session is logically associated with, but separate from, the sessions being tunneled through a PPTP tunnel.

The first set of control connection messages are used to maintain the control connection itself. The control connection is initiated by either the PNS or the PAC after they establish the underlying TCP connection. The control connection is responsible for the establishment, management, and release of sessions carried through the tunnel. It is the means by which a PNS is notified of an incoming call at an associated PAC, as well as the means by which a PAC is instructed to place an outgoing dial call. After

the control connection is established, the PAC or PNS may initiate sessions by requesting outbound calls or by responding to inbound requests. The control connection itself is maintained by keep-alive echo messages.

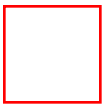
The Tunnel Protocol

PPTP requires the establishment of a tunnel for each communicating PNS-PAC pair. This tunnel is used to carry all user session PPP packets for sessions involving a given PNS-PAC pair.

The user data carried by the PPTP protocol are PPP data packets. PPP packets are carried between the PAC and the PNS, encapsulated in GRE packets, which in turn are carried over IP. The encapsulated PPP packets are essentially PPP data packets without any media-specific framing elements.

Figure 2-27 shows the general packet structure that is transmitted over the tunnels between a PAC and a PNS:

Figure 2-27: The PPTP Tunneled Packet Structure



The Layer 2 Tunneling Protocol

Because both L2F and PPTP provide similar functionality, Cisco and Microsoft, along with other vendors, have collaborated on a single standard: a track protocol within the IETF, which is now called *Layer 2 Tunneling Protocol (L2TP)*. This protocol is considered a work in progress and addresses the following end user requirements:

- End system transparency. Neither the remote end system nor the home site hosts should require any special software to use this service in a secure manner.
- Authentication as provided by the dial-up PPP CHAP, PAP, EAP, or through other dialogs (such as a textual exchange on V.120 before starting PPP). This includes TACACS+ and RADIUS solutions and also supports smart cards and one-time passwords. The authentication should be manageable by the user independently of the ISP.
- Addressing should be as manageable as dedicated dial-up solutions. The address should be assigned by the home site and not by the ISP.
- Authorization should be managed by the home site as it would in a direct dial-up solution.
- Accounting should be performed both by the ISP (for billing purposes) and by the user (for charge-back and auditing purposes).

Protocol Overview

In a way similar to PPTP, L2TP defines two entities:

- *L2TP Access Concentrator (LAC).* This device is attached to the switched network fabric (for example, PSTN or ISDN) or co-located with a PPP end system capable of handling L2TP. The LAC only has to implement the media over which L2TP is to operate to pass traffic to one or more LNSes. The LAC may tunnel any protocol carried within PPP. The LAC is the initiator of incoming calls and the receiver of outgoing calls.
- *L2TP Network Server (LNS).* This server operates on any platform capable of PPP termination. The LNS handles the server side of the L2TP protocol. Because L2TP relies on only the single media over which L2TP tunnels arrive, the LNS may have only a single LAN or WAN interface yet be able to terminate calls arriving at any LAC's full range of PPP interfaces (ASYNC, synchronous ISDN, V.120, and so on). The LNS is the initiator of outgoing calls and the receiver of incoming calls.

There are two parallel components of L2TP operating over a given tunnel: control messages between each LAC-LNS pair and payload packets between the same LAC-LNS pair. The latter are used to transport L2TP-encapsulated PPP packets for user sessions between the pair.

Control Message Overview

Before PPP tunneling can occur between a LAC and an LNS, control messages must be exchanged between them. Control messages are exchanged over the same tunnel that will be used to forward payload data once L2TP call control and management information have been passed. The control messages are responsible for the establishment, management, and release of sessions carried through the tunnel, as well as the status of the tunnel itself. Control messages are the means by which an LNS is notified of an incoming call at an associated LAC, as well as the means by which a LAC is instructed to place an outgoing call.

A tunnel can be established by either a LAC (for incoming calls) or an LNS (for outgoing calls). Following the establishment of the tunnel, the LNS and LAC configure the tunnel by exchanging control messages. When the control message exchange is complete, either the LAC may initiate sessions by indicating inbound requests, or the LNS can request outbound calls. If both ends of the tunnel have the ability to act as an LAC and LNS concurrently, nothing prohibits the establishment of incoming or outgoing calls from both sides of the same tunnel.

A keep-alive mechanism is employed by the L2TP higher layer to differentiate tunnel outages from extended periods of no control or data activity on a tunnel.

Payload Packet Overview

After a tunnel is established and control messages have completed tunnel setup, the tunnel can be used to carry user-session PPP packets for sessions involving a given LNS-LAC pair. The Call ID field in the L2TP header indicates the session to which a particular PPP packet belongs. In this manner, PPP packets are multiplexed and demultiplexed over a single tunnel between a given LNS-LAC pair. The Call ID field value is established during the exchange of call setup control messages.

It is legal for multiple tunnels to exist between a given LNS-LAC pair. With multiple tunnels, each tunnel can be used for a single user session, and the tunnel media (an SVC, for instance) can have specific QoS attributes dedicated to a given user. L2TP provides a tunnel identifier so that individual tunnels can be identified, even when arriving from a single source LAC or LNS.

L2TP uses the well-known UDP port 1701. The entire L2TP packet, including payload and L2TP header, is sent within a UDP datagram. The initiator of an L2TP tunnel picks an available source UDP port and sends to the desired destination at port 1701. The recipient picks a free port on its own system (which may or may not be port 1701) and sends its reply to the initiator's UDP port, setting its own UDP source port to the free port it found.

It is legal for a peer's IP address or UDP port used for a given tunnel to change over the life of a connection (for example, when a peer with multiple IP interfaces responds to a network topology change). Responses should reflect the last source IP address and the UDP port for that tunnel ID.

Note Port 1701 is used for both L2F and L2TP packets. The Version field in the header differentiates the two; L2F uses version 1 and L2TP uses version 2.

A Sample Scenario

Figure 2-28 shows a sample L2TP scenario of a generic Internet arrangement with PSTN access (that is, async PPP using modems) and ISDN access (synchronous PPP access). Remote users (either async or ISDN PPP) access the home LAN as if they had dialed into the LNS, although their physical dial-up is through the ISP NAS (acting as the LAC).

Figure 2-28: A Sample L2TP Scenario

The steps needed to complete the PPP tunnel are as follows:

Step 1 The remote user initiates a PPP connection to an ISP using either the PSTN or ISDN.

Step 2 The LAC accepts the connection, and the PPP link is established. L2TP also permits the LAC to check with an LNS after call indication before accepting the call. This is useful when *Dialed Number Information String (DNIS)*, an indication to the receiver of a call as to what phone number the caller used to reach it or *Calling Line Identification (CLID)* information is available in the incoming call notification.

Step 3 The ISP can now undertake a partial authentication of the end system or user. Only the username field is interpreted to determine whether the user requires a virtual dial-up service. Alternatively, the ISP may have already determined the target LNS from DNIS. If the LNS is willing to accept tunnel creation without any authentication of the caller, the LAC may tunnel the PPP connection without ever having communicated with the remote user.

Step 4 If no tunnel connection currently exists to the desired LNS, one is initiated. L2TP is designed to be largely insulated from the details of the media over which the tunnel is established; L2TP requires only that the tunnel media provide packet-oriented, point-to-point connectivity. Obvious examples of such media are UDP, Frame Relay PVCs, and X.25 VCs.

Step 5 After the tunnel exists, an unused slot within the tunnel (a *Call ID*), is allocated and a connect indication is sent to notify the LNS of this new dial-up session. The LNS either accepts the connection or rejects it.

Step 6 If the LNS accepts the connection, it creates a virtual interface for PPP in a manner analogous to what it would use for a direct-dialed connection. With this virtual interface in place, Link layer frames can now pass through the tunnel in both directions. Frames from the remote user are received at the POP; stripped of CRC, link framing, and transparency bytes; encapsulated in L2TP; and forwarded over the appropriate tunnel.

The LNS accepts these frames, strips L2TP, and processes them as normal incoming frames for the appropriate interface and protocol. The virtual interface behaves very much like a hardware interface, with the exception that the hardware in this case is physically located at the ISP POP. The other direction behaves analogously, with the LNS encapsulating the packet in L2TP, and the LAC stripping L2TP before transmitting it out the physical interface to the remote user.

Step 7 At this point, the connectivity is a point-to-point PPP session whose endpoints are the remote user's networking application on one end and the termination of this connectivity into the LNS's PPP support on the other. Because the remote user has become simply another dial-up client of the LNS, client connectivity can now be managed using traditional mechanisms with respect to further authorization, protocol access, and packet filtering.

Using VPDN Technologies

Although the L2TP protocol is what is being worked on in the standards track, L2F and PPTP implementations will still be available from a variety of vendors. Effectively, all three technologies support similar functionality. However, L2TP will probably have more vendor support because it is on the standards track. When considering whether to implement any of the Virtual Private Dial-up Network (VPDN) technologies into a corporate network environment, the differences between the standard Internet access service and the virtual dial-up service should be considered. There are significant differences with respect to authentication, authorization, address allocation, and accounting.

The details of the differences between these services and the problems presented by these differences are described in the following sections. The mechanisms used for virtual dial-up service are intended to coexist with more traditional mechanisms; an ISP's POP should simultaneously service ISP clients and virtual dial-up clients.

Authentication

In a traditional dial-up scenario, an ISP using a NAS in conjunction with a security server follows an authentication process by challenging the remote user for both a username and password. If the remote user passes this phase, the authorization phase can begin.

For the virtual dial-up service, the ISP pursues authentication to the extent required to discover the user's apparent identity (and by implication, the desired corporate gateway). No password interaction is performed at this point.

As soon as the corporate gateway is determined, a connection is initiated with the authentication

information gathered by the ISP. The corporate gateway completes the authentication by either accepting or rejecting the connection. (For example, the connection is rejected in a PAP request in which the username or password is found to be incorrect.) After the connection is accepted, the corporate gateway can pursue another phase of authentication at the PPP layer. These additional authentication activities are outside the scope of the specification but can include proprietary PPP extensions or textual challenges carried within a TCP/IP Telnet session.

Note For each L2TP tunnel established, L2TP tunnel security generates a unique random key to resist spoofing attacks. Within the L2TP tunnel, each multiplexed session maintains a sequence number to prevent the duplication of packets.

Authorization

When providing a traditional dial-up service, the ISP is required to maintain per-user profiles defining the authorization. Thus a security server could interact with the NAS to provide policy-based usage to connecting users based on their authentication. These policy statements can range from simple source/destination filters for a handful of sites to complex algorithms that determine specific applications, time of day access, and a long list of permitted or denied destinations. This process can become burdensome to the ISP, especially if it is providing access to remote users on behalf of corporations that require constant change to this policy.

In the virtual dial-up service, the burden of providing detailed authorization based on policy statements is given directly to the remote user's corporation. By allowing end-to-end connectivity between remote users and the corporate gateway, all authorization can be performed as if the remote users had dialed directly into the corporate location. This setup frees the ISP from having to maintain a large database of individual user profiles for many different corporations. More importantly, the virtual dial-up service becomes more secure for the corporations using it because it allows the corporations to quickly react to changes in their remote user community.

Addressing

For a traditional Internet service, the user accepts that the IP address may be allocated dynamically from a pool of service provider addresses. This model often means that remote users have little or no access to their corporate network's resources because firewalls and security policies deny access to the corporate network from external IP addresses.

For the virtual dial-up service, the corporate gateway can exist behind the corporate firewall and allocate addresses that are internal (and that can, in fact, be RFC 1597 addresses or non-IP addresses). Because L2TP tunnels operate exclusively at the frame layer, the actual policies of such address management are irrelevant to correct virtual dial-up service; for all purposes of PPP protocol handling, the dial-in user appears to have connected at the corporate gateway.

Accounting

The requirement that both the NAS and the corporate gateway provide accounting data can mean that they may count packets, octets, and connection start and stop times.

Because virtual dial-up is an access service, accounting of connection attempts (in particular, failed connection attempts) is of significant interest. The corporate gateway can reject new connections based on the authentication information gathered by the ISP, with corresponding logging. For cases where the corporate gateway accepts the connection and then continues with further authentication, the corporate gateway can subsequently disconnect the client. For such scenarios, the disconnection indication back to the ISP can also include a reason for why the disconnect occurred.

Because the corporate gateway can decline a connection based on the authentication infor-mation collected by the ISP, accounting can easily draw a distinction between a series of failed connection attempts and a series of brief successful connections. Lacking this facility, the corporate gateway must always accept connection requests and would have to exchange numerous PPP packets with the remote system.

Advantages of Using VPDNs

Table 2-2 shows the advantages of a virtual dial-up service.

Table 2-2: Advantages of VPDN Services

Features	Benefits
Multiprotocol support	ISP can provide multiprotocol services over an IP-only backbone, leveraging facilities, management techniques, personnel, and applied training of the current infrastructure.
User authentication performed at remote user's corporation	ISP is not required to maintain a per-user authentication database. ISP does not have to respond to organizational changes at the corporate location. Corporations are not required to "trust" the ISP's authentication procedures.
User authorization performed at remote user's corporation	ISP is not required to maintain per-user access lists. Simplified firewall management. Corporations can enforce their own security policies.
Simultaneous support for local access	The NAS can be used by theISP for both standard Internet access and the virtual dial-up service, reducing costs, equipment, and infrastructure requirements.
Address allocation performed by remote user's corporation using end-to-end tunnels	The ISP is not required to maintain the corporation's address space within the ISP network. This minimizes the route table carried by the ISP, improves scalability, and supports the corporate use of unregistered addresses across the Internet and public networks.

Media independence	The ISP can leverage any media (Frame Relay, ATM, Point-to-Point, X.25) in the backbone to support the virtual dial-up service.
Dynamic tunnel	Tunnels are initiated and management torn down based on L2TP management. This setup provides a scalable solution because tunnels are initiated only when user traffic is active. Minimizes the NAS resources required to maintain tunnels.
Multiple remote user sessions are multiplexed over a single L2TP tunnel	This is a scalable solution because it minimizes the number of tunnels required to be open at a given time. PVC-based backbone infrastructures such as Frame Relay need only a single PVC between the NAS and the corporate gateway to manage multiple remote user sessions.
Tunnel security maintains random key and sequence numbers	Tunnel establishment involves a NAS (ISP)-to-corporate-gateway authentication process to protect against attacks. In addition, L2TP resists spoofing by using sequence numbers.
No routing protocol dependencies	Neither the ISP nor the corporate customer is required to manage the other's routing domain to provide access and services, freeing both to use whichever routing protocols suits them best.

Additional Considerations

With any of the VPDN technologies, PPP authentication is used to authenticate users or devices; tunnel endpoints may periodically re-authenticate. However, there is no protection for individual packets (either data or control) that traverse the established tunnel. There is work in progress that proposes using IPsec transport mode to secure the VPDN tunnel traffic. In addition, for individual data packets traveling through the VPDN tunnel, security services including authentication, integrity, replay protection, and confidentiality can be provided by using IPsec in conjunction with L2F, PPTP, or L2TP.

Public Key Infrastructure and Distribution Models

Many security protocols rely on public-key cryptography to provide services such as confidentiality, data integrity, data origin authentication, and non-repudiation. The purpose of a Public Key Infrastructure (PKI) is to provide trusted and efficient key and certificate management to support these protocols.

A PKI is defined by the Internet X.509 Public Key Infrastructure PKIX Roadmap "work in progress"

document as follows:

The set of hardware, software, people, policies, and procedures needed to create, manage, store, distribute, and revoke certificates based on public-key cryptography.

A PKI consists of the following five types of components (taken from NIST Special Publication 800-15, *Minimum Interoperability Specification for PKI Components, Version 1*, September 1997, by William Burr, Donna Dodson, Noel Nazario, and W. Timothy Polk):

- *Certification Authorities (CAs)* that issue and revoke certificates
- *Organizational Registration Authorities (ORAs)* that vouch for the binding between public keys, certificate holder identities, and other attributes
- *Certificate holders* that are issued certificates and that can sign digital documents
- Clients that validate digital signatures and their certification paths from a known public key of a trusted CA
- Repositories that store and make available certificates and Certificate Revocation Lists (CRLs)

Functions of a PKI

The functions of a PKI can be summarized as follows:

- *Registration.* The process whereby a subject first makes itself known to a CA (directly or through a registration authority [RA]) before that CA issues a certificate or certificates for that subject.
- *Initialization.* The point at which the user or client system gets the values it needs to begin communicating with the PKI. For example, initialization can involve providing the client system with the public key or the certificate of a CA, or generating the client system's own public/private key pair.
- *Certification.* The process in which a CA issues a certificate for a subject's public key and returns that certificate to the subject (or posts that certificate in a repository).
- *Key Pair Recovery.* If the CA has generated and issued the key pair, the user's private key can be either backed up by a CA, or by a separate key backup system. If a user or his/her employer wants to recover these backed-up key materials, the PKI must provide a system that permits the recovery *without* providing an unacceptable risk of compromise of the private key.
- *Key Generation.* Depending on the CA's policy, the private/public key pair can either be generated by the user in his local environment, or be generated by the CA. In the latter case, the key material may be distributed to the user in an encrypted file or on a physical token (such as a smart card or PCMCIA card).
- *Key Update.* All key pairs must be updated regularly---that is, replaced with a new key pair---and new certificates must be issued. This happens in two cases: normally, when a key has passed its maximum usable lifetime; and exceptionally, when a key has been compromised and must be replaced.
- *Cross-Certification.* A certificate is issued by one CA to another CA; the certificate contains a public CA key associated with the private CA signature key used for issuing certificates. Typically, a cross-certificate is used to allow client systems and end entities in one administrative domain to communicate security with client systems and end users in another administrative

domain.

- *Revocation.* When a certificate is issued, it is expected to be in use for its entire validity period. However, various circumstances may cause a certificate to become invalid before the expiration of the validity period. Such circumstances include change of name, change of association between subject and CA (for example, an employee terminates employment with an organization), and compromise or suspected compromise of the corresponding private key. Under such circumstances, the CA must revoke the certificate.

A Sample Scenario Using a PKI

Figure 2-29 shows an example of two entities communicating with a common CA, using digital certificates to validate public keys.

Figure 2-29: Digital Certificate Communication

Both routers and the CA have a public/private key pair. Initially, the CA has to enroll an X.509 v3 certificate for both routers in a secure manner. Also, both routers must receive a copy of the CA's public key in a secure manner. Now, if the router in New York has traffic to send to the router in Paris and wants authenticated, confidential delivery of the data, the following steps must occur:

Step 1 The New York router sends a request to the CA (for example, it makes an LDAP query) to obtain the Paris router's public key.

Step 2 The CA sends the Paris router's certificate, signed with its own private key.

Step 3 The New York router verifies the signature with the CA's public key to validate the Paris router's public key.

Step 4 The Paris router sends a request to the CA to obtain the New York router's public key.

Step 5 The CA sends the New York router's certificate, signed with its own private key.

Step 6 The Paris router verifies the signature with the CA's public key to validate the New York router's public key.

Now, both routers have each other's public key and can use public key encryption to send authenticated, confidential data.

Typically, an authenticated Diffie-Hellman exchange, as explained in Chapter 1, would take place to derive a shared key for secret key encryption because secret key encryption is usually used for bulk data encryption (it is much faster computationally).

Note The way certificates are exchanged can vary with implementations. For example, in IPsec, IKE allows the certificate to be accessed independently (for example, through DNSSEC) or by having two devices explicitly exchange certificates as part of IKE.

Certificates

Users of public key-based systems must be confident that, any time they rely on a public key, the associated private key is owned by the subject with which they are communicating. (This applies whether an encryption or digital signature mechanism is used.) This confidence is obtained through the use of *public key certificates*, which are data structures that bind public key values to subjects. The binding is achieved by having a trusted CA verify the subject's identity and digitally sign each certificate. The purpose of a CA, therefore, is to bind a public key to the common name of the certificate and, thus, assure third parties that some measure of care has been taken to ensure that this binding is valid.

The CA paradigm essentially relies on an authentication chain that ends in a CA that eventually certifies itself. The problem is shifted from a local perspective to a global perspective, with the whole chain depending on one final link.

A certificate has a limited valid lifetime, indicated in its signed contents. Because a certificate's signature and timeliness can be independently checked by a certificate-using client, certificates can be distributed using untrusted communications and server systems and can be cached in unsecured storage in certificate-using systems.

Certificates are used in the process of validating signed data. Specifics vary according to which algorithm is used, but the general process works as follows:

1. The recipient of signed data verifies that the claimed identity of the user is in accordance with the identity contained in the certificate.

2. The recipient validates that no certificate in the path has been revoked (for example, by retrieving a suitably current CRL or querying an online certificate status responder), and that all certificates were within their validity periods at the time the data was signed.

3. The recipient verifies that the data does not claim to have any attributes for which the certificate indicates that the signer is not authorized.

4. The recipient verifies that the data has not been altered since it was signed by using the public key in the certificate.

If all of these checks pass, the recipient can accept that the data was signed by the purported signer. The process for keys used for encryption is similar to the preceding process.

NOTE It can, of course, be possible that data was signed by someone very different from the signer (for example, if the purported signer's private key was compromised). Security depends on all parts of the certificate-using system, including but not limited to, the following:

· The physical security of the place in which the computer resides

· Personnel security (the trustworthiness of the people who actually develop, install, run, and maintain the

system)

· The security provided by the operating system on which the private key is used

· The security provided the CA

A failure in any one of these areas can cause the entire security system to fail.

The X.509 Standard

The X.509 standard constitutes a widely accepted basis for a PKI infrastructure, defining data formats and procedures related to the distribution of public keys using certificates digitally signed by CAs. RFC 1422 specified the basis of an X.509-based PKI, targeted primarily at satisfying the needs of Internet privacy-enhanced mail (PEM). Since RFC 1422 was issued, application requirements for an Internet PKI have broadened tremendously, and the capabilities of X.509 have greatly advanced. Much work is being done to use digital certificates in Web, email, and IPsec applications. The current standards define the X.509 Version 3 certificate and Version 2 CRL.

X.509 V3 Certificate

The information contained in the certificate must be uniform throughout the PKI. The current proposed standard to provide a common baseline for the Internet uses a X.509 V3 certificate format (see Figure 2-30).

Figure 2-30: The X.509 V3 Certificate Format

Every certificate contains three main fields:

- The body of the certificate
- The signature algorithm
- The signature itself

The *body* of the certificate contains the version number, the serial number, the names of the issuer and subject, a public key associated with the subject, and an expiration date (not before and not after a specified time/date); some certificate bodies contain *extensions*, which are optional unique identifier fields that associate additional attributes with users or public keys. The *signature algorithm* is the algorithm used by the CA to sign the certificate. The *signature* is created by applying the certificate body as input to a one-way hash function. The output value is encrypted with the CA's private key to form the signature value, as shown in Figure 2-31.

Figure 2-31: Creating a Digital Signature for an X.509 V3 Certificate

X.509 V2 CRL

When a certificate is issued, it is expected to be in use for its entire validity period. However, various circumstances may cause a certificate to become invalid before the validity period expires. Such circumstances might include a change of name, a change of association between the subject and CA (for example, an employee terminates employment with an organization), and the compromise or suspected compromise of the corresponding private key. Under such circumstances, the CA should revoke the certificate.

X.509 defines one method of certificate revocation. This method requires each CA to periodically issue a signed data structure called a *certificate revocation list (CRL)*. A CRL is a timestamped list that identifies revoked certificates. The CRL is signed by a CA and made freely available in a public repository. Each revoked certificate is identified in a CRL by its certificate serial number. When a certificate-using system uses a certificate (for example, to verify a remote user's digital signature), that system not only checks the certificate signature and validity but also acquires a suitably recent CRL and checks that the certificate serial number is *not* on that CRL.

The meaning of "suitably recent" can vary with local policy, but it usually means the most recently issued CRL. A CA issues a new CRL on a regular basis (hourly, daily, or weekly). CAs may also issue CRLs at unpredictable time intervals (for example, if an important key is deemed compromised, the CA may issue a new CRL to expedite notification of that fact, even if the next CRL does not have to be issued for some time).

Note A problem of unpredictable CRL issuance is that end-entities may not know that a new CRL has been issued and, thus, may not retrieve it from a repository.

An entry is added to the CRL as part of the next update following notification of revocation. An entry can be removed from the CRL after it appears on one regularly scheduled CRL issued beyond the revoked certificate's validity period.

An advantage of the CRL revocation method is that CRLs can be distributed in exactly the same way as certificates themselves: using untrusted communications and server systems.

One limitation of the CRL revocation method, using untrusted communications and servers, is that the time granularity of revocation is limited to the CRL issue period. For example, if a revocation is reported now, that revocation will not be reliably notified to certificate-using systems until the next CRL is issued---which may be up to one hour, one day, or one week, depending on the frequency at which the CA issues CRLs.

Certificate Distribution

A variety of protocols are under consideration to facilitate the distribution of digital certificates. These include widely used file retrieval mechanisms (such as FTP and HTTP) or specifically designed directory access protocols (such as LDAP). Because FTP and HTTP are assumed to be understood, only LDAP is discussed (in the following section) to give a high-level view of what it is.

Lightweight Directory Access Protocol

The *Lightweight Directory Access Protocol (LDAP)* is used for accessing online directory services. LDAP was developed by the University of Michigan in 1995 to make it easier to access X.500 directories. X.500 was too complicated and required too much computer power for many users so a simplified version was created. LDAP is specifically targeted at manage-ment applications and browser applications that provide read/write interactive access to directories. When used with a directory that supports the X.500 protocols, LDAP is intended to be a complement to the X.500 DAP. The LDAP V2 protocol is defined in RFC 1777. Currently, work is in progress on Version 3 (an Internet draft).

LDAP runs directly over TCP and can be used to access a stand-alone LDAP directory service or to access a directory service that is back-ended by X.500. The standard defines the following:

- A network protocol for accessing information in the directory
- An information model defining the form and character of the information (called a *schema*)
- A namespace defining how information is referenced and organized
- An emerging distributed operation model defining how data may be distributed and referenced (in Version 3)

The general model adopted by LDAP is one of clients performing protocol operations against servers. In this model, a client transmits a protocol request describing the operation to be performed to a server. The server is then responsible for performing the necessary operation(s) in the directory. After completing the operation(s), the server returns a response containing any results or errors to the requesting client.

In LDAP Versions 1 and 2, no provision was made for protocol servers returning referrals to clients. Rather, if the LDAP server does not know the answer to a query, it goes to another server for the information rather than sending a message to the user telling the user to go to that other server. However, for improved performance and distribution, this version of the protocol permits servers to return client's referrals to other servers. This approach allows servers to offload the work of contacting other servers to progress operations.

The LDAP protocol assumes that there are one or more servers that jointly provide access to a *Directory Information Tree (DIT)*. Each tree is made up of entries that contain names and one or more attribute values from the entry form its *relative distinguished name (RDN)*, which must be unique among all its siblings. The concatenation of the RDNs of the sequence of entries from a particular entry to an immediate subordinate of the root of the tree forms that entry's *distinguished name (DN)*, which is unique in the tree.

Some servers may hold *cache* or *shadow* copies of entries, which can be used to answer search and comparison queries, but will return referrals or contact other servers if modification operations are requested.

Summary

This chapter detailed many of the current and evolving technologies relating to security. One of the most important security considerations is establishing the identity of the entity that wants to access the corporate network. This process usually entails authenticating the entity and subsequently authorizing that entity and establishing access controls. Some protocols are specifically designed to only authenticate end-users (people) or end-devices (hosts, routers). Frequently, you have to combine the two protocols so that both end-users and the end-devices they are using to access the network are authenticated.

In addition to establishing identity, you must ensure data integrity and confidentiality; that is, you must protect the data traversing the corporate network. Many technologies exist to provide security services for various TCP/IP layers. Although Application layer security protocols provide the most flexibility for application-specific parameters, using a different security protocol for every application is not practical. Transport security protocols such as SSL and SSH are widely deployed. SSL is bundled into many Web servers and clients and has become a *de facto* standard in securing Web transactions; SSH is most often used for securing Telnet or FTP transactions. IPsec is becoming widely deployed and can offer security services for the Transport and Application layer traffic on a per-packet basis. IPsec should be able to secure Telnet, FTP, and Web traffic but may be harder to scale until client support is more readily available on many platforms.

For dial-in security, protocols such as L2F, PPTP, and L2TP can offer many advantages for corporations. These protocols can provide a way for dial-in users to use the Internet to securely communicate back to the corporate network. However, the packets traversing the secured tunnels are not protected, and it is prudent to add more security with Transport or Network layer security protocols to protect the traffic.

Many of the security protocols discussed in this chapter require either an exchange of crypto-graphic keys or digital certificates. A PKI is required to provide trusted and efficient key and certificate management. PKIs are being implemented in corporations or in a more global fashion, but this particular area is still developing and should be watched carefully in the upcoming years.

All the technologies discussed in this chapter will keep evolving; those readers interested in additional technical details and the latest developments should refer to the work performed by the IETF working groups, which is listed in Appendix A, "Sources of Technical Information."

Posted: Wed Jun 14 11:41:05 PDT 2000

Table of Contents

Export Controls on Cryptography

3

Export Controls on Cryptography

Historically, cryptography has been used as a way to send secret messages between warring nations; as such, it became an important instrument in national security. With the increasing need for secure transactions for data traversing computer networks for medical, financial, and other critical applications, cryptography is now becoming a necessity for nongovernmental, nonmilitary applications.

All over the globe, the laws and regulations concerning cryptography are undergoing a vast change. Legal restrictions on the import and export of cryptographic products are being debated and modified. Here's a list of the major issues being debated:

- *Key length*. The combination of the algorithm and the key length are factors of cryptographic strength. The algorithm is usually well known. The longer the key, the stronger the cryptographic strength of a given algorithm. Some countries have export laws that limit the key length of a given cryptographic algorithm.
- *Key recovery*. In recent years, export laws have been modified if the cryptographic algorithm

includes the capability of incorporating key recovery methods. These modified laws enable governments to wire-tap for encrypted electronic data if they deem it necessary to do so.

- *Cryptography use*. A distinction is sometimes made about whether cryptography is used for authentication and integrity purposes or for confidentiality purposes. When used for confidentiality, the export laws are typically much more stringent.

Historical Perspective on U.S. Policy

In the United States, cryptography export used to be controlled by the International Traffic in Arms Regulation (ITAR) because cryptography was deemed to serve both civilian and military purposes and was placed on the United States Munitions List (USML). If an article or service is placed on the USML, its export is regulated exclusively by the State Department. ITAR controls software that "includes but is not limited to the system functional design, logic flow, algorithms, application programs, operating systems and support software for design, implementation, test, operation, diagnosis, and repair" of equipment controlled by the USML. Except for Canada in most cases, export of software controlled by ITAR requires a validated export license issued by the Office of Defense Trade Controls (DTC) of the State Department. With respect to Canada, most software for unclassified defense items is excluded from the validated license requirement.

In the past, ITAR had jurisdiction over all software that had data-encryption capability except for commercial software with encryption limited to these functions:

- Decryption-only capability for encrypted proprietary software, fonts, or other computer-related proprietary information for the purpose of maintaining vendor control over such information.
- Restrictions on calculating a Message Authentication Code (MAC) or similar result to ensure that no alteration of text, or authentication of users, had taken place; these restrictions did not allow for encryption of data, text, or other media other than that needed for authentication.
- Restrictions to protecting passwords and personal identification numbers (PINs) or similar data to prevent unauthorized access to computing facilities; these restrictions did not allow for encryption of files or text, except as directly related to password and PIN protection.
- Specifically designed and limited to the issuance of cash or travelers' checks, acceptance of deposits, account-balance reporting, and similar financial functions.
- Software for personalized smart cards restricted for uses described in the preceding bullet items.

Software that performed any of these functions---as well as any additional encryption functions---came under the jurisdiction of ITAR. For example, a software package that encrypted passwords as well as data files on a hard disk was under ITAR jurisdiction.

The State Department, which had the sole authority to determine the export licensing jurisdiction of a product, controlled all software in source code form with encryption capability under ITAR jurisdiction, even if the encryption functions were limited to those identified in the preceding list.

Personal-Use Policy

In February 1996, the ITAR rules were amended regarding personal use of cryptography. Temporary export of products for personal use was exempted from the need for a license---provided that the exporter

take normal precautions to ensure the security of the product, including locking the product in a hotel room or safe. The product must not be intended for copying, demonstration, marketing, sale, re-export, or transfer of ownership or control. In transit, the product must remain with the exporter's accompanying baggage. The exporter must keep records of each export for five years. Export to embargoed countries (such as Cuba, Libya, or Syria) is prohibited. In February 1997, the Department of Commerce announced that it would revise the new regulations to clarify (among other things) the personal-use exemption for laptop computers.

An interesting account of the "personal-use policy" rules can be found in a paper written by Matt Blaze, titled "My Life as an International Arms Courier." This paper can be found at the following site: ftp://ftp.research.att.com/dist/mab/export.txt.

On October 1, 1996, the Clinton Administration announced a new policy for exporting encryption technology. Much of this policy was based on the idea of creating a worldwide key management infrastructure that used key escrow and key recovery mechanisms to promote electronic commerce and to secure communications while protecting national security and public safety. The policy allows vendors to export encryption products with a strength of 56-bit DES (or equivalent), provided that vendors make satisfactory commitments to build key recovery mechanisms and to help build the supporting international infrastructure. Temporary export licenses would be granted for periods from six months up to two years, with renewals issued after meeting specified commitments and milestones. This policy applied to hardware and software and would last for two years. After two years, 56-bit products without key recovery would no longer be exportable.

A major progressive step was taken when President Clinton signed an executive order on November 15, 1996, that shifted jurisdiction over encryption export controls (all encryption items controlled on the U.S. Munitions List, except those specifically designed, developed, configured, adapted, or modified for military applications) from the State Department to the Commerce Department and gave the FBI new authority over exports.

The Commerce Department created a draft of the Export Administration Regulations the following month. The new export rules distinguish five categories of *encryption items (EIs)*:

- Certain mass-market encryption software may be released from EI controls after a one-time review.
- *Data recovery crypto* (meaning that the government can access keys or plaintext with a lawful warrant) will be eligible for an export license to non-embargoed countries.
- After a one-time review, (up to) 56-bit cryptography can be granted a six-month export license, provided that the exporting business commits itself to incorporating a data-recovery feature in its products within the next two years. This relaxation of controls lasted until January 1, 1999. After two years, the export of non-recovery 56-bit cryptography will be prohibited again, and the same situation as before will hold (maximum 40-bit key length, with exceptions for financial institutions).
- All other encryption items may be eligible for encryption licensing arrangements; items not authorized under a licensing arrangement will be considered on a case-by-case basis.
- Encryption "technology" may be licensed for export on a case-by-case basis. In April 1997, the president's Export Council Subcommittee on Encryption was established to advise the Secretary for Export Administration on the implementation of crypto export policy; the committee consists

of approximately 25 members from the exporting community and government agencies.

Additional liberalization of export controls were announced in May 1997. The new policy allowed the export of products without key recovery requirements and an unlimited key length if the products were used solely for financial transactions. This policy was extended in July 1998 to allow the general exportability of products without key recovery mechanisms to financial institutions in 45 countries. (These 45 countries agreed to take steps against money laundering.) These financial institutions would be able to distribute the encryption technology among their international branches (except in terrorist states) after the initial approval was granted.

The export policy was further liberalized in September 1998 to include export of 56-bit DES and equivalent products, without requiring key recovery mechanisms, to approved industries worldwide, including subsidiaries of U.S. firms, insurance companies, health and medical organizations, and online merchants. In all cases however, there is the requirement of a one-time product technical review before the export is allowed.

The changes in U.S. encryption export policy will most likely continue. To get more information on updates on U.S. encryption export regulations, refer to the following Web sites:

- Code of Federal Regulations---International Traffic in Arms Regulations: http://www.epic.org/crypto/export_controls/itar.html
- Current and historical information on US commercial encryption export controls: http://www.bxa.doc.gov/Encryption
- Press release from the Department of Commerce---Bureau of Export Administration (BXA) on Update on Export Controls of Encryption Products (12/30/98): http://www.bxa.doc.gov/PRESS/98/1230Encryption.html

Historical Perspective on International Policy

Internationally, encryption export and import controls are also undergoing vast changes. The Coordinating Committee for Multilateral Export Controls (COCOM) was an international organization that provided common export controls of strategic products and technical data from country members to prescribed destinations. COCOM provided an agreement to control the export and handling of sensitive technologies including supercomputers, fast DSP chips, crypto, lasers, precision CNC milling/machining equipment, and so on. COCOM was, in effect, a generalized effort by the West during the Cold War to prevent the flow of technology into communist countries (not just the USSR). It maintained, among others lists, the International Industrial List and the International Munitions List.

COCOM's 17 members were Australia, Belgium, Canada, Denmark, France, Germany, Greece, Italy, Japan, Luxembourg, the Netherlands, Norway, Portugal, Spain, Turkey, the United Kingdom, and the United States. Cooperating members included Austria, Finland, Hungary, Ireland, New Zealand, Poland, Singapore, Slovakia, South Korea, Sweden, Switzerland, and Taiwan.

One of the goals of the COCOM regulations was to prevent cryptography from being exported to "dangerous" countries---usually, countries thought to maintain friendly ties with terrorist organizations, such as Libya, Iraq, Iran, and North Korea. Exporting to other countries is usually allowed although states often require that a license be granted.

In 1991, COCOM adopted the General Software Note (GSN), which effectively allowed export of mass-market cryptographic software (including public domain software). Most member countries of COCOM followed its regulations, but the United States, France, and the United Kingdom maintained separate regulations. COCOM was dissolved in March 1994. Pending the signing of a new treaty, most members of COCOM agreed in principle to maintain the status quo, and cryptography remained on export control lists.

In 1995, 28 countries decided to establish a follow-up to COCOM, the Wassenaar Arrangement on Export Controls for Conventional Arms and Dual-Use Goods and Technologies. The negotiations on the treaty were finished in July 1996, and the agreement was signed by 31 countries (Argentina, Australia, Austria, Belgium, Canada, the Czech Republic, Denmark, Finland, France, Germany, Greece, Hungary, Ireland, Italy, Japan, Luxembourg, the Netherlands, New Zealand, Norway, Poland, Portugal, the Republic of Korea, Romania, the Russian Federation, the Slovak Republic, Spain, Sweden, Switzerland, Turkey, the United Kingdom, and the United States). Later, Bulgaria and Ukraine also signed the treaty.

The Wassenaar Arrangement controls the export of weapons and of *dual-use goods*, that is, goods that can be used both for a military and a civil purpose. Cryptography is such a dual-use good. The provisions are largely the same as those set forth by the COCOM regulations.

Because it was generally recognized that an internationally coordinated approach to encryption policy would help the development of a secure global electronic infrastructure, the Organization for Economic Cooperation and Development (OECD) in 1996 issued guidelines for cryptography policy. It issued a news release in March 1997 that included the eight basic principles for cryptography policy. (Read the complete release at http://www.oecd.org/news_and_events/release/nw97-24a.htm.) The guidelines are nonbinding recommendations to member governments, meaning that they will not be part of international law. The guidelines provide principles that countries should take into account when developing a national crypto policy. The principles are given here:

1. Cryptographic methods should be trustworthy in order to generate confidence in the use of information and communications systems.

2. Users should have a right to choose any cryptographic method, subject to applicable law.

3. Cryptographic methods should be developed in response to the needs, demands, and responsibilities of individuals, businesses, and governments.

4. Technical standards, criteria, and protocols for cryptographic methods should be developed and promulgated at the national and international levels.

5. The fundamental rights of individuals to privacy---including secrecy of communications and protection of personal data---should be respected in national cryptography policies and in the implementation and use of cryptographic methods.

6. National cryptography policies may allow lawful access to plaintext and cryptographic keys of encrypted data. These policies must respect the other principles contained in the guidelines to the greatest extent possible.

7. Whether established by contract or legislation, the liability of individuals and entities that offer cryptographic services and those that hold or access cryptographic keys should be clearly stated.

8. Governments should cooperate to coordinate cryptography policies. As part of this effort, governments should remove, or avoid creating in the name of cryptography policy, unjustified obstacles to trade.

Table 3-1 shows a synopsis of the worldwide encryption policies as known to date. The information is taken from work done by Bert-Jaan Koops and is the most current survey of its kind. More complete information and recent updates are located at http://cwis.kub.nl/~frw/people/koops/lawsurvy.htm.

Table 3-1: Worldwide Encryption Policies

Country	Export/Import Controls	Domestic Laws and Regulations
Argentina	There are no import controls.	There are no controls on crypto use.
Australia	Written permission is needed for exporting cryptographic equipment (this excludes crypto-software transmitted electronically) designed to ensure the secrecy of communications or stored information. There is a personal-use exemption. (Export is allowed for lawful permanent residents, provided that they keep control of the crypto and make sure that it is not transferred anywhere; a record must be kept for three years.)	None.
Austria	Export rules follow EU regulations.	Laws forbid encryption in internal company and organization radio transmissions.
Bangladesh	None.	None.

Belgium	Belgium requires a license for exporting cryptography outside of the Benelux	An unclear 1994 law was amended in 1997 to remove any confusion. The new amendment explicitly states that the use of encryption is free. The provision of indicated encryption services to the public is subject to prior notification (four weeks in advance) to the Belgian Institute of Post and Telecommunications.
Brazil	There are no export or import controls.	There are no controls on crypto use.
Burma	None.	Cryptography is restricted through a licensing regime.
Belarus	None.	For manufacture, repair, and operation of cryptography, a license by the State Security Committee is required. Cryptography use by business people is restricted.
Canada	Canada follows the Wassenaar regulations. The export of items from Canada may be subject to restriction if they are included on the Export Control List. No restrictions exist on imports or exports to the United States.	None.
People's Republic of China	China restricts the import and export of voice-encoding devices.	China may be restricting cryptography, but the authorities have not made a statement about encryption. The legal situation is unclear.
Czech Republic	No export/import controls.	None.
Denmark	There are export controls according to the Wassenaar Arrangement.	None.

Estonia	There are no import controls; however, a permit from the Ministry of Foreign Affairs is required for export and for transit through Estonia.	None.
Finland	For export, a license is required (based on an August 1996 law that implements the EU recommendation on export of dual-use goods). A license is not required if the crypto product is sold freely in retail and does not require extensive vendor support.	None.
France	The import from countries outside the EU and the European Economic Area (EEA) and export of cryptography is regulated according to the law of 1996 (in English; or see the French original) and the decrees implementing it of February 1998. (See the *Journal Officiel* of February 25, 1998, and March 13 and 23, 1998.) Import from within the EU/EEA is free.	Cryptography that does not provide confidentiality can be used without restriction; supply of authentication-only cryptography must still be declared. Use and supply of confidentiality cryptography require authorization. The use of cryptography with key lengths limited to 40 bits is exempted from declaration or authorization if ciphertexts can be cracked in a maximum of 240 trials.
Germany	Export is regulated according to the EU regulation and the Wassenaar Arrangement. The general software note applies. So, mass-market or generally available crypto software does not require a license to export.	None.
Greece	None.	None.

Hungary	There are export controls according to the Wassenaar Arrangement. There are import controls mirroring the export controls, requiring an import license if an export license would be needed in Hungary. Import and export of mass-market encryption software is exempted.	None.
Iceland	None.	None.
India	India requires an import license for encryptors.	None.
Indonesia	The import/export regulation is unclear.	Use of cryptography is said to be prohibited.
Ireland	None.	None.
Israel	There are export controls, but there is no specific limit on key size. Licenses seem to be granted on a case-by-case basis. The import of cryptography also seems to be regulated. The Ministry of Defense is revising the regulations, creating different classes of encryption with different levels of control; certain cryptography products (such as authentication only) would be released from controls.	Israel imposes restrictions on encryption through the Encryption Order, but their scope is not clear. Use of strong encryption may require a license from the army, to be decided on a case-by-case basis; the license is virtually always granted. However, it is unclear whether and how one is required to request a license. No prosecutions for using unlicensed crypto are known, and strong encryption is widely used in Israel.
Italy	COCOM/EU regulations.	Italy has a law that demands encrypted records to be accessible to the treasury.

Japan	Export regulations are designed to implement the Wassenaar Arrangement. Decisions are made on an individual basis. Japan seems to have tightened its export controls in 1996. Now businesses must acquire approval for a cryptography export order larger than 50,000 yen.	None.
Kazakhstan	For importing or exporting cryptographic products, a license is required from the Licensing Commission of the Committee of National Security. The decisions of this committee are subject to judicial review.	A license from the Committee of National Security is required for the development (including research), manufacture, repair (including technical support), and sale (including use and advertising) of cryptographic products.
Latvia	Export controls are in line with the EU dual-use regulation. Import controls mirror the export controls (that is, an import license is required if export is controlled).	There is no domestic restriction of cryptography.
Malaysia	There are no export or import restrictions.	There is no regulation of crypto use.
Mexico	None.	None.

The Netherlands	Export controls are regulated according to the Wassenaar Arrangement. The export of public-domain and mass-market software generally does not require a validated license. Items capable of file encryption do require a validated license for export.	If encrypted files are found in a computer during a house search, the police cannot order or coerce a suspect to decrypt his own material. They can ask/order someone else, who is not under arrest, to provide what information they can about the encrypted material, or they can ask a specialist in the field to decrypt the material.
New Zealand	A license is required to export cryptography. Export of crypto software by electronic means does not fall under the export regulation.	None.
Norway	Norway abides by the COCOM regulations.	None.
Pakistan	None.	The sale and use of encryption hardware and software require approval by Pakistani telecommunications officials.
Philippines	None.	E-mail encryption is not restricted.
Poland	A license is required for exporting encryption software or hardware, in accordance with the EU dual-use goods regulation. A general authorization or import certificate is required to buy encryption products abroad. The end-user must detail the kind of information to be encrypted and where the cryptography is to be installed.	None.
Portugal	None.	None.
Romania	There are no import controls.	None.

Russia	A license is required for the import of encryption facilities manufactured abroad. The export of cryptography is subject to a tightened state control.	In April 1995, President Boris Yeltsin issued a decree prohibiting unauthorized encryption. State organizations and enterprises need a license to use encryption (for both authentication and secrecy, for storage as well as for transmission). The development, production, implementation, or operation of cryptography without a license is prohibited.
Saudi Arabia	None.	It is reported that Saudi Arabia prohibits the use of encryption but that this prohibition is widely ignored.
Singapore	There are no export restrictions. The Trade Development Board (TDB) imposes an import regulation, which requires people to obtain prior approval for importing cryptography hardware or software. The TDB tends to approve crypto software more than crypto hardware. The unofficial policy seems to readily permit import of crypto for banking and credit card transactions.	There is no legislation on cryptography. However, subscribers have to obtain prior approval from the Telecommunications Authority of Singapore (TAS) to send encrypted messages across the Singapore telecommunications lines. TAS is informed by the TDB of its import permits. The use permit may restrict the address from which the encryption product can be used. Selling of the crypto is prohibited without approval from the TDB/TAS.
South Africa	None.	Use of encryption is free for commercial or private organizations. However, any electronic device connected to a telecommunications line must first be approved by Telkom, the tele-communications regulatory body. This restriction does not apply if the device is connected between a modem or router and the computer. Use or provision of cryptography by government bodies requires approval from the relevant agency; likewise,

		crypto systems approved for government use require approval from the relevant agency to be used by commercial or private organizations.
South Korea	South Korea prohibits the import of encryption devices, even for banks.	Encryption policy is not published. It seems that encryption services provided within a public switch telephone network can be restricted on an *ad-hoc* basis. There is no regulation governing the use of encryption.
Spain	Export of cryptography is controlled according to the Wassenaar and EU regulations. The regulation does not make distinctions based on key lengths.	None.
Sweden	Since 1995, Sweden restricts export of encryption according to the Wassenaar Arrangement. Export through the Internet seems to be included in the restrictions. There are no import restrictions.	None.
Switzerland	Switzerland maintains COCOM rules. An export license is required for non-OECD countries.	There is a restriction on the use of certain cryptography: radio communications must remain understandable. If a person wants to encrypt messages for profes-sional reasons, he or she must obtain specific permission.
Turkey	Follows the Wassenaar Agreement.	None.

United Kingdom	Export is controlled in accordance with the EU dual-use regulation; it includes export of crypto software to other EU member states. Crypto export by intangible means (such as over the Internet) is not covered by the regulation and is therefore free if one does not export to embargoed countries and conforms to the Official Secrets Act (copyright, patents, and contracts).	None.
United States of America	There are no import restrictions on cryptography. Cryptography exporting is described at the beginning of the chapter.	None.

A different survey was undertaken by the Electronic Privacy Information Center (EPIC), on behalf of the Global Internet Liberty Campaign (GILC), to provide a comprehensive review of the cryptography policies of virtually every national and territorial jurisdiction in the world. The information can be found at http://www.gilc.org/crypto/crypto-survey.html.

Digital Signatures

Digital signatures will be one of the key elements for the development of (online) financial and business transactions as well as electronic mail. A *digital signature* is an electronic identifier that uses cryptography to ensure the integrity, authenticity, and nonrepudiation of the information to which it corresponds. The legal requirements of a signature or other paper-based method of authentication is often perceived as an obstacle to the use of electronic technologies. Legislation efforts are being made in various parts of the world to eliminate mandatory requirements for handwritten signatures. In spite of these efforts, the most common form of authentication required by national laws remains a *signature*, which is commonly understood to mean the manual writing by an individual of his or her name or initials. Such an interpretation of *signature* is not necessarily supported by the actual words used to define *signature* in legislation.

The following sections look at the issues surrounding the use of digital signatures and how laws are changing to make digital signatures legally binding and admissible in court.

Legal Proof of Authenticity

Authentication of an original document is fundamental to the admissibility of the original document in a court of law. Any copying or conversion process (photocopy, microfilm, electronically scanned image, and so on) must be proven reliable---as must the authenticity of the original document. If there is no capability to authenticate the original document, no amount of reliability evidence with respect to the conversion process will serve to support credibility.

If a court admits a record into evidence, opposing parties are free to dispute the reliability and relevance of the record, and the court can decide that a record is unreliable or relatively irrelevant.

In addition to admissibility and reliability, there are issues relating to burdens and standards of proof and principles of interpretation. A *burden of proof* refers to the persons required to prove the facts (normally the plaintiff or the prosecution). Assuming that the plaintiff or prosecution do have some facts to support their arguments, the defense will want or need to produce some facts to support their own arguments. Generally, the burden of proving a particular fact falls on the party trying to prove that fact.

A *standard of proof* refers to the degree of conviction the judge (or jury) must have about the facts being presented. Generally, there are two standards of proof:

- Beyond a reasonable doubt (the standard for criminal prosecutions)
- The balance of probabilities (the standard for civil proceedings)

All relevant evidence must exhibit two characteristics:

- First, a document must be accepted by the court as being authentic. That is, it must not be counterfeit or have been modified in any way.
- In addition to the document's authenticity, the statements or information contained in the document must be capable of being accurately and independently tested.

When computer-produced evidence takes the form of a printout, it is considered *documentary evidence*. When you submit a printout as evidence, it must satisfy the same statutory or common-law rules pertaining to documentary evidence. That is, the printout must satisfy the same rules as a document with a more traditional origin. However, computer-generated evidence has evolved to the extent that concepts such as "original," "record," and "copy" are challenged. Problems of authentication become more difficult to resolve.

In cases of dispute, proving the origin of a message provides evidence about what actually happened in the disputed circumstances. Proving the origin of a message often means demonstrating that a given message is authentic and not a forgery. In addition, the law may require that there be a way to authenticate a person's intent to enter into certain binding agreements before the law will enforce that agreement. The most common form of authentication required by law is a manual or handwritten signature.

Consider an electronic commerce situation where all financial transactions must be properly authorized, validated, and safeguarded against loss, modification, and destruction. The integrity and confidentiality of the electronic authorization and authentication system must be main-tained at all times. Electronic authorizations of financial transactions must be authenticated. The authentication process must effectively and positively identify the authorizer in such a

way that he or she cannot credibly deny having authorized a transaction (the feature of nonrepudiation).

How Authentic Are Written Signatures?

A written signature is not always authentic. There have been many cases of forged handwritten signatures. The person relying on the document often has neither the names of the persons authorized to sign, nor sample signatures available for comparison. Even when a sample of the authorized signature is available for comparison, only an expert may be able to detect a careful forgery. Paper documents can be lost or destroyed. Even when there is an original handwritten signature, the contract can still be repudiated or declared void for a variety of reasons relating to the law of contract. Thus, handwritten signatures do not by themselves create binding, enforceable agreements.

Note A common method currently used to verify authentication of a written signature is the witness of notary public officials. These officials are considered to be trusted third parties who verify that the person signing a document provided some documentary evidence he or she is, in fact, the person claiming to produce that signature on a document. The electronic equivalent of a notary public is an organization such as Verisign, Inc.; Entrust; or any other digital certificate authority.

Digital Signature Legislation

In the United States, many states are forming digital signature legislation to provide a way to give documents that exist only in electronic form the same legal status as paper documents. This legislation is aimed at providing a secure, reliable, and legally sanctioned method for "signing" electronic documents.

Utah was the first jurisdiction in the United States to enact a statute that puts the force of law behind an electronic signature method. The legislation is known as the Utah Digital Signature Act and was signed by the governor of Utah in March 1995. California passed a digital signature statute in October 1995.

The Utah and the California statutes have different approaches.

The Utah Statute

The Utah statute is detailed and comprehensive and will be supplemented with regulations. It is valid for public and private transaction and the technology used is limited to public key cryptography. For Utah, the definition of digital signature is as follows:

"Digital signature" means a transformation of a message using an asymmetric cryptosystem such that a person having the initial message and the signer's public key can accurately determine whether: (a) the transformation was created using the private key that corresponds to the signer's public key; and (b) the message has been altered since the transformation was made.

In Utah, the Department of Commerce, Division of Corporations and Commercial Code (the "Division") is the agency designated to implement the statute. The Division is a certification authority and may issue, suspend, and revoke certificates in the same way as do licensed certification authorities. In effect, the Division is the certification authority at the top of the chain. The Division is given the power to govern licensed certification authorities, to determine appropriate amounts for "suitable guaranties," to specify various requirements, and to otherwise give effect to and implement the statute.

The Utah statute sets evidentiary presumptions against private key holders and grants statutory liability limits in favor of certification authorities. Regulations to be promulgated by the state government must conform to a detailed set of standards that have been set by statute.

The California Statute

The California statute is much shorter than the Utah statute; it sets forth certain basic principles and then empowers a government agency to create comprehensive regulations. The California statute is for government transaction only and does not specify any specific technology. For California, the definition of a digital signature is as follows:

"Digital signature" means an electronic identifier, created by computer, intended by the party using it to have the same force and effect as the use of a manual signature. The use of a digital signature shall have the same force and effect as the use of a manual signature if, and only if, it embodies all of the following attributes:

1. It is unique to the person using it.
2. It is capable of verification.
3. It is under the sole control of the person using it.
4. It is linked to data in such a manner that if the data is changed, the digital signature is invalidated.
5. It conforms to regulations adopted by the Secretary of State.

The California statute must conform with regulations adopted by the Secretary of State. This statute does not create presumptions or apportion liability. Nor does it provide for the licensing of certification authorities.

Statutes in Other States

In the United States, many state legislatures as well as varying government agencies are putting in place rules and regulations governing the use of digital signatures. Many are following either the Utah or the California model, although serious questions have been raised about whether legislation should be less detailed and less specific about a particular technology, be less pro-industry by giving away liability limits to certification authorities, and pose less burden and risks on the consumer.

As with export regulations, the policies on digital signatures in the United States and in varying countries are changing. A matrix of digital and electronic signature legislation for each of the states in the United States can be found at the following site:

http://www.magnet.state.ma.us/itd/legal/sigleg7.htm

A summary of electronic commerce and digital signature legislation worldwide can be found at this site:

http://www.mbc.com/ds_sum.html

Note I strongly urge any corporation thinking of using digital signatures as legally binding contracts to

consult with their corporate lawyer(s) to fully understand the current U.S. and international laws regarding this area.

Summary

This chapter explored the legal restrictions on the import and export of cryptographic products. These laws are currently in a state of flux as government officials worldwide try to understand the implications of electronic technology on the rapidly evolving Internet-based business models. Around the globe, digital signature legislation is also evolving as a way to give documents that exist only in electronic form the same legal status as paper documents and to provide a secure, reliable, and legally sanctioned method for "signing" electronic documents. You should follow the news in these areas carefully to ensure that any electronic business your corporation is part of follows the current laws on cryptographic export/import and on the use of digital signatures.

continues

continues

continues

Posted: Wed Jun 14 11:30:40 PDT 2000

Table of Contents

Threats in an Enterprise Network

Threats in an Enterprise Network

Today, there is an ever-growing dependency on computer networks for business transactions. With the free flow of information and the high availability of many resources, managers of enterprise networks have to understand all the possible threats to their networks. These threats take many forms, but all result in loss of privacy to some degree and possibly malicious destruction of information or resources that can lead to large monetary losses.

Knowing which areas of the network are more susceptible to network intruders and who is the common attacker is useful. The common trend is to trust users internal to the corporate network and to distrust connections originating from the Internet or from dial-in modem and ISDN lines. It is important to place trust in the employees internal to the network and in authorized people trying to use internal network resources from outside the corporation. Trust must also be weighed with reality. Restricted use of network infrastructure equipment and critical resources is necessary. Limiting network access to only those who require access is a smart way to deter many threats that breach computer network security.

Not all threats are intended to be malicious, but they can exhibit the same behavior and can cause as much harm---whether intended or not. It is important to understand what types of attacks and vulnerabilities are common and what you can do at a policy level to guarantee some degree of safe networking.

This book does not address the many common host application vulnerabilities in detail; instead, it is more concerned with securing the networking infrastructure. In discussions of areas in which host vulnerabilities can be deterred or constrained in the network infrastructure, more details are given.

Types of Threats

Many different types of threats exist, but many threats fall into three basic categories:

- Unauthorized access
- Impersonation
- Denial of service

Unauthorized Access

Unauthorized access is when an unauthorized entity gains access to an asset and has the possibility to tamper with that asset. Gaining access is usually the result of intercepting some information in transit over an insecure channel or exploiting an inherent weakness in a technology or a product.

The ease or difficulty of packet snooping (also known as *eavesdropping*) on networks depends largely on the technology implemented. Shared media networks are particularly susceptible to eavesdropping

because this type of network transmits packets everywhere along the network as they travel from the origin to the final destination. When concentrators or hubs are used in a shared media environment (such as FDDI, 10Base-T, or 100Mbps Ethernet), it can be fairly easy to insert a new node with packet-capturing capability and then snoop the traffic on the network. As shown in Figure 4-1, an intruder can tap into an Ethernet switch and, using a packet-decoding program, such as EtherPeek or TCPDump, read the data crossing the Ethernet.

Figure 4-1: Unauthorized Access Using an Ethernet Packet Decoder

In this example, the intruder gains access to user name/password information and sensitive routing protocol data using an Ethernet packet decoder such as EtherPeek. The data packets being sent are captured by the laptop running EtherPeek; the program decodes the hex data into human-readable form. After access to information is attained, the intruder can use this information to gain access to a machine and then possibly copy restricted, private information and programs. The intruder may also subsequently have the capability of tampering with an asset; that is, the intruder may modify records on a server or change the content of the routing information.

In recent years, it has been getting much easier for anyone with a portable laptop to acquire software that can capture data crossing data networks. Many vendors have created user-friendly (read *easy-to-use*) packet decoders that can be installed with minimal cost. These decoders were intended for troubleshooting purposes but can easily become tools for malicious intent.

Packet snooping by using these decoding programs has another effect: The technique can be used in impersonation attacks, which are discussed in the next section.

Packet snooping can be detected in certain instances, but it usually occurs without anyone knowing. For packet snooping to occur, a device must be inserted between the sending and receiving machines. This task is more difficult with point-to-point technologies such as serial line connections, but it can be fairly easy with shared media environments. If hubs or concentrators are used, it can be relatively easy to insert a new node. However, some devices are coming out with features that remember MAC addresses and can detect if a new node is on the network. This feature can aid the network manager in noticing whether any suspicious devices have been added to the internal network.

In Figure 4-2, a 10Base-T Ethernet switch provides connectivity to several hosts. The switch learns the source MAC addresses of the connecting hosts and keeps an internal table representing the MAC address and associated ports. When a port receives a packet, the switch compares the source address of that packet to the source address learned by the port. When a source address change occurs, a notification is sent to a management station, and the port may be automatically disabled until the conflict is resolved.

Figure 4-2: Port Security on Ethernet Switches

The best way to deter unauthorized access is by using confidentiality and integrity security services to ensure that traffic crossing the insecure channel is scrambled and that it cannot be modified during transit.

Table 4-1 lists some of the more common access breaches and how they are a threat to corporate networks.

Table 4-1: Common Unauthorized Access Scenarios

Ways of Obtaining Unauthorized Access	Ways to Use Unauthorized Access
Establishing false identity with false credentials	Sending email that authorizes money transfers or terminating an employee
Physical access to network devices	Modifying records to establish a better credit rating
Eavesdropping on shared media networks	Retrieving confidential records, such as salary for all employees or medical histories

Impersonation

Impersonation is closely related to unauthorized access but is significant enough to be discussed separately. *Impersonation* is the ability to present credentials as if you are something or someone you are not. These attacks can take several forms: stealing a private key, gaining access to a cleartext user name/password pair, or even recording an authorization sequence to replay at a later time. In large corporate networks, impersonation can be devastating because it bypasses the trust relationships created for structured authorized access.

Impersonation can come about from packet spoofing and replay attacks. *Spoofing attacks* involve providing false information about a principal's identity to obtain unauthorized access to systems and their services. A *replay attack* can be a kind of spoofing attack because messages are recorded and later sent again, usually to exploit flaws in authentication schemes. Both spoofing and replay attacks are usually a

result of information gained from eavesdropping. Many packet snooping programs also have packet-generating capabilities that can capture data packets and then later replay them.

Impersonation of individuals is common. Most of these scenarios pertain to gaining access to authentication sequences and then using this information to attain unauthorized access. Once the access is obtained, the damage created depends on the intruder's motives. If you're lucky, the intruder is just a curious individual roaming about cyberspace. However, most of us will not be that lucky and will find our confidential information compromised and possibly damaged.

With the aid of cryptographic authentication mechanisms, impersonation attacks can be prevented. An added benefit of these authentication mechanisms is that, in some cases, nonrepudiation is also achieved. A user participating in an electronic communication exchange cannot later falsely deny having sent a message. This verification is critical for situations involving electronic financial transactions or electronic contractual agreements because these are the areas in which people most often try to deny involvement in illegal practices.

Impersonation of devices is largely an issue of sending data packets that are believed to be valid but that may have been spoofed. Typically, this attack causes unwanted behavior in the network. The example in Figure 4-3 shows how the unexpected changed behavior changes the routing information. By impersonating a router and sending modified routing information, an impostor was able to gain better connectivity for a certain user.

Figure 4-3: Impersonation of Routing Updates

In this example, the intruder was connected to a corporate LAN and did a lot of work with another researcher on a different LAN. The backbone was set up in such a way that it took five hops and a 56Kb line to get to the other research machines. By capturing routing information and having enough knowledge to change the routing metric information, the intruder altered the path so that his or her access became seemingly better through a backdoor connection. However, this modification resulted in all traffic from the intruder's LAN being rerouted, saturating the backdoor link, and causing much of the traffic to be dropped.

This is an extreme and premeditated example of impersonation. But impersonation can also occur as an accident through unknown protocol and software behavior. For example, old versions of some operating systems have the innocuous behavior of acting as routers if more than one interface was connected; the OS would send out RIP (Routing Information Protocol) updates pointing to itself as the default. Figure 4-4 shows an example of this behavior.

The routed network running RIP is set up to source a default RIP advertisement to all the hosts connected to the engineering lab's LAN. Hosts running RIP typically send all traffic destined to other IP subnets to the default router. If one of the workstations connected to this LAN had a second interface connected to

another LAN segment, it would advertise itself as the default router. This would cause all hosts on the engineering LAN to send traffic destined to other IP subnets to the misguided workstation. It can also cause many wasted hours troubleshooting routing behavior that can be avoided through the use of route authentication or the configuration of trusted sources for accepting routing updates. In the network infrastructure, you have to protect yourself from malicious impersonations as well as accidental ones.

Figure 4-4: Default Route Impersonation

Note Many current networks use the Dynamic Host Configuration Protocol (DHCP), which provides a host with an IP address and an explicit default router. RIP is not used in these environments.

Impersonations of programs in a network infrastructure can pertain to wrong images or configurations being downloaded onto a network infrastructure device (such as a switch, router, or firewall) and, therefore, running unauthorized features and configurations. Many large corporate networks rely on storing configurations on a secure machine and making changes on that machine before downloading the new configuration to the device. If the secure machine is compromised, and modifications are made to device access passwords, downloading this altered configuration to a router, switch, or firewall results in an intruder being able to present false credentials---the modified password---and thereby gain access to critical network infrastructure equipment.

Impersonation can be deterred to some degree by using authentication and integrity security services such as digital signatures. A *digital signature* confirms the identity of the sender and the integrity of the contents of the data being sent.

Denial of Service

Denial of service (DoS) is an interruption of service either because the system is destroyed, or because it is temporarily unavailable. Examples include destroying a computer's hard disk, severing the physical infrastructure, and using up all available memory on a resource.

Many common DoS attacks are instigated from network protocols such as IP. Table 4-2 lists the more common DoS attacks.

Table 4-2: Common Denial of Service Attacks

Name of DoS Attack	Vulnerability Exploited

TCP SYN attack	Memory is allocated for TCP connections such that not enough memory is left for other functions
Ping of Death	Fragmentation implementation of IP whereby large packets are reassembled and can cause machines to crash
Land.c attack	TCP connection establishment
Teardrop.c attack	Fragmentation implementation of IP whereby reassembly problems can cause machines to crash
SMURF attack	Flooding networks with broadcast traffic such that the network is congested

Some DoS attacks can be avoided by applying vendor patches to affected software. For example, many vendors have patched their IP implementations to prevent intruders from taking advantage of the IP reassembly bugs. A few DoS attacks cannot be stopped, but their scope of affected areas can be constrained.

TCP SYN flooding attack effects can be reduced or eliminated by limiting the number of TCP connections a system accepts as well as by shortening the amount of time a connection stays half open (that is, the time during which the TCP three-way handshake has been initiated but not completed). Typically, limiting the number of TCP connections is performed at the entry and exit points of corporate network infrastructures. A more detailed explanation of the most common denial of service attacks is given in "Common Vulnerabilities," later in this chapter.

Motivation of Threat

Understanding some of the motivations for an attack can give you some insight about which areas of the network are vulnerable and what actions an intruder will most likely take. The perception is that, in many cases, the attacks occur from the external Internet. Therefore, a firewall between the Internet and the trusted corporate network is a key element in limiting where the attacks can originate. Firewalls are important elements in network security, but securing a network requires looking at the entire system as a whole.

Some of the more common motivations for attacks are listed here:

- *Greed.* The intruder is hired by someone to break into a corporate network to steal or alter information for the exchange of large sums of money.
- *Prank.* The intruder is bored and computer savvy and tries to gain access to any interesting sites.
- *Notoriety.* The intruder is very computer savvy and tries to break into known hard-to-penetrate areas to prove his or her competence. Success in an attack can then gain the intruder the respect

and acceptance of his or her peers.

- *Revenge*. The intruder has been laid off, fired, demoted, or in some way treated unfairly. The more common of these kinds of attacks result in damaging valuable information or causing disruption of services.
- *Ignorance*. The intruder is learning about computers and networking and stumbles on some weakness, possibly causing harm by destroying data or performing an illegal act.

There is a large range of motivations for attacks. When looking to secure your corporate infrastructure, consider all these motivations as possible threats.

Common Vulnerabilities

Attacks exploit weaknesses in systems. These weaknesses can be caused by poorly designed networks or by poor planning. A good practice is to prevent any unauthorized system or user from gaining access to the network where weaknesses in products and technologies can be exploited.

Spoofing attacks are well known on the Internet side of the world. *Spoofing* involves providing false information about a person or host's identity to obtain unauthorized access to a system. Spoofing can be done by simply generating packets with bogus source addresses or by exploiting a known behavior of a protocol's weakness. Some of the more common attacks are described in this section. Because understanding the IP protocol suite is a key element in most attacks, this section describes the protocol suite along with the weaknesses of each protocol (such as TCP, ICMP, UDP, NNTP, HTTP, SMTP, FTP, NFS/NIS, and X Windows). A more thorough study of these protocol weaknesses can be found in *Firewalls and Internet Security: Repelling the Wily Hacker* by William Cheswick and Steven Bellovin (Addison-Wesley Press).

The TCP/IP Protocol

Internet Protocol (IP) is a packet-based protocol used to exchange data over computer networks. IP handles addressing, fragmentation, reassembly, and protocol demultiplexing. It is the foundation on which all other IP protocols (collectively referred to as the *IP protocol suite*) are built. As a network-layer protocol, IP handles the addressing and controls information to allow data packets to move around the network (commonly referred to as *IP routing*). Figure 4-5 shows the IP header format.

Figure 4-5: The IP Header Format

The *Transmission Control Protocol (TCP)* is built on the IP layer. TCP is a connection-oriented protocol

that specifies the format of data and acknowledgments used in the transfer of data. TCP also specifies the procedures that the computers use to ensure that the data arrives reliably. TCP allows multiple applications on a system to communicate concurrently because it handles all demultiplexing of the incoming traffic among the application programs. Figure 4-6 shows the TCP header format, which starts at the data portion immediately following the IP header.

Figure 4-6: The TCP Header Format

Six bits (flags) in the TCP header tell how to interpret other fields in the header. These flags are listed in Table 4-3.

Table 4-3: TCP Flags

Flag	Meaning
URG	Urgent pointer field is valid.
ACK	Acknowledgment field is valid.
PSH	This segment requests a push.
RST	Resets the connection.
SYN	Synchronizes sequence numbers.
FIN	Sender has reached the end of its byte stream.

The SYN and ACK flags are of interest in the following section.

TCP/IP Connection Establishment

To establish a TCP/IP connection, a three-way handshake must occur between the two communicating machines. Each packet of the three-way handshake contains a sequence number; sequence numbers are unique to the connection between the two communicating machines. Figure 4-7 shows a sample three-way handshake scenario.

Figure 4-7: Establishing a TCP/IP Connection

The steps for establishing the initial TCP connection are as follows:

Step 1 The client initiates a TCP connection to the server. This packet has the SYN bit set. The client is telling the server that the sequence number field is valid and should be checked. The client sets the sequence number field in the TCP header to its initial sequence number.

Step 2 The server responds by sending a packet to the client. This packet also has the SYN bit turned on; the server's initial sequence number is the client's initial sequence number plus 1.

Step 3 The client acknowledges the server's initial sequence number by sending the server's initial sequence number plus 1.

Step 4 The connection is established and data transfer takes place.

TCP uses a sequence number for every byte transferred and requires an acknowledgment of the bytes received from the other end on receipt. The request for acknowledgment enables TCP to guarantee reliable delivery. The receiving end uses the sequence numbers to ensure that the data is in proper order and to eliminate duplicate data bytes.

You can think of TCP sequence numbers as 32-bit counters. These counters range from 0 to 4,294,967,295. Every byte of data exchanged across a TCP connection (as well as certain flags) is sequenced. The sequence number field in the TCP header contains the sequence number of the first byte of data in the TCP segment. The acknowledgment (ACK) field in the TCP header holds the value of next expected sequence number, and also acknowledges all data up through this ACK number minus 1.

TCP uses the concept of *window advertisement* for flow control. That is, TCP uses a sliding window to tell the other end how much data it can buffer. Because the window size is 16 bits, a receiving TCP can advertise up to a maximum of 65,535 bytes. Window advertisement can be thought of as an advertisement from one TCP implementation to the other of how high acceptable sequence numbers can be.

Many TCP/IP implementations follow a predictable pattern for picking sequence numbers. When a host is bootstrapped, the initial sequence number is 1. The initial sequence number is incremented by 128,000 every second, which causes the 32-bit initial sequence number counter to wrap every 9.32 hours if no connections occur. However, each time a connection is initiated, the counter is incremented by 64,000.

If sequence numbers were chosen at random when a connection arrived, no guarantees could be made that the sequence numbers would be different from a previous incarnation.

If an attacker wants to determine the sequencing pattern, all he or she has to do is establish a number of

legitimate connections to a machine and track the sequence numbers used.

TCP/IP Sequence Number Attack

When an attacker knows the pattern for a sequence number, it is fairly easy to impersonate another host. Figure 4-8 shows such a scenario.

Figure 4-8: TCP/IP Sequence Number Spoofing

The steps for impersonating a host are as follows:

Step 1 The intruder establishes a valid TCP connection to the server to figure out the sequence number pattern.

Step 2 The intruder starts the attack by generating a TCP connection request using a spoofed source address. Often, the intruder will pick a trusted host's address and initiate a DoS attack on that host to render it incapacitated.

Step 3 The server responds to the connection request. However, because the trusted host is under a DoS attack, it cannot reply. If it actually could process the
SYN/ACK packet, it would consider it an error and send a reset for the TCP connection.

Step 4 The intruder waits a certain amount of time to ensure that the server has sent its reply and then responds with the correctly guessed sequence number.

Step 5 If the intruder is correct in guessing the sequence number, the server is compromised and illegal data transfer can begin.

Because the sequence numbers are not chosen randomly (or incremented randomly), this attack works---although it does take some skill to carry out. Steven M. Bellovin, coauthor of *Firewalls and Internet Security*, describes a fix for TCP in RFC 1948 that involves partitioning the sequence number space. Each connection has its own separate sequence number space. The sequence numbers ware still incremented as before, however, there is no obvious or implied relationship between the numbering in these spaces.

The best defense against spoofing is to enable packet filters at the entry and exit points of your networks. The external entry point filters should explicitly deny any inbound packets (packets coming in from the external Internet) that claim to originate from a host within the internal network. The internal exit point filters should permit only outbound packets (packets destined from the internal network to the Internet) that originate from a host within the internal network.

TCP/IP Session Hijacking

Session hijacking is a special case of TCP/IP spoofing, and the hijacking is much easier than sequence number spoofing. An intruder monitors a session between two communicating hosts and injects traffic that appears to come from one of those hosts, effectively stealing the session from one of the hosts. The legitimate host is dropped from the connection and the intruder continues the session with the same access privileges as the legitimate host.

Session hijacking is very difficult to detect. The best defense is to use confidentiality security services and encrypt the data for securing sessions.

TCP SYN Attack

When a normal TCP connection starts, a destination host receives a SYN (synchronize/start) packet from a source host and sends back a SYN/ACK (synchronize acknowledge) packet. The destination host must then hear an ACK (acknowledge) of the SYN/ACK before the connection is established. This exchange is the TCP three-way handshake, described earlier in this chapter.

While waiting for the ACK to the SYN/ACK, a connection queue of finite size on the destination host keeps track of connections waiting to be completed. This queue typically empties quickly because the ACK is expected to arrive a few milliseconds after the SYN/ACK is sent.

The TCP SYN attack exploits this design by having an attacking source host generate TCP SYN packets with random source addresses toward a victim host. The victim destination host sends a SYN/ACK back to the random source address and adds an entry to the connection queue. Because the SYN/ACK is destined for an incorrect or nonexistent host, the last part of the three-way handshake is never completed, and the entry remains in the connection queue until a timer expires---typically for about one minute. By generating phony TCP SYN packets from random IP addresses at a rapid rate, an intruder can fill up the connection queue and deny TCP services (such as e-mail, file transfer, or WWW service) to legitimate users.

There is no easy way to trace the originator of the attack because the IP address of the source is forged. In the network infrastructure, the attack can be constrained to a limited area if a router or firewall intercepts the TCP connection and proxies on behalf of the connection-initiating host to make sure that the connection is valid.

Note A *proxy* is a device that performs a function on behalf of another device. For example, if the firewall proxies TCP connections on behalf of a Web server, then the firewall intercepts the TCP connections from a host trying to access the Web server and ensures that valid connection requests are made. After it validates the connection requests (usually by completing the connection by proxy), it initiates its own TCP connection request to the Web server on behalf of the host. The connection is established and normal data transfer between the client and server can start without further interference from the proxy. If a TCP SYN attack occurs, the proxy is attacked but it is not a critical device.

The land.c Attack

The *land.c attack* is used to launch DoS attacks against various TCP implementations. The land.c program sends a TCP SYN packet (a connection initiation), giving the target host's address as both the

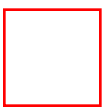
source and destination and using the same port on the target host as both the source and destination. This can cause many operating systems to hang in some way.

In all cases, the TCP ports reached by the attack must be ports on which services are actually being provided (such as the Telnet port on most systems). Because the attack requires spoofing the target's own address, systems behind effective antispoofing firewalls are safe.

The UDP Protocol

Like TCP, the *User Datagram Protocol (UDP)* is a transport layer protocol. However, UDP provides an unreliable, connectionless delivery service to transport messages between machines. It does not offer error correction, retransmission, or protection from lost and duplicated packets. UDP was designed for simplicity and speed and to avoid costly overhead associated with connection establishment and teardown. Figure 4-9 shows the UDP header format.

Figure 4-9: The UDP Header Format

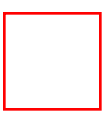


Because there is no control over how fast UDP messages are sent, and there are no connection establishment handshakes or sequence numbers, UDP packets are much easier to spoof than TCP packets. Therefore, it is wise to set up packet filters at the entry and exit points of a campus network to specifically permit and deny UDP-based applications.

The ICMP Protocol

The *Internet Control Message Protocol* (*ICMP*) is used by the IP layer to exchange control messages. ICMP is also used for some popular diagnostic tools such as Ping and traceroute. An example of an ICMP packet is shown in Figure 4-10.

Figure 4-10: An ICMP Packet



The ICMP message is encapsulated within the IP packet. As provided by RFC-791, IP packets can be up to 65,535 (2^{16}-1) octets long; this packet length includes the header length (typically 20 octets if no IP

options are specified). Packets bigger than the maximum transmission unit (MTU) are fragmented by the transmitter into smaller packets, which are later reassembled by the receiver. The MTU varies for different media types. Table 4-4 shows sample MTUs for different media types.

Table 4-4: MTUs for Varying Media Types

Media Type	MTU (in Bytes)
ISDN BRI/PRI	1,500
10M/100M Ethernet	1,500
Hyperchannel	65,535
FDDI	4,352
X.25	576
16MB IBM Token Ring	17,914
SLIP	1,006
Point-to-Point	1,500

The Ping of Death

The Ping of Death is an attack that exploits the fragmentation vulnerability of large ICMP ECHO request (that is, "ping") packets. A sample ICMP ECHO request packet is shown in Figure 4-11.

Figure 4-11: An ICMP ECHO Request Packet

The ICMP ECHO request packet consists of eight octets of ICMP header information followed by the number of data octets in the ping request. The maximum allowable size of the data area is therefore calculated this way:

(65,535 - 20 - 8) = 65,507 octets

The problem is that it is possible to send an illegal ICMP ECHO packet with more than 65,507 octets of data because of the way the fragmentation is performed. The fragmentation relies on an offset value in each fragment to determine where the individual fragment goes when it is reassembled. Therefore, on the last fragment, it is possible to combine a valid offset with a suitable fragment size so that the following is true:

(offset + size) > 65,535

Because typical machines don't process the packet until they have all the fragments and have tried to reassemble them, there is the possibility of the overflow of 16-bit internal variables, which can lead to system crashes, reboots, kernel dumps, and other unwarranted behavior.

Note This vulnerability is not restricted to the ping packet. The problem can be exploited by sending any large IP datagram packet.

A temporary fix to prevent the Ping of Death is to block ping packets at the ingress points to the corporate network. The ideal solution is to secure the TCP/IP implementation against overflow when reconstructing IP fragments.

SMURF Attack

The SMURF attack starts with a perpetrator sending a large number of spoofed ICMP ECHO requests to broadcast addresses, hoping that these packets will be magnified and sent to the spoofed addresses. If the routing device delivering traffic to those broadcast addresses performs the Layer 3 broadcast to Layer 2 broadcast function, most hosts on that IP network will reply to the ICMP ECHO request with an ICMP ECHO reply each, multiplying the traffic by the number of hosts responding. On a multiaccess broadcast network, there could potentially be hundreds of machines replying to each ECHO packet.

Turning off directed broadcast capability in the network infrastructure is one way to deter this kind of attack.

The teardrop.c Attack

teardrop.c is a program that results in another fragmentation attack. It works by exploiting a reassembly bug with overlapping fragments and causes the targeted system to crash or hang. A specific instance of a teardrop program is newtear.c, which is just a specific case in which the first fragment starts at offset 0 and the second fragment is within the TCP header.

The original teardrop.c program used fragmented ICMP packets, but people seem to have created all kinds of variants. The basic attack works for any IP protocol type because it hits the IP layer itself.

If broadcast addresses are used, turning off directed broadcast capability in the network infrastructure is one way to deter this kind of attack. However, the ideal solution is to secure the TCP/IP implementation against problems when reassembling overlapping IP fragments.

The NNTP Protocol

All Usenet traffic uses the *Network News Transfer Protocol (NNTP)* to send messages between news servers and between servers and newsreaders. Because the control protocol used for NNTP does not provide for any authentication, it can be easy to cancel messages before they are posted, create new unauthorized newsgroups, or delete existing newsgroups from the server.

Servers exist that can provide restrictions on who can post to a group based on their user ID or network address. These servers can be used for authenticated access to read and receive news. Local newsgroups should be placed on an internal secure news server; updates from other news services should be received through packet filters that can restrict which machines communicate to it from outside the corporate infrastructure.

The SMTP Protocol

All electronic mail on the Internet is based on the *Simple Mail Transfer Protocol (SMTP)*. Most email programs lack authentication, integrity, and confidentiality services unless special programs such as S/MIME or Pretty Good Privacy (PGP) programs are used. If these programs are not used, authentication, integrity, and confidentiality services can still be provided by using IP Security (IPsec, RFC 1825-1829) on routers and firewalls and by specifying that all e-mail traffic be authenticated and encrypted.

Spam Attack

A large contingency of e-mail attacks are based on e-mail bombing or spamming. E-mail *bombing* is characterized by abusers repeatedly sending an identical e-mail message to a particular address. E-mail *spamming* is a variant of bombing; it refers to sending e-mail to hundreds or thousands of users (or to lists that expand to that many users). E-mail spamming can be made worse if recipients reply to the e-mail, causing all the original addresses to receive the reply.

When large amounts of e-mail are directed to or through a single site, the site may suffer a denial of service through loss of network connectivity, system crashes, or failure of a service because of these factors:

- Overloading network connections
- Using all available system resources
- Filling the disk as a result of multiple postings and resulting syslog entries

Spamming or bombing attacks cannot be prevented, but you can minimize the number of machines available to an intruder for an SMTP-based attack. If your site uses a small number of e-mail servers, you may want to configure your *ingress* (entry from the Internet to the corporate network) and *egress* (exit from the corporate network to the Internet) points to ensure that SMTP connections from the outside can be made only to your central email hubs and to none of your other systems.

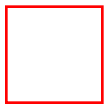
More detailed information on SPAM attacks and deterrents can be found at the following addresses: http://spam.abuse.net/ and http://www.cauce.org/.

The FTP Protocol

The *File Transfer Protocol (FTP)* is a TCP-based application program often used to store and retrieve large data files. The protocol uses two TCP connections (see Figure 4-12):

- One connection for the initial FTP control connection, which is initiated by the client to the server.
- The other connection for the FTP data connection, which is initiated from the server back to the client.

Figure 4-12: FTP Operation



Most common FTP implementations create a new FTP data connection for each file transfer and also require a new port number to be used for each of these new FTP data connections. These requirements can cause problems for restricted environments that want to block externally initiated FTP connections. The packet filters will block the incoming data connection back from the server so that file transfer no longer works.

To circumvent this problem, passive mode FTP was developed. With *passive mode FTP*, the client initiates both the control connection and the data connection so that a packet-filtering firewall can provide some protection and not block data transfers.

The NFS/NIS Services

The *Network File System (NFS)* and the *Network Information System (NIS)* are commonly used services in UNIX environments. NFS is used to access remote file systems by allowing users to mount remote file systems so that they can be accessed locally. NIS is used to establish central services and databases in client/server relationships (typically, these services include user account information and passwords). NIS and NFS are often used together to help enforce file permissions on mounted systems.

Both NFS and NIS use UDP as their underlying protocol. In typical configurations, there is limited authentication on either end of the connection. These services are extremely insecure; this kind of traffic should never be allowed through the entrance or exit points of the corporate network.

X Window System

X Window System is one of the most commonly used windowing systems. The X server offers resources such as the keyboard, the mouse, and the windows on the screen to X clients. The server accepts requests from the client for keyboard input, screen output, or mouse movement and returns the results of these requests. The X11 protocol has been adopted by many of the major workstation vendors for displaying network graphics and is the common element upon which each vendor's graphical user interface is based.

X Window System requires a reliable bi-directional stream protocol such as TCP. The communication between the client and the server consists of 8-bit bytes exchanged across a TCP connection.

Because of limited authentication inherent in the X11 protocol, it is possible for someone with access to the network to connect directly to the X server and either view, or modify ongoing communication between the server and the X client.

In a network infrastructure, limiting X11 traffic to only internal hosts is one way to limit these kinds of attacks.

NOTE The attacks and weaknesses described in this chapter are only some of the more common vulnerabilities to which current networks are susceptible. For current listings of vulnerabilities and technical tips, refer to the many advisories available on the Internet:

ftp://info.cert.org/pub/cert_advisories

www.rootshell.com

www.secnet.com/advisories

www.cert.dfn.de/eng

Social Engineering

Lastly, it is important to remember the importance of social engineering when considering threats to the corporate network. Consider a scenario in which a financial administrator in a large corporate network gets a phone call from someone saying he or she is part of the IS department and wants to verify users and passwords. An unwitting employee may think this is a valid request and submit his or her user name and password over the phone to the intruder impersonating someone from the IS department. The intruder can now impersonate the financial administrator and gain access to very confidential data and possibly alter it for his or her personal gain.

Although some threats to network security are quite sophisticated, it can be very simple to gain access to networks through seemingly innocent social means. Corporate employees should be educated about the company security policy procedures and the importance of authentication methodologies. Employees must understand the ramifications of security breaches so that they are aware of the importance of security procedures. It is the responsibility of the corporation to establish a network security policy and then establish a way to implement that policy.

Summary

This chapter examined the varying threats to a corporate network by detailing which types of attacks and vulnerabilities are common and what you can do at a policy level to guarantee some degree of safe networking. The types of threats usually come in the form of unauthorized access, impersonation, or DoS. Understanding some of the motivations for an attack can give you insight about which areas of the network are vulnerable and what actions an intruder may take. The more common vulnerabilities were detailed to help you evaluate your susceptibility---this can be invaluable in determining what steps you should take to safeguard your most exposed areas.

Posted: Wed Jun 14 11:33:33 PDT 2000

Table of Contents

Considerations for a Site Security Policy

5

Considerations for a Site Security Policy

Defining a site security policy is one of the basic building blocks of designing an enterprise network. It is as critical as defining bandwidth requirements or redundancy needs. As defined in RFC 2196, *The Site Security Handbook*:

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

The policy should be formed with representation from key corporate individuals: management members

who have budget and policy authority, technical staff who know what can and cannot be supported, and legal personnel who know the legal ramifications of various policy choices.

Benefits of creating a corporate security policy include the following:

- Providing a framework for implementing security features in the network infrastructure
- Providing a process by which you can audit existing network security
- Identifying procedures that are considered expedient, prudent, advantageous, and productive
- Enabling global security implementation and enforcement
- Creating a basis for legal action if necessary

A successful security policy must be committed to paper and show that the issues have been well thought out. Following are some key characteristics of a good security policy:

- It must be capable of being implemented technically.
- It must be capable of being implemented organizationally.
- It must be enforceable with security tools where appropriate and with sanctions where prevention is not technically feasible.
- It must clearly define the areas of responsibility for the users, administrators, and management.
- It must be flexible and adaptable to changing environments.

A security policy should not determine how a business operates; the nature of the business should dictate the security policy. Defining a company's security policy can seem difficult, but by defining the policy before choosing security methods, organizations can avoid having to redesign security methodologies after they are implemented.

This chapter focuses on how to start the process of defining a corporate security policy. After you have identified the global corporate security considerations, you can define a security policy specific to the corporate network and determine the implementation details.

Where to Begin

Many companies have existing guidelines for security procedures in a corporate environment. These can be in the form of a statement of conduct rules for employees---which, to some extent, outlines how employees are to deal with confidential technology, intellectual property rights, and other confidential corporate information. These guidelines can be a basis for establishing a strategy for an enterprise network security policy because they establish corporate rules for what information is valuable to the company from a business point of view. The following is an example of a corporate statement of conduct.

Sample Corporate Standard of Conduct

Scope

Clearly articulated and consistently administered standards of conduct form the basis for behavioral expectations within a corporate community. The enforcement of such standards should be accomplished in a manner that protects the rights, health, and safety of the corporate members so that they can pursue

their goals without undue interference.

As a way of supporting our individual commitments to fairness, honesty, equity, and respon-sibility, the members of this corporation subscribe to the following ethical principles and standards of conduct in their professional practice. Acceptance of membership signifies that the individual member agrees to adhere to the principles in this statement.

Use of This Statement

The purpose of this statement is to assist corporate personnel in regulating their own behavior by providing them with standards commonly held by practitioners in the industry. Self-regulation is preferred. However, if an individual observes conduct that may be contrary to established principles, she or he is obligated to bring the matter to the attention of the person allegedly committing the breach of ethics. If unethical conduct continues, the matter may be referred to the offender's superiors for appropriate action.

Signing this document implies agreement with and adherence to the following ethical principles and standards of conduct:

1 *Professional Responsibility*. Corporate employees have a responsibility to support both the general mission and goals of the employing company. All employees shall make every effort to balance the developmental and professional needs of employees with the obligation of the company to protect the safety and welfare of the corporate community.

2 *Legal Authority*. Employees respect and acknowledge all lawful authority. Employees refrain from conduct involving dishonesty, fraud, deceit, misrepresentation, or unlawful discrimination.

3 *Conflict of Interest*. Employees shall seek to avoid private interests, obligations, and transactions that are, or appear to be, in conflict of interest with the mission, goals, policies, or regulations of this company. Members shall clearly distinguish between those public and private statements and actions that represent their personal views and those that represent the views of this company. Further, if employees are unable to perform their duties and responsibilities in a fair and just manner because of previous involvement with a party or parties, they shall remove themselves from the decision-making process.

4 *Confidentiality*. Employees ensure that confidentiality is maintained with respect to all privileged communications and confidential corporate information and professional records. Employees inform all parties of the nature and/or limits of confidentiality.

For existing computer networks, in addition to the corporate statement of conduct, an anonymous user survey can be conducted to gather information on the possible circumvention of security procedures. This survey can result in invaluable information from people who may be circumventing security procedures for productivity reasons without any malicious intent. The circumvented security procedures can then be re-evaluated to determine how the policy can reflect security measures that can practically be implemented. Following is a sample survey questionnaire you can use.

It is important to recognize that the business opportunities are what drive the need for security procedures in the first place. If a corporation does not have many secrets to guard---perhaps because all the information and data available on the network is nonconfidential and freely available---then security procedures may be minimal. However, the more likely it is that a security breach will have negative

business implications resulting in lost revenues, the more stringent the security policies should be.

Sample Security Survey Questionnaire

The corporate Information Systems (IS) department is currently conducting a review of current security procedures to identify areas that may need improvement. Please answer the following questions to the best of your knowledge. All information will be kept confidential to the IS task force performing this survey. Please drop completed forms into the box marked "IS Survey" in the building lobby. Thank you for your participation.

1 I use the following systems (circle all that apply):

Windows UNIX Macintosh Other(specify):

2 Rate the percentage of time spent accessing the corporate network using the following mechanisms:

Corporate LAN:

Corporate frame relay (remote branch office):

Internet:

Modem dial-in:

ISDN:

3 The applications I use most often are (circle all that apply):

Web browsers E-mail Other (specify):

4 Rate the existing security measures:

too restrictive just right too loose

5 Have you discovered any security problems in the last 12 months? If so, what?

.

.

.

6 Are you aware of any back-door accesses to the corporate network? If so, what?

.

.

.

7 Any additional comments on security issues:

.

.

.

. Name (optional):

Risk Management

Risk management is a systematic approach to determine appropriate corporate security measures. How to address security, where to address security, and the type and strength of security controls requires considerable thought.

Before the proliferation of computer networks, confidential data was kept under lock and key, and people were trusted to keep confidential documents in a safe place. In extremely secure environments of the past, such as where classified work for the Department of Defense (DoD) was carried out, your briefcase, purse, and so on were inspected every night on the way out the door. You could not leave the building with *any* magnetic media or classified computer printouts (the printers attached to secured machines used specially colored paper).

In today's environments, all those physical security checks are made obsolete by the computer network. Why try to smuggle a magnetic tape out of the building when you can encrypt it and send it out in e-mail? Computer networks have created an environment in which data can be accessed, moved, or destroyed electronically if there are no electronic lock-and-key mechanisms in place to safeguard the corporation's secrets. New avenues of risk are created and must be managed.

Risk Assessment

Risk assessment is a combination of identifying critical assets, placing a value on the asset, and determining the likelihood of security breaches. When the critical resources have been identified and the likelihood and costs associated with the compromise, destruction, or unavailability of these critical resources have been assessed, a decision can be made as to what level of risk is acceptable to the company. The result of the risk assessment is unique to the organization because it depends on the business needs, trustworthiness of its users, and the location of critical assets.

Identify Network Assets

It is impossible to know who might be an organization's potential enemy. A better approach is for the organization to know itself. Companies must understand what they want to protect, what access is needed to those assets, and how these considerations work together. Companies should be more concerned about their assets and their associated value than about an attacker's motivation.

The corporation must identify the things that require protection. Table 5-1 lists some possible network assets to take into consideration.

Table 5-1: Network Assets

Asset	Description

Hardware	Workstations, personal computers, printers, routers, switches, modems, terminal servers, and firewalls
Software	Source programs, object programs, utilities, diagnostic programs, operating systems, and communication programs
Data	Data stored online and archived offline, backups, audit logs, databases, and data in transit over communication media
People	Users, administrators, and hardware maintainers
Documentation	Software programs, internal hardware and software evaluations, systems, and local administrative procedures

The inventory of the corporation's assets should be conducted globally to ensure consistent handling and evaluation of corporate assets.

Value of Assets

Placing values on corporate assets can be a very subjective process. For intangible assets---usually some form of software, data, or documentation---it can be useful to represent the value in terms of importance or criticality. In this way, the relative loss of the asset becomes more important than placing a "correct" value on it. The value of tangible assets can be based on replacement value and, as in the case of intangible assets, the immediate impact of the loss and the consequences of a loss.

The *replacement value* can encompass the monetary cost of purchasing security hardware (such as firewalls and encrypting devices) and software (such as one-time password generators and audit tools) and the cost of retraining security personnel. For data loss, the immediate impact caused by inaccessible or corrupt data may be a missed presentation deadline that consequently results in the account being lost.

Estimating the worth of data can be difficult in some situations---especially when an established research environment has to evolve to meet changing business needs. Business needs may place a higher value on some data because of its potential patent royalty or other monetary gains. Classifying data according to varying levels of criticality can be a preliminary step in establishing its value. A simple rating system of *high*, *medium*, and *low* can be the starting point for evaluating the relative criticality of data. The data can take many forms, including the following:

- *Administrative data*. Correspondence and such information as property records and personnel information that is generally available to the public.
- *Financial data*. Budgeting and expenditure information relating to corporate operations.
- *Client data*. Information relating to the client that is of a personal nature, or information developed as a result of tests, observations, or counseling.
- *Research data*. Information resulting from, or used to support, any corporate research activity.

- *Proprietary data*. Information that cannot be released to the public without the permission of the owner.

Table 5-2 shows an example of how you can classify different types of data and apply a criticality rating.

Table 5-2: Data Classification

Type of Data	Classification	Criticality
Clinical trial result	Research	High
Market trends	Research	Low
Pending patents	Proprietary	High
Corporate memos	Administrative	Low
Employee locator file	Administrative	Low
New product features	Proprietary	Medium
Trade secrets	Proprietary	High
Acquisition data	Financial	High
Employee salaries	Financial	Medium

Note Some data is more critical because of its time sensitivity. For example, impending patent data and new product data are highly sensitive either until the patent is applied for, or until the product is announced.

When the assets have been identified and valued, it is time to start looking at the likelihood of security breaches.

Threats and Vulnerability

After you have identified the network assets, you have to determine the possible threats to the assets and the likelihood that the asset is vulnerable to a given threat. A *threat* can be any person, object, or event that, if realized, can potentially cause damage to the network or networked device. Threats can be malicious (such as the intentional modification of sensitive information) or accidental (such as an error in

a calculation or the accidental deletion of a file).

A *vulnerability* is a weakness in a network that can be exploited by a threat. For example, unauthorized access (the threat) to the network can occur by an outsider guessing an obvious password. The vulnerability exploited is the poor password choice made by a user. Reducing or eliminating the vulnerable aspects of the network can reduce or eliminate the risk of threats to the network. For example, a tool that can help users choose robust passwords may reduce the chance that users will select poor passwords and, thus, reduce the threat of unauthorized network access.

The threats, as discussed in Chapter 4, "Threats in an Enterprise Network," are usually in the following forms:

- Eavesdropping and information theft
- Disabling access to network resources (DoS attacks)
- Unauthorized access to resources
- Data manipulation

If these threats are realized and networking devices or data is compromised, what are the immediate impacts and further consequences? Will it result in embarrassment or bankruptcy? The greater the possibility of bankruptcy, the more stringent the security measures should be.

Let's take a look at some corporate impacts and consequences in the event of data compromise, loss of data integrity, and unavailability of networked resources.

Data Compromise

Any information stored or transferred electronically can potentially be stolen. Data can be stolen if an intruder has unauthorized access to a system or can eavesdrop on confidential data exchanges. Depending on the type of information disclosed, the results can range from inconsequential to catastrophic. In financial institutions, monetary transactions can cause great loss to the institution itself or to customers who may represent loss of revenue if they take their business elsewhere.

You should create a priority list of the information that is most valuable to the corporation. Data pertaining to customer accounts, personnel data, and data related to finances is almost always extremely sensitive and, therefore, valuable. The security policy should reflect where different classes of sensitive data are stored, how the data is stored, and who has access to the different classes of data.

Loss of Data Integrity

Loss of data integrity can be extremely costly to many corporations. Loss of integrity can result in negative press and, therefore, loss of reputation---which translates into loss of customers and revenue. An obvious example is in the financial environment: A bank or other financial institution has a large probability of bankruptcy if any account data were ever publicly known to be compromised. The public would be hard-pressed to place trust in that institution to reliably handle its financial business.

In addition to the losses incurred from a negative reputation, the costs are extremely high to investigate and restore the compromised data. The data has to be restored from a backup, if a backup exists, and an investigation must be performed to determine if, when, and how the data was compromised. The hours of work required to analyze and restore any compromised data can be quite numerous.

When determining possible security risks, the corporation should take into account all the ways that integrity can be compromised. Data integrity goes right to the heart of your operation:

- How you perform backups
- Where you store the backups
- How you physically secure live data
- Who has physical access to the media that contains your data

Insurance underwriters, for example, confirm that four out of five companies that lose files in a fire go out of business because they cannot recover from the loss. Because so many businesses are now running with all their "actual" data on magnetic media, imagine what someone could do to your business simply by noting that you don't make regular backups and that you leave your computers out in the open where he/she can crash your disks? The security policy should clearly state how to best preserve data integrity for its valuable assets.

Unavailability of Resources

When networked resources become unavailable, the resulting business losses can be catastrophic. In today's environment in which businesses rely more and more on business transactions over computer networks; if critical systems are inaccessible, losses can be tallied in the millions of dollars.

Businesses must estimate the costs of possible system downtime caused by equipment failure, acts of nature (such as flooding, fire, and lightning), or some DoS attack. Network resources can become unavailable because of system upgrades that introduced new software bugs, faulty configurations, or inadequate capacity planning. This area is closely coupled with system reliability and redundancy, which is why a security policy should be established while the network is being designed.

Evaluating Risk

For all possible threats, you must evaluate the risk. Many methodologies are available to measure risk. The common approaches are to define risk in quantitative terms, qualitative terms, or a combination of both. *Quantitative* risk evaluation uses empirical data and known probabilities and statistics. *Qualitative* risk analysis uses an intuitive assessment. Regardless
of the mechanism you use, the important aspect is that how you quantify the loss and the likelihood of the loss occurring should be consistent and meaningful to the people who make the decisions about how to guard against the risks.

Note Automatic risk analysis tools are available in many sophisticated spreadsheet software packages. However, because of a lack of standards in how to perform risk analysis, the manner in which most losses and the likelihood of the losses are quantified and are represented should be clearly understood. If the methodology is fully understood and acceptable, an automatic risk analysis tool may be an adequate solution for evaluating risk.

Figure 5-1 shows a simple example of calculating risk by using the relative likelihood that the threat can occur and the value of the expected incurred loss.

Figure 5-1: A Simple Risk Calculation

A more specific example (taken from an existing LAN administration guide used at The National Institutes of Health) is given in Table 5-3. This table tries to determine: how critical security considerations are for different LANs using a combination of network importance, the probability of a harmful occurrence, and the probability that a degradation of LAN performance will occur after the harmful occurrence is in effect.

Table 5-3: Relative Risk Calculation for LANs

LAN	A[1]	I[1]	C[1]	NI[2]	PO[3]	PD[4]	RR[5]
Admin	2	3	1	6	Very Low 0.1	Low 0.3	3.8
Eng	2	3	2	8	Moderate 0.5	Moderate 0.5	2.0
Finance	2	3	3	18	Low 0.3	Low 0.3	8.8

[1]A = Availability, I = Integrity, and C = Confidentiality
[2]NI = Network Importance. NI is the value of A multiplied by the value of I multiplied by the value of C.
[3]PO= Prevent an Occurence. PO is determined by considering the number of users, previous accreditation, frequency of backups, and compliance with mandatory safeguards requirements.
[4]PD = Prevent Degradation. The capability to PD of A, I, and C for a LAN in the event of a harmful occurrence is determined using the relative need to protect the LAN's availability, integrity, and confidentiality with regard to the sensitivity of data and the criticality of the data-processing capability.
[5]RR = Relative Risk. RR equals NI multiplied by (1-PO) multiplied by (1-PD).

Establishing network importance is significant to managers because doing so facilitates the allocation of resources (to implement additional security services) to protect the assets that are part of the LAN. In terms of potential vulnerability, the more important a network is to a corporation, the greater the percentage of available resources that should be devoted to its protection.

Network importance is a term used to describe the relative importance of a LAN with regard to other corporate LANs. A measure of the RR associated with a harmful occurrence can be expressed as follows:

RR = NI * [(1-PO) * (1-PD)]

In this expression, NI is network importance, (1-PO) is proportional to the probability of a harmful occurrence, and (1-PD) is proportional to the probability that a degradation in LAN performance will result after an occurrence has been initiated.

The importance of the RR calculation is that it provides management with the information required to rank the risk associated with the various corporate networks relative to one another. This ranking of network importance can facilitate the allocation of resources for the implementation of additional safeguards.

In Table 5-3, the left column identifies the evaluated LANs. The next three columns record the ratings for availability, integrity, and confidentiality. The NI column is completed by multiplying the values in the previous three columns. This number establishes the relative importance of each LAN based on the need to protect the LAN. The numbers recorded under both the PO and PD columns are determined using those qualitative ratings (very low, low, moderate, high, very high) and the following scale:

very low 0.1

low 0.3

moderate 0.5

high 0.7

very high 0.9

The RR column is calculated using the following equation:

RR = NI * [(1-PO) * (1-PD)]

It should be noted that the magnitude of the difference in RR between the various LANs is not important. What is important is the relative value. The number reflected in the right column of Table 5-3 represents relative risk such that the higher the number, the greater the relative risk. Thus, the LAN with the highest number represents the greatest relative risk to the corporation.

For the LANs listed in Table 5-3, the financial LAN has the most risk and the engineering LAN has the least risk. Under normal circumstances, the higher the position of the LAN on the relative scale, the higher its priority should be for allocation of protection resources to implement additional safeguards. In some cases, however, resources are not available to implement all the needed upgrades. So, a balance must be achieved in which the resources available are applied to achieve the greatest risk reduction.

Risk Mitigation and the Cost of Security

When all the risks have been assessed, the corporation must determine how much risk it is willing to accept and to what degree the assets should be protected. *Risk mitigation* is the process of selecting appropriate controls to reduce risk to an acceptable level. The *level of acceptable risk* is determined by comparing the risk of security hole exposure to the cost of implementing and enforcing the security policy. If some threats are highly unlikely, it may not be worth the cost of creating a tight security policy to protect the assets. A good rule to follow is to assess the cost of certain losses and not to spend more to

protect something than it is actually worth.

To develop an acceptable security policy, you must consider a number of costs to ensure that the policy is enforceable. There are performance costs to be considered because both encryption and decryption take time and processing power. An offhand decision to encrypt *all* traffic may result in severely degraded performance everywhere; that policy may have to be reevaluated. There are also opportunity costs to be considered. What are the lost opportunities if your company moves more slowly than the competition because of hampered communication and the increased overhead that security procedures---not to mention security audits---impose?

The costs of implementing and managing security procedures must be weighed against the cost of potential benefits. It must be understood that security measures do not make it impossible for an unauthorized user to access information and perform unauthorized tasks to a network computer system; security measures can only make it harder for unauthorized access to occur. A very simplistic example is that of packet filters. Even if the corporation simply implemented packet filters to accept outside data traffic from specified networks, the intruder must first find out which IP network addresses were accepted before attempting to gain access to the internal corporate network through those addresses. Perhaps, this may not be very difficult, but it can be a deterrent to some random, bored intruders looking for something to do in their spare time.

A more sophisticated example is that of trying to crack encrypted traffic. Even a weak algorithm and a short key can stop some attackers from gaining access to valued information. A stronger algorithm and a longer key takes more sophisticated machines and more time to break. The point is to slow down the attack and increase the cost of the attack until it becomes too expensive to be worthwhile for the intruder.

Need-to-Know Policy

Consider the costs associated with the need-to-know policy that are common in the military and defense industries. In these cases, you are given only the information required to accomplish your job. You don't get to see the "big picture" unless you are a project leader or a higher-level manager in the company working on the project in question.

Highly secret facts, issues, and details are compartmentalized so that the left hand literally doesn't know what the right hand is doing. Although this level of classification and compart-mentalization certainly increases security, it also results in some huge cost overruns on projects: Consider the situation in which one engineer finally sees his design mated with the results of another engineer, and they both realize that a little communication two years ago would have eliminated some obvious incompatibilities and resulted in a more efficient overall design that are too late to remedy in the final stages of the project. Was security maintained? Yes, but at an enormous cost.

Those who violate security policy in a defense shop face *serious* consequences---starting with being fired and possibly ending with criminal prosecution and incarceration.

Consider a private company in a highly competitive market that pursues the same level of compartmentalization and need-to-know policy as the Department of Defense. The results are also most likely unfavorable: late products that don't take into account a change in market requirements and a great deal of wasted effort and duplication. The costs for a need-to-know policy in a corporate environment are largely negative; in this case, the policy itself is extremely hard to enforce largely because people violate the policy for all the right reasons at the wrong times.

Consider an example of a salesperson and an engineer who have become friends. The engineer tells the salesperson the results of a highly secret project. Now we all know that people in sales will go to great lengths to close a sale. Ten minutes into a highly competitive sale, out of the salesperson's mouth comes the statement, "Why yes, I have it on good authority that we will be coming out with a product to do XYZ later this year." Your competitor, who might be able to deliver the same product in half the time, now has from the prospective sales account everything they need to outflank you in other accounts.

Is your company going to fire the salesperson? Most likely not: He closed the sale and booked millions in business. Is your company going to fire the engineer? Probably not because he didn't violate security, and he didn't reveal proprietary information to anyone outside the company. Yet the company just lost a big competitive advantage in the marketplace: *market timing*.

As this example shows, trying to duplicate need-to-know policies in corporate environments is very difficult and probably should be avoided.

A Security Policy Framework

Now that you have learned to deal with risk management, it is time to start looking at additional issues for creating the security policy for an enterprise network infrastructure. Special areas of more stringent security needs are places most vulnerable to attacks, such as network inter-connections, dial-up access points, and critical network infrastructure devices and servers.

It is helpful to divide the corporate network into separate components that can be addressed separately. You also need a framework for the security policy that addresses all the elements of a security architecture. The framework must be adhered to by all areas of the corporation to ensure a consistent security approach throughout the enterprise environment.

Components of an Enterprise Network

Traditionally, in the days when network environments consisted primarily of a centralized point-to-point architecture with predetermined information paths, security was fairly straightforward. Securing the link itself provided reasonable assurance of maintaining the integrity, access, and privacy of the information.

Modern enterprise internetworks provide a tremendous opportunity for corporations to remain competitive while increasing overall efficiency. This opportunity comes with a cost. Today's open networking technologies pose a threat to the overall security of the enterprise. This openness can mean that a corporation has little control over who accesses its information resources and the path over which that information flows. Traditional security systems based on point-to-point, nonpacketized transmission media simply were not designed to address the evolving WAN and LAN technologies at the heart of today's enterprise network in which data travels across public networks.

When creating a security policy, you must balance easy accessibility of information with adequate mechanisms of identifying authorized users, ensuring data integrity, and confidentiality. A security policy must be enforceable, both technically and organizationally. It is usually easiest to break an enterprise network into three distinct components (see Figure 5-2):

- The main campus infrastructure

- Dial-up connectivity
- Internet connectivity

Figure 5-2: The Components of an Enterprise Network

The *main campus infrastructure* typically is located within a constrained geographic area and is the core of the enterprise network. The *dial-up access* consists of either PSTN or ISDN service, which connects remote branches, telecommuters, and mobile dial-up users. The *Internet access* connects the main campus through a local Internet service provider (ISP) to the Internet.

Each of these three components may have different security needs. It is important to have a global corporate security framework in place that addresses all the elements of a security architecture so that individual policies can be consistent with the overall site security architecture. For example, having a strong policy with regard to Internet access and having weak restrictions on modem usage is inconsistent with an overall philosophy of strong security restrictions on external access.

Elements of a Security Architecture

The global framework must include the following elements of a security architecture:

- Identity
- Integrity
- Confidentiality
- Availability
- Audit

Each of these elements must be taken into consideration when determining the corporate policy.

Identity

In this book, *identity* is defined as the element of the security architecture that encompasses both authentication and authorization. *Authentication* answers the question, "Who are you and where are you?" *Authorization* answers the question, "What are you allowed to access?" Identity mechanisms must be carefully deployed because even the most careful of security policies can be circumvented if the implementations are hard to use. A classic example is that of passwords or *personal identification code (PIN)* numbers scribbled on a sticky pad and attached to the computer monitor or telephone---a real solution for the user who has to remember a multitude of passwords.

Another example of poorly implemented security is when employees use an easily guessed password so that they don't have to write it down. An ad hoc study at Bell Labs some years ago found that a

surprisingly high percentage of the people logging onto systems chose a password that was a child's name, dog's name, wife's name, and so on. Corporations can install systems that ensure that the passwords selected by its employees are not proper names, words found in the dictionary, or other logical sequences of characters. However, verification and authorization systems that are cumbersome or unnecessarily redundant can frustrate users and should be avoided.

Companies must create appropriate barriers inside their systems so that if intruders do access one part of the corporate environment, they do not automatically have access to the rest of it. Just as the creation of security barriers applies to physical buildings (access to the building itself does not let you access every room in the building), it should also apply to network access. That is, the computer network infrastructure should be partitioned to provide as much protection as necessary for specific components of the network. Although maintaining a high level of security on the entire corporate environment is difficult, it is often possible to do so for a smaller sensitive component.

Integrity

Integrity is the element of the security architecture that encompasses network infrastructure device security (physical and logical access) and perimeter security. Physical access to a computer (or router or switch or firewall) usually gives a sufficiently sophisticated user total control over that device. Physical access to a network link usually allows a person to tap into that link, jam it, or inject traffic into it. Software security measures can often be circumvented when physical access to the hardware is not controlled. Therefore, for corporate facilities, physical security should be based on security guards, closed circuit television, and card-key entry systems. With these measures in place, organizations can feel confident that within their physical facilities, assets are protected and high user productivity is maintained.

Logical access security refers to providing identity mechanisms (authentication and authorization) that must be satisfied before the user is allowed Telnet or console access to integral network infrastructure components (such as routers and firewalls). *Perimeter security* deals with firewall-type functionality, determining which traffic is permitted or denied from various areas of the network. Often, firewalls are placed between the Internet and the main campus or between the dial-up connection and the main campus.

Confidentiality

Confidentiality is the element of the security architecture that ensures that data communication is kept private between the sender and receiver of information. A strong policy statement should dictate to users the types of information deemed sensitive enough to warrant encryption. A program-level policy may dictate the broad categories of information that must be stringently protected, while a system-level policy may detail the specific types of information and the specific environments that warrant encryption protection.

At whatever level the policy is dictated, the decision to use encryption should be made by the authority within the organization charged with ensuring protection of sensitive information. If a strong policy that defines what information to encrypt does not exist, then the owner of the data should ultimately make the decision about whether or not to encrypt information.

Availability

Availability is the process of ensuring that all critical resources are accessible when needed. Keeping data available means that you must have planned system upgrades and configuration changes that are fully tested to avoid catastrophic surprises caused by software bugs or misconfigurations.

Physical security and logical security are also part of ensuring availability. *Physical security* ensures that no malicious tampering can take place and that acts of nature will not cause systems to be inaccessible. It also ensures that hardware failures are handled in a timely manner. *Logical security* ensures that traffic can be rerouted and that malicious software threats can be deterred.

Audit

The *audit* element of the security architecture is necessary to verify and monitor the corporate security policy. A software audit verifies the correct implementation of the security policy in the corporate network infrastructure. Subsequent logging and monitoring of events can help detect any unusual behavior and possible intrusions.

To test the effectiveness of the security infrastructure, security auditing should occur frequently and at regular intervals. Auditing should include new system installation checks, methods to discover possible malicious insider activity, possible presence of a specific class of problems (DoS attacks), and overall compliance with the site security policy.

An audit log, generated by all the various operating systems running in your infrastructure, can be used to determine the extent of the damage from a successful attack. Audit trails are most often put to use after the fact to reconstruct what happened during damage assessment. The problem to avoid is logging *every* event such that the amount of data to sift through becomes insurmountable. If you log too much data and an intrusion does occur, that intrusion will definitely be logged---along with hundreds of other insignificant events. The intrusion will most likely remain undetected by the people responsible for detecting such things because the intrusion is hidden under a mountain of other data being generated by the system.

NOTE If your network or system is designed and implemented well, think about logging the kinds of activity that would most likely indicate a first-stage attack. Don't log every event---just the unusual ones. This information can give you a warning that something is amiss without burying you in too much inconsequential detail.

When creating data log files, consider the following points:

· Use a program to filter through the audit data and bring to your attention the truly serious issues.

· Do not audit every little issue in your network or system.

Understanding how a system normally functions, knowing what is expected and unexpected behavior, and being familiar with how devices are usually used can help the organization detect security problems. Noticing unusual events can help catch intruders before they can damage the system. Software auditing tools can help companies detect, log, and track those unusual events.

Additional Considerations

The security policy should address personnel security considerations as well. Personnel security issues include processes and procedures for establishing identity confirmation, privilege rights required to access certain information, accountability for the proper use and security of the systems being accessed, and proper training to make sure that employees understand and fulfill their security responsibilities.

The most serious breaches of corporate security come from the inside (for example, a disgruntled employee). Internal security breaches can take the form of intellectual property being leaked or disseminated to competitors, employees quitting and going to competitors with proprietary material, or a consultant simply selling off your company's materials for fiscal gain. A serious example of the last scenario happened in the mid-1980s. A consultant sold off proprietary details of IBM's then latest storage management system to Fujitsu. IBM sued Fujitsu and won a considerable sum of money as a result. Yet the amount IBM won in the settlement could not come close to the estimated $1 billion in lost revenues as a result of Fujitsu stealing IBM's technology.

Disgruntled employee problems are the hardest for corporate management to handle in this litigious age because there are so many lawyers who will take the flimsiest of employee termination cases on contingency in hopes of obtaining a tidy out-of-court settlement from the company. Companies know that it costs more to fight the suit in court than to pay the malcontent $25K to get rid of him or her, so the majority of companies will pay some sum of money to get rid of a problem employee who makes a legal threat. Sadly, the disgruntled employee is now loose on the job market again---without the public record of a court case for future employers to find in any background check.

Are background checks even performed? Not every company does them. This brings us to the topic of personnel security audits, a controversial topic because it can infringe on a person's right to privacy. Procedures for background checks should be included in the security policy---the level of screening required may vary from minimal to full background checks, depending on the sensitivity of the information the individual will handle or the risk and magnitude of loss that can be caused by the individual. Beyond that, any subsequent personal auditing is a sensitive area.

In some industry sectors, such as the financial and legal sectors, it is widely accepted that phone conversations are recorded to deter insider trading or client confidentiality infringements. Employees sign a waiver accepting this policy. A corporation should get legal advice about the latest rulings for personal privacy legislation because it relates to the workplace before putting any auditing mechanisms in place.

Companies must be firmer in handling insider security breaches and take corrective action on what is uncovered in a policy or personnel audit. The security policy must be reflected in corporate human resources policies. It is not enough to say, "Our proprietary information is ours, and you can't go around disclosing it." The company must give explicit examples with explicit consequences, putting a clause in an employment agreement that reads, "Give away even one corporate secret to an outsider, and you will be summarily fired." By doing this, a problem employee can be dealt with more quickly, with more confidence that most lawyers won't accept the case. Establish a clear standard of behavior as well as penalties for violations of that standard and make them part of the employee handbook. If the employee then violates the standard, the employee can't claim that he or she didn't know about the standard or the penalties.

Summary

This chapter detailed the process of defining a corporate security policy. The first step is identifying the global corporate security considerations. Second, critical resources need to be identified and the likelihood and costs associated with the compromise, destruction, or unavailability of these critical resources have to be assessed. Third, a decision can be made as to what level of risk is acceptable to the company. Once the acceptable risk is determined for given vulnerabilities, a security policy specific to the corporation can be defined that includes the security services of identity, integrity, confidentiality, availability, and auditing.

Posted: Wed Jun 14 11:32:45 PDT 2000

Table of Contents

Design and Implementation of the Corporate Security Policy

Design and Implementation of the Corporate Security Policy

The design and implementation of a corporate security policy is site-specific. After you have identified the critical assets and analyzed the risks, it is time to design the policy by defining the guidelines and procedures to be followed by corporate personnel.

To be effective, the procedures should be concise and to the point. Don't write a large cumbersome document few people will actually read. A short document of less than 10 pages should suffice as a start. Technical implementation details should not be included because they can change over time. If a corporate network infrastructure is already in place, you might have to modify the existing *ad-hoc* security procedures to align more closely with the newly created policy. The design of the policy takes careful planning to ensure that all security-related issues are adequately addressed.

This chapter discusses the following areas that you must consider before you can design a security policy for the corporate networking environment:

- Defining the physical security controls
- Defining the logical security controls
- Ensuring system and data integrity
- Ensuring data confidentiality
- Developing policies and procedures for the staff that is responsible for the corporate network
- Developing appropriate security awareness training for users of the corporate network

Some implementation details are given as examples of how to carry out part of the policy. Most of the implementation details are found in Chapters 8, 9, and 10, which detail specific features and considerations.

Note Incidence response handling is also part of the planning and implementation phase but, because of its importance and breadth, it is detailed separately in Chapter 7, "Incident Handling."

Physical Security Controls

Physical security controls are those controls pertaining to the physical infrastructure, physical device security, and physical access. Do you expect intruders to tap into your infrastructure to eavesdrop on transmitting data? How easy or difficult is it for intruders to gain physical access to the important network infrastructure devices? If the corporate network has not yet been created at an existing site, you should consider the physical security controls available in its planning phase.

For existing networks, if a security policy is being created or modified to accommodate changing environments, it might be necessary to change the physical infrastructure or the locations of some critical pieces of equipment to ensure an easier security policy imple-mentation. After you have incorporated the physical security controls into the policy, as the corporation grows and new sites are added, you should consider the network physical security controls as the site is constructed.

Physical Network Infrastructure

The physical network infrastructure encompasses both the selection of the appropriate media type and the path of the physical cabling (the network topography). You want to ensure that no intruder is able to eavesdrop on the data traversing the network and that all critical systems have a high degree of availability.

Physical Media Selection

From a security point of view, the type of cable chosen for various parts of the network can depend on the sensitivity of the information traveling over that cable. The three most common cable types used in networking infrastructures are twisted pair, coax, and optical fiber. Optical fiber is most often used in high-bandwidth and long-haul environments. Unlike either twisted pair or coax, optical fiber does not radiate any energy and, therefore, provides a very high degree of security against eavesdropping. Optical fiber is also much more difficult to tap into than either twisted pair or coax cable.

Wire taps can sometimes be detected by using tools to measure physical attenuation of cable. Typically, a time domain reflectometer (TDR) tool is used to check coax cable, and an optical time domain reflectometer (OTDR) tool is used for optical fiber cable. These devices are used mainly to measure signal attenuation and the length of an installed cable base; sometimes, however, they can also detect illegal wire taps.

Let's take a look at how you can detect taps in fiber optic cable using an OTDR. One of the things an eavesdropper needs when tapping into an optical cable is an *optical splitter*. The insertion of an optical splitter into an optical cable allows the tap to be made, but it also affects the signal level in the media.

This level can be measured. If a benchmark optical signal level is observed at several points along the topology of an optical media network, any conventional optical tap inserted into the network should be observable. Figure 6-1 shows an initial OTDR fiber optic cable trace between two buildings.

Figure 6-1: A Baseline OTDR Measurement

Figure 6-2 shows the fiber optic trace taken after an optical splitter was inserted into the length of the fiber cable.

Figure 6-2: The OTDR Measurement After the Fiber Optic Splitter Is Inserted

Although these types of traces can be an indication that an illegal tap might be in place, they are most useful in detecting cable degradation problems.

Note An expert can insert a tap in a way that isn't easily detectable by a TDR or OTDR. However, it is good practice to initially take a baseline signal level of the physical cable infrastructure and periodically verify the integrity of the physical cable plant. Even if it doesn't detect unauthorized media taps, the measurement will provide you with some confidence in the integrity of the cable infrastructure.

When choosing the transmission media to install for various segments of the network infra-structure, it is important to ensure that eavesdropping on the physical wire is proportionally more difficult as the data on that wire becomes more sensitive. In addition, if it is important that the transmission media be secure, the entire data path must be secure (see Figure 6-3).

Figure 6-3: An Example of Consistent Transmission Media Use

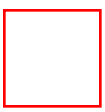


Figure 6-3 shows a large medical facility with two buildings connected by a FDDI ring. Because the server holding the patient records is located in the administrative building, and the doctor retrieving the information is located in the hospital building, both the backbone segment and the LAN segments of the network use optical fiber. It is very difficult for someone to gain access to patient information by tapping into optical fiber.

NOTE Although it is useful to keep the possibility of tapping in mind, in today's typical corporate network, there is very little need to use an "unauthorized" tap. Why bother with all the cloak-and-dagger stuff when there are all these PCs and workstations already attached to the network? All the thief has to do is run a program on any authorized workstation/PC to put its network controller into promiscuous mode; then the thief can "sniff" the network at his or her leisure.

Several shareware programs can do this now; they are available for Windows, Linux, Solaris, and others. There is no need for a thief to set up an actual sniffer on the network anymore. Because there is no way to prevent anyone from running such a program on a Macintosh or a PC running Windows 95/98, there isn't much point in actually worrying about restricting the ability to sniff. Even a policy stating that anyone caught sniffing the corporate network will be fired probably won't be very helpful because this is very hard to detect.

The issue therefore is reversed: The question you ask now is, "How do we prevent people who *are* sniffing the network from reading the contents of the packets they've sniffed?" The answer is obviously some form of encryption.

Network Topography

The physical path of the media, also known as the *network topography*, is a concern for the availability of the network and its attached devices. It touches on the reliability and security of the infrastructure. It is important to have a structured cabling system that minimizes the risk of downtime.

Imagine a large campus environment with multiple buildings. If the topography of the backbone network infrastructure is not a true starred network with common conduits at the base of the star, a construction worker with a backhoe could bring down large portions of the network (see Figure 6-4). However, if alternative physical paths are made available (that is, if you create a true starred network), only small portions of the network might become inaccessible if the physical cable fails (see Figure 6-5).

Figure 6-4: A Sample Physical Topography

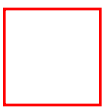


Figure 6-5: A True Starred Physical Topography

The cable infrastructure should also be well secured to prevent access to any part of it. If cables installed between buildings are buried underground, they must be buried a minimum of 40 inches, although local regulations might dictate other guidelines. Sometimes, cables can be encased in concrete to provide maximum protection. The International Telecommunication Union has a number of recommendations (the Series L Recommendations) that cover the construction, installation, and protection of cable plants. These guidelines can be found at http://info.itu.ch/itudoc/itu-t/rec/l.html.

Physical Device Security

Physical device security is sometimes understated. Intruders with enough incentive will think of anything to get at what they want. Physical device security includes identifying the location of the devices, limiting physical access, and having appropriate environmental safeguards in place.

Physical Location

The location of critical network resources is extremely important. All network infrastructure equipment should be physically located in restricted access areas to eliminate the possibility of unauthorized access by physical proximity. Facility issues can be a horrific nightmare, but when it comes to creating space for wiring closets that house critical infrastructure equipment, such as switches, firewalls, modems, and routers, it is imperative that you fight for whatever autonomous space there is. Don't overlook any aspect of the physical facility. Having a secure lock on a wiring closet does not provide much protection if you can go through the ceiling panels to get into the room.

The infrastructure equipment includes more than just the networks and the routers, firewalls, switches, and network access servers that interconnect the networks. Infrastructure equipment also includes the servers that provide the various network services:

- Network management (SNMP)
- Domain Name Service (DNS)
- Network time (NTP)
- Network File System (NFS)
- HyperText Transfer Protocol (HTTP)
- User authentication and authorization (TACACS+, RADIUS, Kerberos)
- Network audit and intrusion detection

Most of these servers can be segmented into a common area to provide easier access control measures. However, you must also be sure that adequate redundancy needs are met to ensure the availability of these critical services.

Note Whenever possible, incorporate security controls for cases in which physical access might be compromised. For example, protect console access using authentication mechanisms and use screen savers with authentication mechanisms for critical servers.

Here is another area of concern that is sometimes overlooked. When printing confidential configuration files or faxing configurations, there is the possibility that the printouts from printers or fax machines might fall into the wrong hands. You might want to make it a requirement to put all sensitive printers and fax machines on a LAN segment that is physically located in a room with controlled access. Also, you must have a way to dispose of the printouts and documents securely. Shredding is not out of the question.

Physical Access

Who has access to the wiring closets and restricted locations? The physical access requirements of controlled areas are determined largely by the results of the risk analysis or a physical security survey. It is good practice to restrict physical access to wiring closets and locations of critical network infrastructure equipment. Access to these areas should not be permitted unless the person is specifically authorized or requires access to perform his or her job.

Note The following is a true story. Although it might represent a rare occurrence, it is best to avoid any such instances if possible. A network connection was down, and some resources were unavailable. After some time spent analyzing possible problems, the equipment closet was inspected. It turns out that the cable connecting the LAN to the router had been disconnected. A maintenance worker had been working in another part of the closet, found the wire to be in the way, and disconnected it. When his work was finished, he forgot to reconnect it. A more devious example is that of a competitor posing as a maintenance worker and gaining access to confidential information.

Part of the physical security policy should be to have contract maintenance personnel or others who are not authorized with unrestricted access, but who are required to be in the controlled area, to be escorted by an authorized person or to sign in before accessing the controlled area.

To ensure an enforceable physical security policy, it is essential to ensure that people's work areas mesh well with access restrictions. If these conditions are not met, well-meaning employees will find ways to circumvent your physical security (for example, they will jam doors open rather than lock and unlock them 15 times per hour).

If your facility is providing temporary network access for visitors to connect back to their home networks (for example, to read e-mail), plan the service carefully. Define precisely where you will provide it so that you can ensure the necessary physical access security. A typical example is at large industry meetings; if these meetings are hosted at a corporate facility, the host corporation usually has a network for guests. This network should reside in a single area and access should be given only to conference attendees.

Environmental Safeguards

Adequate environmental safeguards must be installed and implemented to protect critical networked resources. The sensitivity or criticality of the system determines whether security is "adequate." The more critical a system, the more safeguards must be put in place to ensure that the resource is available at

all costs. At a minimum, you should consider the following environmental safeguards:

- Fire prevention, detection, suppression, and protection
- Water hazard prevention, detection, and correction
- Electric power supply protection
- Temperature control
- Humidity control
- Natural disaster protection from earthquakes, lightning, windstorms, and so on
- Protection from excessive magnetic fields
- Good housekeeping procedures for protection against dust and dirt

The last item might seem a little extreme, but anyone who has worked with fiber optic equipment knows that is has been prone to network degradation and downtime caused by dust particles and will recognize the usefulness of this seemingly inane point.

The following lists identify a sample physical security control policy for a university.

Construction and Location of Premises:

- All university buildings must have network closets built in accordance with relevant fire and safety standards.
- All network closets must be protected from potential sources of man-made or natural hazards, such as floods, earthquakes, and lightning.

Maintenance of Equipment:

- All network infrastructure equipment must be connected to backup power supplies.
- All network infrastructure equipment must be in locked cabinets with keys that only maintenance staff can access.

Physical Access:

- Access to network closets and equipment racks is authorized only for people in the network infrastructure operations group.
- Other personnel may access network closets only in the company of a member of the network infrastructure operations group.
- Surveillance cameras must be installed in all network closets.
- In the event of personnel changes, the locks to the network closets must be changed.

Logical Security Controls

Logical security controls create boundaries between network segments. As such, they control the flow of traffic between different cable segments. When traffic is logically filtered between networks, logical access controls provide security.

The example in Figure 6-6 shows three university buildings each connected by a router. The administration building has a LAN that allows only specific IP addresses from the engineering building

(144.254.3.3 and 144.254.3.4) and the liberal arts building (144.254.7.3 and 144.254.7.4) to access the LAN. These addresses are permitted access because they are known to belong to hosts in the faculty room, to which only faculty members have access.

Figure 6-6: Security Through Logical Access Controls

Note Although traffic filtering provides some measure of security, it is easy to spoof IP addresses. Filtering should be used in conjunction with other security measures.

Because logical boundaries are not as secure as physical boundaries, you must fully understand the path the data is taking from one point to another. Although logical boundaries usually exist between separate subnets, routing policies and virtual local-area networks (VLANs) can obfuscate the logical traffic flow.

Tips The only way to detect unauthorized traffic on the network is through the use of a packet analyzer or an intrusion detection system. It is prudent to place intrusion detection systems at critical network access points.

Subnet Boundaries

A characterization is sometimes made that traffic on different subnets is secure because the traffic is constrained to a single subnet domain. The thinking is that there is a logical separation between different groups of addresses that make up the different network access domains. You can provide filters to permit or deny traffic based on subnet addresses. However, as was pointed out in the preceding section, IP addresses are easy to spoof; other security measures should always be used in conjunction with filtering mechanisms. (Readers not familiar with IP addressing and subnetting can refer to the following sidebar.)

IP Addressing

An *IP address* is a 32-bit address represented by a dotted decimal notation of the form X.Y.Z.K (for example, 6.0.0.6). The following chart lists how the IP address space is divided by function.

Address Range	Functionality
1.0.0.0-223.255.255.255	IP unicast address
244.0.0.0-239.255.255.255	IP multicast address

240.0.0.0-255.255.255.254	Reserved for future use
0.0.0.0	An unknown IP address
255.255.255.255	Local segment broadcast

The IP unicast addresses are divided into three classes:

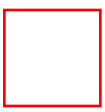
Class	Address Range	Number of Networks	Approximate Number of Hosts Per Single Network
A	1.0.0.0-126.255.255.255	127	16 million
B	128.0.0.0-191.255.255.255	64	65,000
C	192.0.0.0-223.255.255.255	32	254

The 32-bit IP address contains a network portion and a host portion, as shown in Figure 6-7.

Figure 6-7: A Bitmap of Class A, Class B, and Class C Addresses

A *network mask* is used to separate the network information from the host information. The mask is represented in binary notation as a series of contiguous 1s followed by a series of contiguous 0s. The network mask of the class A, class B, and class C networks in their binary and dotted decimal format is shown in Figure 6-8.

Figure 6-8: An Example of Natural Network Masks



A *subnet* is a subset of the class A, class B, or class C network. Subnets are created by further extending the network portion of the address into the host portion. The use of subnets increases the number of subnetworks and reduces the number of hosts on each subnetwork. The following chart shows an example of a class C network 192.150.42.0 and the possible ways you can create subnetworks with contiguous subnet masks:

Bits in Subnet Mask	Dotted Decimal Format	Number of Networks	Number of Hosts in Each Network
0	255.255.255.0	1	254
1	255.255.255.128	2	126
2	255.255.255.192	4	62
3	255.255.255.224	8	30
4	255.255.255.240	16	14
5	255.255.255.248	32	6
6	255.255.255.252	64	2

Let's take the specific example of a 3-bit subnet mask used on the 192.150.42.0 network. This network yields eight separate subnetworks with 30 hosts on each network, as listed here:

Subnet	Network Address	Broadcast Address	Host Address Range
0	192.150.42.0	192.150.42.31	192.150.42.1-192.150.42.30
1	192.150.42.32	192.150.42.63	192.150.42.33-192.150.42.62
2	192.150.42.64	192.150.42.95	192.150.42.65-192.150.42.94
3	192.150.42.96	192.150.42.127	192.150.42.97-192.150.42.126
4	192.150.42.128	192.150.42.159	192.150.42.129-192.150.42.158

5	192.150.42.160	192.150.42.191	192.150.42.161-192.150.42.190
6	192.150.42.192	192.150.42.223	192.150.42.193-192.150.42.222
7	192.150.42.224	192.150.42.255	192.150.42.225-192.150.42.254

Subnetting gives the network administrator several benefits: It provides extra flexibility, makes more efficient the use of network address utilization, and contains broadcast traffic because a broadcast does not cross a router.

Because subnets are under local administration, the outside world sees an organization as a single network and has no detailed knowledge of the organization's internal structure. However, internally, each subnet constitutes a separate LAN, possibly on a separate physical cable segment (see Figure 6-9).

Figure 6-9: An Example of Subnet Boundaries

The logical infrastructure of any network depends largely on how networks are logically separated into groups using subnets and how traffic is controlled between these subnets. Routing (also known as *Layer-3 switching*) is how traffic is controlled between subnets. Where routing information is distributed and accepted plays a large role in how you gain access to data on various networks. VLANs can also modify traditional subnet physical boundaries.

Routing Boundaries

Routing involves two basic activities:

- Determining optimal routing paths
- Transporting *packets* through an internetwork

The latter activity is typically referred to as *Layer-3 switching*. Switching is relatively straightforward: It involves looking up the destination address in a table that specifies where to send the packet. The table is created as a result of determining the optimal path to a given destination. If the table entry for a given destination is not there, the optimal path must be computed. The computation of the optimal path depends on the routing protocol used and can be a very complex process.

Note Routing fundamentals are beyond the scope of this book. Read *Internet Routing Architectures*, published by Cisco Press/MTP, for a more detailed discussion on routing.

A security policy can incorporate detailed routing policies, where routes for separate networks and subnets are announced and accepted on an as-needed basis. Most routers, regardless of the routing protocol used, have features that suppress the announcement of specified routes and can ignore certain received routes and not incorporate them into their tables. Usually, there are many ways to accomplish the same goal. It is best to first design the logical boundaries, decide how open or closed an environment you want, and then implement the policy accordingly.

Filtering routes is one way of exerting some control over who can source traffic and to what destination. Filtering does not protect you from spoofing attacks, but it can make spoofing attacks harder to carry out.

Figure 6-10 shows a common scenario of creating logical routing boundaries.

Figure 6-10: An Example of Logical Routing Boundaries

In this scenario, the corporate network is divided into three distinct components:

- Corporate campus network
- Internet access
- Dial-in access

The campus network has a class B address of 144.254.0.0, which is subnetted into 256 distinct networks using an 8-bit subnet mask of 255.255.255.0. The Internet access is provided by an unnumbered interface. The dial-in access is provided by a subnetted class C address of 192.150.42.0 with a 5-bit subnet mask of 255.255.255.248. This corporation allows free access to all corporate campus servers but allows only the branch office network 192.150.42.32 to access the Internet through the campus network. The policy can be implemented as follows:

- Allow all 144.254.0.0 routes to be announced everywhere
- Announce all 192.150.42.0 networks to the main campus
- Announce the 192.150.42.32 network to the Internet
- Suppress all other 192.150.42.0 network announcements

Static routing protocols offer the ultimate control of routes. However, the management of static routes in environments that exceed 10 or more entries can become an administrative nightmare. A dynamic routing protocol is much more flexible and can offer similar control for larger environments.

Note Routing can be a very complex subject; it is strongly recommended that you fully understand the routing protocols used in a given corporate environment and draw out the logical infrastructure before implementing any filtering commands. Where possible, use a modeling tool as a sanity check to verify the assumed logical path of network traffic.

VLAN Boundaries

A VLAN is a group of hosts or network devices---such as routers (running transparent bridging) and bridges---that form a single bridging domain. Layer-2 bridging protocols, such as IEEE 802.10 and Inter-Switch Link (ISL), allow a VLAN to exist across a variety of equipment, such as LAN switches.

VLANs are formed to group related users regardless of the physical connections of their hosts to the network. The users can be spread across a campus network or even across geographically dispersed locations. A variety of strategies can be used to group users. For example, the users can be grouped according to their department or functional team. In general, the goal is to group users into VLANs so that most of the user traffic stays within the VLAN. If you do not include a router in a VLAN, no users outside that VLAN can communicate with the users in the VLAN and vice versa.

Typically, although not necessarily, a VLAN corresponds to a particular subnet. Because a VLAN allows you to group end stations even if they are not located physically on the same LAN segment, you must ensure that the VLAN boundaries are properly understood and configured.

Logical Access Control

Access to equipment and network segments should be restricted to individuals who require access. Two types of controls should be implemented:

- *Preventative controls*, which are designed to uniquely identify every authorized user and to deny access to unauthorized users.
- *Detective controls*, which are designed to log and report the activities of authorized users and to log and report unauthorized access or attempted access to systems, programs, and data.

The correct technical solution is one that will be followed and not circumvented. You must strive to strike a balance between what authentication methods users will actually use and the methods that provide adequate security for a given system.

Control and Limit Secrets

When providing access control, it never works to have a different password for every router, switch, firewall, and other device. Probably the easiest approach is to use one password for console access and another for logical Telnet access to the devices. These passwords should be changed on a monthly basis (or whatever time table is comfortable). The passwords should definitely change when a person leaves the group.

Authentication Assurance

Some organizations still base their authentication mechanisms on standard, reusable passwords. Any reusable password is subject to eavesdropping attacks from sniffer programs. It is recommended that, if possible, these environments change to a more robust authentication scheme, such as any of the one-time passwords described in Chapter 2, "Security Technologies." However, if this is not possible, here are some recommendations for using traditional passwords:

- Choose passwords that cannot be guessed easily. Many automated password-cracking programs

use a very large dictionary and can crack passwords in a matter of seconds. Passwords should also be as long as the system supports and as users can tolerate.

- Change default passwords immediately when you install new network infrastructure equipment. Don't forget to change the passwords for console access and passwords used for maintenance purposes. For any product you buy, find out from the manufacturer whether there are ways to recover passwords and whether there are any ways to access configurations using these passwords (usually through undocumented means).
- Restrict access to the password when possible. Many vendors now have features that encrypt the password portion of configuration files. Use these features whenever they are available.
- Provide guidelines for how often a user should change his or her password. It is recommended that passwords be changed at least whenever a privileged account is compromised or when there is a critical change in personnel.

Choosing Passwords

Here are some guidelines for choosing appropriate passwords:

- Do not use your logon name in any form (as-is, reversed, capitalized, doubled, and so on).
- Do not use your first, middle, or last (current or former) name in any form.
- Do not use any of your immediate family's names (spouse, offspring, parents, pets, and so on).
- Do not use other information easily obtained about you, including license plate numbers, telephone numbers, social security numbers, the brand of automobile you drive, the name of the street you live on, and so on.
- Do not use a password of all digits or of all the same letter. These types of passwords significantly decrease the search time for a cracker.
- Do not use a word contained in any English or foreign language dictionaries, spelling lists, or other lists of words.
- Do not use a password shorter than six characters.
- Never give your network password to anyone. Securing your password is *your* responsi-bility. The whole purpose of having a password in the first place is to ensure that no one other than you can use your logons. Remember that the best kept secrets are those you keep to yourself.
- Never e-mail your password to anyone.
- Use a password with mixed-case alphabetics, if possible (some systems use passwords that are case sensitive).
- Use a password that includes some nonalphabetic characters, such as digits or punctuation marks.
- Use a password that is easy to remember, because you don't want to write it down.
- Use a password you can type quickly without having to look at the keyboard. This makes it harder for someone to steal your password by watching over your shoulder. Be wary of typing passwords in front of others.
- Change your password on a regular basis. Try to change it every three months.

Note Many authentication mechanisms have automated password protocol enforcement. For instance, an

initial password is marked as expired in the account record, either forcing the user to change the password when he or she logs in, or disabling the account if the user doesn't change the password. Users can be forced to change their passwords at regular intervals. If your authentication mechanism has these provisions (many TACACS+ and RADIUS implemen-tations do), use them.

System Greeting Messages

Many systems offer the capability to configure a greeting or banner message when accessing the system. Never include location information or the type of system in greeting or login banner messages. The system announcement messages must not welcome the user or identify the company, neither must it identify the equipment vendor or the type of operating system in use. Savvy intruders can easily reference databases of vendor or system hacks and bugs that they then can exploit. Make intruders work to get into the system before they learn what type of system it is; this gives you an additional chance to detect them breaking in.

Here is an example of a good banner message:

****WARNING**WARNING**WARNING**WARNING**WARNING****

YOU HAVE ACCESSED A RESTRICTED DEVICE. USE OF THIS DEVICE WITHOUT AUTHORIZATION

OR FOR PURPOSES FOR WHICH AUTHORIZATION HAS NOT BEEN EXTENDED IS PROHIBITED.

LOG OFF IMMEDIATELY.

WARNINGWARNING**WARNING**WARNING**WARNING**

Remember the Human Factor

Any security implementation is only as secure as its weakest link. If the security mechanisms you put in place are too complex for the users, they will find a way to circumvent the security practices, thereby creating more vulnerabilities.

The following lists provide an example of a logical security control policy for a university.

Logical Network Layout:

- All connections to which students have easy access (student housing, classrooms, labs, libraries) will be on VLANs.
- The VLANs a student can access will be determined by the curriculum in which the student is enrolled.

- The faculty rooms in each building will be connected to subnets specified solely for faculty use.
- The administration building will be on its own subnet.
- All infrastructure devices and critical services will be on their own subnets.

Access to Networks:

- All VLAN traffic will be cross-routed to each other so that all students have access to all classroom, housing, and lab computing facilities.
- Only faculty members will be allowed access to the faculty subnets.
- Only faculty and administrative personnel will be allowed access to the administration building LAN.

Access to Infrastructure Devices:

- Telnet and modem access to network infrastructure equipment is allowed only for network infrastructure operations personnel. (This equipment includes routers, firewalls, switches, and critical servers.)
- All infrastructure device access will be based on one-time password authentication technology.
- All infrastructure devices will have a generic login prompt with no information pertaining to system type or vendor name.
- All activity on infrastructure devices will be logged (such as configuration changes or new image loading).

Infrastructure and Data Integrity

On the network infrastructure, you want to ensure as best you can that any traffic on the network is valid traffic. *Valid traffic* can be categorized as expected network traffic, such as the following:

- Supported services
- Unspoofed traffic
- Data that has not been altered

Firewalls control the flow of traffic between networks and are often used to control the flow of supported network services. Authenticating data in the network infrastructure gives reasonable security against altered packets. Putting safeguards in place to deploy methods to deter attacks might help deter spoofed traffic.

Firewalls

A common way to ensure infrastructure integrity is with firewalls. A firewall, in its most simplistic sense, controls the flow of traffic. Rules are created to permit or deny various types of traffic and parallel any routing decisions made. The permission or denial of traffic can include specific network services. Many books have been written on firewalls and firewall design; some are referenced at the end of this chapter. Typically, firewalls are deployed at critical ingress and egress points of the network infrastructure, as shown in Figure 6-11.

Figure 6-11: Firewall Deployment

Currently, there are three classifications of firewalls that encompass different filtering characteristics:

- *Packet filtering*. These firewalls rely solely on the TCP, UDP, ICMP, and IP headers of individual packets to permit or deny traffic. The packet filter looks at a combination of traffic direction (inbound or outbound), IP source and destination address, and TCP or UDP source and destination port numbers.
- *Circuit filtering*. These firewalls control access by keeping state information and reconstructing the flow of data associated with the traffic. A circuit filter won't pass a packet from one side to the other unless it is part of an established connection.
- *Application gateway*. These firewalls process messages specific to particular IP applications. These gateways are tailored to specific protocols and cannot easily protect traffic using newer protocols.

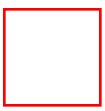
Before determining which classifications best fit your environment, examine the traffic flow control you can exert in your environment. Most of the control is based on a combination of the following characteristics:

- Direction of traffic
- Traffic origin
- IP address
- Port numbers
- Authentication
- Application content

Direction of Traffic

Traffic can be filtered in either the inbound or outbound direction. Generally, *inbound* traffic comes from an outside untrusted source to the inside trusted network. *Outbound* traffic comes from inside the trusted network to an outside untrusted network (see Figure 6-12).

Figure 6-12: Traffic Direction



Traffic Origin

Whether traffic was initiated from the inside (trusted) network or the outside (untrusted) can be a factor in managing traffic flow. For example, you might want to allow certain UDP packets to originate from inside the trusted network (DNS), but might not allow DNS requests to come in from the outside untrusted network. Alternately, you might want to restrict TCP traffic to outside untrusted networks if the TCP session was initiated from the inside trusted network.

IP Address

The source or destination address can be used to filter certain traffic. This approach is useful for implementing precursory controls to help avoid spoofing attacks.

Port Numbers

TCP and UDP source and destination port numbers are used to recognize and filter different types of services. Which services you support is a key question and is discussed in more detail in "Network Services," later in this chapter.

Authentication

At some ingress points to trusted networks, you might want to authenticate users before they can access particular services, such as Telnet, FTP, or HTTP. Available authentication mechanisms vary, but they all aid in controlling use and auditing who is accessing which services. As an aside, authentication can also help service providers with billing and accounting information.

Application Content

It can be useful to look at applications and determine certain controls. You might want to look into filtering certain Uniform Resource Locators (URLs) or filtering specific content types (such as Java applets).

Network Services

Choosing which services and protocols you support can be a daunting task. An easy approach is to permit all and deny as needed. This policy is easy to implement because all you have to do is turn on all services and allow all protocols to travel across network boundaries. As security holes become apparent, you restrict or patch those services at either the host or network level.

This approach is fairly simple, but it is also vulnerable to a multitude of attacks. A more secure approach is to deny all and permit as needed. With this method, you turn off all services and selectively enable services on a case-by-case basis as they are needed. The deny-all model is generally more secure than the permit-all model, but it requires more work to successfully implement. It also requires a better understanding of the services. If you allow only known services, you provide for a better analysis of a particular service or protocol and you can design a security mechanism suited to the security level of the

site.

Note Security complexity can grow exponentially with the number of services provided. Evaluate all new services with a skeptical attitude to determine whether they are actually needed.

It is beyond the scope of this book to provide a detailed list of all the network services available. However, the books recommended in Appendix A, "Sources of Technical Information," provide you with all the detail necessary to choose the services appropriate for your specific environment. To summarize, the services most commonly required in environments include SNMP, DNS, NTP, WWW, Telnet, FTP, NNTP, and SMTP.

It can be a daunting task to figure out which services to filter. At a minimum, you should follow the CERT recommendations, which strongly suggest that you filter the services listed in Table 6-1. The Computer Emergency Response Team (CERT) from Carnegie Mellon University collects reports of computer crime, provides this information to vendors, and distributes information from vendors regarding vulnerabilities of their systems.

Table 6-1: CERT-Recommended Services to Filter

Protocol	Port Number	Description
TCP	53	DNS Zone Transfer
UDP	69	Tftpd
TCP	87	Link (commonly used by intruders)
TCP	111	SunRPC
UDP	111	SunRPC
TCP	2049	NFS
UDP	2049	NFS
TCP	512	BSD UNIX R-command
TCP	513	BSD UNIX R-command
TCP	514	BSD UNIX R-command
TCP	515	lpd

TCP	540	uucpd
TCP	2000	OpenWindows
UDP	2000	OpenWindows
TCP	6000+	X Windows
UDP	6000+	X Windows

The services a site provides will, in most cases, have different levels of access needs and models of trust. Services that are essential to the security or smooth operation of a site are better off on a dedicated machine with very limited access.

Services provided on the same machine can interact in catastrophic ways. For example, allowing anonymous FTP on the same machine as the WWW server might permit an intruder to place a file in the anonymous FTP area and cause the HTTP server to execute it. If possible, each service should run on a different machine. This arrangement helps isolate intruders and limit potential harm.

Authenticated Data

To ensure a reasonable amount of data integrity, you should authenticate most traffic traversing the network. For network infrastructure integrity, traffic specific to the operation of a secure infrastructure (such as routing updates) should also be authenticated.

Routing Updates

If you do not authenticate routing updates, unauthorized or deliberately malicious routing updates can compromise the security of your network traffic. A security compromise can occur if an unfriendly party diverts or analyzes your network traffic. For example, an unauthorized router could send a fictitious routing update to convince your router to send traffic to an incorrect destination. This diverted traffic could be analyzed to learn confidential information about your organization, or merely to disrupt your organization's ability to effectively communicate using the network.

Note The need to authenticate network traffic also applies to bridging spanning-tree and VLAN protocols. If you can spoof a routing update or bridge topology change, you can black-hole various portions of the network, causing denial of service attacks that can be very, very difficult to detect because routing protocols don't keep much information about the device that sent them a routing packet. Bridges and switches keep even less information.

Checksums protect against the injection of spurious packets, even if the intruder has direct access to the physical network. Combined with a sequence number or other unique identifier, a checksum can also

protect against *replay attacks*, wherein an old (but once valid) routing update is retransmitted by either an intruder or a misbehaving router. The most security is provided by complete encryption of sequenced, or uniquely identified, routing updates. This approach prevents an intruder from determining the topology of the network. The disadvantage to encryption is the overhead involved in processing the updates.

Note At a minimum, it is recommended that you authenticate routing updates with a checksum.

Common Attack Deterrents

Most common networking attacks can be made more difficult with firewall-type products.

Attacks Against Any Random Host Behind the Firewall

In many cases, attacks against random hosts behind the firewall can be completely prevented, depending on how you've configured the firewall. The most conservative configuration allows no traffic at all to reach internal hosts unless those hosts initiate an outgoing connection of some kind. If you're set up this way, a number of attacks can be deterred.

Attacks Against Exposed Services

Web servers, mail servers, FTP servers, and so on behind the firewall are at the most risk because any host on the network can send at least some kinds of packets to them at any time. You are generally better off putting those exposed services on a demilitarized zone (DMZ) network, rather than on your internal network. You must also make sure that the server itself is protected. Firewalls do some things to protect exposed services, but that's still where the biggest risks lie.

Attacks Against Internal Client Hosts

If internal client hosts have formed outgoing connections, they are exposing themselves to some return traffic. In general, attacks against internal clients can be conducted only by the server to which the client has connected---which includes someone impersonating that server using IP spoofing. To impersonate the server, the attacker obviously has to know which server the client has connected to.

For any given attack, protection is generally complete for hosts that aren't actively talking to the net, partial for hosts that are actively talking to the net, and minimal for exposed services. However, security depends on how the system is configured.

Spoofing Attacks

No product, even if properly configured, can protect you completely against outside hosts assuming the addresses of your inside hosts. There is no way a firewall, or any other device, can determine whether the source address given in an unauthenticated IP packet is valid---other than to look at the interface on which that packet arrived. Therefore, no firewall can protect against the general case of one outside host spoofing another. If something has to rely on the address of an outside host, you must have control over the *entire* network path to that host, not just a single access point.

Any internetworking product can make spoofing attacks more difficult by making it harder for the

attacker to guess which nodes it's profitable to spoof at any given moment. This protection is not complete; an attacker who can sniff the network at strategic points, or who can make good guesses based on knowledge of the traffic patterns, can get around the internetworking product.

The following lists provide an example of the infrastructure and data integrity section of a sample university security policy.

Infrastructure Security:

- Access to switch LAN ports and router interfaces will be disabled when not in use.
- Firewall functionality will be used at egress points; *egress points* are defined as any connections that provide access anywhere outside the main university campus.
- Only necessary network services will be supported. These services will be defined by the network operations steering group.

Data Integrity:

- Software not related to work will not be used on any computer that is part of the network infrastructure and critical services.
- All software images and operating systems should use a checksum verification scheme before installation to confirm their integrity.
- All routing updates and VLAN updates must be authenticated between sending and receiving devices.

Data Confidentiality

Data confidentiality pertains to encryption. The hardest aspect of this endeavor is deciding which data to encrypt and which to keep as cleartext. This decision should be made using the *risk assessment procedure*, in which data is classified according to various sensitivity levels. It is usually prudent to take a careful look at your data and to encrypt the data that would pose the greatest risk if it should ever be compromised.

Network Address Translation

Network Address Translation (NAT) is often falsely regarded as a security feature. NAT was originally created to help solve the problem of a large corporation having to renumber its thousands of hosts when it connected to the Internet with an illegal address (an *illegal address* being one that is not assigned by the NIC and might therefore already be in use).

The only way NAT serves as a security feature is if no one knows the internal corporate network address being translated into a valid legal address. If the corporate office were the target for an attack, some forms of attacks would be harder to carry out because the corporate network address is unknown. However, in most cases, the network addresses are well known, even if they are illegal. An RFC recommends specific class A, class B, and class C network numbers to be used illegally (that is, three legal, illegal addresses). These addresses, as specified in RFC 1976, are listed here:

10.0.0.0-10.255.255.255

172.16.0.0-172.31.255.255

192.168.0.0-192.168.255.255

If a corporation is using an illegal network address, it is recommended that it go through the grueling process of renumbering to prevent the many Band-Aid solutions it will otherwise encounter with future features and applications that might not be NAT friendly.

There are, of course, some situations in which an illegal network address cannot be changed. For more details on using legal versus illegal network addresses, refer to the section on NAT in Chapter 9, "Securing Internet Access."

The following list provides an example of the data confidentiality section of the university security policy.

Data Confidentiality:

- All information regarding student grades and transcripts must be encrypted.
- All information regarding student financial information must be encrypted.
- All employee salary and benefits information must be encrypted.

Policies and Procedures for Staff

The people responsible for maintaining and upgrading the network infrastructure should have specific guidelines to aid them in carrying out their tasks in accordance with the corporate security policies.

Secure Backups

The procedure of creating backups is an integral part of running a computer environment. For the network infrastructure, backups of all network service servers, as well as backups of the configurations and images of networking infrastructure equipment, is critical. The following list should be included in your backup policy:

- Ensure that your site is creating backups for all network infrastructure equipment configurations and software images.
- Ensure that your site is creating backups for all servers that provide network services.
- Ensure that your site is using offsite storage for backups. The storage site should be carefully selected for both its security and its availability.
- Consider encrypting your backups to provide additional protection for the information once it is offsite. However, be aware that you will need a good key management scheme so that you'll be able to recover data at any point in the future. Also, make sure that you will have access to the necessary decryption programs in the future when you might need to perform the decryption.
- Don't always assume that your backups are good. There have been many instances of computer security incidents that have gone on for long periods of time before a site has noticed the incident. In such cases, backups of the affected systems are also tainted.
- Periodically verify the correctness and completeness of your backups.

Keep original and backup copies of data and programs safe. Apart from keeping them in good condition for recovery purposes, you must protect the backups from theft. It is important to keep backups in a separate location from the originals, not only for damage considerations but also to guard against theft. Media used to record and store sensitive software or data should be externally identified, protected, controlled, and secured when not in actual use.

Equipment Certification

All new equipment to be added to the network infrastructure should adhere to specified security requirements. Each specific site must determine which security features and functionality are necessary to support its security policy. These features can be as simple as providing TACACS+ or RADIUS support and can progress to the more specific requirements of providing TACACS+ and RADIUS support with token card authentication integration and time-of-day support.

Use of Portable Tools

Portable hosts pose some risk. Make sure that the theft of one of your staff's portable computers won't cause problems. Consider developing guidelines for the kinds of data allowed to reside on the hard disks of portable computers, as well as how the data should be protected (for example, whether encryption should be used) when it is on a portable computer.

Audit Trails

Keeping logs of traffic patterns and noting any deviation from normal behavior can be your first clue to a security breach. Cliff Stoll relayed his experience in *The Cuckoo's Egg*, in which he helped catch some cyberspies by noting a two-cent discrepancy in some accounting data.

What to Collect

The actual data you collect for your logs will differ for different sites and for different types of access changes within a site. In general, the information you want to collect includes:

- User name
- Host name
- Source and destination IP address
- Source and destination port numbers
- Timestamp

Of course, you can gather much more information, depending on what the system makes available and how much space is available to store that information.

Note Do not record passwords that might be sent in cleartext. Doing so creates an enormous potential security breach if the audit records should be improperly accessed.

Storing the Data

Depending on the importance of the data and the need to have it local in instances in which services are being denied, data could be kept local to the resource until needed or be transmitted to storage after each event. Consider how secure the path is between the device generating the log and the actual logging device (the file server, tape or CD-ROM drive, printer, and so on). If that path is compromised, logging can be stopped or spoofed or both.

In an ideal world, the logging device would be directly attached to the device generating the log by a single, simple, point-to-point cable. Because that is usually impractical, the data path should pass through the minimum number of networks and routers. Even if logs can be blocked, spoofing can be prevented with cryptographic checksums (it probably isn't necessary to encrypt the logs because they should not contain sensitive information in the first place).

The storage device should also be carefully selected. Consider *Write-Once, Read Many (WORM) drives* for storing audit data. With these drives, even if an attacker can get to the data (with the exception of the physical media), they cannot change or destroy the data.

Because collecting audit data can result in a rapid accumulation of bytes, you must consider the availability of storage for this information in advance. By compressing data or keeping data for a short period of time, you can reduce the required storage space. It is useful to determine a time frame with which everyone is comfortable and for which you will keep detailed audit logs for incident response purposes.

Audit data should be some of the most carefully secured data at the site and in the backups. If an intruder were to gain access to audit logs, the systems themselves---in addition to the data---would be at risk. Audit data can also become the key to the investigation, apprehension, and prosecution of the perpetrator of an incident. For this reason, it is advisable to seek the advice of legal counsel when deciding how you will treat audit data. Get legal counsel *before* an incident occurs. If your data-handling plan is not adequately defined before an incident occurs, it might mean that there is no recourse in the aftermath of an event, and it might create a liability resulting from the improper treatment of the data.

Legal Considerations

Because of the nature of the content of audit data, a number of legal questions arise that you might want to bring to the attention of your legal counsel. If you collect and save audit data, be prepared for consequences resulting from both its existence and its content. One area of concern is the privacy of individuals. In certain instances, audit data might contain personal information. Searching through the data, even for a routine check of the system's security, might represent an invasion of privacy.

A second area of concern involves your having knowledge of intrusive behavior originating from your site. If an organization keeps audit data, is it responsible for examining it to search for incidents? If a host in one organization is used as a launching point for an attack against another organization, can the second organization use the audit data of the first organization to prove negligence on the part of that organization?

The following lists provide an example of the policies and procedures for the staff aspect of the university security policy.

Personnel Security Controls:

- Key positions must be identified, and potential successors should always be identified.
- Recruiting employees for positions in the implementation and operation of the network infrastructure requires a thorough background check.
- All personnel involved with implementing and supporting the network infrastructure must attend a two-day security seminar, which has been developed internally.

Equipment Acquisition and Maintenance:

- All infrastructure equipment must pass the acquisition certification process before purchase.
- All new images and configurations must be modeled in a test facility before deployment.
- All major scheduled network outages and interruptions of services must be announced to those affected.

Backup Procedures:

- All software images and configurations will be backed up in infrastructure devices when modified.
- The previous image and configuration file will be kept until another change is made. Therefore, there should always be available the current and the previous image and configuration.
- All backups will be stored in a dedicated locked area.

Security Awareness Training

Users are typically not aware of security ramifications caused by certain actions. People who use computer networks as a tool to get their job done want to perform their job functions as efficiently as possible---and security measures often are more of a nuisance than a help. It is imperative for every corporation to provide employees with adequate training to educate them about the many problems and ramifications of security-related issues.

The security training should be provided to all personnel who design, implement, or maintain network systems. This training should include information regarding the types of security and internal control techniques that should be incorporated into the network system development, operations, and maintenance aspects.

Individuals assigned responsibilities for network security should be provided with in-depth training regarding the following issues:

- Security techniques
- Methodologies for evaluating threats and vulnerabilities
- Selection criteria and implementation of controls
- The importance of what is at risk if security is not maintained

For large corporate networks, it is good practice to have a LAN administrator for each LAN that connects to the corporate backbone. These LAN administrators can be the focal point for disseminating information regarding activities affecting the LAN.

Rules to abide by typically should exist before connecting a LAN to the corporate backbone. Some of these rules are as follows:

- Provide documentation on network infrastructure layout
- Provide controlled software downloads
- Provide adequate user training

Training is also necessary for personnel in charge of giving out passwords. This personnel should ensure that proper credentials are shown before reinstating a "forgotten" password. There have been many publicized incidents in which people received new passwords simply by acting aggravated enough but without presenting adequate credentials. Giving out passwords in this fashion can have serious-enough ramifications that the person who bypasses known regulations should be terminated.

Social Engineering

Many intruders are far more successful using social engineering than they are with a technical hack. A critical training requirement should be that employees and users are not to believe anyone who calls them on the phone and asks them to do something that might compromise security. Would you give any caller your personal financial information and accept a new PIN number over the phone? Hopefully not---you have not absolutely established the inquirer's identification. The same is true for passwords and any kind of confidential corporate infor-mation requested over the phone. Before divulging any kind of confidential information, you must positively identify the person to whom you are talking.

Summary

In this chapter, we start to look at what is needed to create the guidelines and procedures that are part of a corporate security policy. To be effective, the procedures should be concise and to the point. They should encompass rules that define physical security controls---these pertain to the physical infrastructure, physical device security, and physical access. These rules should include guidelines to secure data integrity and confidentiality to ensure that data has not been altered in transit and is only understandable by the sender and intended receiver of the information. It also is imperative for every corporation to provide employees with adequate training to educate them about the many problems and ramifications of security-related issues.

continues

continues

Posted: Wed Jun 14 11:36:48 PDT 2000

Table of Contents

Incident Handling

A *security breach* is often referred to as an incident. An *incident* is any breach that is the result of an external intruder attack, unintentional damage, an employee testing some new program and inadvertently exploiting a software vulnerability, or a disgruntled employee causing intentional damage. Each of these possible events should be addressed in advance by adequate contingency plans.

The time to think about how to handle a security incident is *not* after an intrusion has occurred. Planning and developing procedures to handle incidents before they occur is
a critical piece of any security policy. The procedures should be detailed enough to encompass the practical steps in recognizing that a breach has occurred, evaluating the breach, and restoring and

recovering from your losses.

Fearing unknown intrusion threats to their computer systems, some corporations restrict access to their systems and networks. Consequently, these organizations spend far too much time reacting to recurring incidents at costs to convenience and productivity. What is needed is a form of computer security response that can quickly detect and respond to incidents in a way that is both cost-efficient and cost-effective.

Several factors have contributed to the growing presence of computer network security incidents:

- *Reliance on computers.* An increasing number of corporations rely on computers and networks for communications and critical business transactions. Consequently, many corporations would suffer great losses to productivity should their systems become unavailable. Because of system complexity, reliance on computer networks often presents unanticipated risks and vulnerabilities.
- *Use of large networks.* Large networks that link governments, businesses, and academia are growing by leaps and bounds. Efficient response to computer security incidents is very important for anyone relying on a large network. Compromise of one computer network can affect a significant number of other systems connected to the network but located in different organizations---with resulting legal and financial ramifications. Incident response teams note that intruder attempts to penetrate systems occur daily at numerous sites throughout the United States, and that many corpor-ations are often unaware that their systems have been penetrated or have been used as springboards for attacks on other systems.

How bad is the problem? The first major publicized incident in 1988, the Internet Worm caused shutdowns and Denial of Service (DoS) problems for weeks to over 3,000 sites. In 1989, the NASA WANK (Worms Against Nuclear Killers) Worm caused a major loss of availability on two large government networks, resulting in significant expense and investigations by the U.S. Government Accounting Office (GAO) into network management and security. More recent years have seen attacks such as Kevin Mitnick's theft of numerous credit card numbers in California from 1992 to 1995 (refer to http://kevinmitnick.com/indictment.html) and the widely publicized attempted attack against a telecommunications infrastructure initiated by the Chaos Computer Club (CCC) in Germany in September 1995. The CCC called for a DoS attack against the French telecommunications systems to protest French nuclear testing in the Pacific (refer to Chaos Computer Club, "Stop the Test," http://www.zerberus.de/texte/aktion/atom/, September 1, 1995).

In an executive report dated May 22, 1996, the GAO reported the following:

The [U.S. Department of] Defense may have experienced as many as 250,000 attacks in 1995. DISA [the Defense Information Systems Agency] information also shows that attacks are successful 65 percent of the time, and that the number of attacks is doubling each year.

Recent surveys show the same trend in the private sector. For example, in March 1999 the FBI released an annual survey it conducts with the San Fransisco-based Computer Security Institute that reports that criminal security breaches cause 123 million dollars in losses in 1998. Many incidents include intruders using international networks to target numerous worldwide systems.

Accurate accounting costs and profit losses caused by security incidents are rather difficult to obtain. This information is extremely sensitive to corporations whose business relies more and more on reliable

computing services. Many times, computer incidents are kept under cover and are not even reported. The problem, however, is real; corporations should have procedures in place to recover from a security breach should one occur.

Building an Incident Response Team

An organization must first create a centralized group to be the primary focus when an incident happens. This group is usually a small core team whose responsibilities include the following:

- Keeping up to date with the latest threats and incidents
- Being the main point of contact for incident reporting
- Notifying others of the incident
- Assessing the damage and impact of the incident
- Finding out how to avoid further exploitation of the same vulnerability
- Recovering from the incident

Establishing the Core Team

The core incident response team should consist of a well-rounded representation from the corporation. Essential are people who can diagnose and understand technical problems; thus, technical knowledge is a primary qualification. Good communication skills are equally important. Because computer security incidents can provoke emotionally charged situations, a skilled communicator must know how to resolve technical problems without fueling emotions or adding complications. In addition, the individuals may spend much of their time communi-cating with affected users and managers, either directly or by preparing alert information, bulletins, and other guidance.

News about computer security incidents can be extremely damaging to an organization's stature among current or potential clients. Therefore, a company spokesperson is also needed to interact with the press or media. If the incident is significant, the corporation will want to represent itself clearly without worrying its customers or the stock market and causing negative business repercussions. You must find personnel who have the correct mix of technical, communication, and political skills.

A member of the core incident response team should have many of the following qualifications:

- Comprehensive networking knowledge
- Good communication skills
- Good interpersonal skills
- Understanding of company business
- Good analytical skills
- Even temperament

Detecting an Incident

Determining whether or not some suspicious system or user behavior is really an incident is tricky. When looking for signs of a security breach, some of the areas to look for from a network viewpoint are

- Accounting discrepancies
- Data modification and deletion
- Users complaining of poor system performance
- Atypical traffic patterns
- Atypical time of system use
- Large numbers of failed login attempts

Detecting any anomalies in normal network behavior requires a knowledge of what is "normal" behavior. Using auditing tools that keep track of traffic patterns and historical trends can be one of the many ways you can determine normal behavior. Realistically, a corporation should not delude itself in thinking it can detect and stop all intrusions from occurring. Rather, it should put procedures in place that limit any impact of an intrusion.

Because of the multitude of existing known attacks and new ones cropping up on a regular basis, the use of automated tools is essential. Many intrusion-detection systems are based on a combination of statistical analysis methods and rule-based methods:

- The *statistical analysis method* maintains historical statistical profiles for each user or system that is monitored. The method raises an alarm when observed activity departs from established patterns of use. This type of analysis is intended to detect intruders masquerading as legitimate users. Statistical analysis may also detect intruders who exploit previously unknown vulnerabilities that cannot be detected by any other means.
- The *rule-based analysis method* uses rules that characterize known security attack scenarios and raise an alarm if observed activity matches any of its encoded rules. This type of analysis is intended to detect attempts to exploit known security vulnerabilities of the monitored systems. This analysis can also detect intruders who exhibit specific patterns of behavior known to be suspicious or in violation of site security policy. Most rule-based systems are user configurable so that you can define your own rules based on your own corporate environment.

When looking for an intrusion system to deploy in your environment, the following characteristics are an indication of a good system (a complete list can be found at http://www.cs.purdue.edu/coast/intrusion-detection/detection.html):

- It must run continually without human supervision. The system must be reliable enough to allow it to run in the background of the system being observed. However, it should not be a black box---that is, its internal workings should be verifiable from the outside.
- It must be fault tolerant. The system must survive a system crash without rebuilding its knowledge base at restart.
- It must impose minimal overhead on the system. An intrusion system that slows a computer to a crawl will simply not be used.

- It must observe deviations from normal behavior and have timely alerting mechanisms.
- It must be easily tailored to fit into varying corporate environments. Every network has a different usage pattern, and the statistical analysis database or rule database should adapt easily to these patterns.
- It must cope with changing system behavior over time as new applications are added.
- It must be difficult to bypass. The intrusion-detection system should itself be secure and not open to compromise in any way.

Note Intrusion-detection systems that fill all of the preceding requirements are few; those that do exist are expensive. Before spending a lot of money on any intrusion system, make sure you understand what the system can detect and how easy the software is to modify to handle new attack scenarios.

Handling an Incident

You must follow certain steps when you handle an incident. These steps should be clearly defined in security policies to ensure that all actions have a clear focus. The goals for handling any security breaches should be defined by management and legal counsel in advance.

One of the most fundamental objectives is to restore control of the affected systems and to limit the impact and damage. In the worst-case scenario, shutting down the system, or disconnecting the system from the network, may be the only practical solution.

Prioritizing Actions

Prioritizing actions to be taken during incident handling is necessary to avoid confusion about where to start. Priorities should correspond to the organization's security policy and may be influenced by government regulations and business plans. The following are things to be considered:

- *Protecting human life and people's safety.* Systems should be implemented that control plant processes, medical procedures, transportation safety, or other critical functions that affect human life and safety and are required by law to be operational (as per OSHA and other governmental safety regulations).
- Protecting sensitive and/or classified data.
- *Protecting data that is costly in terms of resources.* With any security incident, you want to reduce the loss as much as possible.
- Preventing damage to systems.
- *Minimizing the disruption of computing resources.* You want to reduce the spread of any damage across additional parts of the network.

Assessing Incident Damage

A very time-consuming task is initially determining the impact of the attack and assessing the extent of any damages. When a breach has occurred, all parts of the network become suspect. You should start the process of a systematic check through the network infrastructure to see how many systems could have

been impacted. Check all router, switch, network access server, and firewall configurations as well as all servers that have services that support the core network infrastructure. Traffic logs must be analyzed to detect unusual behavior patterns. The following checklist may be a useful starting point:

- Check log statistics for unusual activity on corporate perimeter network access points, such as Internet access or dial-in access.
- Verify infrastructure device checksum or operating system checksum on critical servers to see whether operating system software has been compromised.
- Verify configuration changes on infrastructure devices and servers to ensure that no one has tampered with them.
- Check sensitive data to see whether it was accessed or changed.
- Check traffic logs for unusually large traffic streams from a single source or streams going to a single destination.
- Run a check on the network for any new or unknown devices.
- Check passwords on critical systems to ensure that they have not been modified (it would be prudent to change them at this point).

Reporting and Alerting Procedures

You should establish a systematic approach for reporting incidents and subsequently notifying affected areas. Effective incident response depends on the corporate constituency's ability to quickly and conveniently communicate with the incident response team. Essential communi-cations mechanisms include a central telephone "hotline" monitored on a 24-hour basis, a central electronic-mail (e-mail) address, or a pager arrangement. To make it easy for users to report an incident, an easy-to-remember phone number such as XXX-HELP (where XXX is the company internal extension) should be used. Users should have to remember only this one number; technology can handle the issues of call forwarding and sending out email and pager alerts.

Who to alert largely depends on the scope and impact of the incident. Because of the widespread use of worldwide networks, most incidents are not restricted to a single site. In some cases, vulnerabilities apply to several million systems, and many vulnerabilities are exploited within the network itself. Therefore, it is vital that all sites with involved parties be informed as soon as possible. The incident response team should be able to quickly reach all users by sending to a central mailing list or, alternatively, sending telephone voice mailbox messages or management points-of-contact lists.

Although you want to inform all affected people, it is prudent to make a list of points of contacts and decide how much information will be shared with each class of contact. The classes of contact include people within your own organization (management, users, network staff), vendors and service providers, other sites, and other incident response teams. Here is an example of a message that can be sent to corporate employees in some situations:

We are currently experiencing a possible security breach and have disconnected all outside corporate connections. Please review the current status at http://corporate/security.info.

We will let you know as soon as connectivity is restored.

Efficient incident handling minimizes the potential for negative exposure. Some guidelines for the level

of detail to provide are given here (taken from RFC 2196):

- *Keep the technical level of detail low.* Detailed information about the incident may provide enough information for others to launch similar attacks on other sites, or even damage the site's ability to prosecute the guilty party after the event has ended.
- *Work with law enforcement officials to ensure that evidence is protected.* Many times, you may have to show law enforcement officials why they should be involved in your case---they are not yet equipped to handle an initial response to an electronic security incident. If prosecution is involved, ensure that the evidence collected is not divulged to the public.
- *Delegate all handling of the public to in-house PR people who know how to handle the press.* These PR people should be trained professionals who know how to handle the public diplomatically.
- *Do not break or halt lines of communication with the public.* Bad PR can result if the public doesn't hear anything or is speculating on its own.
- *Keep the speculation out of public statements.* Speculation of who is causing the incident or the motives behind the incident are very likely to be in error and can give a poor impression of the people handling the incident (for example, that they are given to speculation rather than to factual analysis).
- *Do not allow the public attention to detract from the handling of the event.* Always remember that the successful closure of an incident is of primary importance.

Never allow anyone within the organization who is not properly trained to talk to the public. The most embarrassing leaks and stories typically originate from employees who are cornered by a persistent press person. Employees are usually instructed to not talk to the public concerning contracts, mergers, financial reports, and so on for the same reasons, so the typical corporate policy need only be extended to include security incidents.

Responding to the Incident

One of the most fundamental objectives is to restore control of the affected systems and to limit the impact and damage. In the worst-case scenario, whether it is an inside or outside attacker, you can usually shut off the attacker's access point. Doing so limits the potential for further loss, damage, or disruption but can have some adverse effects:

- It can be disruptive to legitimate users.
- You cannot obtain more legal evidence against the attacker.
- You may not have enough information to find out who the attacker is or what his or her motivation is.

An alternative is to wait and monitor the intruder's activities. This may provide evidence about who the intruder is and what he or she is up to. This alternative must be considered very carefully because delays in stopping the intruder can cause further damage. Although monitoring an intruder's activities can be useful, it may not be worth the risk of further damage.

Keep Accurate Documentation

Documenting all details relating to the incident is crucial because doing so provides the information necessary to later analyze any cause-and-effect scenarios. Details recorded should include who was notified and what actions were taken---all with the proper date and time. A log book for incident response should be kept that will make it easier to sort through all the details later to reconstruct events in their proper chronological order. For legal purposes, all docu-mentation should be signed and dated to avoid the invalidation of any piece of data that could later be used as evidence if legal action is to be taken.

Real-World Example Scenarios

Let's take a look at two example scenarios of real-world problems.

Scenario #1

The Internet connection mysteriously collapses a number of different times. By looking at the backbone network traffic charts, we see that there is a huge disparity between incoming and outgoing traffic. A huge number of outgoing packets are heading out to the Internet without any responses. Looks like we are generating a significant number of UDP packets. What is going on? Using our cool RMON probes and data, we see that 3 million packets per minute are being generated from an internal computer to some Internet Relay Chat (IRC) site in Russia. We block out the IRC site thinking that it is the problem. Later that night, it happens again to an IRC site in the Netherlands. The corporation again loses connectivity to the Internet.

This time, we are ready and watching for the perpetrator. We catch the intruder in the act: It turns out he or she is using one of the corporate computers to attack the IRC sites! The intruder was running a version of a SPRAY program that floods UDP packets to the victim. A check of the computer being used as the attack launch site finds that its password file has over 200 compromised accounts, so there is little chance of being able to lock the intruder out.

We have a large dilemma. This computer is a critical corporate resource for thousands of users. Do we take down the machine, notify all users, and create new passwords? Do we take down the Internet connection and lock out everyone working on the Internet?

The answer depends largely on the corporation's decision about how to handle this and other types of scenarios. How would you handle this scenario in your environment?

Scenario #2

Some internal intrusions may be completely harmless. Consider the following scenario.

You notice that traffic from an unusual IP address is on the engineering network and is connecting to privileged machines. Instead of hastily turning off the Internet connection and locking 1,000 users out, you take some time to trace the connection. You find no entry for the IP address in DNS and it's not a new corporate IP address (which you probably would have known about). It turns out that the supposed intruder is an engineer who has a separate account at home and has dialed in to the modem he or she set

up on the workstation in his or her office at work. Once logged on to the workstation, the engineer was able to continue gaining access to the rest of the engineering network (see Figure 7-1).

Figure 7-1: An Unintentional Breach

Assuming that the employee was just working overtime, you should put in place procedures about the proper usage of network access. Most likely, the corporation would mandate that all access to the corporate network be achieved through legitimate connections.

Recovering from an Incident

Recovering from an incident involves a post-mortem analysis of what happened, how it happened, and what steps should be taken to prevent a similar incident from occurring again.
A formal report with the correct chronological sequence of events should be presented to management along with a recommendation of further security measures to be put in place. It may be prudent to perform a new risk analysis at this time and to change past security policies if the incident was caused by a poor or ineffective policy.

It is not very productive to turn your computing environment into a virtual fortress after surviving a security breach. Instead, re-evaluate current procedures and prepare yourself before another incident occurs so that you can respond quickly. Having procedures formulated before an incident happens allows the system operators to tell management what is expected should an incident occur. This arrangement aids in setting expectations about how quickly the incident can be handled and which one of many possible outcomes results from a security incident.

Summary

This chapter focused on how to deal with security incidents. Planning and developing procedures to handle incidents before they occur is a critical part of any security policy. If these procedures are in place it will be easier for the group responsible for dealing with security incidents to prioritize its actions. All organizations should create a centralized group to be the primary focus when an incident happens.

The hardest part is actually determining whether or not some suspicious system or user behavior is really an incident. However, after it is determined that a security breach has indeed occurred, one of the most fundamental objectives is to restore control of the effected systems and to limit the impact and damage. Only when control is restored can the recovery process begin. Recovering from an incident involves a post-mortem analysis of what happened, how it happened, and what steps should be taken to prevent a similar incident from occurring again.

Posted: Wed Jun 14 11:34:28 PDT 2000

Table of Contents

Securing the Corporate Network Infrastructure

Securing the Corporate Network Infrastructure

This chapter explains how to secure the corporate campus networking infrastructure. The sample main corporate infrastructure shown in Figure 8-1 is the basis for all configuration examples in this chapter.

Figure 8-1: The Main Corporate Infrastructure

This chapter looks at features specific to equipment provided by Cisco Systems, Inc. and explains how to configure specific devices to incorporate the following elements of a security architecture:

- Identity
- Integrity
- Confidentiality
- Availability
- Audit

Many of these functions can also be used from other products if they are available. Many examples are given with references to commands used for the various devices most commonly found as part of the corporate network infrastructure:

- Routers
- Switches
- Network access servers
- Firewalls

The configuration examples are guidelines; in many instances, you will have to modify them to fit your specific environment. At the end of the chapter is a sample configuration for Cisco IOS devices, switches, and the PIX firewall to ensure a secure network infrastructure.

Identity

Authentication and authorization are considered together under the heading "identity." *Authentication* pertains to users identifying themselves with specified credentials, such as a username and a password.

Authorization refers to the subsequent access rights to which the successfully identified person has privileges. Many times, these processes can be taken as separate entities. However, in most of the cases we will consider, authorization and access privileges are a natural second step after a person has been successfully authenticated. The two processes are therefore considered together under the topic of identity.

Controlling Network Device Access

If an intruder were to gain physical console access or logical terminal access into a networking device (such as a router, switch, firewall, or network access server), that person could do significant damage to your network. The intruder would be able to reconfigure devices or gain information about the device's configuration. Some common ways to get access to network devices are through console ports, virtual terminal (vty) ports, and auxiliary (aux) ports. At a minimum, users should be authenticated before they can gain device access through these ports.

In addition to the traditional passwords, many devices use one-time password schemes and the RADIUS and/or TACACS+ protocol to authenticate users before granting them access to network devices. The advantage of using RADIUS or TACACS+ is to take advantage of the scaleable common database for authentication purposes. Should passwords change, only one database (or possibly a few databases) must be modified when using RADIUS and/or TACACS+; without these protocols, each individual device must be modified.

Note Keep in mind that controlling physical access to any infrastructure device is critical. All network infrastructure equipment should be in areas accessible only by authorized personnel. Without physical access control, the rest of the control measures are useless.

Basic Versus Privileged Access

Most devices typically have two modes of operation:

- Basic
- Privileged

Both modes are password protected. The basic-mode commands are used for everyday system monitoring. The privileged commands are used for system configuration and basic trouble-shooting. The passwords should be different for basic and privileged modes. For all Cisco devices, after you access the system and enter an initial login sequence, the system enters basic mode, which gives you access to only basic-mode commands. You can enter privileged mode by entering the enable command followed by the privilege-mode password.

Cisco IOS Devices

For Cisco IOS devices, basic access mode is denoted by the > prompt after the hostname; privileged access is denoted by the # prompt after the hostname. Table 8-1 shows the commands accessible for basic access; Table 8-2 shows the commands for privileged access (that is, enable mode).

Table 8-1: Cisco IOS Basic Access Commands

Command	Description
Router>?	
<1-99>	Session number to resume
access-enable	Create a temporary access list entry
clear	Reset functions
connect	Open a terminal connection
disable	Turn off privileged commands
disconnect	Disconnect an existing network connection
enable	Turn on privileged commands
exit	Exit from the EXEC
help	Description of the interactive help system
lock	Lock the terminal
login	Log in as a particular user
logout	Exit from the EXEC
mrinfo	Request neighbor and version information from a multicast router
mstat	Show statistics after multiple multicast traceroutes
mtrace	Trace reverse multicast path from destination to source
name-connection	Name an existing network connection
pad	Open an X.29 PAD connection
ping	Send echo messages

ppp	Start IETF Point-to-Point Protocol (PPP)
resume	Resume an active network connection
rlogin	Open an rlogin connection
slip	Start Serial-Line IP (SLIP)
systat	Display information about terminal lines
telnet	Open a Telnet connection
terminal	Set terminal line parameters
traceroute	Trace the route to destination
tunnel	Open a tunnel connection
where	List active connections
x3	Set X.3 parameters on PAD

Table 8-2: Cisco IOS Privileged Access Commands

Command	**Description**
Router#?	
<1-99>	Session number to resume
access-enable	Create a temporary access list entry
access-template	Create a temporary access list entry template
bfe	For manual emergency mode settings
clear	Reset functions

clock	Manage the system clock
configure	Enter configuration mode
connect	Open a terminal connection
copy	Copy configuration or image data
debug	Debugging functions (*see also* undebug)
disable	Turn off privileged commands
disconnect	Disconnect an existing network connection
enable	Turn on privileged commands
erase	Erase flash or configuration memory
exit	Exit from the EXEC
help	Description of the interactive help system
lock	Lock the terminal
login	Log in as a particular user
logout	Exit from the EXEC
mbranch	Trace multicast route down tree branch
mrbranch	Trace reverse multicast route up tree branch
mrinfo	Request neighbor and version information from a multicast router
mstat	Show statistics after multiple multicast traceroutes
mtrace	Trace reverse multicast path from destination to source
name-connection	Name an existing network connection

no	Disable debugging functions
pad	Open an X.29 PAD connection
ping	Send echo messages
ppp	Start IETF Point-to-Point Protocol (PPP)
reload	Halt and perform a cold restart
resume	Resume an active network connection
rlogin	Open an rlogin connection
rsh	Execute a remote command
send	Send a message to other tty lines
setup	Run the SETUP command facility
show	Show running system information
slip	Start Serial-Line IP (SLIP)
start-chat	Start a chat-script on a line
systat	Display information about terminal lines
telnet	Open a Telnet connection
terminal	Set terminal line parameters
test	Test subsystems, memory, and interfaces
traceroute	Trace the route to destination
tunnel	Open a tunnel connection

undebug	Disable debugging functions
verify	Verify checksum of a flash file
where	List active connections
write	Write running configuration to memory, network, or terminal

The authentication of enable mode in Cisco IOS devices can take one of three forms:

- A password
- A secret
- TACACS+

The following example is taken from a router in configuration mode to see the options for configuring authentication for enable mode:

Router(config)#**enable ?**

last-resort Define enable action if no TACACS+ servers respond

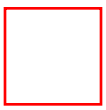
password Assign the privileged level password

secret Assign the privileged level secret

use-tacacs Use TACACS+ to check enable passwords

Both the enable password and enable secret commands allow you to establish an encrypted password that users must enter to access the privileged enable mode.

The difference between the enable password and the enable secret command lies in the encryption algorithm used to encrypt the password or secret. The enable password command uses a reversible encryption algorithm (denoted by the number 7 in the configuration option). This reversible algorithm is necessary to support certain authentication protocols (notably CHAP), where the system needs access to the cleartext of user passwords. However, enable secrets is encrypted using the MD5 algorithm (denoted by the number 5 in the configuration option). This algorithm is not reversible and is more secure. The strength of the encryption used is the only significant difference between the two commands.



Tips It is recommended that you use the enable secret command because it has an improved encryption algorithm over the enable password command.

The following example shows the configuration options for enable password and enable secret:

Router(config)#**enable password ?**

0 Specifies that an unencrypted password will follow

7 Specifies that a hidden password will follow

LINE The unencrypted (cleartext) enable password

level Set exec level password

Router(config)#**enable secret ?**

0 Specifies that an unencrypted password will follow

5 Specifies that an encrypted secret will follow

LINE The unencrypted (cleartext) enable secret

level Set exec level password

You can enter enable password or enable secret in unencrypted form, as in this example:

Router(config)#**enable secret 0 thisisasecret**

Should you do so, however, enable password or enable secret is shown in the configuration file as follows:

enable secret 5 1dLOD$QR.onv68q3326pzM.Zexj1

You can also enter the secret in encrypted form, as in this example:

Router(config)#**enable secret 5 1dLOD$QR.onv68q3326pzM.Zexj1**

To do so, however, the encrypted secret would have to be copied from a previously encrypted secret. For this example, the (unencrypted) secret the user would type is **thisisasecret**.

You cannot recover a lost encrypted password. You must clear nonvolatile random-access memory (NVRAM) and set a new password. Entering enable password or enable secret in encrypted form should be done with caution.

The following example shows the configuration file after both enable secret and enable password have been configured:

hostname Tallinn

!

enable secret 5 1dLOD$QR.onv68q3326pzM.Zexj1

enable password 7 047E050200335C465817

TIP If you configure both the enable secret and the enable password commands, the enable secret command takes precedence.

It is recommended that you use enable secret instead of enable password because the former command

provides a more secure encryption algorithm for the secret in the configuration. The enable secret command provides more security for your configuration files should they be stored remotely on a TFTP server. Passwords should never be seen in cleartext when you view any configuration files.

If TACACS+ authentication is chosen for enable mode, you can specify a back-up authentication mechanism in the event that connection to the TACACS+ server is not available.

If you use the enable use-tacacs command, you must also specify tacacs-server authentication enable or you will be locked out of the privileged enable mode.

The Cisco IOS software has incorporated additional user controls through which privilege levels can be assigned to various commands to further limit administrative access. Many times, you may want to assign particular members of the staff only a subset of the privileged enable commands. Cisco IOS allows 16 privilege levels, numbered 0 through 15. Level 1 is the current basic mode, and level 15 is the current privileged mode accessible through the enable command.

Note There are five commands associated with privilege level 0: disable, enable, exit, help, and logout. If you configure TACACS+ authorization for a privilege level greater than 0, these five commands are not included.

Both enable password and enable secret can be configured to provide for the privilege level authentication. The following examples show how to configure either enable password or enable secret to gain access to a specific privilege level:

Router(config)#**enable password level 10 ?**

0 Specifies that an unencrypted password will follow

7 Specifies that a hidden password will follow

LINE The unencrypted (cleartext) enable password

Router(config)# enable secret level 10 ?

0 Specifies that an unencrypted password will follow

5 Specifies that an encrypted secret will follow

LINE The unencrypted (cleartext) enable secret

Here is a specific example of the privilege level command used in conjunction with enable secret to assign different commands to different privilege levels. In this case, network operators can log in with a secret configured for level 9 privilege access; once properly authenticated, these operators are allowed to reload the routers and look at statistics using the show command. Such a configuration would look like this:

Hostname Tallinn

!

privilege exec level 9 show

privilege exec level 9 reload

enable secret level 9 5 1dLOD$QR.onv68q3326pzM.Zexj1

The network operators are given the secret; then, they can access the appropriate commands using the following command at the router prompt:

router> enable 9

password: **<secret for level 9>**

NOTE The write terminal/show running-config command displays all the commands the current user can modify (that is, all the commands at or below the user's current privilege level). The command does not display commands above the user's current privilege level because of security considerations.

The show config/show startup-config command does not really show the configuration. It simply prints out the contents of NVRAM, which just happens to be the configuration of the router at the time the user does a write memory.

To enable a privileged user to view the entire configuration in memory, the user must modify the privileges for all commands configured on the router. This approach is not recommended because it is quite cumbersome. Instead, the following alternative configuration is suggested:

username showconfig password foo

username showconfig priv 15 autocommand write terminal

With this approach, anyone who knows the foo password can show the configuration by doing an extra login on a spare vty.

Using TACACS+ Authorization to Control Access to Specific Commands on IOS Routers

Instead of using privilege levels to define varying command privileges, you can achieve the same result using TACACS+ authorization. On the router, you would use this command:

aaa authorization command 15 tacacs+ none

On the TACACS+ server, you have

group=partner_company {

default service = permit

cmd = crypto {

deny .*

}

cmd = aaa {

```
deny .*
}
cmd = tacacs-server {
deny .*
}
cmd = no {
deny crypto.*
deny aaa.*
deny tacacs.*
}
}
user = luser {
login = des slslkdfjse
member=partner_company
}
```

The first portion of the cmd = crypto ... statement denies any crypto, aaa, and tacacs configuration commands. The second portion of the statement does not allow the group to remove the crypto, aaa, or tacacs commands.

Cisco Switches

For Cisco switches, basic access mode is denoted by the > prompt after the system prompt; privileged access is indicated by the word (enable) in the system prompt. Table 8-3 displays the basic mode commands; Table 8-4 shows the privileged mode commands (both tables list the commands available when your system is equipped with a Supervisor Engine I or II module).

Table 8-3: Cisco Switch Basic Access Commands

Command	**Description**
Switch> ?	
enable	Enable privileged mode
help	Show this message

history	Show contents of history substitution buffer
ping	Send echo packets to hosts
quit	Exit from the administration session
session	Tunnel to ATM or router module
set	Set, use set help for more information
show	Show, use show help for more information
wait	Wait for *x* seconds

Table 8-4: Cisco Switch Privilege Access Commands

Command	**Description**
Switch> (enable) ?	
clear	Clear, use clear help for more information
configure	Configure system from terminal/network
disable	Disable privileged mode
disconnect	Disconnect user session
download	Download code to a processor
enable	Enable privileged mode
help	Show this message
history	Show contents of history substitution buffer
ping	Send echo packets to hosts

quit	Exit from the administration session
reconfirm	Reconfirm VMPS
reset	Reset system or module
session	Tunnel to ATM or router module
set	Set, use set help for more information
show	Show, use show help for more information
slip	Attach/detach Serial Line IP (SLIP) interface
switch	Switch to standby <*clock*|*supervisor*>
telnet	Telnet to a remote host
test	Test, use test help for more information
upload	Upload code from a processor
wait	Wait for *x* seconds
write	Write system configuration to terminal/network

To authenticate a user for privileged access on Cisco switches, two forms of authentication are possible:

- Using a TACACS+ server
- Using a locally defined password

The command to specify the authentication is as follows:

set authentication enable {tacacs | local} {enable | disable}

The locally defined enable password is configured using the set enablepass command. The command prompts you for the old password. If the password you enter is valid, you are prompted to enter a new password and to verify the new password. A zero-length password is allowed.

Console> (enable) set enablepass

Enter old password: <old_password>

Enter new password: <new_password>

Retype new password: <new_password>

Password changed.

Console> (enable)

Tips At this time, the passwords on the switch are not encrypted. Care should be taken when changing or viewing configurations to ensure that no unauthorized person can view the password.

Cisco PIX Firewall

The *Private Internet Exchange (PIX) firewall* contains a command set based on Cisco IOS technologies. The unprivileged basic mode is available when you first access the PIX firewall; this mode displays the > prompt. The basic mode lets you view restricted settings. Privileged mode displays the # prompt and lets you change current settings. Any unprivileged command also works in privileged mode. Tables 8-5 and 8-6 display the basic mode and privileged mode commands for the PIX firewall.

Table 8-5: Cisco PIX Firewall Basic Access Commands

Command	Description
pixfirewall>?	
enable	Enter privileged mode or change privileged mode password
quit	Quit
uptime	Show system uptime
who	Show active administration sessions on PIX

Table 8-6: Cisco PIX Firewall Privileged Access Commands

Command	Description
pixfirewall# ?	

configure	Configure from terminal, floppy, or memory
debug	Enable debugging ICMP trace
disable	Exit from privileged mode
enable	Modify enable password
groom	Groom the flash by rewriting it
http	Add authorized IP addresses for HTTP access to PIX
kill	Terminate a Telnet session
passwd	Change Telnet and HTTP console access password
ping	Test connectivity from specified interface to <ip>
quit	Quit
radius-server	Configure a RADIUS server
reload	Halt and reload system
session	Internal router console
syslog	Log messages to SYSLOG server
tacacs-server	Configure a TACACS+ server
telnet	Add authorized IP addresses for Telnet access to PIX
uptime	Show system uptime
who	Show active administration sessions on PIX
write	Write configuration to flash, floppy, or terminal; or erase the flash

Currently, the PIX allows only the privileged mode password to be specified locally with this command:

enable password <*password*> [encrypted]

The *password* is encrypted in the configuration file using MD5. If you use the word encrypted during configuration, you are specifying that the password you enter is already encrypted. The encrypted password must be 16 characters in length.

Following is an example of entering the enable password command unencrypted:

enable password thisisasecret

show enable password

enable password feCkwUGktTCAgIbD encrypted

Following is an example of entering the enable password command encrypted:

enable password thisisgibberish encrypted

show enable password

enable password thisisgibberish encrypted

Line Access Controls

The console is a terminal line port. Most devices have console ports that allow physical access to a given device. A console port is extremely useful in cases where the network is down; it is often a last-resort method of communicating with the device. The vty ports are usually for remote console access. Administrators can remotely Telnet into a device to access and perform all commands as if they were attached to the device with a physical console connection. Auxiliary ports can be used for modem support and asynchronous connections.

Authentication and authorization capabilities for console, Telnet, or auxiliary lines vary by product. At a minimum, users should be authenticated before gaining device access. Consistent authentication mechanisms should be used if possible to simplify keeping track of passwords.

Cisco IOS

For line access in Cisco IOS devices (which can be a console, a vty, or an aux port), you can set a simple password, where no username has to be entered. The passwords are either cleartext or encrypted in the configuration file:

Router# Configure terminal

Router(config)# line console 0

Router(config-lin)# **password ?**

0 Specifies that an unencrypted password will follow

7 Specifies that a hidden password will follow

LINE The unencrypted (cleartext) line password

For additional control, the line access can be controlled by authentication with a username and a corresponding password. The username can be set either from a database local to the IOS device or using TACACS+. TACACS+ has the advantage of more flexibility and an easier database management control mechanism:

Router# Configure terminal

Router(config# line console 0

Router(config-lin)# **login ?**

local Local password checking

tacacs Use TACACS+ server for password checking

The local keyword designates using the local database for authentication. Therefore, a local database must be configured on the router. This is done using the following commands:

Router# Configure terminal

Router(config)# **username staff password ?**

0 Specifies that an unencrypted password will follow

7 Specifies that a hidden password will follow

LINE The unencrypted (cleartext) line password

A sample configuration in which the console access is secured using a simple password but access to the vty and aux ports are secured by a local database of users is shown here:

username staff password 7 082C495C0012001E010F02

username admin password 7 0574837212001E010F0296

!

line con 0

password 7 047E050200335C465817

line aux 0

login local

line vty 0 4

login local

!

You can limit the access to inbound and outbound Telnet connections on vty ports by putting in access lists (that is, filters) that only permit or deny access from or to specified networks or hosts. A thorough explanation of Cisco IOS access lists is given in Chapter 9, "Securing Internet Access." The following example allows only incoming Telnet access from hosts on network 144.254.5.0:

access-list 3 permit 144.254.5.0 0.0.0.255

!

line vty 0 4

access-class 3 in

Interactive access can be completely prevented by applying the configuration command no exec to any asynchronous line. This command allows only an outgoing connection for a line. When a user tries to Telnet to a line with the no exec command configured, the user gets no response when he or she presses the Return (Enter) key at the login screen. To define which protocols to use to connect to a specific line of the router, use the transport input line configuration command:

transport input {all | lat | mop | nasi | none | pad | rlogin | telnet | v120}

Note The none option became the default in Cisco IOS Release 11.1. Before Release 11.1, the default was all.

The following example shows a configuration in which the vty lines accept only incoming Telnet connections, and the aux port does not accept any incoming connections and will not even give a login prompt:

line vty 0 4

transport input telnet

line aux 0

no exec

transport input none

Another useful command is to use session timeouts for unattended console or vty ports. By default, the timeout is 10 minutes and can be modified with the exec-timeout command. Here is an example in which

console and Telnet port accesses time out after 1 minute and 30 seconds of inactivity:

line console 0

exec-timeout 1 30

line vty 0 4

exec-timeout 1 30

Cisco Switches

You can access the switch command line interface (CLI) from a console terminal connected to an EIA/TIA-232 port or through a Telnet session. The CLI allows fixed baud rates. Telnet sessions automatically disconnect after remaining idle for a user-defined period of time.

The set authentication enable command is used to designate authentication using either the TACACS+ server or a local password authentication to determine whether a user has privileged access permission:

set authentication enable {tacacs | local} {enable | disable}

The set authentication login command is used to designate either TACACS+ authentication or local password authentication for Telnet access:

set authentication login {tacacs | local} {enable | disable}

The Cisco switches also have a limited authorization capability through the IP permit list command. When you enable this feature, Telnet access and SNMP services are authorized only for the IP addresses of the hosts configured on the permit list. The IP permit list is the first level of security for the Telnet and SNMP protocols. All other Transmission Control Protocol/Internet Protocol (TCP/IP) services continue to work for any hosts when you enable the IP permit list. Outbound Telnet, Trivial File Transfer Protocol (TFTP), and other IP-based services remain unaffected by the IP permit list. SNMP from nonpermitted IP addresses have no response---that is, the request times out. Notifications of unauthorized access attempts are available through SNMP traps and syslog options.

Before enabling the IP permit feature, be sure that you configure your IP address in the permit list, especially when configuring through SNMP. Failure to do so results in immediate disconnection from the system being configured. I recommend that you disable the IP permit feature before clearing the IP permit entries or host addresses.

Here is an example that allows IP addresses from network 144.254.5.0 and the hosts 144.254.7.10 and 144.254.7.20 to have either Telnet or SNMP access to the switch:

ip permit 144.254.5.0 0.0.0.255

ip permit 144.254.7.10

ip permit 144.254.7.20

ip permit enable

Cisco PIX Firewall

The PIX firewall CLI can be accessed using the console connector or over Telnet. The following command enables authentication using either TACACS+ or RADIUS:

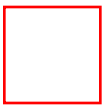
aaa authentication [any|telnet] console tacacs+|radius

When used with the console option, this command enables authentication service for access to the PIX firewall console over Telnet or from the console connector on the PIX firewall unit.

If used with the any keyword, access to the serial console or Telnet to the PIX's console must be authenticated with the authentication server. If used with the telnet keyword, only Telnet access to the PIX firewall console requires authentication from the authentication server.

Telnet access to the PIX firewall console is available only from the inside interface (*inside interface* refers to the interface connected to the corporate network), as shown in Figure 8-2.

Figure 8-2: The Inside and Outside Interfaces of the PIX Firewall



Telnet access requires use of the telnet command:

telnet local_ip [*netmask*]

The telnet command lets you decide who can access the PIX firewall with Telnet. Up to 16 hosts or networks are allowed access to the PIX firewall console with Telnet, 5 simultaneously.

To set up a password for Telnet access to the console, you must configure the passwd command:

passwd password [*encrypted*]

The default password is cisco. The passwd command sets a password for Telnet and gives the PIX Firewall Manager access to the firewall console. An empty password can be used and is also changed into an encrypted string. Any use of a write command displays or writes the password in encrypted form. After passwords are encrypted, they cannot be changed back to plaintext. Consider this example:

passwd secretforpix

show passwd

passwd jMorNbK0514fadBh encrypted

Note Default password or passwords should be changed before any network infrastructure device is put into production use.

Authentication of the serial console creates a potential deadlock situation if the authentication server requests are not answered and you must access the console to attempt diagnosis. If the console login request to authentication times out, you can gain access to the PIX firewall from the serial console by entering the PIX username and the enable password.

SNMP Security

The Simple Network Management Protocol (SNMP) is often used to gather statistics and remotely monitor network infrastructure devices. It is a very simplistic protocol and therefore has virtually no security built into its original version. In SNMP Version 1 (SNMPv1), *community strings* (passwords) are sent in cleartext. These community strings are used to authenticate messages sent between the SNMP manager and the agent. These community strings can easily be stolen by someone eavesdropping on the wire.

SNMPv2 addresses some of the known security weaknesses of SNMPv1. Specifically, Version 2 uses the MD5 algorithm to authenticate messages between the SNMP server and the agent.

In most devices, SNMP has two options:

- The read-only (ro) option specifies that you can only read any SNMP MIB objects.
- The read-write (rw) option specifies that the SNMP manager can read and/or modify SNMP MIB objects.

Whenever possible, configure filters to allow only specified hosts to have SNMP access to devices.

HTTP Security

To facilitate configuration and management of network devices, many manufacturers are implementing HTTP servers into devices to create cross-platform, easy management solutions. Keep in mind that the communication between the HTTP client and the HTTP server embedded into the network infrastructure devices should be secured. This means that you must use some of the technologies (such as HTTPS, SSL, SSH, or possibly even IPsec) discussed in Chapter 2, "Security Technologies."

Cisco IOS Devices

You can issue most of the Cisco IOS commands using a Web browser. The Cisco IOS feature is accessed by using the Cisco Web browser interface, which is accessed from the router's home page. All Cisco routers and access servers running Cisco IOS Release 11.0(6) or later have an HTTP server, which is an embedded subcomponent of the Cisco IOS software. The HTTP server allows users with a privilege level of 15 (or any other configured privilege level) to access the Cisco Web browser interface.

Router(config)#**ip http ?**

access-class Restrict access by access class

authentication Set HTTP authentication method

port HTTP port

server Enable HTTP server

Note Before Cisco IOS Release 11.3, only users with privilege level 15 could use this feature, and the only authentication mechanism was the enable password.

You can use three different methods to authenticate HTTP:

Router(config)#**ip http authentication ?**

enable Use enable passwords

local Use local username and passwords

tacacs Use TACACS+ to authorize user

To permit users to have access to the HTTP server, you must enable the Cisco Web browser interface with the following command:

ip http server

After you have enabled the Cisco Web Browser interface, users can use a Web browser to access Web pages associated with the router and to issue commands. Cisco IOS software currently allows only users with a privilege level of 15 to access the predefined home page for a router or access server. If you have a privilege level other than 15, you can issue Cisco IOS commands from a Web page where the commands defined for your specific user privilege level are displayed.

To request a router Web page for a privilege level other than the default of 15, perform the following steps:

Step 1 Type the following command in the URL field of your Web browser and press Enter:

`http://router-name/level/mode/command`

The browser prompts you for the password.

Step 2 Type the password and press Enter. The Web browser should display a Web page specific to your user privilege level, mode, and the command you have requested.

The following example shows what you would type in the URL field of your Web browser to request a user privilege level of 9 on a Cisco router named Tallinn:

http://Tallinn/level/9/exec

The following example shows a configuration in which HTTP access is allowed only from specific hosts and in which the enable password is the method of authenticating HTTP server users:

access-list 6 permit 144.254.5.0 0.0.0.255

!

ip http server

ip http access-class 6

ip http authentication enable

Cisco PIX Firewall

The PIX firewall can also be accessed using a Web-based configuration tool. The commands required on the PIX are as follows:

http ip_address [netmask]

passwd password

After you have enabled the Cisco PIX Web browser interface by specifying a host IP address, users can use a Web browser on the specified host to access Web pages associated with the PIX firewall console HTML management interface.

Integrity

Integrity in the campus infrastructure requires that any software image (where *image* is the executable binary images of programs) running on a device must be valid, and that none of the configurations have been altered by any person other than permitted personnel. You want to ensure that only permitted devices are connected to the network, and that no one is injecting any unwanted data. Integrity is largely available through the use of data-authentication methods such as checksums and hash functions.

Image Authentication

When downloading images onto any network infrastructure device, you may want to ensure that the images have not been modified or changed in transit. Most devices have a checksum verification to ensure that the image will load correctly when the device is rebooted. Any time the checksum does not verify correctly, the image should be erased and replaced with an image containing a successful checksum.

All Cisco software releases on Cisco Connection Online (CCO) and all floppy-based Cisco IOS releases subsequent to and including 10.2(5) are protected by MD5 image authentication. MD5, defined in RFC 1321, scans the image and produces a unique 128-bit checksum. The mathe-matics of the MD5 algorithm give you a checksum mechanism that makes it computationally not feasible to create a substitute file with the same checksum as a chosen target file and therefore helps prevent a specific, targeted attack on a specific file (in this case, the router image).

MD5 allows CCO users to verify that no bits in the image were corrupted during file transfer, reducing the possibility of loading corrupted software onto their routers. MD5 floppy verification ensures image integrity on diskette-based shipments.

Secure Workgroup

In corporate campus networks, there are increasingly more requests to provide integrity at the workgroup level. On the Catalyst 5000 series switch, the MAC address security feature allows the switch to block input traffic to an Ethernet or Fast Ethernet port when the MAC address of a station attempting to access the port is different from the configured MAC address (see Figure 8-3).

Figure 8-3: MAC Address Security

When a port receives a packet, the module compares the source address of that packet to the secure source address learned by the port. When a source address change occurs, the port is disabled, and the LED for that port turns orange. When the port is re-enabled, the port LED turns green.

Secure port filtering does not apply to trunk ports, where the source addresses change frequently.

MAC address security is configured with the following command:

set port security modNum/portNum(s) <*enable*|*disable*> [mac_addr]

If the MAC address is not given, the address is learned. After the address is learned, the address remains unchanged until the system relearns it when you reenter the command. The MAC address is stored in NVRAM and is maintained even after a reset. When a packet's source address does not match the allowed address, the port through which the packet came is disabled, and a link-down trap is sent to the SNMP manager.

If there were two critical servers connected to ports 2/1 and 2/2 on a switch, and port security was enabled, the configuration commands would be as follows:

Console> set port security 2/1 enable

Port 2/1 port security enabled with the learned mac address.

Console> set port security 2/2 enable 01-02-03-04-05-06

Port 2/2 port security enabled with 01-02-03-04-05-06 as the secure mac address.

Routing Authentication

Secure routing encompasses all areas that ensure routing integrity. The simplest way to create complete havoc in a network is to inadvertently inject bogus routes into the core network. This problem can be minimized through the use of route authentication and route filtering.

Route authentication ensures that routing updates come from a trusted source and that none of the data has been tampered with. It uses a *cryptographic checksum*, the one-way hash function, to ensure the authentication of a peer and the integrity of the contents of the routing update.

All peer routers must be configured with a specific key and encryption algorithm. The typical hash algorithms used are MD5, SHA-1, and IDEA. Cisco routers use MD5. The IP routing protocols that currently support route authentication include the following:

- Routing Information Protocol Version 2 (RIPv2)
- Border Gateway Protocol (BGP)
- Open Shortest Path First (OSPF)
- Enhanced Interior Gateway Routing Protocol (Enhanced IGRP)
- Intermediate System-to-Intermediate System (IS-IS)

For the IP routing protocols in Cisco routers that do not support MD5 route authentication, RIPv1, or IGRP, the command validate-update-source is used to ensure that the source IP address of incoming routing updates is on the same IP network as one of the addresses defined for the receiving interface. This feature is on by default.

Following is a sample configuration for two routers using Enhanced IGRP and route authentication (see Figure 8-4).

Hostname Building1

!

key chain To-Bldg2

key 1

key-string secretkey

accept-lifetime 08:30:00 June 6 1998 infinite

send-lifetime 08:30:00 June 6 1998 infinite

!

interface FE 1

ip address 144.254.4.2 255.255.255.0

ip authentication mode eigrp 109 md5

ip authentication key-chain eigrp 109 toBuilding1

!

router eigrp 109

network 144.254.0.0

Hostname Building2

!

key chain To-Bldg1

key 1

key-string secretkey

accept-lifetime 08:30:00 June 6 1998 infinite

send-lifetime 08:30:00 June 6 1998 infinite

!

interface FE 1

ip address 144.254.4.3 255.255.255.0

ip authentication mode eigrp 109 md5

ip authentication key-chain eigrp 109 toBuilding2

!

router eigrp 109

network 144.254.0.0

Note Router clocks should be synchronized with Network Time Protocol (NTP) if route authentication is to work properly.

Figure 8-4: Route Authentication

Route Filters and Routing Believability

By default, all dynamic routing protocols propagate routing information. At times, you may not want certain other devices or portions of your network to learn your network topology from the routing protocol. If this is the case, you must take explicit steps to prevent route propagation.

To prevent routing updates through a specified router interface, use the following command in router configuration mode:

passive interface [*interface type and number*]

To prevent other routers from learning one or more routes, you can suppress routes from being advertised in routing updates. You do this to prevent other routers from learning a particular device's interpretation of one or more routes. To suppress routes from being advertised in routing updates, use the following command in router configuration mode:

distribute-list {*access-list-number* | *name*} out [*interface-name*]

You may want to avoid processing certain routes listed in incoming updates. This feature does not apply to OSPF or IS-IS. To avoid processing certain routes, configure the following command in router configuration mode:

distribute-list {*access-list-number* | *name*} in [*interface-name*]

It is also possible to filter sources of routing information. You can do this to prioritize routing information from different sources, because some pieces of routing information may be more accurate than others. For Cisco IOS routers, an administrative distance is used to rate the trustworthiness of a routing information source, such as an individual router or a group of routers. In a large network, some routing protocols and some routers can be more reliable than others as sources of routing information. Also, when multiple routing processes are running in the same router for IP, it is possible for the same route to be advertised by more than one routing process. By specifying administrative distance values, you enable the router to intelligently discriminate between sources of routing information. The router will always pick the route with a routing protocol that has the lowest administrative distance. To filter sources of routing information, give the following command in router configuration mode:

distance *weight* [*address mask* [*access-list-number* | *name*]] [ip]

The *weight* argument can be an integer from 10 to 255. (The values 0 to 9 are reserved for internal use.) Used alone, the *weight* argument specifies a default administrative distance that the Cisco IOS software uses when no other specification exists for a routing information source. Routes with a distance of 255 are not installed in the routing table.

None of the preceding commands with the distribution and attributes attached to distributed routes should be played with lightly. In nontrivial networks, unpredictable and disruptive behavior can result.

Figure 8-5 shows a sample scenario in which the core network is comprised of 100BaseT switches connected to routers. 10BaseT interfaces are used to connect the local LANs using the RIP routing protocol. Route filtering is used to ensure that only default routes are announced on the local LANs and that no inadvertent misbehaving host can source a default route to the backbone.

Figure 8-5: Controlling Routing Information

Note If you are using DHCP on the LAN, the hosts on the LAN have a default router configured and there is no need to use RIP.

---

router eigrp 109

network 144.254.0.0

distance 255

distance 100 144.254.5.0 0.0.0.255

!

router rip

network 144.254.0.0

passive interface FE 1/0

distribute list 11 out

distance 255

!

access-list 11 permit 0.0.0.0

In this configuration, the passive interface command is not required for the LAN interface for Enhanced IGRP because Enhanced IGRP requires a neighbor before sending out routing updates. Because no other

Enhanced IGRP neighbors exist on the LAN, no routing updates will be sent out.

Data Confidentiality

Data confidentiality pertains to encryption. Whether or not you encrypt traffic within the main corporate infrastructure depends largely on how sensitive the information is and how likely it is that the data can be intercepted. In many environments, encryption of sensitive data occurs mostly between dial-in access points and Internet access points.

Within the corporate network infrastructure, confidentiality is important when accessing device information. Typically, it is prudent to encrypt the following:

- Telnet sessions to devices
- TFTP configuration downloads
- SNMP transactions to and from network devices
- HTTP access to device information

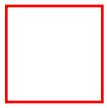
Kerberos-encrypted Telnet can be used to provide confidential Telnet access (the Cisco IOS software has included this feature since IOS version 11.2). Figure 8-6 shows some likely traffic that would require confidentiality for corporate infrastructure devices: Telnet sessions, SNMP sessions, TFTP sessions, and HTTP sessions. SSH and IPsec offer such support. The choice of which technology to use depends largely on what is currently supported in various products.

Network Availability

Network availability ensures that redundancy measures are in place and that features are configured to deter most common attacks. For critical devices, redundant power supplies are a must (don't forget to put them on separate circuits---not just separate power outlets---leading to a distinctly separate circuit breaker at the distribution panel). There are two reasons for incurring the expense of this kind of redundant power supply:

- Should a power supply fail, it might cause the circuit to go dead (that is, it will cause the breaker back at the distribution panel to blow open).
- The startup inrush current of two supplies in parallel can cause a breaker to open up.

Figure 8-6: Secure Access to Corporate Infrastructure Devices



You also may want to consider uninterruptable power supplies (UPSs) as insurance against catastrophic power outages. The UPSs should be rated to carry the maximum load for at least 10 minutes, and the

UPS should be able to deliver a notification or warning to the operator when the UPS senses that the primary power has failed for more than approximately 30 seconds. If there is a possibility of severe electrical outages, you may want to consider a back-up generator if your network requires continual uptime.

Note To determine whether the additional cost of a UPS makes sense, consider the impact of a power loss or down time for each piece of equipment. If a 100BaseT switch goes down in a building and is on the same power supply as most of the building's users, it may not make sense to keep the switch operational. However, if a router is on the same power supply, and the design of the network is such that the router must stay up to continue giving valid routing information to other routers, a UPS for the router would be necessary.

Redundancy Features

Equipment redundancy is largely an issue of how quickly the outage of a piece of equipment can be resolved. Any network infrastructure device that must be available 100 percent of the time is an obvious candidate for complete redundancy to cover the worst possible scenario. Many devices have incorporated redundant processor cards in high-performance equipment to ensure a smooth, dynamic failover in the event of single-card failures. In addition, new protocols or enhancements to existing protocols have been developed to ensure that redundancy with multiple boxes have failover capability without user intervention. To have redundant coverage, make sure that failover to the backup system happens automatically.

Cisco IOS

For critical network segments that cannot have any routing outages, the Cisco IOS devices supporting these segments should be configured with the *Hot Standby Router Protocol (HSRP)*. HSRP provides high network availability because it routes IP traffic from hosts on Ethernet, FDDI, or Token Ring networks without relying on the availability of any single router.

When HSRP is configured on a network segment, it provides a virtual MAC address and an IP address that is shared among routers in a group of routers that is running the HSRP. One of these devices is selected by the protocol to be the active router. The active router receives and routes packets destined for the group's MAC address. For n routers running the HSRP, there are n + 1 IP and MAC addresses assigned.

The HSRP detects when the designated active router fails, at which point a selected standby router assumes control of the hot standby group's MAC and IP addresses. A new standby router is also selected at that time.

Devices running the HSRP send and receive multicast UDP-based hello packets to detect router failure and to designate active and standby routers.

Note When the HSRP is configured on an interface, ICMP redirect messages are disabled by default for the interface.

The HSRP feature is configured with the following interface command:

standby [*group-number*] ip [*ip-address* [*secondary*]]

A number of group attributes can be configured to affect how the local router participates in the HSRP. Here is an example of these attributes:

Router(config)#int e 0

Router(config-if)#**standby ?**

<0-255> Group number

authentication Authentication string

ip Enable hot standby protocol for IP

mac-address Specify virtual MAC address for the virtual router

preempt Overthrow lower priority designated routers

priority Priority level

timers Hot standby timers

track Priority tracks this interface state

use-bia Hot standby uses interface's burned-in address

Consider the scenario shown in Figure 8-7.

Figure 8-7: An Example of HSRP Implementation

The configuration of a primary router is as follows:

hostname Primary

!

interface Ethernet1

ip address 144.254.1.1 255.255.255.0

no ip redirects

standby priority 200

standby preempt

standby ip 144.254.1.3

The configuration of a standby router is as follows:

hostname Standby

!

interface Ethernet1

ip address 144.254.1.2 255.255.255.0

no ip redirects

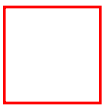
standby priority 101

standby ip 144.254.1.3

Cisco Switches

Switches are normally connected hierarchically, as shown in Figure 8-8.

Figure 8-8: An Example of Switch Hierarchy



In simple networks, the upper two levels of the hierarchy can be collapsed into a single backbone layer. Figure 8-8 shows the network topology after the spanning tree converges into a loop-free topology. The spanning tree has blocked the redundant links to avoid loops. Every access switch and distribution switch in the figure has a redundant uplink.

The Spanning Tree Protocol

The Spanning Tree Protocol (STP; IEEE 802.1D bridge protocol) is a link-management protocol that provides path redundancy while preventing undesirable loops in the network. For an Ethernet network to function properly, only one active path must exist between two stations. In STP, an algorithm calculates the best loop-free path through a switched network. Switches send and receive spanning-tree packets at regular intervals. The switches do not forward the packets, but use the packets to identify a loop-free path.

To provide path redundancy, STP defines a tree that spans all switches in an extended network. STP forces certain redundant data paths into a standby (blocked) state. If one network segment in the STP becomes unreachable, or if STP costs change, the spanning-tree algorithm reconfigures the spanning-tree topology and reestablishes the link by activating the standby path.

STP operation is transparent to end stations, which do not detect whether they are connected to a single LAN segment or to a switched LAN of multiple segments.

Election of the Root Switch

All switches in an extended LAN participating in STP gather information on other switches in the network through an exchange of data messages called *Bridge Protocol Data Units (BPDUs)*. This exchange of messages results in the following actions:

· The election of a unique root switch for the stable spanning-tree network topology.

· The election of a designated switch for every switched LAN segment.

· The removal of loops in the switched network by placing redundant switch ports in a backup state.

The STP root switch is the logical center of the spanning-tree topology in a switched network. All paths that are not needed to reach the root switch from anywhere in the switched network are placed in STP blocked mode.

Bridge Protocol Data Units

BPDUs contain information about the transmitting switch and its ports, including switch and port Media Access Control (MAC) addresses, switch priority, port priority, and port cost. The STP uses this information to elect the root switch and root port for the switched network, as well as the root port and designated port for each switched segment.

The stable active topology of a switched network is determined by the following:

· The unique switch identifier (MAC address) associated with each switch.

· The path cost to the root associated with each switch port.

· The port identifier (MAC address) associated with each switch port.

Each configuration BPDU contains the following minimal information:

· The unique identifier of the switch that the transmitting switch believes to be the root switch.

· The cost of the path to the root from the transmitting port.

· The identifier of the transmitting port.

The switch sends configuration BPDUs to communicate with and compute the spanning-tree topology. A MAC frame conveying a BPDU sends the switch group address to the destination address field. All switches connected to the LAN on which the frame is transmitted receive the BPDU. BPDUs are not directly forwarded by the switch, but the receiving switch uses the information in the frame to calculate a BPDU, and, if the topology changes, to initiate a BPDU transmission.

A BPDU exchange results in the following:

· One switch is elected as the root switch.

· The shortest distance to the root switch is calculated for each switch.

· A designated switch is selected. This is the switch closest to the root switch through which frames will be forwarded to the root.

· A port for each switch is selected. This is the port providing the best path from the switch to the root switch.

· Ports included in the STP are selected.

Creating a Stable STP Topology

If all switches are enabled with default settings, the switch with the lowest MAC address in the network becomes the root switch. In some cases, however, (because of traffic patterns, the number of forwarding ports, or line types), the switch picked as the root may not be the ideal root switch. By increasing the priority (that is, by lowering the numerical priority number) of the ideal switch so that it becomes the root switch, you force an STP recalculation to form a new, stable topology.

The time it takes to detect and correct failures is important. For Cisco switches, the Spanning Tree Protocol UPlinkFast and BackboneFast features reduce spanning-tree convergence times. *UPlinkFast* provides fast convergence after a spanning-tree topology change and achieves load balancing between redundant links using uplink groups. An *uplink group* is a set of ports (per VLAN), only one of which is forwarding at any given time. Specifically, an uplink group consists of the root port (which is forwarding) and a set of blocked ports, except for self-looping ports. The uplink group provides an alternative path in case the currently forwarding link fails.

Note The UPlinkFast feature is most useful in wiring-closet switches. This feature may not be useful for other types of applications.

To configure a switch as the primary root switch, enter this command:

set spantree root vlans [diameter *network_diameter*] [*hello hello_time*]

This command reduces the bridge priority (the value associated with the switch) from the default (32,768) to a significantly lower value, which allows the switch to become the root switch.

Note Run the set spantree root command on backbone switches or distribution switches only; do not run it on access switches.

To configure a switch as the secondary root switch, enter this command:

set spantree root [*secondary*] vlans [*dia network_diameter*] [*hello hello_time*]

You can run this command on more than one switch to create multiple backup switches in case the primary root switch fails.

The *BackboneFast Convergence feature* reduces the time needed for the spanning tree to converge after experiencing a topology change caused by indirect link failures. This feature complements the UPlinkFast feature just described. However, the BackboneFast Convergence feature is designed for all switches that experience indirect link failures.

Note For the BackboneFast feature to work, you must enable it on all switches in the network.

To configure the BackboneFast Convergence feature, enter this command:

set spantree backbonefast enable

The *Multiple Default IP Gateways feature* allows you to configure up to three default IP gateways. Defining multiple default IP gateways provides redundancy. In the event that the primary gateway cannot be reached, the switch uses the secondary default IP gateways in the order in which they are configured. This feature is configured with the following command:

set ip route destination gateway [metric] [primary]

Use the primary keyword to give a default IP gateway higher priority than the other default IP gateway(s). If you do not designate a primary default IP gateway, the system chooses the default IP gateway based on the order in which the gateways were configured. If two or more gateways are designated as primary gateways, the system chooses the *last* primary gateway configured to be the default IP gateway.

Cisco PIX Firewall

The Cisco PIX firewall is usually a critical device in most corporate infrastructures. To eliminate it being a single point of failure, it is prudent to install a redundant PIX firewall and to use the failover command to ensure fast dynamic recovery in the event that the primary PIX has a power failure or some other type of failure. Use the failover command without an argument after you connect the optional failover cable between your primary firewall and a secondary firewall.

Note Failover is supported only between identical PIX firewall models running the same software version.

Failover IP addresses must be configured on each interface card. The active unit of the failover pair uses the system IP addresses and the primary unit's MAC address; the standby unit uses the failover IP addresses and the secondary unit's MAC address. The system IP addresses and the failover IP addresses must be on the same subnet with no router between them.

When a failover occurs, each unit changes state. The newly active unit assumes the IP and MAC addresses of the previously active unit and begins accepting traffic. The new standby unit assumes the failover IP and MAC addresses of the unit that was previously the active unit. Because network devices see no change in these addresses, no ARP entries change or time out anywhere on the network.

Note Both PIX firewall units in a failover pair must have the same configuration. To accomplish this, always enter configuration changes on the active unit in a PIX firewall failover configuration. Use the write memory command on the active unit to save configuration changes to flash memory (nonvolatile

memory) on both the active and the standby units. Changes made on the standby unit are not replicated on the active unit.

Both units in a failover pair communicate through the failover cable. The two units send special failover hello packets to each other over all network interfaces and the failover cable every 15 seconds. The failover feature in PIX firewall monitors failover communication, the power status of the other unit, and the 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 standby unit.

Common Attack Deterrents

A multitude of types of attacks can bring a network to its knees. Many can be avoided or constrained with features that have been specifically developed to deter some of the better-known attacks.

Spoofed Packets

Although it is very difficult to actually recognize spoofed packets, some mechanisms can be used to help prevent some more obvious spoofs. Some of these packets may be caused by simple misconfigurations and routing loops. Whenever possible, filters should be put into place to ensure that only valid network addresses are permitted past the routers. All corporate infrastructure routers should have filters in place to disallow any obviously bogus traffic. For example, any edge router should deny traffic whose source address is one of the RFC reserved addresses shown in Table 8-7.

Table 8-7: RFC Reserved Addresses

Network IP Address	Mask
127.0.0.0	0.255.255.255
10.0.0.0	0.255.255.255
172.16.0.0	0.240.255.255
192.168.0.0	0.0.255.255

These IP addresses are specified for special use and are therefore designated as nonroutable in the Internet infrastructure. (That is, no Internet Service Provider will route these networks; therefore, no edge routers connecting to the Internet should receive packets with these addresses as a source.)

Some devices also have features to assist in tracking down the source of packets with bogus source addresses. For Cisco IOS devices, this is an extension of the access list logging feature that will show the

input interface for packets. (For a detailed discussion on Cisco IOS access lists, refer to Chapter 9, "Securing Internet Access.") It is enabled by adding log-input to an access list entry:

access-list 100 permit ip any host 171.69.233.3 log-input

The output from this command looks like this:

%SEC-6-IPACCESSLOGP: list 100 permitted udp 171.69.2.132(53)

(Ethernet0/0)-> 171.69.233.3(5775), 1 packet

%SEC-6-IPACCESSLOGDP: list 100 permitted icmp 171.69.2.75

(Ethernet0/0) -> 171.69.233.3 (0/0), 1 packet

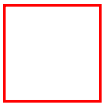
Fragmentation Attacks

To deter any attack based on fragments, the device must have an option to reassemble the original packet, ensure that the packet is valid, and then fragment the packet again before forwarding it. This check can severely limit system performance; think carefully before rushing off to implement this feature on every device. It is best to determine the most critical, vulnerable area and then place the deterrent there. In most instances in a large corporate network, the most vulnerable areas are at the network access points such as Internet access or dial-in access.

TCP SYN Attack

It is important to recognize that it is nearly impossible to stop a TCP SYN flooding attack. What can be done, however, is to constrain its impact on critical parts of the network. Typically, a firewall is set up to act as a proxy when a TCP connection is established. The firewall checks for incoming TCP connection requests and proxy answers on behalf of the destination device to ensure that the request is valid before connecting to the server (see Figure 8-9).

Figure 8-9: A TCP Proxy



After the firewall has established a genuine connection with the client and the server, it then merges these two connections into a single source/destination session. In the case of bogus requests, the firewall usually also has parameters to set very aggressive timeouts on half-open connections; it also has parameters to set threshold levels for the number of both outstanding connections and incoming rate of TCP-connection requests.

You should be careful when changing any TCP timer parameters. You don't want them so short that valid

connections from slower links time out.

On the Cisco IOS devices, the command to employ against TCP SYN attacks is this one:

ip tcp intercept <*access-list-number*>

ip tcp intercept mode watch

This command keeps track of the following information:

- How many session requests in the last one minute?
- How many incomplete sessions are there?
- How long is the wait for the final acknowledgment?

For the PIX firewall, you can issue the following command to limit the number of half-open TCP connections and total number of TCP connections allowed:

static 172.17.1.12 10.1.1.2 [*max_conns*] [*em_limit*]

In this syntax, *max_conns* is the maximum number of TCP connections allowed, and *em_limit* is the embryonic connection (half-open connection) limit. Refer to Chapter 9, "Securing Internet Access," for a more complete description.

Audit

The Audit function ensures that the network infrastructure is configured as expected. The function can also actively monitor network activity and includes the capability of intrusion detection.

All communication between auditing servers and network infrastructure devices should be authenticated and confidential (that is, encrypted) whenever possible. Audit logs should also be saved on write-once media (for example, WORM drives) or should be sent over a network to a trusted system that is inaccessible by the administrators of the system being audited. This way, if a break-in occurs, the intruder cannot erase his or her tracks.

Configuration Verification

It is important to verify that network infrastructure device configurations are valid to ensure proper implementation behavior. Verification of configurations is usually performed with some kind of modeling or simulation tool that can access all the infrastructure device configurations and then provide a simulation model that can be tested. Here is a list of some areas to be modeled:

- Mapping current network topology
- Identifying services on hosts
- Performing "what-if" scenarios to detect filtering problems
- Performing sample attack scenarios to find vulnerabilities

Monitoring and Logging Network Activity

This area intersects with network management; you can monitor system usage and traffic patterns to help determine what normal behavior is. There are numerous ways to accomplish this, but the main focus should be what to monitor and log. At the very least, you want to keep track of network usage and any high volumes of data traffic.

Event logging is very important in keeping track of various system information. Event logging automatically logs output from system error messages and other events to the console terminal. You may want to redirect these messages to another destination (such as syslog servers that can be used as a single destination point for all infrastructure system information). You should be able to specify the severity of the event to be logged; you also want to configure the logged output to be timestamped. The logged output can be used to assist real-time debugging and management and to track potential security breaches or other nonstandard activities throughout a network.

Note If you are using Cisco IOS routers, features such as Netflow and IP accounting may be useful to keep traffic statistics information. The IP accounting support also provides information identifying IP traffic that fails IP access lists. Identifying IP source addresses that violate IP access lists alerts you to possible attempts to breach security. To use this feature, you must enable IP accounting of access list violations using the ip accounting access-violations command. Users can then display the number of bytes and packets from a single source that attempted to breach security against the access list for the source destination pair.

Syslog Management

Syslog messages are based on the User Datagram Protocol (UDP) and are received on UDP port 514. The message text is kept under 512 bytes to ensure that the UDP packet is smaller than 576 bytes---the smallest packet that must be accepted by a host without packet fragmentation.

Syslog messages are categorized by eight priority levels, shown in Table 8-8.

Table 8-8: Syslog Priority Codes

Priority	Code	Description
LOG_EMERG	0	Emergency or panic condition messages
LOG_ALERT	1	Conditions that should be corrected immediately
LOG_CRIT	2	Critical conditions
LOG_ERR	3	Errors
LOG_WARNING	4	Warnings

LOG_NOTICE	5	Not error conditions, but may require special handling
LOG_INFO	6	Informational messages
LOG_DEBUG	7	Debugging messages

Syslog messages generated by various devices can be logged locally or redirected to a log file or syslog management server. A syslog management server can be used to collect all syslog information that is deemed critical as part of the corporate network for auditing purposes.

Intrusion Detection

Intrusion detection refers to the real-time monitoring of network activity and the analyzing of data for potential vulnerabilities and attacks in progress. Internal, authorized users conducting unauthorized activity on the network---such as trying to transmit confidential documents over the Internet or illegally modifying network access privileges---can be detected in real time and stopped immediately. An external intruder trying to break into the network can be handled in the same manner.

Real-time capability (as opposed to a periodic review of log files) can significantly reduce potential damage and recovery costs of an attack by eliminating the intruder from the network.

As mentioned in Chapter 7, "Incident Handling," a good intrusion system should have the following characteristics:

- It must run continually without human supervision. The system must be reliable enough to allow it to run in the background of the system being observed.
- It must be fault tolerant in the sense that it must survive a system crash and not require its knowledge-base to be rebuilt at restart.
- It must resist subversion. The system can monitor itself to ensure that it has not been subverted.
- It must impose minimal overhead on the system. A system that slows a computer to a crawl will simply not be used.
- It must observe deviations from normal behavior and immediately alert someone in the event of abnormal behavior.
- It must cope with changing system behavior over time as new applications are being added.

The ability to write customized detection rules for proprietary purposes is also of interest. You may want to write customized detection rules to prevent a document labeled "confidential" from being emailed outside the network or to address vulnerabilities for custom or legacy systems. Such customization allows the system to be modified for use in almost any environment, even if those uses are not common enough to be included as standard features of a commercial product.

Cisco Auditing Products

A number of products are available that can be used to model and simulate network infrastructures and to provide monitoring capability and intrusion detection. The products available through Cisco Systems include the following:

- Netsys
- NetSonar
- Cisco Resource Manager
- Cisco Enterprise Accounting for NetFlow
- NetRanger

Implementation Example

This section shows configurations for the firewall, routers, and switches shown in Figure 8-1. These configurations show the commands that should be used for most Cisco infrastructure equipment to ensure security in the devices and the network infrastructure itself. Some features are shown that have not been discussed in detail in this chapter; disable them if they are not used because they can cause some security risks. The authentication method for device access is TACACS+ wherever available to take advantage of a single centralized authentication database. All comments in the configuration files are preceded by a ! symbol.

Router configuration:

hostname Router

! ensure all vty login, line and username password are encrypted

! with minimal encryption (7) unless configured as a secret

! which uses MD5 encryption

service password-encryption

! configure enable password and enable secret - the enable secret takes

! precedence

enable password 7 047E050200335C465817

enable secret 5 1dLOD$QR.onv68q3326pzM.Zexj1

! finger is often used to get user information - this should be disabled

no service finger

```
no service pad
! disables access to minor TCP services such as echo,
! chargen, discard and daytime
no service tcp-small-servers
! disables access to minor UDP services such as echo,
! chargen and discard
no service udp-small-servers
!
no ip bootp server
! prevents client applications from using source routes
no ip source-route
! configure TACACS+ authentication as default - for users logging in as
! staff, there is a local database authentication in the event that the
! TACACS+ server is unavailable
aaa new-model
aaa authentication login default tacacs+
aaa authentication login staff tacacs+ local
! authorize running exec shell when authenticated - if TACACS+
! server is not available, commands associated with privilege
! levels 0 and 1 don't require authentication commands associated
! with privilege level 15 require local authentication
aaa authorization exec tacacs+ local
aaa authorization commands 0 tacacs+ none
```

```
aaa authorization commands 1 tacacs+ none

aaa authorization commands 15 tacacs+ local

! interim accounting records will be sent every time there is

! new information to report

! accounting for all exec terminal sessions

aaa accounting update newinfo

aaa accounting exec start-stop tacacs+

! set local database authentication

username staff password 7 082C495C0012001E010F02

!

interface ethernet 0/0

! The router uses proxy ARP (defined in RFC 1027) to help hosts with

! no knowledge of routing determine the MAC addresses of hosts on other

! networks or subnets. This feature can cause a potential security

! hole and should be disabled

no ip proxy arp

! disable the forwarding of directed broadcasts to avoid

! unnecessary denial-of-service attacks

no ip directed broadcast

! disable the Cisco Discovery Protocol (CDP) for this interface.

! CDP could provide sensitive information such as configuration

! and routing tables to a potential attacker

no cdp enable
```

```
! filter such that only devices on this network have SNMP access
access-list 6 ip permit 144.254.9.0 0.0.0.255
! configure TACACS+ server and encryption key
tacacs-server host 144.254.5.9
tacacs-server key thisisakey
! SNMP access is read-only and can only be accessed by devices
! associated with access-list 6
snmp-server community public RO 6
! physical console access accessible via staff login and
! appropriate local password - the session times out after
! 2 minutes and 30 seconds of idle time
line con 0
exec-timeout 2 30
login authentication staff
! no login prompt and no input access allowed through auxiliary port
line aux 0
no exec
transport input none
! telnet access requires default authentication (TACACS+) and upon
! successful authentication commands associated with privilege
! level 15 are accessible. The session times out after 2 minutes
! and 30 seconds of inactivity
line vty 0 4
```

exec-timeout 2 30

login authentication default

privilege level 15

! turn on syslog and define console information to be logged

service timestamps log datetime localtime show-timezone

logging on

logging 144.254.5.5

logging console information

Switch configuration:

hostname Switch

! define Telnet and console authentication to be via TACACS+

set authentication login tacacs enable

set authentication enable tacacs enable

! define TACACS+ server and encryption key

set tacacs key secretkey

set tacacs server 144.254.5.9

! define syslog logging server and enable system logging messages

! to the current login session

set logging server 144.254.5.5

set logging server enable

set logging session enable

PIX firewall configuration:

! define enable password and Telnet password

enable password BjeuCKspwqCc94Ss encrypted

passwd nU3DFZzS7jF1jYc5 encrypted

! define TACACS+ server and encryption key

tacacs-server host 144.254.5.9 <key>

!

no snmp-server location

no snmp-server contact

! allow only these hosts to Telnet into the PIX

telnet 144.254.7.10 255.255.255.255

! define syslog messages to be logged and the syslog host

syslog output 23.4

syslog host 144.254.5.5

Summary

This chapter explained what you should consider to secure your networking infrastructure. It is important to control all device access---both physical and logical---to ensure that no one can tamper with the network by reconfiguring the devices. General concepts and specific features used in Cisco devices were shown to incorporate additional elements of a security architecture, including integrity, confidentiality, availability, and audit. You must use all these concepts together to obtain the most effective security controls for your network infrastructure.

continues

continues

Posted: Wed Jun 14 11:37:13 PDT 2000

Table of Contents

Securing Internet Access

Securing Internet Access

This chapter examines how to secure Internet access to the corporate network. This is accomplished using some type of firewall functionality. Firewalls have become an integral component of perimeter network access such as the boundary between the trusted corporate network and the less-trusted Internet. On this perimeter, traffic can be analyzed and controlled according to parameters such as specific applications, addresses, and users, for both incoming traffic from remote users and outgoing traffic to the Internet.

Note Constructing a firewall policy for your corporate environment was discussed in Chapter 6, "Design and Implementation of the Corporate Security Policy." If you are new to firewalls, turn to Appendix A, "Sources of Technical Information," and read the books listed under "Firewall Books" to get a good understanding of firewalls and their function.

A firewall device should be as impenetrable as possible; therefore, it should be one of the most secure devices in your infrastructure. In this chapter, we'll look at sample firewall design implementations to control Internet access and will refer to features specific to equipment provided by Cisco Systems. The chapter explains how to configure Cisco IOS devices and the Cisco PIX firewall to provide necessary security controls for Internet access. Many of the functions shown can also be used from other products if they are available. All the controls described in Chapter 8, "Securing the Corporate Network Infrastructure," should be used in these devices to provide appropriate security controls.

Internet Access Architecture

When the decision is made to connect the corporate network to the Internet, it is important to recognize the additional security exposures. In most cases, you make a decision of how open an environment you can tolerate. In a very open environment, you may impose limited restrictions on access; in a more secure environment, you may impose more stringent access controls for traffic entering or leaving the main corporate network.

There are many variations on how to design access to the Internet. A common scenario is to construct a firewall between the internal corporate network and the external Internet connection (see Figure 9-1).

Figure 9-1: Internet Access with a Firewall

The firewall can be a single device, such as a screening router with limited firewall capabilities. Often, the firewall has at least three interfaces. One of these interfaces is used as a perimeter network to isolate services (such as e-mail, FTP, DNS, and HTTP) offered to Internet users. Internet connections may be restricted solely to these services. This may be a sufficient model for a small corporation. However, the downfall is that if this single device is compromised, the entire network is open to exposure.

Another scenario, used most often in large-traffic environments, uses an exterior screening router along with a more robust firewall (see Figure 9-2).

Figure 9-2: Internet Access with a Screening Router and a Firewall

This second model is much more secure because it offers multiple levels of security to the corporation. The exterior screening router acts as a first-level filter to permit or deny traffic coming in from the Internet to the internal campus. It validates most incoming traffic before passing it on to the firewall. The firewall then provides the more CPU-intensive function of packet-by-packet inspection. In this scenario, it is also effective to include an *active audit device* that includes network traffic monitoring and intrusion detection on the network segment connecting the firewall to the exterior router. This device can verify adherence to the corporate security policy and can pin-point and isolate any attacks from the Internet to the corporate network---or any attacks instigated from your internal network out to the Internet.

Note Intrusion detection and active audit capabilities should be incorporated at network perimeter points to provide added security measures and to verify proper traffic behavior. A combination of intrusion detection, active audit, and a firewall at the network perimeter is the best defense against most known attacks.

External Screening Router Architecture

If your corporate network is small, the screening router model may be a sufficient solution to providing secure access to the Internet. It is possible that the security measures used will not always catch spoofed traffic, but at least it should provide a reasonable level of a basic buffer from the Internet.

Note The screening router solution can also be used in larger networks to define a logical separation internally between some sensitive areas of your network---for example, using a firewall between the finance building and the rest of a large campus, or using firewalls at all network perimeter points (including dial-in points and branch office connections).

Most screening routers use filtering capabilities to act as a firewall. How filters are created and to what extent they look at traffic is largely vendor dependent. The following sections examine how Cisco IOS routers provide filtering; most other vendors' devices have similar capabilities.

Cisco IOS Filters

The Cisco IOS software has an extended filtering capability to permit or deny specific traffic from entering or leaving the corporate network. These filters are called *access lists*.

Access lists filter network traffic by controlling whether routed packets are forwarded or blocked at the router's interfaces. Your router examines each packet to determine whether to forward or drop the packet, based on the criteria specified within the access lists. Access list criteria can include the source address of the traffic, the destination address of the traffic, the upper-layer protocol, or other information.

Note Sophisticated users can sometimes successfully evade or fool basic access lists because no host-to-host authentication is required.

If the access list is inbound, when the router receives a packet, the Cisco IOS software checks the access list's criteria statements for a match. If the packet is permitted, the software continues to process the packet. If the packet is denied, the software discards the packet.

If the access list is outbound, after receiving and routing a packet to the outbound interface, the software checks the access list's criteria statements for a match. If the packet is permitted, the software transmits the packet. If the packet is denied, the software discards the packet.

Cisco IOS Release 11.1 and later releases introduced substantial changes to IP access lists. These extensions are backward compatible; migrating from a release earlier than Release 11.1 to the current image will convert your access lists automatically. However, previous releases are not upwardly compatible with these changes. Thus, if you save an access list with the current image and then use older software, the resulting access list may not be interpreted correctly. This error can cause severe security problems. Save your old configuration file before booting Release 11.1 or later images.

The access lists can be specified for a number of different protocols, as shown here:

Router(config)#**access-list ?**

<1-99> IP standard access list

<100-199> IP extended access list

<200-299> Protocol type-code access list

<300-399> DECnet access list

<600-699> Appletalk access list

<700-799> 48-bit MAC address access list

<800-899> IPX standard access list

<900-999> IPX extended access list

<1000-1099> IPX SAP access list

<1100-1199> Extended 48-bit MAC address access list

<1200-1299> IPX summary address access list

Because we deal mainly with the IP protocol for Internet access, we will restrict the discussion to IP standard and IP extended access lists. For details on other protocols, refer to the Cisco online documentation.

Standard IP Access Lists

Standard IP access lists use the source IP addresses for matching operations. The configuration command takes the following syntax:

access-list *access-list-number* {deny | permit} source [*source-wildcard*]

Note The abbreviation any can be used to specify a source and source mask of 0.0.0.0255.255.255.255.

The following is an example in which we create a standard access list and apply it to the incoming Internet traffic interface. The access list denies all inbound traffic from the Internet that contains a source address from known reserved RFC addresses and permits any other traffic from the Internet to the corporate campus.

access-list 9 deny 127.0.0.0 0.255.255.255

access-list 9 deny 10.0.0.0 0.255.255.255

access-list 9 deny 172.16.0.0 0.240.255.255

access-list 9 deny 192.168.0.0 0.0.255.255

access-list 9 permit any

!

! apply the access-list 9 to the incoming Internet interface

interface Serial 0/0

description to the Internet

ip address 161.71.73.33 255.255.255.248

ip access-list 9 in

! outgoing

interface Ethernet 1/0

description to the Corporate Network

ip address 144.254.1.1 255.255.255.0

Extended Access Lists

Extended IP access lists use source and destination addresses for matching operations; they use optional protocol-type information for finer granularity of control.

The following command defines an extended IP access list number and its access conditions:

access-list *access-list-number* **{deny | permit} protocol source** *source-wildcard*

destination *destination-wildcard* **[operator] [operand][precedence** *precedence***]**

[tos *tos*] [established] [log]

Note The abbreviation host can be used for a specific source and for a specific destination without having to include the source wildcard or the destination wildcard.

For IP extended access lists, there are a number of well-known protocols you can define:

Router(config)#**access-list 101 permit ?**

<0-255> An IP protocol number

eigrp Cisco's Enhanced IGRP routing protocol

gre Cisco's GRE tunneling

icmp Internet Control Message Protocol

igmp Internet Gateway Message Protocol

igrp Cisco's IGRP routing protocol

ip Any Internet protocol

ipinip IP in IP tunneling

nos KA9Q NOS-compatible IP over IP tunneling

ospf OSPF routing protocol

tcp Transmission Control Protocol

udp User Datagram Protocol

The most common protocols to filter are the TCP and UDP protocols. For the TCP protocol, the following parameters (operators) can be filtered on:

Router(config)#**access-list 101 permit tcp any any ?**

eq Match only packets on a given port number established

established Match established connections

gt Match only packets with a greater port number

log Log matches against this entry

lt Match only packets with a lower port number

neq Match only packets not on a given port number

precedence Match packets with a given precedence value

range Match only packets in the given range of port numbers

tos Match packets with the given TOS value

Here is a list of the more commonly used TCP port numbers (operands):

Router(config)#**access-list 101 permit tcp any any eq ?**

<0-65535> Port number

bgp Border Gateway Protocol (179)

chargen Character generator (19)

cmd Remote commands (rcmd, 514)

daytime Daytime (13)

discard Discard (9)

domain Domain Name Service (53)

echo Echo (7)

exec Exec (rsh, 512)

finger Finger (79)

ftp File Transfer Protocol (21)

ftp-data FTP data connections (used infrequently, 20)

gopher Gopher (70)

hostname NIC hostname server (101)

ident Ident Protocol (113)

irc Internet Relay Chat (194)

klogin Kerberos login (543)

kshell Kerberos shell (544)

login Login (rlogin, 513)

lpd Printer service (515)

nntp Network News Transport Protocol (119)

pop2 Post Office Protocol v2 (109)

pop3 Post Office Protocol v3 (110)

smtp Simple Mail Transport Protocol (25)

sunrpc Sun Remote Procedure Call (111)

syslog Syslog (514)

tacacs TAC Access Control System (49)

talk Talk (517)

telnet Telnet (23)

time Time (37)

uucp UNIX-to-UNIX Copy Program (540)

whois Nicname (43)

www World Wide Web (HTTP, 80)

For UDP, here is a list of the more commonly used port numbers:

Router(config)#**access-list 101 permit udp any any eq ?**

<0-65535> Port number

biff Biff (mail notification, comsat, 512)

bootpc Bootstrap Protocol (BOOTP) client (68)

bootps Bootstrap Protocol (BOOTP) server (67)

discard Discard (9)

dnsix DNSIX security protocol auditing (195)

domain Domain Name Service (DNS, 53)

echo Echo (7)

mobile-ip Mobile IP registration (434)

nameserver IEN116 name service (obsolete, 42)

netbios-dgm NetBios datagram service (138)

netbios-ns NetBios name service (137)

ntp Network Time Protocol (123)

rip Routing Information Protocol (router, in.routed, 520)

snmp Simple Network Management Protocol (161)

snmptrap SNMP Traps (162)

sunrpc Sun Remote Procedure Call (111)

syslog System Logger (514)

tacacs TAC Access Control System (49)

talk Talk (517)

tftp Trivial File Transfer Protocol (69)

time Time (37)

who Who service (rwho, 513)

xdmcp X Display Manager Control Protocol (177)

Note When dealing with TCP or UDP port numbers, remember that these commonly known port numbers are always used as the *destination* port number. The *source* port number is more or less arbitrarily picked by the originating host from the range of numbers 0 to 65535.

There are a few things to keep in mind when configuring these access lists on Cisco IOS devices:

- By default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.
- After the access list is created on the router, any subsequent additions (possibly entered from the terminal) are placed at the end of the list. In other words, you cannot selectively add or remove access list command lines from a specific access list.
- The order of access lists is important. The entries are searched in sequential order; the first match is the one acted on.

Note It is recommended that you create your access lists on a TFTP server and then download the access lists to your router. This approach can considerably simplify maintenance of your access lists because the order of access list criteria statements is important and because you cannot reorder or delete criteria statements on your router. The TFTP server should be well protected from both read and write access to ensure that only authorized personnel have access to the files.

An example of an extended IP access list follows. It meets the following criteria:

- It allows all incoming TCP traffic if the session was initiated within the internal corporate network.
- It allows FTP control and FTP data traffic to the FTP server with the address 144.254.1.4.
- It allows HTTP traffic to the Web server with the address 144.254.1.3.
- It denies all other traffic from entering the corporate network.
- It logs all access list violations.

```
access-list 101 permit tcp any any established

access-list 101 permit tcp any host 144.254.1.4 eq ftp

access-list 101 permit tcp any host 144.254.1.4 eq ftp-data

access-list 101 permit tcp any host 144.254.1.3 eq www

access-list 101 deny ip any any log

!

interface Serial 0/0

description to the Internet

ip address 161.71.73.33 255.255.255.248

ip access-list 101 in
```

The order in which you create your access list entries is important in Cisco IOS devices. When the router is deciding whether to forward or block a packet, the Cisco IOS software tests the packet against each criteria statement in the order the statements were created. After a match is found, no more criteria statements are checked. From a performance standpoint, if you have the need for very long lists (for example, more than 100 entries), you should place the more-common scenarios at the beginning of the list. For example, if most of your incoming traffic is Web related, the criteria for Web traffic should be placed at the top of the list.

Note The established keyword can be used only for the TCP upper-layer protocol filters. The expectation is that a session has been started from an internal source to an external source and this permitted traffic is the reply; therefore, it is an "established" session. The manner in which the established keyword filters TCP packets is based on whether the ACK or RST bit is set. (Set ACK and RST bits indicate that the packet is not the first in the session and, therefore, that the packet belongs to an established session.) Reflexive access lists provide a more robust session-filtering mechanism and is described later in this chapter.

Named Access Lists

Named access lists were introduced in Cisco IOS Release 11.0. You can identify IP access lists with an alphanumeric string (a name) rather than a number (1 to 199). Named access lists allow you to configure

more than 99 standard IP (and 100 extended IP) access lists in a router. Currently, only packet and route filters can use a named list.

The advantage of using a named access list is that you can selectively remove entries. However, you still cannot selectively add access list command lines to a specific access list---subsequent additions are still placed at the end of the list.

Consider the following before configuring named access lists:

- Access lists specified by name are not compatible with older releases.
- Not all access lists that accept a number will accept a name. Access lists for packet filters and route filters on interfaces can use a name.
- A standard access list and an extended access list cannot have the same name.

An example of a named access list is shown in the next section (reflexive access lists *require* the use of extended named access lists).

Reflexive Access Lists

Reflexive access lists were introduced in Cisco IOS Release 11.3. They allow IP packets to be filtered based on upper-layer session information.

A common requirement for filtering is to permit IP traffic for sessions originating from within your network but to deny IP traffic for sessions originating from outside your network. Using basic extended access lists, you can approximate session filtering by using the established keyword with the permit command. The established keyword filters TCP packets based on whether the ACK or RST bits are set. This method of using the established keyword is available only for the TCP upper-layer protocol. For the other upper-layer protocols (such as UDP, ICMP, and so forth), you would have to either permit all incoming traffic or define all possible permissible source/destination host/port address pairs for each protocol.

Reflexive access lists are much more suitable for true session filtering. After it is configured, a reflexive access list is triggered when a new IP upper-layer session (such as TCP or UDP) is initiated from inside your network, with a packet traveling to the external network. When triggered, the reflexive access list generates a new, temporary entry. This entry permits traffic to enter your network if the traffic is part of the session, but will not permit traffic to enter your network if the traffic is not part of the session. The filter criterion is based on the ACK and RST bits as well as the source and destination addresses and port numbers. Session filtering uses temporary filters that are removed when a session is over, limiting the hacker's attack opportunity to a smaller time frame.

Temporary reflexive access list entries are removed at the end of the session. For TCP sessions, the entry is removed 5 seconds after two set FIN bits are detected, or immediately after matching a TCP packet with the RST bit set. (Two set FIN bits in a session indicate that the session is about to end; the 5-second window allows the session to close gracefully. A set RST bit indicates an abrupt session close.) Alternatively, the temporary entry is removed after no packets of the session have been detected for a configurable length of time (called the *timeout period*).

For UDP and other protocols, the end of the session is determined differently than it is for TCP. Because other protocols are considered to be connectionless (sessionless) services, they have no session tracking

information embedded in their packets. Therefore, the end of a session is considered to be when no packets of the session have been detected for a configurable length of time (the timeout period).

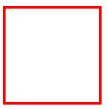
There are two restrictions on using the reflexive access list feature:

- Reflexive access lists can be defined with extended named IP access lists only. You cannot define reflexive access lists with numbered or standard named IP access lists or with other protocol access lists.
- Reflexive access lists do not work with some applications that use port numbers that change during a session. If the port numbers for a return packet are different than those of the originating packet, the return packet will be denied, even if the packet is actually part of the same session.

The TCP-based application, FTP, is an example of an application with changing port numbers. With reflexive access lists, if you start an FTP request from within your network, the request will not complete. Instead, you must use passive FTP when originating requests from within your network.

Figure 9-3 shows an example of how to use reflexive access lists. Notice that the Ethernet interface connects to the internal corporate networks, and that the serial interface connects to the Internet. The configuration example permits both inbound and outbound TCP traffic on the serial interface---but only if the first packet in a given session originated from inside your corporate network.

Figure 9-3: Reflexive Access Lists



This is what the example configuration looks like:

! Define the interface where the session-filtering configuration is

! to be applied and apply access lists to the interface, for inbound

! traffic and for outbound traffic.

interface Serial 1

description Access to the Internet

ip access-group inboundfilters in

ip access-group outboundfilters out

! Define the global idle timeout value for all reflexive access lists.

```
! If for 120 seconds there is no TCP traffic that is part of an
! established session, the corresponding reflexive access list
! entry will be removed.
ip reflexive-list timeout 120
! Define the outbound access list. This is the access list that
! evaluates all outbound traffic on interface Serial 1.
ip access-list extended outboundfilters
! Define the reflexive access list tcptraffic. This entry permits all
! outbound TCP traffic and creates a new access list named tcptraffic.
permit tcp any any reflect tcptraffic
! Define the inbound access list. This is the access list
! that evaluates all inbound traffic on interface Serial 1.
ip access-list extended inboundfilters
! Define the inbound access list entries. Permit BGP and EIGRP
! but deny ICMP. The last entry points to the reflexive access list.
! If a packet does not match the first three entries, the packet will
! be evaluated against all the entries in the reflexive access list
! named tcptraffic.
permit bgp any any
permit eigrp any any
deny icmp any any
evaluate tcptraffic
!
```

Advanced Firewall Architecture

Although a screening router is a good first step at providing Internet access security, a more secure solution relies on a more robust firewall architecture. Typically, this is accomplished with both a screening router and more intense firewall capabilities. In addition to primitive filtering capabilities, a firewall typically has the capability to provide:

- Advanced packet-by-packet inspection
- Application content filtering
- Application authentication/authorization
- Encryption technology
- Network Address Translation (NAT)

The traffic-filtering capabilities must incorporate state information and often must also be able to filter on application content. E-mail virus scanning, Java applet filtering, and URL logging or blocking are some of the commonly implemented advanced functions in a firewall. Sometimes these application-specific functions are offloaded to separate devices to save CPU processing cycles on the firewall device itself.

Packet authentication and confidentiality using encryption is becoming a strong requirement. Implementing this functionality in a multivendor interoperable way has become much easier with the emergence of IPsec products. IPsec authentication and confidentiality capabilities can be applied to many Internet access architectures to provide for authenticated, confidential traffic flow.

NAT is also commonly used but, as mentioned in Chapter 5, "Considerations for a Site Security Policy," a legitimate NIC assigned address should be used whenever possible to avoid any application or feature restrictions in the future. This is strictly the author's opinion and mileage may vary depending on specific requirements.

Advanced Packet Session Filtering

A robust firewall must have the capability to do packet-by-packet inspection and filtering on specific packet session information. The firewall should inspect traffic that travels through it to discover and manage state information for TCP and UDP sessions. For many corporate environments, FTP, Telnet, HTTP traffic, Java applets, e-mail, DNS, and some popular voice and video applications must be supported. Controls must be in place to ensure as best as possible that any such traffic is valid traffic.

Most advanced session filters keep information relating to the following questions:

- How long ago was the last packet in this session transmitted?
- Are the sequence/acknowledgment numbers climbing as expected?
- Was the session initiated from the inside or outside?
- Is the session still open or has it been closed?
- What port or ports are the return data channels using?

TCP Protocol Traffic

For IP traffic using the TCP protocol, advanced packet-session filtering inspects all IP and TCP headers in every packet based on a combination of the following fields:

- IP destination address
- IP source address
- IP protocol field
- TCP source port
- TCP destination port
- TCP flags field:
  - SYN alone for a request to open a connection
  - SYN/ACK for a connection confirmation
  - ACK for a session in progress
  - FIN for session termination

These fields allow the monitoring of connection state information and provide a reasonable amount of certainty about when valid connections are in progress.

UDP Protocol Traffic

For IP traffic using the UDP protocol, advanced packet-session filtering inspects all IP and UDP headers in every packet based on a combination of the following fields:

- IP destination address
- IP source address
- IP protocol field
- UDP source port
- UDP destination port

Because UDP is a connectionless service, there are no actual UDP "sessions," *per se*. Most systems approximate sessions by examining UDP packet information and determining whether the packet is similar to other UDP packets recently seen.

Application Content Filtering

Application content filtering refers to examining packet contents in more detail to ensure validity of the content. Some of the more common applications whose content you may want to control or examine are described in the following sections.

World Wide Web

The World Wide Web (WWW) has played a large role in making the Internet a place to conduct business. Chapter 2, "Security Technologies," described technologies, such as S-HTTP and SSL, that can be used to secure Web transactions. However, program languages, such as Java and JavaScript, that are

used to write interactive programs for Web applications remain a security issue. Some corporations also want to restrict specific URL sites from their employees because of the sites' possibly illegal content.

Java Applets

When WWW users download a Web page with embedded Java or JavaScript programs (called *applets*), there is no control over whether it has approved content or if it contains a virus or malicious program. You must prevent users from inadvertently downloading destructive Java applets into your network. To protect against this risk, you could require all users to disable Java in their browser. If this is not an agreeable solution, you can use methods to filter Java applets at the firewall, allowing users to download only those applets residing within the firewall as well as trusted applets from outside the firewall.

Many firewalls do not detect or block encapsulated Java applets. Therefore, Java applets that are wrapped or encapsulated (for example, those in .zip or .jar format) are usually not blocked at the firewall. In addition, some firewalls may not detect or block applets loaded using FTP, Gopher, or HTTP on a nonstandard port.

URL Filtering/Blocking

Some corporations may want to restrict access to certain URLs because of the sites' inappropriate content (for example, pornographic material). By providing URL filtering and blocking capabilities, the firewall can be used to restrict access to specified Web sites.

E-mail and SMTP

Simple Mail Transfer Protocol (SMTP) is used to handle e-mail exchange between mail servers on the Internet. Many firewalls have the capability to check SMTP messages for illegal commands. Any packets with illegal commands are usually dropped, and the SMTP session will hang and eventually time out.

For example, in a Cisco PIX firewall, an *illegal command* is any command except for the following legal commands:

DATA EHLO EXPN

HELO HELP MAIL

NOOP QUIT RCPT

RSET SAML SEND

SOML VRFY

Other Common Application Protocols

Many multimedia applications used for videoconferencing---for example CU-SeeMe, H.323 (for NetMeeting and ProShare), and RealAudio---use the TCP control channel to establish media channels. This control channel contains information that opens new media channels. Firewalls should have the capability to watch these control channels, to identify those ports that media channels use, and to open additional channels on a dynamic basis.

Table 9-1 lists the most common applications that should be supported in firewalls to have extensive

application control support:

Table 9-1: Common Application Protocols

Protocol	Description
Audio/Video Streaming	
CU-SeeMe by White Pine	Application that supports live audio/videoconferencing and text chat across the Internet
H.323	New standard in audio/videoconferencing
Internet Phone by Intel	Voice communication application above H.323 protocol stack
NetMeeting by Microsoft	Audio, video, and application sharing implemented over T.120 and H.323
RealAudio and RealVideo by Progressive Networks	Protocol for the transmission of high-quality streaming sound and video on the Internet
StreamWorks by Xing	Protocol for the transmission of high-quality streaming sound and video on the Internet
VDOLive by VDOnet	Application for transmitting high-quality video over the Internet
Information Seeking	
Archie	Standard tool for searching Internet file servers
Gopher	Application that provides a menu-driven front-end to Internet services
HTTP	Primary protocol used to implement the WWW
Network News Transfer Protocol (NNTP)	Protocol used to transmit and receive network news

Pointcast by Pointcast (HTTP)	Protocol for viewing news in TV-like fashion
Wide Area Information Servers (WAIS)	Tool for keyword searches (based on database content) of databases on the Internet
Security and Authentication	
HTTPS	Secured (that is, encrypted) HTTP; an implementation of SSL
TACACS+	Authentication protocol
Kerberos	Authentication service
LDAP	Standard for Internet directory services
RADIUS	A widely adopted authentication protocol
Secure ID	Protocol used by an authentication service product of Security Dynamics Technologies, Inc.
Databases	
Lotus Notes	Proprietary protocol developed by Lotus to implement its Notes application
SQL Server by Microsoft	A data replication server
SQLNet Version 1	Oracle protocol for transmission of SQL queries
SQLNet Version 2	Extension of SQLNet Version 1; adds support for port redirection
Mail	
Comsat	Mail notification protocol
Imap	Internet mail access protocol

POP Version 2	Mail protocol that allows a remote mail client to read mail from a server
POP Version 3	Modified version of POP Version 2
SMTP	Protocol widely used for the transmission of e-mail
Other TCP and UDP Services	
Chargen	TCP Chargen server sends a continual stream of characters until the client terminates the connection; UPD Chargen servers send a datagram containing a random number of characters in response to each datagram sent by a client
Daytime	Daytime server returns the date and the time of day in text format; can be run over TCP or UDP
Discard	Discard server discards whatever is sent to it by a client; can be run over TCP or UDP
DNS	Distributed database used by TCP/IP to map names to IP addresses
Finger	Protocol that provides information about users on a specified host
FTP	Protocol for copying files between hosts
Identd (auth)	Protocol used for user identification
Internet Relay Chat (IRC)	Protocol for online "chat" conversations over the Internet
NetBIOS over TCP/IP (NBT)	NetBIOS name, datagram, and session services encapsulated within TCP/IP
Network Time synchronization Protocol (NTP)	Protocol providing time across a network with precise clocks; implemented over TCP and UDP

RAS	Remote access service
Rexec	Protocol that provides remote execution facilities
Rlogin	Protocol that enables remote login between hosts
Rsh	Protocol that allows commands to be executed on another system
Simple Network Management Protocol(SNMP)	Protocol used for managing network resources
SNMP Trap	Notification by SNMP to the manager of some event of interest
Syslog	Protocol that allows a computer to send logs to another computer
Telnet (Telecommunications Network Protocol)	Remote terminal protocol enabling any terminal to log in to any host
TFTP	Small, simple FTP used primarily in booting diskless systems
Time	Service that returns the time of day as a binary number
UNIX-to-UNIX Copy Program (UUCP)	UNIX file-copying protocol
Who	Service that uses local broadcasts to provide information about who is logged on to the local network
X11	Windowing system protocol
Remote Procedure Call Services	
Lockmanager (nlockmgr)	Protocol used for the transmission of lock requests

Mountd	Protocol used for the transmission of file mount requests
Network File System (NFS)	Protocol that provides transparent file access over a network
Network Information Service (NIS)	Protocol that provides a network-accessible system-administration database, widely known as the Yellow Pages
Rstat	Protocol used to obtain performance data from a remote kernel
Rwall	Protocol used to write to all users in a network

Application Authentication/Authorization

Authentication and authorization controls for device access should be configured on all infrastructure devices, including the routers and firewalls that provide Internet access. These controls were discussed in Chapter 8, "Securing the Corporate Network Infrastructure." In addition, there may be a requirement to authenticate based on application access. For example, you may have a policy in place that requires all incoming HTTP sessions to be authenticated before they can access a specific Web server.

A sample scenario is shown in Figure 9-4.

Figure 9-4: HTTP Authentication Through a Firewall

In this figure, the following steps are carried out:

Step 1 The user from the Internet initiates an HTTP request to a specified corporate Web server.

Step 2 The firewall intercepts the connection and initiates the authentication process (shown here using TACACS+).

Step 3 If the user authenticates successfully, the firewall completes the HTTP connection to the corporate Web server.

Step 4 The firewall forwards requests and responses without further intervention.

In addition to authentication, if you are using TACACS+ or RADIUS (which also include authorization methods), you can usually configure the firewall to permit access to specific hosts or services depending on user or host identity. It is up to the corporation to determine which users can access the network, which services those individuals can use, and which hosts they can access.

Encryption

With the emergence of IPsec in many products, it is easier for corporate networks to implement authenticated and confidential data transfer sessions. Ideally, encrypted traffic (encrypted for the sake of authentication or for confidentiality) should stay encrypted from the sender to the recipient. However, if the corporate Internet access policy includes firewall operations, the firewall may have to look at the contents of the packet to carry out its function. The following three scenarios must be considered:

- *Encryption/decryption is performed from a host on the Internet to the screening router*. This approach allows the screening router to decrypt the packet and perform the filter check before passing the packet on to the firewall for more complete inspection. This scenario assumes that the internal network is secure and that no encryption is required within the corporate network.
- *Encryption/decryption is performed on the screening router*. After the initial filter check is made, the packet is sent to the firewall for inspection. The firewall then encrypts the packet to the destination host. This scenario assumes that the network between the screening router and the firewall is secure enough to handle the unencrypted traffic. This scenario is not very likely; if the firewall is to encrypt the traffic to the corporate recipient, the following scenario may be a better candidate.
- *Encryption/decryption is performed on the firewall*. This approach bypasses the screening router but a complete packet-by-packet inspection can be performed by the firewall, and then the firewall can initiate the encrypted session with the internal host. This approach ensures that, on the wire, the data is always encrypted.

Note Many current implementations that use SSL for secure Web traffic tunnel SSL through the firewall. Future secure Web transactions may use IPsec where firewalls will not be bypassed.

Figure 9-5 shows a sample scenario in which Web traffic is encrypted using IPsec but the firewall can still enforce authenticated Web transactions.

In the figure, the following steps are carried out:

Step 1 The user from the Internet initiates an encrypted HTTP request to a specified corporate Web server.

Step 2 The firewall decrypts the packet and recognizes that it is an HTTP request.

Step 3 The firewall intercepts the connection and initiates the authentication process (shown here using TACACS+).

Step 4 If the user authenticates successfully, the firewall completes the HTTP connection to the corporate Web server.

Step 5 The firewall subsequently decrypts all incoming Web traffic and forwards the unencrypted packet to the internal corporate host. (All new Web sessions are authenticated.)

Step 6 The firewall takes all responses from the Web server going to the host on the Internet and encrypts them before forwarding them to the Internet host.

Figure 9-5: Encrypted HTTP Authentication Through a Firewall

If there is a requirement to encrypt the Web traffic from the firewall to the internal Web server, that can be accomplished at the expense of more CPU cycles. Because IPsec allows for authenticated traffic, confidential traffic, or both, there are a variety of choices to be made about where to provide authentication and where to provide confidentiality measures. Figure 9-6 shows the network, and Table 9-2 lists the various combinations to be considered.

Figure 9-6: Using IPsec Through a Firewall

Table 9-2: Possible IPsec Use Through a Firewall

Traffic From/To Internet	Traffic To/From Corporate Network
Authentication and confidentiality	Authentication and confidentiality
Authentication and confidentiality	Authentication
Authentication and confidentiality	No IPsec
Authentication	Authentication
Authentication	Authentication

No IPsec	No IPsec

In all cases, Table 9-2 assumes that the firewall is an IPsec endpoint and, therefore, that it operates in IPsec transport mode. Refer to Chapter 2 for a more detailed explanation of IPsec.

Note When using IPsec, authentication is always recommended. Authentication ensures a high probability that the sender and recipient of the packet are who they should be. Confidentiality measures should be employed in addition to authentication for very sensitive data as determined by risk analysis.

Network Address Translation

Network Address Translation (NAT) is often used in environments that have private address space as opposed to a globally unique address. Private address space is not a security feature because, more often than not, the private address will be known. The Internet Assigned Numbers Authority (IANA) has reserved the following three blocks of IP address space for private networks:

10.0.0.0-10.255.255.255

172.16.0.0-172.31.255.255

192.168.0.0-192.168.255.255

The first block is a single class A network number; the second block is a set of 16 contiguous class B network numbers; and the third block is a set of 255 contiguous class C network numbers.

If a corporation decides to use private addressing, these blocks of addresses should be used---unless, of course, there is a historical reason why some other private address space is being used. The more common reason is the infamous "We'll never need to connect to the Internet!"

Public Versus Private IP Addresses

The question of whether to use private addressing is becoming a part of network design for many corporate TCP/IP application users. The Internet has grown beyond anyone's expectations. With this growth came concerns about Internet address depletion and, more importantly, address allocation procedures and their impact on the Internet routing system. There are a few reasons why private address space could and should be used (such as for environments where external connectivity may not be required or is extremely limited). Some examples are listed here:

- A large organization where only a small number of specific hosts are allowed access to the Internet.
- A large organization where outside Internet access is allowed only to a few specified hosts, such as Web servers.
- An organization that may have had to switch Internet service providers and received a new block of address space. It may be too cost prohibitive or disruptive to change to the new address space at the time.

The cost of using private Internet address space is the potentially costly effort to renumber hosts and networks between public and private. If the company has any thought about opening up to global Internet access, it is recommended that the corporation start a transition plan for renumbering to a globally unique address if it is currently using private address space. The widespread use of DHCP is making this process much easier and will avoid possible future issues with non-NATable and non-proxyable protocols.

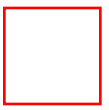
However, if you decide to use private address space, you don't have to coordinate with IANA or an Internet registry. Addresses within this private address space are unique only within your network.

Note Remember, if you need globally unique address space, you must obtain addresses from an Internet registry. This is necessary if any part of your network is connected to the Internet.

To use private address space, determine which hosts do not need to have network layer connectivity to the outside. These hosts are private hosts and will use private address space. *Private hosts* can communicate with all other hosts within the corporate network, both public and private, assuming that all addressing within the corporate network is unique. However, the private hosts cannot have IP connectivity to any host external to the corporate network without the use of NAT or proxies.

All public hosts use the globally unique address space assigned by an Internet registry. *Public hosts* can communicate with other hosts within the network and can have IP connectivity to external public hosts. Figure 9-7 shows the differences between public and private address spaces (assuming that NAT is not being used).

Figure 9-7: Private and Public Address Spaces



Because private addresses have no global meaning, routing information about private networks is not propagated on outside links, and packets with private source or destination addresses should not be forwarded across such links. Routers in networks that don't use private address space---especially those of Internet service providers---should be configured to reject (filter out) routing information about private networks.

With this scheme, many large networks need only a relatively small block of addresses from the globally unique IP address space. The Internet at large benefits through conservation of globally unique address space, and the corporate networks benefit from the increased flexibility provided by a relatively large private address space.

NAT Functionality

In its simplest configuration, NAT operates on the router or firewall connecting the inside corporate

network to the outside Internet (see Figure 9-8).

Figure 9-8: NAT on a Router or Firewall

The inside corporate network is addressed with private addresses that must be converted into legal addresses before packets can be forwarded to the outside Internet.

NAT functionality can become quite complex, depending on the applications it has to support. Here are some of the functionalities you should consider when opting for NAT for Internet access:

- *Static Address Translation.* The user can establish a one-to-one mapping between the inside local addresses and the global addresses.
- *Dynamic Source Address Translation.* The user can establish dynamic mapping between the inside local addresses and the global addresses. This is done by describing the local addresses to be translated and the pool of addresses from which to allocate global addresses, and then associating the two.
- *Dynamic Port Translation.* The user can conserve addresses in the global address pool by allowing source ports in TCP connections or UDP conversations to be translated. Different local addresses then map to the same global address, with port translation providing the necessary uniqueness. When translation is required, the new port number is picked out of the same range as the original, following the convention of Berkeley Standard Distribution (BSD):

(1-511, 512-1023, 1024-4999, 5000-65535)

- *Destination Address Rotary Translation.* A dynamic form of destination translation can be configured for some outside-to-inside traffic. After a mapping is set up, a destination address matching one of those addresses on an access list is replaced with an address from a rotary pool. Allocation is done in a round-robin basis, performed only when a new connection is opened from the outside to the inside. All non-TCP traffic is passed untranslated (unless other translations are in effect).

When using NAT, it is always important to ensure that all corporate applications that require Internet access are supported through the firewall. Typically, the following protocols and applications are supported:

- Any TCP-based protocol that does not carry the source or destination IP address in the data portion of the segment. These protocols include ICMP, HTTP, SMTP, and others.
- Any UDP-based protocol that does not carry the source or destination IP address in the data portion of the datagram. These protocols include TFTP, NTP, and others.

Note Many applications embed IP addresses into the data portion of the packet. If you are using NAT,

ensure that the firewall you are using supports the translation of IP addresses within the specific applications you are using.

Implementation Examples

Now let's consider two scenarios:

- A Cisco IOS firewall
- A PIX firewall used in conjunction with a screening Cisco IOS router

In both cases, an intrusion detection system should be used to help get more information in case an attack is attempted and to keep active audit logs of traffic coming into or leaving the corporate network.

Note The intent of the following sections is to point out practical design examples for implementing robust firewall designs. Sample scenarios are given with configuration commands that may not have been covered in detail in this text. For detailed configuration command information, refer to the Cisco product documentation.

Cisco IOS Firewall

The Cisco IOS firewall includes features that enable the required functionality of a robust firewall. The advanced traffic session filtering is performed using the Content-Based Access Control (CBAC) mechanism (explained in the next section). The sample configuration is based on the network shown in Figure 9-9.

Figure 9-9: The Sample Cisco IOS Firewall Implementation

Content-Based Access Control

Advanced packet session filtering in Cisco IOS software is supported as of Version 11.2 with the CBAC feature. By default, Cisco routers pass all routable traffic between all router interfaces. By configuring access control lists (ACLs), traffic can be permitted and denied from being processed and forwarded.

CBAC not only examines network layer and transport layer information, it also examines the application layer protocol information (such as FTP connection information) to learn about the state of the TCP or UDP session. In this way, CBAC allows support of protocols that involve multiple channels created as a result of negotiations in the control channel. Most of the multimedia protocols---as well as some other protocols (such as FTP, RPC, and SQL*Net)---involve multiple channels.

CBAC inspects traffic that travels through the firewall to discover and manage state information for TCP and UDP sessions. This state information is used to create temporary openings in the firewall's access lists to allow return traffic and additional data connections for permissible sessions (sessions that originated from within the protected internal network).

In many cases, you will configure CBAC in one direction only at a single interface. This arrangement causes traffic to be permitted back into the internal network only if the traffic is part of a permissible (valid, existing) session.

To define a set of inspection rules, use the ip inspect name global configuration command:

ip inspect name *inspection-name protocol* [timeout *seconds*]

You can configure CBAC to inspect the following types of sessions:

- All TCP sessions, regardless of the application layer protocol (sometimes called *single-channel* or *generic* TCP inspection)
- All UDP sessions, regardless of the application layer protocol (sometimes called *single-channel* or *generic* UDP inspection)

You can also configure CBAC to specifically inspect certain application layer protocols. The following application layer protocols can all be configured for CBAC:

- CU-SeeMe (only the White Pine version)
- FTP
- H.323 (such as NetMeeting, ProShare)
- Java
- UNIX R commands (such as rlogin, rexec, and rsh)
- RealAudio
- RPC (Sun RPC, not DCE RPC or Microsoft RPC)
- SMTP
- SQL*Net
- StreamWorks
- TFTP
- VDOLive

When a protocol is configured for CBAC, the protocol's traffic is inspected, state information is maintained, and, in general, packets are allowed back through the firewall only if they belong to a permissible session.

NOTE Before you configure Java inspection, you must configure a standard access list that defines "friendly" and "hostile" external sites. You configure this access list to permit traffic from friendly sites and to deny traffic from hostile sites. If you do not configure an access list but use a "placeholder" access list in the ip inspect name *inspection-name* http command, all Java applets are blocked.

Create a standard access list that permits traffic only from friendly sites and that denies traffic from hostile sites. Block all Java applets except for applets from the friendly sites defined in the access list.

Java blocking works only with standard access lists.

access-list *access-list-number* {deny | permit} source [source-wildcard]

ip inspect name *inspection-name* http [java-list access-list] [timeout seconds]

CBAC does not provide intelligent filtering for all protocols; it works only for the protocols you specify. If you don't specify a certain protocol for CBAC, the existing access lists determine how that protocol is filtered. No temporary openings are created for protocols that have not been specified for CBAC inspection.

Sample IOS Firewall Configuration

The sample firewall configuration shown in Listing 9-1 is an implementation of the following policy:

- Device access is limited to the username security_geeks
- Device authentication is performed from the local database
- Anti-spoofing filters are in place for Internet connections
- Only services initiated within the corporate environment are allowed except for FTP and WWW services to the FTP server and WWW server
- Some special debugging tools are allowed to be initiated from the Internet to the corporate network to make troubleshooting available for traveling staff
- SNMP access is allowed only from specified internal corporate SNMP servers

---

Note The filters that allow incoming IP traffic are constructed in such a way that specified services are denied unless specifically permitted. This arrangement allows for more control of traffic coming into the corporate network and is the recommended method for environments with severe security concerns.

Listing 9-1 The IOS Firewall Configuration

```
! ensure all vty login, line, and username passwords are encrypted

! with minimal encryption (7) unless configured as a secret

! which uses MD5 encryption

service password-encryption

! disables access to minor TCP services such as echo,

! chargen, discard, and daytime

no service udp-small-servers

! disable access to minor UDP services such as echo,

! chargen, and discard
```

```
no service tcp-small-servers
!
hostname imafirewall
!
enable secret 5 $1$dLOD$QR.onv68q3326pzM.Zexj1
no service finger
no service pad
no ip bootp server
!
! set local database authentication
username security_geeks password 7 082C495C0012001E010F02
ip subnet-zero
no ip source-route
!
! The following commands define the inspection rule "primaryfw",
! allowing the specified protocols to be inspected. Note that Java
! applets will be permitted according to access list 66, defined later
! in this configuration.
!
ip inspect name primaryfw cuseeme timeout 3600
ip inspect name primaryfw ftp timeout 3600
ip inspect name primaryfw http java-list 51 timeout 3600
ip inspect name primaryfw rcmd timeout 3600
```

```
ip inspect name primaryfw realaudio timeout 3600
ip inspect name primaryfw smtp timeout 3600
ip inspect name primaryfw tftp timeout 30
ip inspect name primaryfw udp timeout 15
ip inspect name primaryfw tcp timeout 3600
!
! The following interface configuration applies the "primaryfw"
! inspection rule to inbound traffic at Ethernet 0. Since this interface
! is connected to the internal corporate network side of the firewall,
! traffic entering Ethernet 0 is actually exiting the trusted internal
! network. Applying the inspection rule to this interface causes all
! traffic going from the corporate network to the
! Internet to be inspected; return traffic will only be
! permitted back through the firewall if it is part of a session that
! began from within the corporate network. Also note that access list
! 101 is applied to inbound traffic at Ethernet 0. Any traffic that
! passes the access list will be inspected by CBAC. (Traffic blocked by
! the access list will not be inspected.)
!
! Access list 108 prevents spoofing by allowing only the traffic destined to
! the corporate network to go out the Ethernet 0 interface.
```

Listing 9-1 Continued

```
!
```

interface Ethernet0

description To Corporate Network

ip address 144.254.1.1 255.255.255.0

no ip directed-broadcast

no ip proxy-arp

ip inspect primaryfw in

ip access-group 101 in

ip access-group 108 out

no ip route-cache

no cdp enable

!

interface Ethernet 1

description DMZ for ftp and www servers

ip address 144.254.2.1 255.255.255.0

no ip directed broadcast

ip access-group 102 in

no ip route-cache

no cdp enable

!

interface Serial0

description Frame Relay to Internet

no ip address

ip broadcast-address 0.0.0.0

```
encapsulation frame-relay IETF
no ip route-cache
no arp frame-relay
bandwidth 56
service-module 56k clock source line
service-module 56k network-type dds
frame-relay lmi-type ansi
!
! Note that the following interface configuration applies access list
! 111 to inbound traffic at the external serial interface. (Inbound
! traffic is entering the network.) When CBAC inspection occurs on
! traffic exiting the network, temporary openings will be added to
! access list 111 to allow returning traffic that is part of existing
! sessions.
!
interface Serial0.1 point-to-point
ip unnumbered Ethernet0
ip access-group 111 in
no ip route-cache
bandwidth 56
no cdp enable
frame-relay interface-dlci 16
!
```

ip classless

ip route 0.0.0.0 0.0.0.0 Serial0.1

! filter such that only devices on this network have SNMP access

access-list 6 permit 144.254.9.0 0.0.0.255

!

! The following access list defines "friendly" and "hostile" sites for

! Java applet blocking. Because Java applet blocking is defined in the

! inspection rule "primaryfw" and references access list 66, applets

! will be actively denied if they are from any of the "deny" addresses

! and allowed only if they are from either of the two "permit" networks.

!

access-list 66 deny 172.19.1.203

access-list 66 deny 172.19.2.147

access-list 66 permit 172.18.0.0 0.1.255.255

access-list 66 permit 192.168.1.0 0.0.0.255

access-list 66 deny any

!

! The following access list 101 is applied to interface Ethernet 0

! above. This access list permits all traffic that should be CBAC

! inspected, and also provides anti-spoofing. The access list is

! deliberately set up to deny unknown IP protocols because no such

```
! unknown protocols will be in legitimate use.
!
access-list 101 permit tcp 144.254.0.0 0.0.255.255 any
access-list 101 permit udp 144.254.0.0 0.0.255.255 any
access-list 101 permit icmp 144.254.0.0 0.0.255.255 any
access-list 101 deny ip any any
!
! Anti-spoof filters
access-list 102 permit ip 144.254.2.0 0.0.0.255 any
access-list 108 permit ip any 144.254.0.0 0.0.255.255
!
! The following access list 111 is applied to interface Serial 0.1
! above. This access list filters traffic coming in from the external
! Internet side. When CBAC inspection occurs, temporary openings will be
! added to the beginning of this access list to allow return traffic
! back into the internal network. This access list should restrict
! traffic that will be inspected by CBAC.
!
! Anti-spoofing filters
access-list 111 deny ip 127.0.0.0.0 0.255.255.255 any
access-list 111 deny ip 10.0.0.0 0.255.255.255 any
access-list 111 deny ip 172.16.0.0 0.240.255.255 any
access-list 111 deny ip 192.168.0.0 0.0.255.255 any
```

```
access-list 111 deny ip 144.254.0.0 0.0.255.255 any
! allow Internet traffic for ftp, ftp-data and www to ftp server
! and www server on dmz
access-list 111 permit ip any host 144.254.2.2 0.0.0.0 eq ftp
access-list 111 permit ip any host 144.254.2.2 0.0.0.0 eq ftp-data
access-list 111 permit ip any host 144.254.2.3 0.0.0.0 eq http
! Port 22 is SSH... encrypted, RSA-authenticated remote login. Can be
! used to get to specified corporate host(s) from the Internet
access-list 111 permit tcp any 144.254.9.0 0.0.0.255 eq 22
!
! Sometimes Enhanced IGRP is run on the Internet link. When you use
! an input access list, you have to explicitly allow control
! traffic. This could be more restrictive, but there would have to be
! entries for the Enhanced IGRP multicast as well as for the corporation's
! own unicast address.
access-list 111 permit eigrp any any
```

Listing 9-1 Continued

```
!
! These are the ICMP types actually used...
! administratively-prohibited is useful when you're trying to figure out
! why you can't reach something you think you should be able to reach.
access-list 111 permit icmp any 144.254.0.0 0.0.255.255 administratively-prohibited
!
```

```
! This allows network admins who may be traveling or otherwise coming
! in through the Internet to ping hosts at the corporate
! office:
access-list 111 permit icmp any 144.254.0.0 0.0.255.255 echo
! This allows outgoing pings
access-list 111 permit icmp any 144.254.0.0 0.0.255.255 echo-reply
!
! Path MTU discovery requires too-big messages
access-list 111 permit icmp any 144.254.0.0 0.0.255.255 packet-too-big
!
! Outgoing traceroute requires time-exceeded messages to come back
access-list 111 permit icmp any 144.254.0.0 0.0.255.255 time-exceeded
!
! Incoming traceroute
access-list 111 permit icmp any 144.254.0.0 0.0.255.255 traceroute
!
! Permits all unreachables because if you are trying to debug
! things from the corporate network, you want to see them.
! If no debugging was ever done from the network, it would be more
! appropriate to permit only port unreachables or no unreachables at
! all.
access-list 111 permit icmp any 144.254.0.0 0.0.255.255 unreachable
!
```

```
! Final deny all which logs all access list violations via syslog
access-list 111 deny ip any any log
!
no cdp run
snmp-server community public RO 6
!
line con 0
exec-timeout 2 30
login authentication security_geeks
line aux 0
no exec
transport input none
line vty 0 4
exec-timeout 2 30
login authentication security_geeks
!
service timestamps log datetime localtime show-timezone
!
logging on
logging 144.254.5.5
```

logging console information

PIX Firewall with Screening IOS Router

In this scenario, the Cisco IOS router is used as the screening router to provide basic filtering of traffic coming from the Internet. The PIX firewall provides the more robust firewall features (see Figure 9-10).

Figure 9-10: Sample Cisco PIX Firewall with Cisco IOS Screening Router

The sample configurations in Listings 9-2 and 9-3 depict the implementation of the following Internet access security policy:

- Device (screening router and firewall) access is through TACACS+ authentication and authorization
- The screening router has simple anti-spoofing filters
- Two illegal networks (192.168.0.0 and 10.0.0.0) must make use of NAT to convert to the legal address given by the ISP of 192.150.50.0
- Hosts on the 10.0.0.0 network can access everything
- Hosts on the 192.168.0.0 network can access the Internet but cannot access hosts on the 10.0.0.0 network
- Only Internet traffic from 144.254.0.0 can access the FTP server whose illegal 192.168.0.6 address must be assigned the legal address 192.150.50.6
- The FTP traffic must be authenticated using TACACS+
- All Internet Web (HTTP) traffic is directed to host 192.168.0.2 (it must be assigned the legal address of 192.150.50.9)
- All outbound Web traffic is sent to do a URL check by way of the WebSense server
- All Internet mail (SMTP) traffic is directed to host 10.0.1.99 (it must be assigned the legal address of 192.150.50.7)

Listing 9-2 Configuration of Cisco IOS Screening Router

```
! ensure all vty login, line, and username passwords are encrypted

! with minimal encryption (7) unless configured as a secret

! that uses MD5 encryption

service password-encryption

! disables access to minor TCP services such as echo,
```

! chargen, discard, and daytime

no service udp-small-servers

! disable access to minor UDP services such as echo,

! chargen, and discard

no service tcp-small-servers

!

hostname screen

!

enable secret 5 1dLOD$QR.onv68q3326pzM.Zexj1

no service finger

no service pad

no ip bootp server

!

no ip source-route

!

! configure TACACS+ authentication as default - for users logging in as

! staff, there is a local database authentication in the event that the

! TACACS+ server is unavailable

aaa new-model

aaa authentication login default tacacs+

aaa authentication login staff tacacs+ local

aaa authorization exec tacacs+ local

! interim accounting records will be sent every time there is

```
! new information to report
! accounting for all exec terminal sessions
aaa accounting update newinfo
aaa accounting exec start-stop tacacs+
!
! set local database authentication
username staff password 7 082C495C0012001E010F02
!
interface Serial 0/0
description to the Internet
ip address 161.71.73.33 255.255.255.248
ip access-group 109 in
!
interface Ethernet1/0
description To Corporate Network
ip address 192.150.50.1 255.255.255.0
no ip directed-broadcast
no ip proxy-arp
ip access-group 108 in
no ip route-cache
no cdp enable
!
access-list 108 permit ip 192.150.50.0 0.255.255.255 any
```

```
!
! anti-spoof filters
access-list 109 deny ip 127.0.0.0 0.255.255.255 any
access-list 109 deny ip 10.0.0.0 0.255.255.255 any
access-list 109 deny ip 172.16.0.0 0.240.255.255 any
access-list 109 deny ip 192.168.0.0 0.0.255.255 any
!
! Allow any tcp traffic that has been established from the corporate network
access-list 109 permit tcp any any established
!
! allow Internet traffic for ftp and ftp-data only from network 144.254.0.0
access-list 109 permit tcp 144.254.0.0 0.0.255.255 host 192.150.50.8 0.0.0.0 eq ftp
access-list 109 permit tcp 144.254.0.0 0.0.255.255 host 192.150.50.8 0.0.0.0 eq ftp-data
!
! allow Internet traffic for smtp and www server to specific servers
access-list 109 permit tcp any host 192.150.50.9 0.0.0.0 eq http
access-list 109 permit tcp any host 192.150.50.7 0.0.0.0 eq smtp
!
! Sometimes Enhanced IGRP is run on the Internet link. When you use
! an input access list, you have to explicitly allow control
! traffic. This could be more restrictive, but there would have to be
! entries for the Enhanced IGRP multicast as well as for the corporation's
! own unicast address.
```

```
access-list 109 permit eigrp any any
!
! These are the ICMP types actually used...
! administratively-prohibited is useful when you're trying to figure out
! why you can't reach something you think you should be able to reach.
access-list 109 permit icmp any 192.150.50.0 0.0.0.255 administratively-prohibited
!
! This allows network admins who may be traveling or otherwise coming
! in through the Internet to ping hosts at the corporate
! office:
access-list 109 permit icmp any 192.150.50.0 0.0.0.255 echo
!
! This allows outgoing pings
access-list 109 permit icmp any 192.150.50.0 0.0.0.255 echo-reply
!
! Path MTU discovery requires too-big messages
access-list 109 permit icmp any 192.150.50.0 0.0.0.255 packet-too-big
!
! Outgoing traceroute requires time-exceeded messages to come back
access-list 109 permit icmp any 192.150.50.0 0.0.0.255 time-exceeded
!
```

Listing 9-2 Continued

```
! Incoming traceroute
```

```
access-list 109 permit icmp any 192.150.50.0 0.0.0.255 traceroute
!
! Permits all unreachables because if you are trying to debug
! things from the corporate network, you want to see them.
! If no debugging was ever done from the network, it would be more
! appropriate to permit only port unreachables or no unreachables at
! all.
access-list 109 permit icmp any 192.150.50.0 0.0.0.255 unreachable
!
! Final deny all which logs all access list violations via syslog
access-list 109 deny ip any any log
!
no cdp run
!
tacacs-server host 192.150.50.10
tacacs-server key thisisakey
!
line con 0
exec-timeout 2 30
login authentication staff
!
line aux 0
no exec
```

transport input none

line vty 0 4

exec-timeout 2 30

login authentication default

!

service timestamps log datetime localtime show-timezone

!

logging on

logging 192.150.50.11

logging console information

Listing 9-3 Configuration of a PIX Firewall

! Sets the security levels for each interface, specifies that each

! interface uses Ethernet, and assigns IP addresses and network

! masks.

nameif ethernet0 outside security0

nameif ethernet1 inside security100

nameif ethernet2 dmz security50

interface ethernet0 auto

interface ethernet1 auto

interface ethernet2 auto

!

ip address outside 192.150.50.3 255.255.255.255

ip address inside 10.0.0.1 255.255.255.0

```
ip address dmz 192.168.0.1 255.255.255.0
!
! Specifies the host name for the PIX firewall.
hostname pixfirewall
!
! define enable password and Telnet password
enable password BjeuCKspwqCc94Ss encrypted
passwd nU3DFZzS7jF1jYc5 encrypted
!
! the following performs defined protocol security checks
fixup protocol ftp 21
fixup protocol http 80
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol smtp 25
fixup protocol sqlnet 1521
!
! Enables use of text strings instead of IP addresses. This makes your
! configuration files more readable.
names
!
! Enables paging so that if 24 lines of information
! display, PIX firewall pauses the listing and prompts you
```

```
! to continue.
pager lines 24
!
! The logging host command specifies which host runs a syslog server.
! This command also causes the PIX firewall to start sending syslog
! messages to that host. The logging trap command sets syslog to send
! all possible messages to the syslog host. The no logging console
! command disables displaying messages to the console.
logging on
logging host 10.0.1.100
logging trap 7
logging facility 20
no logging console
!
! Sets the ARP timeout to 14,400 seconds (four hours).
! Entries are kept in the ARP table for four hours before
! they are flushed. Four hours is the standard default value
! for ARP timeouts.
arp timeout 14400
!
! create a pool of addresses to be used with NAT
global (outside) 1 192.150.50.15-192.150.50.250 netmask 255.255.255.0
!
```

```
! enable IP communications between hosts on the 10.0.0.0 network and host on
! either the Internet or the 192.168.0.0 network. For communication to the
! Internet, the source IP address gets substituted with an address from the
! global pool
nat (inside) 1 10.0.0.0 255.0.0.0 0 0
!
! enables IP communications between hosts on the 192.168.0.0 network and
! the Internet. Any address starting with 192.168.0 will be substituted
! with an address from the global pool
nat (dmz) 1 192.168.0.0 255.255.255.0 0 0
!
! define static translations for the FTP server, Web server, SMTP server,
! TACACS+ server, and syslog server
static (dmz, outside) 192.150.50.6 192.168.0.6 netmask 255.255.255.255 0 0
static (dmz, outside) 192.150.50.9 192.168.0.2 netmask 255.255.255.255 0 0
static (inside, outside) 192.150.50.7 10.0.1.99 netmask 255.255.255.255 0 0
```

Listing 9-3 Continued

```
static (inside, outside) 192.150.50.10 10.0.0.100 netmask 255.255.255.255 0 0
static (inside, outside) 192.150.50.11 10.0.6.50 netmask 255.255.255.255 0 0
!
! allows packets from 10.0.0.0 network to go to the 192.168.0.0 network
statix (inside, dmz) 10.0.0.0 192.168.0.0 netmask 255.0.0.0 0 0
!
```

! enables www access to 192.168.0.2 - this command requires the static command

! above to know proper translated address

conduit permit tcp host 192.150.50.9 eq www any

!

! enables SMTP access to 10.0.1.99 - this command requires the static command

! above to know proper translated address

conduit permit tcp host 192.150.50.7 eq smtp any

!

! allow FTP access from hosts from 144.254.0.0 network

conduit permit tcp host 192.150.50.6 eq ftp 144.254.0.0 255.255.0.0

!

! Sets RIP listening attributes. The three no rip interface passive lines

! cause the PIX firewall to not listen to RIP broadcasts on each interface.

! The no rip interface default lines causes PIX firewall to not

! broadcast a default route on any interface.

No rip inside passive

No rip outside passive

No rip dmz passive

no rip inside default

no rip outside default

no rip dmz default

!

```
! Sets the outside default route to the router attached to the Internet.

route outside 0.0.0.0 0.0.0.0 192.150.50.1 1

!

! Default values for the maximum duration that PIX firewall resources

! can remain idle until being freed. To improve system performance,

! you can set the xlate and conn timers from 24 hours to 1 hour.

timeout xlate 24:00:00 conn 12:00:00 udp 0:02:00

timeout rpc 0:10:00 h323 0:05:00 uauth 0:05:00

!

! use WebSense server which has address 10.0.6.80 - all outbound URL requests are

! sent to the WebSense server

url-server (inside) host 10.0.6.80 timeout 5

filter url http 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0

!

! authenticate FTP traffic via TACACS+

tacacs--server host 10.0.6.50 thisisakey

aaa authentication ftp inbound 192.168.0.6 255.255.255.255 144.254.0.0 255.255.0.0

!

! Give Telnet access to PIX firewall console to inside hosts on 10.0.8.0 subnet.

telnet 10.0.8.0 255.255.255.0

!

! Sets the maximum transmission unit value for Ethernet access.

mtu outside 1500
```

mtu inside 1500

mtu dmz 1500

PIX Fundamentals

By default, the PIX firewall prevents all outside Internet connections from accessing inside corporate hosts or servers. As of Release 4.1, security levels are used that allocate a numeric security value (ranging from 0 to 100) to an interface. This value is configured with the following command:

nameif hardware_id if_name security_level

This is to help identify default behavior in a multi-interface firewall. The behavior is as follows:

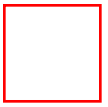
- Traffic going from an interface with a higher security level to a destination interface with a lower security level: Allow all IP-based traffic unless restricted by access lists, authentication, or authorization.
- Traffic going from an interface with a lower security level to a destination interface with a higher security level: Drop all packets unless specifically allowed by the conduit command. Further restriction is needed if authentication and authorization are used.
- Traffic going from an interface with same security level as destination interface security level: No communication between the two networks.

In addition, there are some further considerations:

- The first interface has a default security level of 100 and is named inside.
- The second interface has a security level of 0 and is named outside.
- Only one network should have a security level of 100.
- Only one network should have a security level of 0.
- Multiple perimeter networks can exist.
- If a command requires two interface names, always specify the more secure name first and the less secure name second (for example, Static (inside, outside)).

Figure 9-11 shows how different security levels can be deployed on a PIX firewall with multiple interfaces.

Figure 9-11: PIX Firewall Security Levels



The inside network has a security level of 100; the outside interface has a security level of 0. In addition, there are two separate perimeter networks: one with a security level of 60 and another with a security

level of 30.

The configuration for this example is shown here:

nameif ethernet0 outside security0

nameif ethernet1 inside security100

nameif ethernet2 staff security60

nameif ethernet3 partners security30

Controlling Inbound Access

In many corporate environments, internal users are allowed access to all Internet resources, but traffic coming in from the Internet undergoes closer scrutiny. If your security policy requires that outside users access inside hosts and servers, use the static command to specify which IP addresses are visible on the outside interfaces for users to access. The static command must be followed by the conduit command to specify which services users can access on the servers. These commands take the following form:

static [(internal_if_name, external_if_name)] global_ip local_ip

[netmask network_mask] [max_conns [em_limit]] [norandomseq]

conduit permit|deny protocol global_ip global_mask [operator port [port]]

foreign_ip foreign_mask [operator port [port]]

Together, a static and conduit statement pair create an exception to the PIX Firewall Adaptive Security mechanism by permitting connections from one firewall network interface to access hosts on another.

Controlling Outbound Access

Outbound access control is accomplished using access lists. The access lists are created with the outbound command and are based on the following information:

- IP source address
- IP destination address
- IP protocol type
- Destination port number

The use of an outbound command requires use of the apply command. The apply command lets you specify whether the ACL applies to inside users' ability to start outbound connections with the apply command's outgoing_src option, or whether the access list applies to inside users' ability to access servers on the outside network with the apply command's outgoing_dest option.

The commands take the following form:

outbound list_ID permit|deny ip_address [netmask [java|port[-port]]] [protocol]

outbound list_ID except ip_address [netmask [java|port[-port]]] [protocol]

apply [(if_name)] list_ID outgoing_src|outgoing_dest

The outbound controls are typically used for the following situations:

- Whether one or more inside users can create outbound connections (single IP address, single subnet, or all IP addresses)
- Whether inside users can access specific outside servers
- What services inside users can use for outbound connections and to access outside servers
- Whether outbound connections can execute Java applets on the inside network

The following example permits only outbound HTTP traffic from a specified source address:

outbound 1 deny 0.0.0.0 0.0.0.0 0 tcp

outbound 1 except 192.168.0.2 255.255.255.255 http

apply (inside) 1 outgoing_src

Cut-Thru-Proxy Feature

Whenever you permit outside users access to your network, you should establish a user authentication and authorization system. The PIX has a feature called *Cut-Thru-Proxy* that enables authentication based on FTP, HTTP, or Telnet traffic and subsequent authorization for any allowed application traffic. The example in Figure 9-12 shows the use of this feature.

In the figure, any outbound FTP or HTTP traffic must be successfully authenticated before the connection is established.

aaa authentication ftp, http inbound 0.0.0.0 0.0.0.0 tacacs+

aaa authorization ftp, http inbound 0.0.0.0 0.0.0.0

tacacs-server host 144.254.5.9 sharedsecret

When an outside user tries to access the corporate FTP server, the following sequence of steps occur:

Step 1 The user from the Internet initiates an HTTP or FTP request to a specified corporate server.

Step 2 The firewall intercepts the connection and initiates the authentication process (in this case, using TACACS+).

Step 3 If the user authenticates successfully, the firewall completes the HTTP or FTP connection to the specified corporate server.

Step 4 The firewall forwards requests and responses without further intervention.

Figure 9-12: The PIX Cut-Thru Proxy Feature (FTP and HTTP)

A corresponding sample user profile on the TACACS+ server that authorizes an authenticated user to use FTP on 144.254.1.4 and HTTP on 144.254.1.3 is as follows:

```
{

Profile_cycle = 11

Profile_id = 8

Password = clear "abcd"

Set Server current failed_login = 0

Service = Shell {

cmd = ftp {

permit 144.254.1.4

}

cmd = http {

permit 144.254.1.3

}

}

}
```

PIX and Mail

The mailhost command lets you create an SMTP mail host on an internal secure interface that can be accessed safely from an unprotected or less secure external interface. It is configured with the following command:

mailhost [(internal_if_name, external_if_name)] global_ip local_ip [max_conns [em_limit]]

The mailhost command imposes a security check and translation of the SMTP protocol with the PIX Firewall Adaptive Security enroute. The mailhost command limits what connections from less secure interfaces can do to the mail host itself. Only the seven SMTP commands specified in RFC 821, section 4.5.1 (HELO, MAIL, RCPT, DATA, RSET, NOOP, and QUIT) are permitted. Any other commands are treated as NOOP and discarded, with OK returned to the sender. This command creates its own implied conduit.

The mailhost command removes the need for an external mail relay in the perimeter network---also known as the *DMZ (demilitarized zone)*---that section of the network outside the firewall but "in front of" the Internet. The mailhost command is also known as the *Mail Guard feature*. However, if support for SMTP Service Extensions (RFC 1869, also known as ESMTP or HELO) is required by the customer, the Mail Guard feature should not be used. A mail host on a DMZ network with a static conduit to an internal mailer is a better solution in these cases.

Note The fixup protocol smtp command enables the Mail Guard feature in later versions of the PIX firewall.

Summary

This chapter discussed general architectures for securing your Internet access to the corporate network and showed two specific implementations that illustrate some configurations when using Cisco devices. Many variants are possible, depending on how open or restrictive your corporate environment is. It is often best to permit only IP services that are supported in the corporation and to deny all others. This arrangement allows for fairly strict control of traffic entering or leaving the corporate network through the Internet connection. For more robust Internet access monitoring and control, it is usually a good idea to include some kind of intrusion detection system and active audit component into the Internet access implementation architecture.

continues

continues

continues

continues

continues

Posted: Wed Jun 14 11:39:09 PDT 2000

Table of Contents

Securing Dial-In Access

Securing Dial-In Access

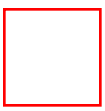
This chapter examines how to secure the dial-in connections coming into the corporate network. Often, corporate networks encompass both privately connected dial-in infrastructures (direct dial-in) and public data infrastructures (virtual dial-in) from Internet service providers (ISPs) to deliver remote access to corporate users. Dial-in access for a corporate network usually includes access between corporate branches located in different geographic regions, telecommuters, and mobile users.

The direct dial-in access can be by way of public switched telephone networks (PSTN)---for example, modem lines, frame relay, ATM, T1/T3 circuits, or ISDN. A sample dial-in environment is shown in Figure 10-1; notice that there are branch offices connected with T1 lines, mobile users dialing in with modems, and telecommuters dialing in using ISDN BRI.

Figure 10-1: A Sample Dial-In Access Environment

Another way corporations provide dial-in access is by partnering with an ISP and using the ISP's public infrastructure to provide network access. This concept of virtual dial-in is shown in Figure 10-2. For this model to work in a secure manner, tunneling technologies, such as GRE, L2F, L2TP, or IPsec, must be used to provide secure access back to the corporate network.

Figure 10-2: Dial-In Access Using the Internet



The following sections look at both the direct dial-in and the virtual dial-in scenarios and examine ways that various protocols can be applied.

Note The example configurations given are specific to Cisco Systems equipment; however, many of the functions shown can also be used with other vendors' products if they are available.

Dial-In Security Concerns

The dial-in environment has security considerations similar to those involved in securing a corporation's Internet access, discussed in the preceding chapter. It may be necessary to restrict access to certain areas of the corporate network depending on who the remote user is and from where they are trying to obtain the connection. It is usually a good idea to incorporate firewall functionality into the dial-in access perimeters and to implement some kind of auditing and intrusion detection system to keep accurate connection and traffic statistics.

Regardless of how dial-in access is provided to the corporate network (as an extension using leased lines, ISDN, or POTS networks, or as a connection from remote parts of the Internet), the main security concerns lie in the following areas:

- Identifying the caller
- Identifying the location of the caller
- Identifying the destination of the call
- Keeping track of accessed applications and data
- Keeping track of the duration of a connection
- Ensuring authenticated communication
- Ensuring private communication

Note For all equipment that is part of the dial-in infrastructure, the same security precautions should be used on the devices composing the corporate dial-in infrastructure as described in Chapter 8, "Securing the Corporate Network Infrastructure."

Authenticating Dial-In Users and Devices

A key element in allowing dial-in connectivity is to know who is accessing your corporate network by establishing an initial authentication mechanism. Authentication can be performed at the device level or at the user level.

Serial Line Internet Protocol (SLIP) and *Point-to-Point Protocol (PPP)* are two common methods of sending IP packets over standard asynchronous serial lines with minimum line speeds of 1,200 baud. Using SLIP or PPP encapsulation over asynchronous lines is an inexpensive way to connect PCs to a network. SLIP and PPP over asynchronous dial-up modems allow a home computer to be connected to a network without the cost of a leased line. Dial-up SLIP and PPP links can also be used for remote sites that need only occasional remote node or backup connectivity. Both public-domain and vendor-supported SLIP and PPP implementations are available for a variety of computer applications.

Note PPP is a newer, more robust protocol than SLIP and provides more built-in security mechanisms. PPP is much more prevalent than SLIP in modern networks.

Simple Dial-In Environments

Most serial line connections make use of PPP encapsulation, which can use a variety of authentication mechanisms to establish the identity of a peer device (refer to Chapter 2, "Security Technologies"). An example of a simple dial-in environment is shown in Figure 10-3. Notice that there are only two remote branch offices that need non-permanent low-bandwidth or variable-bandwidth connections to the corporate network (and can therefore connect to the corporate campus using ISDN). The corporate network also accommodates a few mobile users dialing in with modem connections. The configurations of these routers are shown in Listings 10-1 and 10-2.

Figure 10-3: An Example of a Small Company's Dial-In Environment

Listing 10-1 Configuration of the Corporate Access Router

```
hostname CORPORATE-NAS

!

! ensure all vty, login, line, and username passwords are encrypted

! with minimal encryption (7) unless configured as a secret

! that uses MD5 encryption

service password-encryption

! disables access to minor TCP services such as echo,

! chargen, discard, and daytime

no service tcp-small-servers

! disables access to minor UDP services such as echo,

! chargen, and discard

no service udp-small-servers

!
```

```
!define privileged access password
enable secret letmedostuff
!
! define modem usernames and passwords
username merike password ilikeAbsolut
username toivo password joekeg
username staff password iamincontrol
! define shared passwords for CHAP authentication with Branch routers
username BRANCH1 password letmein
username BRANCH2 password knockknock
!
! define ISDN switch type
isdn switch-type primary-5ess
!
! loopback interface is 'logical' subnet to which
! all dial-in users belong
interface loopback 0
ip address 144.254.200.253 255.255.255.0
!
! define local LAN interface
interface Ethernet 0/0
description Corporation LAN
ip address 144.254.166.6 255.255.255.0
```

```
!
controller T1 1/0
framing esf
clock source line primary
linecode b8zs
pri-group timeslots 1-24
!
! configure PRI
interface Serial1/0:23
description To Branch Routers
no ip address
encapsulation ppp
! route incoming ISDN modem calls to the modem module
isdn incoming-voice modem
! to use dialer profiles just in case we expand to
! another PRI in the future
dialer rotary-group 0
dialer-group 1
no fair-queue
no cdp enable
!
! set up a dialer profile
interface Dialer0
```

```
!
!users will be on subnet defined under loopback 0
ip unnumbered Loopback0
no ip mroute-cache
encapsulation ppp
!
! assign IP addresses from pool named 'dialup'
peer default ip address pool dialup
dialer in-band
!
! define which packets keep link up as defined by dialer-list
dialer-group 1
no fair-queue
no cdp enable
```

Listing 10-1 Continued

```
!
! define CHAP authentication with PAP as fallback
ppp authentication chap pap
ppp multilink
!
! modem access configuration
interface Group-Async1
!
```

```
! users will be on subnet defined by loopback0
ip unnumbered loopback0
encapsulation ppp
!
! user interactively selects to use box as a
! terminal server or a ppp router
async mode interactive
!
! assign IP address from pool named dialup
peer default ip address pool dialup
no cdp enable
!
! define CHAP authentication with PAP as fallback
ppp authentication chap pap
!
!define all async lines to belong to this interface
group-range 1 16
!
! address pool for dial-in users
ip local pool dialup 144.254.200.20 144.254.200.50
!
!configure routing
router eigrp 109
```

```
redistribute static
passive-interface Dialer0
network 144.254.0.0
no auto-summary
!
ip route 192.150.41.0 255.255.255.0 Dialer0
ip route 192.150.42.0 255.255.255.0 Dialer0
!
! permit dialing and keep line up for IP traffic
dialer-list 1 protocol ip permit
!
! physical console access accessible with any login name
! but requires correct password
line con 0
login
password igetfullcontrol
!
! modem RS-232 interface configuration
line 1 16
! use local database to authenticate users
login local
! present a login prompt but monitor packets
autoselect during-login
```

```
! if ppp packet detected, shift automatically into ppp mode
autoselect ppp
! selects state machine for CD and DTR modem signals
modem InOut
! allow connections to modem using any transport
transport input all
!
! no login prompt and no input access allowed through auxiliary port
line aux 0
no exec
transport input none
!
! virtual terminal line (Telnet) access using any login name
! but requires correct password
line vty 0 4
exec-timeout 20 0
login
password letmein

!
```

Listing 10-2 Configuration of the Branch Routers

```
hostname BRANCH1
!BRANCH2: hostname BRANCH2
!
```

```
service password-encryption
no service udp-small-servers
no service tcp-small-servers
!
!define shared passwords for CHAP authentication with Corporate NAS
username CORPORATE-NAS password letmein
! BRANCH 2: username CORPORATE-NAS password knockknock
!
isdn switch-type basic-5ess
!
interface Ethernet0
ip address 192.150.41.1 255.255.255.0
! BRANCH2: ip address 192.150.42.1 255.255.255.0
!
interface BRI0
description ISDN TO CORPORATE
ip unnumbered Ethernet0
encapsulation ppp
dialer wait-for-carrier-time 60
dialer map IP 144.254.166.6 name CORPORATE-NAS speed 56 5551234
dialer load-threshold 100 either
dialer-group 1
ppp authentication chap pap
```

```
!
ip classless
ip route 0.0.0.0 0.0.0.0 144.254.166.6
ip route 144.254.166.6 255.255.255.255 BRI0
!
dialer-list 1 list protocol ip permit
!
! physical console access accessible using any login name
! but requires correct password
```

Listing 10-2 Continued

```
line con 0
login
password igetfullcontrol
! no login prompt and no input access allowed through auxiliary port
line aux 0
no exec
transport input none
!
! virtual terminal line (Telnet) access using any login name
! but requires correct password
line vty 0 4
exec-timeout 20 0
login
```

password letmein

!

Note Because the branch routers connect to the same device at the corporate office and provide the same functionality, their configurations are nearly identical. Only one configuration is given for the branch routers; differences follow as a comment.

Complex Dial-In Environments

Configuring PAP or CHAP authentication on individual devices is manageable in simple environments. However, in corporations with hundreds or thousands of dial-in connections, a more scaleable approach must be used. To scale to a large number of users, consider incor-porating either TACACS+ or RADIUS as a better way to provide a manageable database of users. Both TACACS+ and RADIUS provide for separate authentication, authorization, and accounting facilities. When using either TACACS+ or RADIUS, the authentication mechanisms can take multiple forms, including these:

- Static password
- Changeable password
- One-time password
- NT database authentication
- UNIX /etc/password authentication
- Kerberos
- Digital certificates

TACACS+ and RADIUS Authentication

To enable TACACS+ on a Cisco Network Access Server (NAS), enter the following commands:

aaa new-model

tacacs-server host <ip address of tacacs server>

tacacs-server key <key>

The key must be specified both here and in the TACACS+ server configuration file if you want the packets to be encrypted between the server and the client (the NAS).

To enable RADIUS on a Cisco NAS, enter the following commands:

aaa new-model

radius-server host <ip address of radius server>

radius-server key <key>

The key must be specified both here and in the RADIUS server configuration file if you want the password in the packet to be encrypted between the server and the client (the NAS).

Step 1: Define a Method List

The first step in configuring either TACACS+ or RADIUS authentication is to define a method list. A *method list* is simply a list describing the authentication methods to be queried, in sequence, to authenticate a user. Method lists enable you to designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. Cisco IOS software uses the first method listed to authenticate users; if that method fails to respond, the Cisco IOS software selects the next authentication method listed in the method list. This process continues until there is successful communication with a listed authentication method---or until all methods defined are exhausted.

Note The Cisco IOS software attempts authentication with the next listed authentication method only when there is no response from the previous method. If authentication fails at any point in this cycle---meaning that the security server or local username database responds by denying the user access---the authentication process stops, and no other authentication methods are attempted.

The syntax for specifying a method list on the access server is as follows:

aaa authentication <service> {default | list-name} <method-type1>

<method-type2><method-type3> <method-type4>

(method-type 2 thru 4 are optional)

The authentication services that can be defined are listed here:

Service Description

arap Set authentication list for AppleTalk Remote Access (ARA) users' attempts to log in to the router.

nasi Set authentication list for NetWare Asynchronous Services Interface (NASI) users' attempts to log in to the router.

enable Set authentication list for enable mode.

login Set authentication lists for character mode connections.

ppp Set authentication lists for PPP connections.

You can specify up to four different authentication methods per method list for backup purposes. The methods that can be used to authenticate a user for the defined services are listed here. Although all supported methods are listed, we will concentrate specifically on TACACS+ and RADIUS as the primary authentication methods.

Service Description

enable Use the enable password for authentication.

line Use the line password for authentication.

local Use the local database for authentication.

none No authentication.

tacacs+ Use TACACS+ for authentication.

radius Use RADIUS for authentication

krb5 Use Kerberos 5 for authentication.

krb5-telnet Use Kerberos 5 Telnet authentication protocol when using Telnet to connect to the router. If selected, this keyword must be listed as the first method in the method list.

auth-guest Allow guest logins only if the user has already logged into EXEC.

guest Allow guest logins.

if-needed Do not authenticate if the user has already been authenticated on a tty line.

Not all services can use all methods. Table 10-1 shows which authentication methods can be defined for which services in Cisco IOS devices.

Table 10-1: Authentication Methods and Their Corresponding Services

Method	ARAP	NASI	Enable	Login	PPP
enable	N/A	X	X	X	N/A
line	X	X	X	X	N/A
local	X	X	N/A	X	X
none	N/A	X	X	X	X
tacacs+	X	X	X	X	X
radius	X	N/A	X	X	X
krb5	N/A	N/A	N/A	X	X
krb5-telnet	N/A	N/A	N/A	X	N/A

auth-guest	X	N/A	N/A	N/A	N/A
guest	X	N/A	N/A	N/A	N/A
if-needed	N/A	N/A	N/A	N/A	X

After the authentication method list is defined, a name is defined within the command, which is used later to link the command to an interface or line configuration. Although any name can be used, there is a reserved name known as default. The service and authentication methods defined within default are applied to any interface or line that does not have any other list linked to it.

Note To configure the Cisco IOS software to check the local user database for authentication before attempting another form of authentication, use the aaa authentication local-override command. This command is useful when you want to configure an override to the normal authentication process for certain personnel (such as system administrators).

The following examples show some typical uses of the aaa authentication command:

- The following command states that a user trying to make a character mode login to the router must be authenticated by the TACACS+ server; if that server fails to respond, use the local database instead:

aaa authentication login ADMIN tacacs+ local

> **Note** The local database is checked only if the TACACS+ server fails to respond, not if the user fails authentication with the TACACS+ server.

- The following command states that a user attempting a PPP connection to the router must authenticate with the RADIUS server; if that fails, the user must provide the enable password:

aaa authentication ppp USER radius enable

- The following command states that the default for character mode access is to use RADIUS unless otherwise stated:

aaa authentication login default radius

- The following command states that the default for packet mode access is to use TACACS+ authentication:

aaa authentication ppp default tacacs+

Step 2: Link the Method List to a Line or Interface

After a method list is created, the next step is to link the method list to a line or interface. The following examples provide some typical uses.

The first example configures TACACS+ as the security protocol to be used for PPP authen-tication using

the method list dialusers:

```
aaa new-model

! defines a method list, "dialusers", to be used on serial interfaces running PPP.

! The keyword tacacs+ means that authentication will be done through TACACS+. If ! TACACS+
returns an ERROR of some sort during authentication, the keyword local ! indicates that
authentication will be attempted using the local database on the ! network access server.

aaa authentication ppp dialusers tacacs+ local

tacacs-server host 144.254.9.5

tacacs-server key iamasecret

! select line and apply the test method list to this line

interface serial 0
```

ppp authentication chap pap dialusers

The second example configures RADIUS as the security protocol to be used for PPP authentication using the method list default:

```
aaa new-model

! defines a method list, "default," to be used on serial interfaces running PPP.

! The keyword default means that PPP authentication is applied by default to all

! interfaces. The if-needed keyword means that if the user has already

! authenticated by going through the ASCII login procedure, then PPP
! authentication is not necessary and can be skipped.

! If authentication is needed, the keyword radius means that authentication

! will be done through RADIUS. If RADIUS returns an ERROR of some sort during

! authentication, the keyword local indicates that authentication will be
! attempted using the local database on the network access server.

aaa authentication ppp default if-needed radius local

radius-server host 144.254.9.5
```

radius-server key iamasecret

! select line and apply the default method list to this line.

interface serial 0

ppp authentication default

If you do not include the enable method for system administrator logins, you will no longer be able to log in to your network access server unless you have a functioning TACACS+ server appropriately configured with usernames and passwords. The addition of the enable method ensures that you can still log in to the NAS if the NAS cannot contact a TACACS+ server. The network access server tests the enable method only if it cannot contact a TACACS+ server.

To avoid being locked out of your NAS, you can do the following:

- Set the default method to none and create a secondary list with another name that applies only to the lines or interfaces for which you want to use TACACS+:

aaa authentication login default none

aaa authentication login USERS tacacs+

line 1

login authentication USERS

- Create a special list called CONSOLE and set the authentication to none; apply that to the console port or vtys so that you can access it using the console (or using Telnet if you do not have console access):

aaa authentication login default tacacs+

aaa authentication login CONSOLE none

line con 0

login authentication CONSOLE

Authorization

Authorization is the process by which you can control what users can and cannot do. Often it is not enough to simply establish a link connection on authentication. After the device or user has been authenticated, a subsequent authorization step may be required to permit access to a specified area of the network. Many corporate environments restrict access to some company branches or limit certain users to only particular areas of the network or particular applications.

Here are some reasons to use authorization requests:

- If you chose to assign a particular IP address or an access list to a particular user or group of users
- If you choose to allow a particular user or group of users to use Telnet but not to use rlogin

- If you want a user to get his or her IP address from an address pool on the NAS
- If you want to add callback functionality for added security and accounting

Any and all of these reasons require authorization for the particular service to be configured on the NAS; you must also configure the appropriate profile in the TACACS+ or RADIUS configuration file.

TACACS+ and RADIUS Authorization

When either TACACS+ or RADIUS authorization is enabled, the NAS uses information retrieved from the user's profile (located either in the local user database or on the security server) to configure the user's session. After this is done, the user is granted access to a requested service only if the information in the user's profile allows it.

Much like configuring authentication, the first step in configuring either TACACS+ or RADIUS authorization is to define a method list. This process continues until there is successful communication with a listed authorization method, or until all defined methods are exhausted.

The syntax for specifying an authorization method list on the access server is as follows:

aaa authorization <service type> {default | list-name}

<method1> <method2><method3> <method4>

Note Authorization is bypassed for authenticated users who log in using the console line, even if authorization has been configured.

The following authorization service types are supported:

Service Description

Network Checks authorization for all network activities, including SLIP, PPP, and ARAP.

EXEC Determines whether the user is allowed to run an EXEC shell when logging into the NAS. This keyword may cause the TACACS+ or RADIUS daemon to return user profile information such as autocommand, acl, and so on.

Commands Checks authorization for all commands at the specified privilege level. Command authorization attempts authorization for all EXEC mode commands (including global configuration commands) associated with a specific privilege level. Valid levels are 0 through 15. Level 1 is normal user EXEC commands; level 15 is the privileged level.

Reverse Access Applies to Reverse Telnet sessions.

Reverse Telnet

Telnet is a standard terminal emulation protocol used for remote terminal connection. Normally, you log in to a NAS through a dial-up connection and then use Telnet to access other network devices from that NAS. There are times, however, when it is necessary to establish the Telnet connection in the opposite

direction---from inside a network to a NAS on the network periphery---to gain access to modems or other devices connected to that NAS. *Reverse Telnet* allows users with dial-out capability to Telnet to modem ports attached to a NAS.

It is important to control access to ports accessible through Reverse Telnet. Failure to do so exposes a security hole through which unauthorized users can gain free access to modems, from which they can trap and divert incoming calls or make outgoing calls to unauthorized destinations.

Authentication during Reverse Telnet is performed through the standard authenticated login procedure for Telnet. Typically, the user has to provide a username and password to establish either a Telnet or a Reverse Telnet session. Reverse Telnet authorization provides an additional (optional) level of security by requiring authorization in addition to authentication. When enabled, Reverse Telnet can use RADIUS or TACACS+ to authorize whether or not this user is allowed Reverse Telnet access to specific asynchronous ports (after the user successfully authenticates through the standard Telnet login procedure).

Reverse Telnet authorization offers the following benefits:

· An additional level of protection by ensuring that users engaged in Reverse Telnet activities are indeed authorized to access a specific asynchronous port using Reverse Telnet.

· An alternative method to using only access lists on an interface to manage Reverse Telnet authorization.

You can specify up to five separate methods to carry out the authorization for the specified service type. The supported methods are listed here:

Method Description

tacacs+ The NAS exchanges authorization information with the TACACS+ security daemon. TACACS+ authorization defines specific rights for users by associating attribute-value pairs (stored in a database on the TACACS+ security server) with the appropriate user.

If-Authenticated The user is allowed to access the requested function provided that the user has been authenticated successfully.

local The router or access server consults its local database, as defined by the username command, to authorize specific rights for users. Only a limited set of functions can be controlled from the local database.

radius The NAS requests authorization information from the RADIUS security server. RADIUS authorization defines specific rights
for users by associating attributes (stored in a database on the RADIUS server) with the appropriate user.

kerberos instance The NAS uses the instance defined by the kerberos instance map map command for authorization.

Once defined, method lists must be applied to specific lines or interfaces before any of the defined methods are performed. The only exception is the default method list (named default). If the aaa authorization command for a particular authorization type is issued without specifying a named method list, the default method list automatically applies to all interfaces or lines except those that have a named

method list explicitly defined. (A defined method list overrides the default method list.) If no default method list is defined, no authorization takes place.

Note If authorization is not explicitly configured on the access server, everything is permitted by default. However, if authorization is configured, the default behavior is to deny everything. Before configuring authorization on the access server, be sure that you have configured an authenticated user who is authorized to do everything, or you may lock yourself out of the NAS.

Now let's look at examples that show some typical uses of the aaa authorization command.

The following example shows how you can configure a NAS to restrict the commands an individual user can execute:

aaa authorization commands 1 tacacs+

When you enter this command in your NAS, you are permitted to execute only NAS commands that are explicitly permitted in the TACACS+ configuration file. Therefore, make sure that you have configured an authenticated user who is authorized to run all commands.

To require that the system administrators be authorized at level 15, enter the following command:

aaa authorization commands 15 tacacs+ if-authenticated

This command uses TACACS+ authorization level 15; if problems arise, you can switch off the TACACS+ server and authorization is then granted to anyone who is authenticated.

The next example shows the configuration on a Cisco IOS NAS for authentication and authorization services to be provided by a RADIUS server. If the RADIUS server fails to respond, the local database is queried for authentication and authorization information:

aaa new-model

! command defines a method list, staff, for login authentication

aaa authentication login staff local

! defines the authentication method list "dialup," which

! specifies that RADIUS authentication then (if the RADIUS server

! does not respond) local authentication will be used on

! serial lines using PPP

aaa authentication ppp dialup radius local

! defines the network authorization method list named

! "dialup2," which specifies that RADIUS authorization will be used

```
! on serial lines using PPP. If the RADIUS server fails
! to respond, then local network authorization will be performed.
aaa authorization network dialup2 radius local
! username and password to be used for the PPP CHAP
username staff password letmein
radius-server host 144.254.9.6
radius-server key myRaDiUSpassWoRd
interface group-async 1
group-range 1 16
encapsulation ppp
! selects CHAP as the method of PPP authentication and applies
! the "dialup" method list to the specified interfaces.
ppp authentication chap dialup
! applies the dialup2 network authorization method list to the
! specified interfaces.
ppp authorization dialup2
line 1 16
! command used to allow a PPP session to start up automatically
autoselect ppp
! command used to display the username and password prompt without
! pressing the Enter key. After the user logs in, the autoselect
! function (in this case, PPP) begins.
autoselect during-login
```

! command used to apply the staff method list for login authentication

login authentication staff

! command to configure modems attached to the selected lines to accept

! only incoming calls

modem dialin

Sample TACACS+ Database Syntax

Listing 10-3 shows the syntax used in CiscoSecure, the Cisco TACACS+ Access Control Server, for its TACACS+ database. The syntax may change as more functionality is added;
this example is given to show what you can configure on the TACACS+ server side. Most TACACS+ servers employ similar functionality and often also have a simple-to-use graphical user interface that creates the appropriate database for you.

Listing 10-3 The Syntax for the CiscoSecure Server

[unknown_user] = {

[user | group] = [<user name> | <group name>] {

password = [clear | chap | arap | pap | des] ["password"]

[from "dd mmm yy" until "dd mmm yy" | until "dd mmm yy"]

password = [skey | system | no_password] [from "dd mmm

yy" until "dd mmm yy" | until "dd mmm yy"]

password = file <"file name"> [from "dd mmm yy" until "dd

mmm yy" | until "dd mmm yy"]

privilege = [clear | des] "<password>" [0-15]

privilege = [skey] [0-15]

Listing 10-3 Continued

default service = [permit | deny]

prohibit service = <service name>

default attribute = [permit | deny]

allow <"nas name"> <"port name"> <"rem_addr">

refuse <"nas name"> <"port name"> <"rem_addr">

expires = [<"month day year"> | <"dd mmm yy">]

valid = [<"month day year"> | < "dd mmm yy">]

member = <group name>

service = shell {

default attribute = [permit | deny]

default cmd = [permit | deny]

prohibit cmd = <command>

set acl = <access-class number>

set autocmd = <"command">

set noescape = [true | false]

set nohangup = [true | false]

set priv-lvl = [0-15]

set timeout = <minutes>

set callback-dialstring = <phone number>

set callback-line = <line number>

set callback-rotary = <rotary number>

set nocallback-verify = 1

cmd = <command> {

[deny | permit] <"command arg">

default attribute = permit

}

time = [<Mo, Tu, We, Th, Fr, Sa, Su 0000 - 2359> | <Any 0000 - 2359>]

}

service = ppp {

default protocol = [permit | deny]

prohibit protocol = <protocol>

protocol = lcp {

default attribute = [permit | deny]

set callback-dialstring = <phone number>

set callback-line = <line number>

set callback-rotary = <rotary number>

set nocallback-verify = 1

time = [<Mo, Tu, We, Th, Fr, Sa, Su 0000 - 2359> | <Any 0000 - 2359>]

}

protocol = vpdn {

set tunnel-id = <NAS name>

set ip-addresses = <"x.x.x.x x.x.x.x">

}

protocol = ip {

default attribute = [permit | deny]

set addr = <ip address>

set addr-pool = <ip local pool name>

set inacl = <input access-list number>

set outacl = <output access-list number>

set route = <"destination_address mask gateway">

set routing = [true | false]

time = [<Mo, Tu, We, Th, Fr, Sa, Su 0000 - 2359> | <Any 0000 - 2359>]

}

protocol = ipx {

default attribute = [permit | deny]

set acl = <access-list number>

time = [<Mo, Tu, We, Th, Fr, Sa, Su 0000 - 2359> | <Any 0000 - 2359>]

}

protocol = atalk {

default attribute = [permit | deny]

set zonelist = <zonelist>

time = [<Mo, Tu, We, Th, Fr, Sa, Su 0000 - 2359> | <Any 0000 - 2359>]

}

}

The Lock-and-Key Feature

Lock-and-key is a traffic-filtering security feature in Cisco IOS devices that dynamically filters IP protocol traffic. It can be used to authorize temporary access to specified areas of a corporate network. Lock-and-key is configured using IP dynamic extended access lists and can be used in conjunction with other standard access lists and static extended access lists.

When triggered, lock-and-key reconfigures the interface's existing IP access list to permit designated users to reach specified areas of the network. When it is finished, lock-and-key reconfigures the interface back to its original state.

For a user to gain access to a host through a router with lock-and-key configured, the user must first Telnet to the router. When a user initiates a standard Telnet session to the router,

lock-and-key automatically attempts to authenticate the user. If the user is authenticated, he or she then gains temporary access through the router and can reach the destination host.

Currently, a user at a remote site can use WAN technology---such as Asynchronous Transfer Mode (ATM), dial-on-demand routing (DDR), Frame Relay, ISDN, PPP, or X.25---to connect to the corporate office using lock-and-key.

The following steps describe the lock-and-key access operation (see Figure 10-4):

Step 1 A user opens a Telnet session to a border (firewall) router configured for lock-and-key. The user connects using the virtual terminal port on the router.

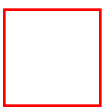
Step 2 The Cisco IOS software receives the Telnet packet, opens a Telnet session, prompts for a password, and performs a user authentication process. The user must pass authentication before access through the router is allowed. The authentication process can be done by the router or by a central access security server, such as a TACACS+ or RADIUS server.

Step 3 When the user passes authentication, he or she is logged out of the Telnet session, and the software creates a temporary entry in the dynamic access list. (Per your configuration, this temporary entry can limit the range of networks to which the user is given temporary access.)

Step 4 The user exchanges data through the router/firewall.

The software deletes the temporary access list entry when a configured timeout is reached or when the system administrator manually clears the entry. The configured timeout can either be an idle timeout or an absolute timeout.

Figure 10-4: A Lock-and-Key Operation



Note The temporary access list entry is not automatically deleted when the user terminates a session. The temporary access list entry remains until a configured timeout is reached or until the entry is cleared by the system administrator.

When lock-and-key is triggered, it creates a dynamic opening in the firewall by temporarily reconfiguring an interface to allow user access. While this opening exists, another host can spoof the authenticated user's address to gain access behind the firewall. Lock-and-key does not cause the address spoofing problem; the problem is only identified here as a concern to the user. Spoofing is a problem inherent to all access lists, and lock-and-key does not specifically address this problem.

To prevent spoofing, you can configure network data encryption as described in the last section of this chapter. Configure encryption so that traffic from the remote host is encrypted at a secured remote router

and is decrypted locally at the router interface that provides the lock-and-key service. You want to ensure that all traffic using lock-and-key is encrypted when entering the router. In this way, no hackers can spoof the source address because they are unable to duplicate the encryption or to be authenticated (a required part of the encryption setup process).

Lock-and-Key Authentication

There are three possible ways to configure an authentication query process:

- *Configure a security server.* Use a network access security server such as a TACACS+ server. This method requires additional configuration steps on the TACACS+ server but allows for stricter authentication queries and more sophisticated tracking capabilities.

Router# login tacacs

- *Configure the username command.* This method is more effective than the preceding one because authentication is determined on a user basis.

Router# username name password password

- *Configure the password and login commands.* This method is less effective than the first method because the password is configured for the port, not for the user. Therefore, any user who knows the password can authenticate successfully.

Router# password password

Router# login local

Note It is recommended that you use the TACACS+ server for your authentication query process. TACACS+ provides authentication, authorization, and accounting services. It also provides protocol support, protocol specification, and a centralized security database.

Lock-and-Key Examples

The first lock-and-key example is shown in Figure 10-5. Here we show how to configure lock-and-key access from a telecommuter to a NAS, with authentication occurring locally at the campus NAS. Lock-and-key is configured on the BRI 0 interface of the NAS.

The configuration looks as follows:

! Telecommuter who will come in using lock-and-key

username telecommuter password 7 0758364708452A

isdn switch-type basic-dms100

!

interface ethernet 0

ip address 144.254.166.6 255.255.255.0

```
interface BRI0
ip unnumbered ethernet 0
encapsulation ppp
dialer idle-timeout 3600
dialer wait-for-carrier-time 100
dialer map ip 171.73.34.33 name merike
dialer-group 1
isdn spid1 8316333715291
isdn spid2 8316339371566
ppp authentication chap
ip access-group 101 in
!
ip classless
ip route 0.0.0.0 0.0.0.0 144.254.166.6
ip route 144.254.166.6 255.255.255.255 BRI0
! allows Telnet from telecommuter to this router
access-list 101 permit tcp any host 144.254.166.6 eq telnet
! allows telecommuter to have access anywhere inside campus after Telneting
! to router and successful authentication
access-list 101 dynamic telecommuter timeout 120 permit ip any any
!
dialer-list 1 protocol ip permit
line vty 0
```

login local

autocommand access-enable timeout 5

Figure 10-5: Lock-and-Key for Telecommuter Access

The first access-list entry allows only Telnet sessions into the router. The second access-list entry is always ignored until lock-and-key is triggered.

After a user Telnets into the router, the router attempts to authenticate the user. If authentication is successful, autocommand executes and the Telnet session terminates. The autocommand command creates a temporary inbound access list entry at the Serial 0 interface, based on the second access-list entry (telecommuter). This temporary entry expires after 5 minutes, as specified by the timeout value.

The second lock-and-key example is shown in Figure 10-6. This example shows how to configure lock-and-key access for a branch router, with authentication on a TACACS+ server. Lock-and-key access is configured on the BRI 0 interface of the NAS.

Figure 10-6: Lock-and-Key for Branch Router Access

The configuration on the NAS is as follows:

aaa new-model

aaa authentication login lockkey tacacs+ enable

aaa authorization exec tacacs+

!

isdn switch-type basic-dms100

```
!
interface Ethernet0
ip address 144.254.166.6 255.255.255.0
!
interface BRI0
ip unnumbered Ethernet0
ip access-group 101 in
no ip mroute-cache
encapsulation ppp
dialer idle-timeout 300
dialer map ip 192.150.42.1 name Branchrouter 97328866
dialer-group 1
isdn spid1 8316333715291
isdn spid2 8316339371566
no fair-queue
compress stac
ppp multilink
!
router eigrp 100
network 144.254.0.0
!
ip classless
ip route 0.0.0.0 0.0.0.0 192.150.42.1
```

```
ip route 192.150.42.1 255.255.255.255 BRI0

! allows Telnet from the branch hosts to this router

access-list 101 permit tcp any host 144.254.166.6 eq telnet

!

! allows anybody inside campus to have access to the branch resources

access-list 101 permit tcp any 144.254.0.0 0.0.255.255 established

!

! allows certain hosts inside to be accessed from the branch without authentication

access-list 101 permit ip any host 144.254.120.6

access-list 101 permit ip any host 144.254.120.8

!

! allows for branch to have access anywhere inside campus after Telneting

! to router and successful authentication

access-list 101 dynamic Branch timeout 5 permit ip any any

!

tacacs-server host 144.254.5.9

tacacs-server key secretkey

!

dialer-list 1 protocol ip permit

!

line con 0

exec-timeout 2 30

password letmein
```

line vty 0 4

exec-timeout 15 0

! once user logs in, authentication is by way of tacacs+

login authentication lockkey

The configuration on TACACS+ is as follows:

user = lockkeyuser {

password = clear "secretword"

service = shell {

! following turns on the dynamic access-list

set autocmd = "access-enable"

}

}

The third lock-and-key example shows a configuration in which two users can have different lock-and-key dynamic access list configurations and different access privileges. If these two users access the device from the same interface, only the first configured dynamic ACL is activated.

interface Ethernet0/0

ip address 144.254.163.2 255.255.255.0

ip access-group 161 in

no ip directed-broadcast

no ip mroute-cache

!

interface Ethernet0/1

ip address 144.254.166.8 255.255.255.0

ip access-group 141 in

no ip directed-broadcast

no ip mroute-cache

!

access-list 141 dynamic genesis permit ip any any log

access-list 141 permit ip any host 224.0.0.10

access-list 141 permit ip any any

access-list 161 dynamic new permit tcp any any log

access-list 161 permit ip any any

Double Authentication/Authorization

When a remote user dials in to a local corporate perimeter host (a NAS or router) over PPP, CHAP or PAP can be used to authenticate the user. However, both of these authentication methods rely on a secret password (the "secret") that must be stored on the local host and either remembered by a user or saved on the remote host. If either host ever comes under the control of a network attacker, the secret password is compromised.

Consider a corporate user who often uses a laptop computer to log in to the corporate enterprise network, which uses only CHAP for authentication. If the laptop computer is ever stolen, the computer can still connect to the corporate network if the correct dial-in script is executed.

With the double authentication feature, there are two authentication/authorization stages. These two stages occur after a remote user dials in and a PPP session is initiated.

In the first stage, the user logs in using the remote host name. CHAP (or PAP) authenticates the remote host and then PPP negotiates with RADIUS or TACACS+ to authorize the remote host. In this process, the network access privileges associated with the remote host are assigned to the user.

Note You should restrict authorization at this first stage to allow only Telnet connections to the local host. This arrangement prevents an attacker from using the device authentication to access the NAS and to then Telnet from the NAS to other parts of the network.

In the second stage, the remote user must Telnet to the NAS to be authenticated. When the remote user logs in, the user must be authenticated with the specified login authentication. The user then must enter the access-profile command to be reauthorized. When this authorization is complete, the user has been double authenticated and can access the network according to per-user network privileges.

WARNING Double authentication can cause certain undesirable events if multiple hosts share a PPP connection to a NAS. If user Belvekdoir initiates a PPP session and activates double authentication at the NAS, any other user automatically has the same network privileges as Belvekdoir until Belvekdoir's PPP

session terminates. This happens because Belvekdoir's authorization profile is applied to the NAS's interface during the PPP session; any PPP traffic from other users uses the PPP session that has already been established.

Another undesirable event can occur if, in the middle of Belvekdoir's PPP session, another user, Jim, executes the access-profile command. This action results in a reauthorization; Jim's authorization profile is applied to the interface, replacing Belvekdoir's profile. This replacement can disrupt or halt Belvekdoir's PPP traffic or grant Belvekdoir additional authorization privileges he or she should not have.

This following example shows the configuration on a Cisco NAS. The first three lines configure a TACACS+ server. The next two lines configure PPP and login authentication, and the last two lines configure network and EXEC authorization. The last line is necessary only if the access-profile command will be executed as an autocommand.

aaa new-model

tacacs-server host 144.254.5.9

tacacs-server key mytacacskey

aaa authentication ppp default tacacs+

aaa authentication login default tacacs+

aaa authorization network tacacs+

aaa authorization exec tacacs+

The sample configuration in Listing 10-4 shows authentication/authorization profiles on the TACACS+ server for the remote host psycho and for three users (usernames Merike-default, Wayne-merge, and Tom-replace). The configurations for these three usernames show different configurations that correspond to the three different forms of the access-profile command. The three user configurations also show how to set up autocommand for each form of the access-profile command.

Listing 10-4 Authentication/Authorization Profiles on the TACACS+ Server

key = "mytacacskey"

default authorization = permit

#

 # This allows the remote host to be authenticated by the local host

 # during fist-stage authentication, and provides the remote host

```
# authorization profile.
#
user = psycho
{
login = cleartext "welcome"
chap = cleartext "welcome"
service = ppp protocol = lcp {
interface-config="ip unnumbered ethernet 0"
}
service = ppp protocol = ip {
# It is important to have the hash sign and some string after
# it. This indicates to the NAS that you have a per-user
# config.
inacl#3="permit tcp any 192.150.42.0 0.0.0.255 eq telnet"
inacl#4="deny icmp any any"
route#5="192.150.42.0 255.255.255.0"
route#6="192.150.41.0 255.255.255.0"
}
}
```

```
# - Without arguments access-profile removes any access-lists it can find
# in the old configuration (both per-user and per-interface), and makes sure
```

```
# that the new profile contains ONLY access-list definitions.
#
user = Merike-default
{
login = cleartext "welcome"
chap = cleartext "welcome"
service = exec
{
# this is the autocommand that executes when Merike-default logs in
autocmd = "access-profile"
}
service = ppp protocol = ip {
# Put whatever access-lists, static routes, and so on here
# If you leave this blank, the user will have NO IP
# access-lists (not even the ones installed prior to
# this)
inacl#3="permit tcp any host 144.254.166.10 eq telnet"
inacl#4="deny icmp any any"
}
}
```

```
# With the 'merge' option, all old access-lists are removed (as before),
```

```
# but then (almost) all AV pairs are uploaded and installed. This

# will allow for uploading any custom static routes, filters, and so on,

# that the user may need in his or her profile. This needs to be used with

# care, as it leaves open the possibility of conflicting configurations.

#

user = Wayne-merge

{

login = cleartext "welcome"

chap = cleartext "welcome"

service = exec

{

# this is the autocommand that executes when Wayne-merge logs in

autocmd = "access-profile merge"

}

service = ppp protocol = ip

{

# Put whatever access-lists, static routes, and so on here
```

Listing 10-4 Continued

```
# If you leave this blank, the user will have NO IP

# access-lists (not even the ones installed prior to

# this)

inacl#3="permit tcp any any"

route#2="144.254.0.0 255.255.0.0"
```

```
}

}


#- With the 'replace' option,

# ALL old configuration is removed and ALL new configuration is installed.

#

# One caveat: access-profile checks the new configuration for address-pool and

# address AV pairs. As addresses cannot be renegotiated at this point, the

# command will fail (and complain) when it encounters such an AV pair.

# Such AV pairs are considered to be "invalid" for this context.

#-------------------------------------------------------------------------

user = Tom-replace

{

login = cleartext "welcome"

chap = cleartext "welcome"

service = exec

{

# this is the autocommand that executes when Tom-replace logs in

autocmd = "access-profile replace"

}

service = ppp protocol = ip

{

# Put whatever access-lists, static routes, and so on here
```

 # If you leave this blank, the user will have NO IP

 # access-lists (not even the ones installed prior to

 # this)

inacl#3="permit tcp any any"

inacl#4="permit icmp any any"

route#2="171.71.73.0 255.255.255.0"

}

}

Automated Double Authentication

You can make the double-authentication process easier for users by implementing automated double authentication. *Automated double authentication* provides all the security benefits of double authentication, but offers a simpler, more user-friendly interface for remote users. With double authentication, a second level of user authentication is achieved when the user Telnets to the NAS or router and enters a username and password. With automated double authentication, the user does not have to Telnet to the NAS; instead, the user responds to a dialog box that requests a username and password or personal identification number (PIN).

Note To use the automated double-authentication feature, the remote user hosts must be running a companion client application. As of Cisco IOS Release 12.0, the only client application software available is the Glacier Bay application server software for PCs.

Listing 10-5 shows a complete configuration file for a Cisco IOS router with automated double authentication.

Listing 10-5 Complete Configuration File for Automated Double Authentication

hostname myrouter

!

no service password-encryption

!

! The following command enables AAA:

aaa new-model

```
! The following command enables user authentication via the TACACS+ server:
aaa authentication login default tacacs+
aaa authentication login console none
! The following command enables device authentication via the TACACS+ server:
aaa authentication ppp default tacacs+
! The following command causes the remote user's authorization profile
! to be downloaded from the TACACS+ server to the Cisco router when required:
aaa authorization exec tacacs+
! The following command causes the remote device's authorization profile
! to be downloaded from the TACACS+ server to the Cisco router when required:
aaa authorization network tacacs+
!
enable secret thisisasecret
!
ip host twiggy 192.150.42.101
ip host minky 192.150.42.103
ip host itchy 192.150.42.105
ip domain-name mycompany.com
ip name-server 144.254.5.27
! The following command globally enables automated double authentication:
ip trigger-authentication timeout 60 port 7500
isdn switch-type basic-5ess
!
```

```
!
interface Ethernet0
ip address 144.254.166.10
no ip route-cache
no ip mroute-cache
no keepalive
ntp disable
no cdp enable
!
interface Virtual-Template1
ip unnumbered Ethernet0
no ip route-cache
no ip mroute-cache
```

Listing 10-5 Continued

```
!
! Automated double authentication occurs via the ISDN BRI interface BRI0:
interface BRI0
ip unnumbered Ethernet0
! The following command turns on automated double authentication at this
! interface:
ip trigger-authentication
! PPP encapsulation is required:
encapsulation ppp
```

no ip route-cache

no ip mroute-cache

dialer idle-timeout 500

dialer map ip 192.150.42.1 name Brabch2 5554768

dialer-group 1

no cdp enable

! **The following command specifies that device authentication occurs via PPP
! CHAP:

ppp authentication chap

!

router eigrp 109

redistribute static

network 144.254.0.0

no auto-summary

!

ip default-gateway 172.18.1.1

no ip classless

ip route 192.150.42.0 255.255.255.0 Bri0

!

! Virtual profiles are required for double authentication to work:

virtual-profile virtual-template 1

dialer-list 1 protocol ip permit

no cdp run

!

```
! The following command defines where the TACACS+ server is:

tacacs-server host 144.254.5.9 port 1049

tacacs-server timeout 90

! The following command defines the key to use with TACACS+ traffic (required):

tacacs-server key mytacacskey

!

line con 0

exec-timeout 10 0

login authentication console

line aux 0

transport input all

line vty 0 4

exec-timeout 10 0

password lab

!
```

Accounting and Billing

In large corporations, accounting and billing are essential for keeping track of who is accessing which corporate resources. Although it is mostly a network management function, keeping a historical database of dial-in usage patterns can alert the network administrator to any unusual activity and can serve as a historical paper trail when an intrusion does occur. The important parameters to keep track of include the following:

- Origin of connection
- Destination of connection
- Duration of connection

TACACS+ and RADIUS Accounting

When aaa accounting is enabled, the NAS reports user activity to the TACACS+ or RADIUS security server (depending on which security method you have implemented) in the form of *accounting records*. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. This data can be analyzed for network management, client billing, and auditing purposes.

Like authentication and authorization method lists, method lists for accounting define the way accounting is performed. Named accounting method lists enable you to designate a particular security protocol to be used on specific lines or interfaces for accounting services.

aaa accounting <event type> {default | list-name}{start-stop | wait-start

|stop-only | none} [method1 [method2]]

Five different event types are supported:

Event Description

system Enables accounting for all system-level events not associated with users (such as reloads).

network Enables accounting (including packet and byte counts) for all network-related requests, including SLIP, PPP, and ARAP sessions.

connection Provides information about all outbound connections made from the NAS, such as Telnet, local-area transport (LAT), TN3270, packet assembler/disassembler (PAD), and rlogin.

exec Enables accounting for EXEC processes (user shells).

command Applies to the EXEC mode commands a user issues. Command authorization attempts authorization for all EXEC mode commands, including global configuration commands associated with a specific privilege level.

You can specify when accounting records are to be sent by using one of the following keywords:

Keyword Description

start-stop An accounting start record is sent to the server as soon as the session begins (it does not wait for an acknowledgment from the server). A stop record is sent when the session ends and includes the session statistics.

wait-start An accounting start record is not sent until an acknowledgment is received from the server that the session has started. A stop record is sent when the session ends and includes the session statistics.

stop-only The NAS sends only an accounting stop at the end of the session; the stop record includes the session statistics.

none Stops all accounting activities on a line or interface.

Cisco IOS software supports the following two methods for accounting:

Method Description

TACACS+ The NAS reports user activity to the TACACS+ security server in the form of accounting records. Each accounting record contains accounting AV pairs and is stored on the security server.

RADIUS The NAS reports user activity to the RADIUS security server in the form of accounting records. Each accounting record contains accounting AV pairs and is stored on the security server.

In the following sample configuration, RADIUS-style accounting is used to track all usage of EXEC commands and network services such as SLIP, PPP, and ARAP:

aaa accounting exec start-stop radius

aaa accounting network start-stop radius

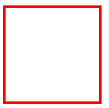
Accounting records are text lines containing tab-separated fields. The first six fields are always the same:

- Timestamp
- NAS name
- Username
- Port
- Address
- Record type

Centralized Billing

For central control of dial-in use and a centralized billing strategy, it is often the requirement of large corporations to use a callback mechanism (see Figure 10-7).

Figure 10-7: A Callback Example



The steps for a callback are as follows:

Step 1 Remote user dials in to network access server.

Step 2 The NAS disconnects the call.

Step 3 The NAS authenticates the remote user.

Step 4 If the user is authenticated, the NAS initiates a call to the remote user and a connection is established.

Configurations for both the NAS and the TACACS+ servers are shown in Listings 10-6 and 10-7.

Listing 10-6 Configuration for NAS Server

```
NAS(config)# aaa new-model

NAS(config)# tacacs-server host 144.254.5.9

NAS(config)# tacacs-server key secretkey

NAS(config)# aaa accounting exec wait-start tacacs+

NAS(config)# aaa accounting network wait-start tacacs+

NAS(config)# service exec-callback

NAS(config)# arap callback

!

NAS(config)# aaa authentication login EXECCHECK tacacs+

NAS(config)# aaa authorization network tacacs+

NAS(config)# aaa authentication arap ARAPCHECK tacacs+

NAS(config)# aaa authorization network tacacs+

NAS(config)# aaa authentication ppp PPPCHECK tacacs+

NAS(config)# aaa authorization network tacacs+

!

NAS(config)# line 4

NAS(config-line)# login authentication EXECCHECK

NAS(config-line)# arap authentication ARAPCHECK
```

Listing 10-6 Continued

```
!

NAS(config)# int async 6

NAS(config-if)# ppp authentication chap PPPCHECK
```

NAS(config-if)# ppp callback accept

Listing 10-7 Configuration for TACACS+ Server

```
user = merike {

arap = cleartext AAAA

login = cleartext LLLL

chap = cleartext CCCC

pap = cleartext PPPP

opap = cleartext OOOO

service = ppp protocol = lcp {

callback-dialstring=67150

}

service = arap {

callback-dialstring=2345678

}

service = exec {

callback-dialstring=3456789

callback-line=7

nocallback-verify=1

}

}
```

Additional Considerations for Virtual Dial-In Environments

When using a virtual dial-in environment in which dial-in access is provided by using an ISP's public infrastructure, additional security measures must be taken to ensure that the data traversing the public

network is not modified in transit and is kept private. These additional security measures are implemented using a combination of various tunneling techniques, including GRE, L2F, L2TP, IPsec, and CET.

Note The PPTP, L2F, and L2TP tunneling technologies were discussed in Chapter 2, "Security Technologies." These techniques were specifically designed to add more security services to virtual dial-in environments. They all accomplish nearly the same functions of providing flexibility of authenticating dial-in clients with an ISP NAS (using either local or remote security servers), creating a virtual tunnel between the ISP NAS and a corporate home gateway, and finally negotiating a virtual PPP session between the originating client and the home gateway. Because extensions for multiprotocol PPP environments exist, using these mech-anisms allows for native routing of non-IP protocols such as AppleTalk and IPX. However, if you want to add IP packet authentication and confidentiality on top of the multiprotocol data tunnel, you must encapsulate these protocols in an IP GRE tunnel.

GRE Tunneling

The *Generic Routing Encapsulation (GRE) protocol* encapsulates various network protocols inside IP tunnels. With GRE tunneling, a router at each site encapsulates protocol-specific packets in an IP header, creating a virtual point-to-point link to routers at other ends of an IP cloud, where the IP header is stripped off. GRE is capable of handling the transportation of multiprotocol and IP multicast traffic between two sites that have only IP unicast connectivity.

GRE tunneling involves three types of protocols:

- *Passenger.* The protocol is encapsulated (for Cisco devices, this includes IP, CLNP, IPX, AppleTalk, DECnet Phase IV, XNS, VINES, and Apollo).
- *Carrier*. GRE protocol provides carrier services.
- *Transport.* IP carries the encapsulated protocol.

GRE tunneling allows non-IP protocols to take advantage of the enhanced security functions built into IP. Specifically, it allows for authenticated data packets and provides confidentiality of these packets using various encryption techniques specifically designed for the IP protocol.

GRE tunneling should be used with care because it can disguise the nature of a link, making it look slower, faster, or more or less costly than it may actually be in reality. This change can cause problems with routing behavior. It also takes up more CPU cycles than does routing a protocol natively.

Cisco Encryption Technology (CET)

CET is a proprietary security solution introduced in Cisco IOS Release 11.2. It provides network data encryption at the IP packet level and implements the following standards:

- Digital Signature Standard (DSS)
- Diffie-Hellman (DH) public-key algorithm
- Data Encryption Standard (DES)

Following is a simple configuration example of two routers that use CET to encrypt/decrypt Telnet and WWW traffic between the branch office and the corporate campus network.

Branch Router Configuration Commands:

```
hostname Branch_router

crypto map toNAS 10

set algorithm 56-bit des cfb-64

!encryption peer is device named NAS

set peer NAS

! encrypt/decrypt what is defined in this access-list

match address 101

!

interface Ethernet 0

ip address 192.150.42.1 255.255.255.0

!

interface Bri 0

ip address unnumbered Ethernet 0

encapsulation ppp

dialer map ip 144.254.5.20 name NAS

dialer-group 1

ppp authentication chap

! associate crypto map with Bri interface

crypto map toNAS

!

! define traffic to start dial or keep isdn line up

dialer-list 1 protocol ip permit
```

! encrypt/decrypt telnet traffic between branch and campus

access-list 101 permit ip 144.254.0.0 0.0.255.255 192.150.42.0 0.0.0.255 eq telnet

! encrypt/decrypt WWW traffic between branch and campus

access-list 101 permit ip 144.254.0.0 0.0.255.255 192.150.42.0 0.0.0.255 eq http

NAS Configuration Commands:

hostname NAS

crypto map toBranch 10

set algorithm 56-bit des cfb-64

! encryption peer is device named Branch_router

set peer Branch-router

! encrypt/decrypt what is defined in this access-list

match address 101

!

interface Ethernet 0

ip address 144.254.5.20 255.255.255.0

!

! configuring PRI

interface Serial0:23

description to the Branch

crypto map toBranch

!

! encrypt/decrypt Telnet traffic between branch and campus

access-list 101 permit ip 192.150.42.0 0.0.0.255 144.254.0.0 0.0.0.255 eq telnet

! encrypt/decrypt WWW traffic between branch and campus

access-list 101 permit ip 192.150.42.0 0.0.0.255 144.254.0.0 0.0.0.255 eq http

IPsec

IPsec is a framework of open standards developed by the IETF that provides security services at the IP level (refer to Chapter 2 for details). IPsec shares the same benefits as CET: Both technologies offer authenticated and confidential IP packet transport. IPsec, however, offers a standards-based solution that provides multivendor interoperability.

Note GRE is required in an IPsec environment if there is a requirement to encrypt an IP routing protocol that uses multicast packets to distribute its routing information (for example, OSPF or EIGRP). In its current form, the IPsec standard supports only unicast IP packets. For small environments, static routes should suffice and GRE tunnels are not necessary.

The following is a simple configuration example of two routers that use IPsec to encrypt/decrypt Telnet and WWW traffic between the branch office and the corporate campus network.

Branch Router Configuration:

hostname Branch_router

crypto ipsec transform-set first ah-md5-hmac

mode tunnel

crypto ipsec transform-set second ah-sha-hmac esp-des

mode tunnel

!

crypto isakmp policy 5

auth rsa-encr

hash md5

lifetime 3600

!

crypto map toNAS 10 ipsec-isakmp

set peer 144.254.5.20

set transform-set first second

match address 106

!

interface Ethernet0

ip address 192.150.42.1 255.255.255.0

!

interface Bri 0

ip address unnumbered Ethernet 0

encapsulation ppp

dialer map ip 144.254.5.20 name NAS

dialer-group 1

ppp authentication chap

! associate crypto map with BRI interface

crypto map toNAS

!

! encrypt/decrypt Telnet traffic between branch and campus

access-list 106 permit ip 144.254.0.0 0.0.255.255 192.150.42.0 0.0.0.255 eq telnet

! encrypt/decrypt WWW traffic between branch and campus

access-list 106 permit ip 144.254.0.0 0.0.255.255 192.150.42.0 0.0.0.255 eq http

NAS Configuration:

hostname NAS

!

```
crypto ipsec transform-set first ah-md5-hmac
mode tunnel
crypto ipsec transform-set second ah-sha-hmac esp-des
mode tunnel
!
crypto isakmp policy 5
auth rsa-encr
hash md5
lifetime 3600
!
crypto map toBranch 10 ipsec-isakmp
set peer 192.150.42.1
set transform-set first second
match address 106
!
interface Ethernet 0
ip address 144.254.5.20 255.255.255.0
!
!configuring PRI
interface Serial0:23
description to the Branch
crypto map toBranch
!
```

! encrypt/decrypt Telnet traffic between branch and campus

access-list 106 permit ip 192.150.42.0 0.0.0.255 144.254.0.0 0.0.0.255 eq telnet

!encrypt/decrypt WWW traffic between branch and campus

access-list 106 permit ip 192.150.42.0 0.0.0.255 144.254.0.0 0.0.0.255 eq http

Implementation Examples

This section shows two comprehensive examples of virtual dial-in environments. The first example uses GRE tunnels with CET; the second example uses L2TP with IPsec.

GRE with CET

The example in Figure 10-8 shows a branch router located in Estonia that is connecting to the corporate network in Vancouver over the Internet.

Figure 10-8: Virtual Dial-In Using GRE with CET

The following security policy is defined for this scenario:

- Private addresses are used for the remote branch router and the corporate network.
- Communications from the remote branch to the corporate network must be private and must be encrypted.
- All infrastructure devices should have authenticated access.

The policy is implemented as follows:

- A tunnel is constructed between the home gateway and the remote branch router. The function of the tunnel is to provide connectivity from the branch office private network address space into the corporate network private address space.
- When configuring the tunnel interface between the two routers, the source and destination of the tunnel must be registered IP addresses.
- The remote router runs NAT so that communications from the remote router to the Internet are routed locally, and only communications to the corporate network go across the encrypted tunnel.
- The corporate home gateway router has two separate links to the firewall: One over a network that

has a registered IP address, and the other over a network with the private network address.

- A filter is placed on the corporate home gateway router to ensure that only virtual private network routes are passed on the private network link to the firewall (access-list 120).
- Device authentication is by way of a local database. Passwords are the same (because we're dealing with only a limited number of devices) but are changed every two months.

Home Gateway Configuration:

hostname Vancouver-gw

!

! ensure all vty, login, line, and username passwords are encrypted

! with minimal encryption (7) unless configured as a secret

! that uses MD5 encryption

service password-encryption

!

! disable access to minor TCP services such as echo,

! chargen, discard, and daytime

no service tcp-small-servers

!

! disable access to minor UDP services such as echo,

! chargen, and discard

no service udp-small-servers

!

!define privileged access password

enable secret letmedostuff

!

! local database for device authentication access

```
username admin password ComeOnIN
!
! Change the encryption key every 24 hours
crypto cisco key-timeout 1440
!
! Public key for the remote router Eesti
crypto key pubkey-chain dss
named-key Eesti signature
serial-number 07124346
key-string
44EF0246 9EF0E99E 79BA3629 142D4C0E 923D02EF 5B358A1C 089468CE 8B3562F8
398692A8 A38D99F8 0703913C 2F51F7B6 9217128C 29BA6251 AA77E442 2EE00A63
quit
!
! Crypto map for the connection between Vancouver-gw and Eesti,
! this defines the remote crypto peer, what traffic to encrypt.
! It is applied to the tunnel and physical interfaces.
!
crypto map Vancouver-to-Eesti 10
set peer Eesti
match address 140
!
! Tunnel interface from Vancouver-gw to remote branch router (Eesti)
```

```
! The tunnel interface is unnumbered to preserve address space; it could also
! use IP addresses from the private network space.
! The source of the tunnel and the destination of the tunnel are ISP registered
! addresses because the tunnel end points must be reachable across the Internet.
!
interface Tunnel100
description tunnel to branch router Eesti
ip unnumbered FastEthernet5/0
no ip directed- broadcast
tunnel source Serial2/0
tunnel destination 207.9.31.1
crypto map Vancouver-to-Eesti
!
! Apply the crypto map to the serial interface
!
interface Serial2/0
description connection to ISP1 - DS3
ip address 207.1.1.1 255.255.255.252
no ip directed-broadcast
framing c-bit
cablelength 50
dsu bandwidth 44210
crypto map Vancouver-to-Eesti
```

```
!
interface FastEthernet3/0
description network for Internet traffic
ip address 207.1.2.1 255.255.255.240
no ip directed-broadcast
full-duplex
!
interface FastEthernet5/0
description network for private network traffic
ip address 172.26.71.1 255.255.255.252
no ip directed-broadcast
full-duplex
! Access list so that only private network traffic traverses this link
ip access-group 120
!
ip classless
!
! Default route to ISP
ip route 0.0.0.0 0.0.0.0 207.1.1.2
! Routes for the corporate intranet for use by the VPN routers
ip route 172.20.0.0 255.255.0.0 172.26.71.2
ip route 172.26.0.0 255.255.128.0 172.26.71.2
! Route to the remote branch network on router Eesti
```

```
ip route 172.26.129.0 255.255.255.0 Tunnel100

! Route to the NAT pool on the firewall

ip route 207.1.2.16 255.255.255.248 207.1.2.2

! ACL list to only allow VPN traffic through the VPN DMZ interface

access-list 120 permit ip 172.26.129.0 0.0.0.255 any

access-list 120 permit ip 172.26.130.0 0.0.0.255 any

! ACL to determine what to be encrypted, packets between

! the two tunnel endpoints which are GRE encapsulated.

access-list 140 permit gre host 207.1.1.1 host 207.9.31.1

!

line con 0

exec-timeout 2 30

login authentication admin

!

line vty 0 4

exec-timeout 2 30
```

login authentication admin

Remote Branch Router Configuration:

```
hostname Eesti

!

! ensure all vty, login, line and username passwords are encrypted

! with minimal encryption (7) unless configured as a secret

! that uses MD5 encryption
```

```
service password-encryption
!
! disable access to minor TCP services such as echo,
! chargen, discard, and daytime
no service tcp-small-servers
!
! disable access to minor UDP services such as echo,
! chargen, and discard
no service udp-small-servers
!
!define privileged access password
enable secret letmedostuff
!
! local database for device authentication access
username admin password ComeOnIN
!
! Change the encryption key every 24 hours
crypto cisco key-timeout 1440
!
! Public key for the home gateway Vancouver-gw
!
crypto key pubkey-chain dss
named-key VancouverESA signature
```

```
serial-number 007462E4
key-string
17C11157 CC640BF3 3DC5B608 C5C60963 C0421A67 D2D7AF70 97728A9A BACA0E07
35288070 AD90A20F 56F1BFE7 D8A4BB68 2C2419E0 26CF8E17 B09CA9A0 3090942E
quit
!
! Crypto map for the connection from Eesti to Vancouver-gw, this defines the remote
! peer, and what traffic to encrypt, which is determined by access list 140
! This gets applied to the tunnel and physical interfaces.
!
crypto map Eesti-to-Vancouver 10
set peer VancouverESA
match address 140
!
! Tunnel interface from remote branch (Eesti) to home gateway (Vancouver-gw)
!
interface Tunnel100
description network connection back to headquarters (Vancouver)
ip unnumbered Ethernet1/0
no ip directed-broadcast
tunnel source 207.9.31.1
tunnel destination 207.1.1.1
crypto map Eesti-to-Vancouver
```

```
!
! Apply the crypto map to the physical interface,
! this is also the outside NAT interface.
!
interface Serial0/0
description frame relay connection to ISP
ip address 207.9.31.1 255.255.255.240
no ip directed-broadcast
ip nat outside
encapsulation frame-relay
frame-relay lmi-type ansi
crypto map Eesti-to-Vancouver
!
! NAT inside interface
!
interface Ethernet1/0
description private IP address for remote site
ip address 172.26.129.1 255.255.255.0
no ip directed-broadcast
ip nat inside
!
! Translate IP addresses matching access list 150 into the IP address
! given to serial interface connected to the ISP
```

ip nat inside source list 150 interface Serial0/0 overload

ip classless

! default route to ISP

ip route 0.0.0.0 0.0.0.0 207.9.31.14

!

! Routes for the networks inside the corporate intranet that

! the remote needs to access

!

ip route 172.26.0.0 255.255.128.0 Tunnel100

ip route 172.20.0.0 255.255.0.0 Tunnel100

!

! Traffic going to any other destination will take the default route and be

! translated by NAT, access list 150 tells NAT what to translate.

!

access-list 150 permit ip 172.26.129.0 0.0.0.255 any

!

! ACL to determine what to be encrypted,

! all packets between the two tunnel endpoints.

!

access-list 140 permit gre host 207.9.31.1 host 207.1.1.1

!

line con 0

exec-timeout 2 30

login authentication admin

!

line vty 0 4

exec-timeout 2 30

login authentication admin

L2TP with IPsec

The example in Figure 10-9 shows the remote connection of a remote branch office in Toronto and a remote branch office in New York connecting back to the corporate network in Denver. Both connections are done through local ISPs and use the Internet as the way to transport the data back to the corporate network in Denver. Mobile users also have access to the corporate network using local ISP dial-up connections.

Figure 10-9: Virtual Dial-In Using L2TP with IPsec

The following security policy is defined for this example:

- The branch office in Toronto is allowed to communicate directly to the Internet but must encrypt all traffic going to the corporate network in Denver.
- All New York branch office traffic must go through the Denver corporate office firewall.
- All mobile users use authenticated and private data connections back to the corporate network through ISP collaborate agreements.
- All corporate infrastructure device access is required to be authenticated and authorized for limited access.

The policy is implemented as follows:

- The branch office router in Toronto allows the users to talk directly to the Internet while using an IPsec-encrypted tunnel to access the corporate network. The serial interface on the router has been assigned an IP address from the ISP's address space. The Ethernet interface uses a private network address, and NAT is used to translate traffic going to the Internet. This router uses static routing.
- The branch router in New York requires that all traffic, even traffic to the Internet, must go through the corporate firewall. The serial interface on the router has been assigned an IP address

from the ISP's address space; the Ethernet interface uses a private network address. This router uses OSPF routing.

- There is an agreement between the ISP and the corporation that if a mobile user presents the ISP's NAS with a username in the format username@mkos.name, the PPP session will be transported to the corporation's home gateway for termination. Using L2TP tunneling with IPsec, a secure tunnel is provided from the NAS (isp-nas) to the home gateway (Denver-gw).

Home Gateway Router Configuration:

```
hostname Denver-gw

!

! In IOS firewall IPsec images "no service tcp & no udp small servers" is the

! default so it does not have to be explicitly defined.

! Turn on timestamps for log and debug information, set to the local time with

! timezone information displayed.

!

service timestamps debug datetime msec localtime show-timezone

service timestamps log datetime msec localtime show-timezone

!

service password-encryption

!

no logging console

!

! Enable TACACS+ to authenticate login, enable any PPP sessions, also enable

! accounting start-stop records for EXEC and PPP sessions

!

aaa new-model

aaa authentication login default tacacs+ enable
```

```
aaa authentication login console none
aaa authentication enable default tacacs+ enable
aaa authentication ppp default tacacs+
aaa authorization network default tacacs+
aaa accounting exec default start-stop tacacs+
aaa accounting network default start-stop tacacs+
!
enable secret 5 $1$xDvT$sT/TGeGrAwfAKbMr4N1NZ1
enable password 7 02050D480809
!
no ip finger
ip domain-name mkos.com
!
! Enable VPDN and tell it to use L2TP. The PPP name of the remote NAS will be
! isp-nas and the local PPP name is Denver-gw. Also for the VPDN, use an
! alternative tacacs+ server. Connections inbound will use virtual-template 1
! as the basis to create to the actual virtual-access interface.
!
vpdn enable
!
vpdn aaa override-server 172.20.24.47
vpdn-group 1
accept dialin l2tp virtual-template 1 remote isp-nas
```

local name Denver-gw

!

! Define the IPsec transform policy set, (ah-sha-hmac) AH with SHA

! authentication algorithm, (esp-des) ESP with 56-bit DES encryption algorithm,

! (esp-sha-hmac) ESP with SHA authentication algorithm. Because a GRE is used,

! run IPsec in transport rather than tunnel mode.

!

crypto ipsec transform-set auth2 ah-sha-hmac esp-des esp-sha-hmac

mode transport

!

! IPsec using certificates: The routers must first obtain certificates from

! the Certificate Authority (CA) server. When both peers have valid certificates,

! they automatically exchange RSA public keys as part of the ISAKMP negotiation.

! All that is required is that the routers register with the CA and obtain

! a certificate. A router does not have to keep public RSA keys for all peers

! in the network.

!

crypto ca identity vpnnetwork

enrollment url http://mkosca

crl optional

cryto ca certificate chain vpnnetwork

certificate 44FC6C531FC3446927E4EE307A806B20

```
! Certificate is multiple lines of hex digits
quit
certificate ca 3051DF7169BEE31B821DFE4B3A338E5F
! Certificate of the CA, multiple of lines hex digits
quit
certificate 52A46D5D10B18A6F51E6BC735A36508C
! Certificate is multiple lines of hex digits
quit
!
! The crypto map determines what to encrypt and to what peer to send the traffic.
! An interface can have only one crypto map applied to it. The crypto map below
! is structured into sections, which apply for the different destinations,
! while still being a single crypto map entity.
!
crypto map Denver-to-remotes local-address Serial2/0
crypto map Denver-to-remotes 100 ipsec-isakmp
set peer 207.9.31.1
set transform-set auth2
match address Denver_gre_Toronto
crypto map Denver-to-remotes 200 ipsec-isakmp
set peer 207.10.31.1
set transform-set auth2
match address Denver_gre_NewYork
```

```
crypto map Denver-to-remotes 500 ipsec-isakmp

set peer 201.1.1.1

set transform-set auth2

match address ISP1_VPDN

!

! Set the timezone and daylight savings time for this router.

!

clock timezone PST -8

clock summer-time PDT recurring

!

! Tunnel interface to router Toronto. The tunnel source is specified as an

! interface with a registered IP address. The crypto map is applied to both

! the tunnel and physical interfaces. The IP precedence of packets being

! tunneled are copied into the IP header of the outbound frame.

! This example uses an IP unnumbered tunnel interface. Only packets destined

! for the intranet arrive on this interface because NAT is used at the remote

! for packets destined for the Internet.

!

interface Tunnel100

description tunnel to branch router Toronto

ip unnumbered FastEthernet5/0

no ip directed-broadcast

tunnel source Serial2/0
```

```
tunnel destination 207.9.31.1
crypto map Denver-to-remotes
!
! Tunnel interface to router New York. The crypto map is applied to both the
! tunnel and physical interfaces. Note that the same crypto map has been used
! on both the tunnels, with different sections of the crypto map applying to each
! tunnel. The IP precedence of packets being tunneled are copied into the IP
! header of the outbound frame. This example uses an IP-numbered tunnel interface
! with OSPF as the routing protocol and routing information authentication
! enabled. The policy for this remote site is that all packets destined to the
! Internet must go through the corporate firewall. This is achieved by using
! policy routing (route-map VPN_InBound).
!
interface Tunnel101
description tunnel to branch router NewYork
ip address 172.26.123.1 255.255.255.252
no ip directed-broadcast
ip ospf message-digest-key 1 md5 7 00071A15075434101F2F
ip policy route-map VPN_InBound
tunnel source Serial2/0
tunnel destination 207.10.31.1
crypto map Denver-to-remotes
!
```

```
! DS3 connection to ISP. Two ACLS are applied here. The inbound ACL stops
! some common protocols and network addresses known to be invalid or harmful.
! The outbound security ACL prevents packets from private network addresses
! that have not been through NAT from leaving. The crypto map is applied
! to the interface.
!
interface Serial2/0
description connection to ISP1 - DS3
ip address 207.1.1.1 255.255.255.252
ip access-group IntSecurity in
ip access-group IntSecurityOut out
no ip directed-broadcast
framing c-bit
cablelength 50
dsu bandwidth 44210
crypto map Denver-to-remotes
!
! This interface is connected to the corporate network Web server and to the
! firewall, which is doing NAT for the corporate network's access to the
! Internet.
!
interface FastEthernet3/0
description network for Internet traffic
```

```
ip address 207.1.2.1 255.255.255.240
no ip directed-broadcast
full-duplex
!
! This interface is connected to the firewall, is treated as an inside interface,
! is for the VPN traffic to access the corporate network, and is using NAT
! on the firewall to the Internet. This route-map on the interface is responsible
! for setting the correct precedence on the IP packets destined for the VPN,
! to gain the QoS agreement with the service provider. The ACL is used to allow
! only known VPN networks on the link.
!
interface FastEthernet5/0
description network for VPN traffic
ip address 172.26.71.1 255.255.255.252
ip access-group 120 out
no ip directed-broadcast
ip policy route-map VPN_QoS
full-duplex
!
! The virtual template is used by the VPDN code as the basis to create the
! virtual-access interface on which the L2TP connections terminate.
!
interface Virtual-Template1
```

```
ip unnumbered FastEthernet5/0
no ip directed-broadcast
peer default ip address pool vpn_users
!
! OSPF for the VPN network, remote branch NewYork is running OSPF.
! The OSPF process is set to redistribute static routes that match
! route-map VPN_ROUTES_OUT, and originate the default route for the
! remote VPN sites running OSPF. Authentication is enabled for routing
! information so that only remotes with the correct key can participate.
!
router ospf 100
redistribute static subnets route-map VPN_ROUTES_OUT
passive-interface FastEthernet5/0
passive-interface Tunnel100
network 172.26.71.0 0.0.0.3 area 0
network 172.26.120.0 0.0.3.255 area 172.26.120.0
default-information originate
area 172.26.120.0 authentication message-digest
!
ip classless
!
! Default route to ISP
ip route 0.0.0.0 0.0.0.0 207.1.1.2
```

```
!
! Corporate network uses 172.20/24 and 172.26/24.
ip route 172.20.0.0 255.255.0.0 172.26.71.2
ip route 172.26.0.0 255.255.0.0 172.26.71.2
!
! Static route to branch in Toronto (Ethernet 0)
ip route 172.26.120.0 255.255.255.0 Tunnel100
!
! Route to the NAT pool on the firewall
ip route 207.1.2.16 255.255.255.248 207.1.2.2
!
! ACL to determine what frames get set specified QoS for ISP1
ip access-list extended Bronze_ISP1_QoS
permit ip 172.26.0.0 0.0.255.255 172.26.120.0 0.0.0.255
permit ip 172.20.0.0 0.0.255.255 172.26.120.0 0.0.0.255
!
! ACL to determine the traffic to encrypt for the VPDN L2TP tunnel
! from ISP NAS "isp-nas"
ip access-list extended ISP1_VPDN
permit ip host 207.1.1.1 host 201.1.1.1
!
! ACL to block any traffic inbound from private addresses
! and some common troublesome services
```

```
ip access-list extended IntSecurity
permit tcp any any established
deny ip 127.0.0.0 0.255.255.255 any
deny ip 10.0.0.0 0.255.255.255 any
deny ip 172.16.0.0 0.15.255.255 any
deny ip 192.168.0.0 0.0.255.255 any
deny udp any any eq snmp
deny udp any any eq 2000
deny udp any any gt 6000
deny tcp any any gt 6000
deny tcp any any eq 2000
deny udp any any eq tftp
deny udp any any eq sunrpc
deny udp any any eq 2049
deny tcp any any eq 2049
deny tcp any any eq sunrpc
deny tcp any any eq 87
deny tcp any any eq exec
deny tcp any any eq login
deny tcp any any eq cmd
deny tcp any any eq lpd
deny tcp any any eq uucp
permit ip any any
```

```
!
! ACL to prevent any packets from private addresses being sent to the Internet.
ip access-list extended IntSecurityOut
deny ip 127.0.0.0 0.255.255.255 any
deny ip 10.0.0.0 0.255.255.255 any
deny ip 172.16.0.0 0.15.255.255 any
deny ip 192.168.0.0 0.0.255.255 any
permit ip 207.0.0.0 0.255.255.255 any
!
! ACL to determine which frames are set to Silver QoS for ISP1
ip access-list extended Silver_ISP1_QoS
permit ip 172.26.0.0 0.0.255.255 172.26.121.0 0.0.0.255
permit ip 172.20.0.0 0.0.255.255 172.26.121.0 0.0.0.255
!
! ACL determines which packets IPsec will look at for tunnel100
ip access-list extended Denver_gre_Tororno
permit gre host 207.1.1.1 host 207.9.31.1
!
! ACL determines which packets IPsec looks at for tunnel101
ip access-list extended Denver_gre_NewYork
permit gre host 207.1.1.1 host 207.10.31.1
!
! Turn on syslog and point it at the management station.
```

```
logging 172.20.18.5
!
! ACL determines which static routes are redistributed into the OSPF VPN process
access-list 18 permit 172.26.0.0 0.0.255.255
access-list 18 permit 172.20.0.0 0.0.255.255
!
! ACL only allows Telnet to the router from particular subnets
access-list 70 permit 172.20.18.0 0.0.0.192
access-list 70 permit 172.20.24.0 0.0.0.255
!
! ACL determines which management stations can access this device using SNMP
access-list 75 permit 172.20.18.0 0.0.0.255
!
! ACL only allows particular networks on the VPN interface to the firewall
access-list 120 permit ip 172.26.120.0 0.0.0.255 any
access-list 120 permit ip 172.26.121.0 0.0.0.255 any
access-list 120 permit ip 172.26.122.0 0.0.0.255 any
access-list 120 permit ip 172.26.123.0 0.0.0.255 any
!
! ACL for route map to policy route all packets to the firewall.
access-list 195 permit ip 172.26.121.0 0.0.0.255 any
access-list 195 permit ip 172.26.123.0 0.0.0.3 any
!
```

```
! Route map determines which routes to distribute into OSPF VPN process
route-map VPN_ROUTES_OUT permit 20
match ip address 18
set metric 1000
set metric-type type-1
!
! Route map used to policy route all specified packets to the corporate firewall
route-map VPN_InBound permit 100
match ip address 195
set ip next-hop 172.26.71.2
!
! Route map used to set the precedence bits on outbound VPN network packets
route-map VPN_QoS permit 100
match ip address Bronze_ISP1_QoS
set ip precedence priority
route-map VPN_QoS permit 200
match ip address Silver_ISP1_QoS
set ip precedence immediate
!
! Configure SNMP, only allow management stations matching access list 75
! to manage this router
snmp-server community public RO 75
snmp-server community private RW 75
```

```
snmp-server trap-source Ethernet1/0
snmp-server packetsize 4096
snmp-server enable traps snmp
snmp-server enable traps frame-relay
snmp-server host 172.20.18.5 traps public
snmp-server tftp-server-list 75
!
! Configure which TACACS server to use and the key.
tacacs-server host 172.20.18.5
tacacs-server key SECRET12345
!
! Console and vty are secured using TACACS+
line con 0
exec-timeout 5 0
transport input none
line aux 0
!
! Only allow Telnet to this router if the source address is in access list 70
line vty 0 4
access-class 70 in
password 7 1511021F0725
transport input telnet
!
```

! Configure NTP so that all the routers have the same time in the network.

ntp clock-period 17179770

ntp server 172.26.71.2

end

Remote Branch Router in Toronto Configuration:

hostname Toronto

!

! In IOS firewall IPsec images "no service tcp & no udp small servers" is the

! default. Turn on timestamps for log and debug information and set to the local

! time with timezone information displayed.

!

service timestamps debug datetime msec localtime show-timezone

service timestamps log datetime msec localtime show-timezone

service password-encryption

!

logging buffered 32000 debugging

no logging console

!

! Enable TACACS+ to authenticate login and enable passwords,

! also enable accounting start-stop records for exec sessions

!

aaa new-model

aaa authentication login default tacacs+ enable

aaa authentication enable default tacacs+ enable

aaa accounting exec default start-stop tacacs+

!

enable secret 5 1SKkd$qbTmOJ9dyffjccNUB0cvn0

enable password 7 02050D480809

!

no ip finger

ip domain-name mkos.com

!

! Define the IPsec transform policy set; because a GRE is used, run IPsec in

! transport rather than tunnel mode.

!

crypto ipsec transform-set auth2 ah-sha-hmac esp-des esp-sha-hmac

mode transport

!

crypto ca identity vpnnetwork

enrollment url http://mkosca

crl optional

cryto ca certificate chain vpnnetwork

certificate 44FC6C531FC3446927E4EE307A806B20

! Certificate is multiple lines hex digits

quit

certificate ca 3051DF7169BEE31B821DFE4B3A338E5F

! Certificate is multiple lines hex digits

quit

certificate 52A46D5D10B18A6F51E6BC735A36508C

! Certificate is multiple lines hex digits

quit

!

! The crypto map determines what packets should be encrypted as determined by

! access list 140, and the crypto peer that is the IP address of Denver-gw,

! along with the transforms that will be allowed. The setting of the local-address

! ensures that if there are multiple paths, the same IP address is always used

! for this crypto pair, no matter what interface a packet arrives on.

!

crypto map ipsec-Toronto-to-Denver local-address Serial0/0

crypto map ipsec-Toronto-to-Denver 10 ipsec-isakmp

set peer 207.1.1.1

set transform-set auth2

match address 140

!

! Set the timezone and daylight savings time for this router

!

clock timezone EST -5

clock summer-time EDT recurring

!

```
! Tunnel interface to transport traffic to Denver-gw, the tunnel source is
! specified as an interface with a registered IP address. The IP address of
! the Ethernet is used, which is a private address; an unnumbered interface
! is used here to show that you do not have to address the tunnel interface.
! The IP precedence of the packets being tunneled are copied into the IP header
! of the outbound frame.
!
interface Tunnel100
description VPN connection back to headquarters (Denver)
ip unnumbered Ethernet1/0
no ip directed-broadcast
tunnel source Serial0/0
tunnel destination 207.1.1.1
crypto map ipsec-Toronto-to-Denver
!
! Serial 0/0 is the connection to the ISP; it has one of the ISP's registered
! addresses. Two access lists are applied to the interface: one inbound and
! one outbound. These are explained where the access list is defined below.
! This interface is specified as the outside interface for NAT.
! Finally, the crypto map is applied to the interface to determine what
! should be encrypted.
!
interface Serial0/0
```

description frame relay connection to ISP

ip address 207.9.31.1 255.255.255.240

ip access-group IntSecurity in

ip access-group IntSecurityOut out

no ip directed-broadcast

ip nat outside

encapsulation frame-relay IETF

no ip mroute-cache

frame-relay lmi-type ansi

crypto map ipsec-Toronto-to-Denver

!

! Ethernet 1/0 is the remote LAN interface; it is assigned a private IP address

! and is a NAT inside interface. A route-map is applied to the interface to set

! the IP precedence to get the ISP Bronze offering of QoS.

!

interface Ethernet1/0

description private IP address for remote site

ip address 172.26.120.1 255.255.255.0

no ip directed-broadcast

ip nat inside

ip policy route-map Bronze_ISP1_QoS

!

! Configure NAT: Any source address matching access list 150,

```
! translate to the IP address of interface serial 0/0. The overload options
! mean that many IP addresses will be translated to serial 0/0 IP addresses
! on different ports.
!
ip nat inside source list 150 interface Serial0/0 overload
ip classless
!
! Static routes: The default is to send all traffic to the ISP. The corporation
! uses networks 172.20/24 and 172.26/24 for its networks, so any traffic
! destined to these addresses should go across the tunnel interface.
!
ip route 0.0.0.0 0.0.0.0 207.9.31.14
ip route 172.20.0.0 255.255.0.0 Tunnel100
ip route 172.26.0.0 255.255.0.0 Tunnel100
!
! ACL to block particular services and networks, inbound from the ISP.
ip access-list extended IntSecurity
permit tcp any any established
deny ip 127.0.0.0 0.255.255.255 any
deny ip 10.0.0.0 0.255.255.255 any
deny ip 172.16.0.0 0.15.255.255 any
deny ip 192.168.0.0 0.0.255.255 any
deny udp any any eq snmp
```

```
deny udp any any eq 2000
deny udp any any gt 6000
deny tcp any any gt 6000
deny tcp any any eq 2000
deny udp any any eq tftp
deny udp any any eq sunrpc
deny udp any any eq 2049
deny tcp any any eq 2049
deny tcp any any eq sunrpc
deny tcp any any eq 87
deny tcp any any eq exec
deny tcp any any eq login
deny tcp any any eq cmd
deny tcp any any eq lpd
deny tcp any any eq uucp
permit ip any any
!
! ACL to prevent packets from private networks leaving by the ISP interface.
ip access-list extended IntSecurityOut
deny ip 127.0.0.0 0.255.255.255 any
deny ip 10.0.0.0 0.255.255.255 any
deny ip 172.16.0.0 0.15.255.255 any
deny ip 192.168.0.0 0.0.255.255 any
```

```
permit ip 207.9.31.0 0.0.0.255 any
!
! Turn on syslog and point it at the management station.
logging 172.20.18.5
!
! ACL to secure why can Telnet to the router
access-list 70 permit 207.1.1.1
access-list 70 permit 172.20.18.0 0.0.0.192
access-list 70 permit 172.20.24.0 0.0.0.255
!
! ACL to determine which management stations can access this device using SNMP
access-list 75 permit 172.20.18.0 0.0.0.255
!
! ACL to determine which frames should be protected (encrypted) with IPsec
access-list 140 permit gre host 207.9.31.1 host 207.1.1.1
!
! What packets are eligible sourced from NAT inside networks for
! address translation
access-list 150 permit ip any any
!
! Access list used in route map to set IP precedence to get specified
! QoS level from ISP.
access-list 175 permit ip any any
```

```
!
! Route map used on Ethernet 1/0 to set the precedence bits of all IP frames
! to priority (1)
route-map Bronze_ISP1_QoS permit 10
match ip address 175
set ip precedence priority
!
! Configure which TACACS server to use and the key.
tacacs-server host 172.20.18.5
tacacs-server key SECRET12345
!
! Configure SNMP for network management. Because only the corporation's
! management stations will manage this router, the trap source is set to use
! Ethernet 1/0.
!
snmp-server community public RO 75
snmp-server community private RW 75
snmp-server trap-source Ethernet1/0
snmp-server packetsize 4096
snmp-server enable traps snmp
snmp-server enable traps frame-relay
snmp-server enable traps syslog
snmp-server host 172.20.18.5 traps public
```

snmp-server tftp-server-list 75

!

! Console and vty are secured using TACACS+

!

line con 0

exec-timeout 5 0

transport input none

line aux 0

!

! Only allow Telnet to this router if the source address is in access list 70

line vty 0 4

access-class 70 in

password 7 1511021F0725

transport input telnet

!

! Configure NTP so that all the routers have the same time in the network.

ntp clock-period 17179770

ntp server 172.26.71.2

end

!

Remote Branch Router in New York Configuration:

Hostname NewYork

!

```
! In IOS firewall IPsec images "no service tcp & no udp small servers" is the
! default. Turn on timestamps for log and debug information, set to the
! local time with timezone information displayed.
!
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
service password-encryption
!
logging buffered 32000 debugging
no logging console
!
! Enable TACACS+ to authenticate login and enable passwords, also enable
! accounting start-stop records for exec sessions
!
aaa new-model
aaa authentication login default tacacs+ enable
aaa authentication enable default tacacs+ enable
aaa accounting exec default start-stop tacacs+
!
enable secret 5 $1$z1c.$vLAcnZ849epT8xLHNeTT0/
enable password 7 110A1016141D
!
ip domain-name mkos.com
```

!

! Define the IPsec transform policy set; because a GRE is used, run IPsec in

! transport rather than tunnel mode.

!

crypto ipsec transform-set auth2 ah-sha-hmac esp-des esp-sha-hmac

mode transport

!

crypto ca identity vpnnetwork

enrollment url http://mkosca

crl optional

cryto ca certificate chain vpnnetwork

certificate 44FC6C531FC3446927E4EE307A806B20

! Certificate is multiple lines of hex digits

quit

certificate ca 3051DF7169BEE31B821DFE4B3A338E5F

! Certificate is multiple lines of hex digits

quit

certificate 52A46D5D10B18A6F51E6BC735A36508C

! Certificate is multiple lines of hex digits

quit

!

! The crypto map determines which packets should be encrypted as determined

```
! by access list 141, and the crypto peer, which is the IP address of Denver-gw,
! along with the transforms that will be allowed. The setting of the
! local-address ensures that if there are multiple paths, the same IP address
! is always used for this crypto pair, no matter what interface a packet arrives ! on.
!
crypto map NewYork-to-Denver local-address Serial0/0
crypto map NewYork-to-Denver 20 ipsec-isakmp
set peer 207.1.1.1
set transform-set auth2
match address 141
!
! Set the timezone and daylight savings time for this router
!
clock timezone est -8
clock summer-time EST recurring
!
! Tunnel interface to transport traffic to Denver-gw, the tunnel source is
! specified as an interface with a registered IP address. The router is
! configured to run OSPF with the home gateway across the tunnel interface.
! OSPF is using message digest 5 to authenticate routing updates.
! The crypto map is applied to both the tunnel and the physical interfaces.
! The IP precedence of packets being tunneled are copied into the IP header
! of the outbound frame.
```

```
!
interface Tunnel101
ip address 172.26.123.2 255.255.255.252
no ip directed-broadcast
ip ospf authentication-key 7 104D000A06182D1D1C
ip ospf message-digest-key 1 md5 7 045802150C2E73581917
tunnel source Serial0/0
tunnel destination 207.1.1.1
crypto map NewYork-to-Denver
!
! Serial 0/0 is the connection to the ISP; it has one of the ISP's registered
! addresses. Two ACLs are applied to the interface: one inbound and one outbound.
! The crypto map is applied to the interface to determine what should
! be encrypted.
!
interface Serial0/0
ip address 207.10.31.1 255.255.255.240
ip access-group IntSecurity in
ip access-group IntSecurityOut out
no ip directed-broadcast
encapsulation frame-relay IETF
frame-relay lmi-type ansi
crypto map NewYork-to-Denver
```

```
!
! Ethernet 1/0 is the remote LAN interface; it is assigned a private IP address.
! A route-map is applied to the interface to set the IP precedence
! level to get the ISP Silver offering of QoS.
interface Ethernet1/0
ip address 172.26.121.1 255.255.255.0
no ip directed-broadcast
ip policy route-map Silver_ISP1_QoS
!
! Configure OSPF for IP routing and authenticate routing updates.
router ospf 100
network 172.26.120.0 0.0.3.255 area 172.26.120.0
area 172.26.120.0 authentication message-digest
!
ip classless
!
! Because all traffic from the remote router must go through the firewall at
! corporate headquarters, a static default route is not used but an explicit
! route for the tunnel destination end point is used. This router gets its
! default route from OSPF.
!
ip route 207.1.1.1 255.255.255.255 207.10.31.14
!
```

```
! ACL to block particular services and networks, inbound from the ISP.

ip access-list extended IntSecurity

permit tcp any any established

deny ip 127.0.0.0 0.255.255.255 any

deny ip 10.0.0.0 0.255.255.255 any

deny ip 172.16.0.0 0.15.255.255 any

deny ip 192.168.0.0 0.0.255.255 any

deny udp any any eq snmp

deny udp any any eq 2000

deny udp any any gt 6000

deny tcp any any gt 6000

deny tcp any any eq 2000

deny udp any any eq tftp

deny udp any any eq sunrpc

deny udp any any eq 2049

deny tcp any any eq 2049

deny tcp any any eq sunrpc

deny tcp any any eq 87

deny tcp any any eq exec

deny tcp any any eq login

deny tcp any any eq cmd

deny tcp any any eq lpd

permit ip any any
```

```
!
! ACL prevents packets from private networks from leaving by the ISP interface.
ip access-list extended IntSecurityOut
deny ip 127.0.0.0 0.255.255.255 any
deny ip 10.0.0.0 0.255.255.255 any
deny ip 172.16.0.0 0.15.255.255 any
deny ip 192.168.0.0 0.0.255.255 any
permit ip 207.0.0.0 0.255.255.255 any
!
! Turn on syslog and point it at the management station.
logging 172.20.18.5
!
! ACL secures who can Telnet to the router
access-list 70 permit 207.1.1.1
access-list 70 permit 172.20.18.0 0.0.0.255
!
! ACL determines which management stations can access this device using SNMP
access-list 75 permit 172.20.18.0 0.0.0.255
!
! ACL determines which frames should be protected (encrypted) with IPsec
access-list 141 permit gre host 207.10.31.1 host 207.1.1.1
!
! ACL used in route map to set IP precedence to get specified QoS level from ISP.
```

```
access-list 175 permit ip 172.26.121.0 0.0.0.255 any
!
! Route map used on Ethernet 1/0 to set the precedence bits of all IP frames
! to immediate (2)
route-map Silver_ISP1_QoS permit 10
match ip address 175
set ip precedence immediate
!
! Configure which TACACS server to use and the key.
tacacs-server host 172.20.18.5
tacacs-server key SECRET12345
!
! Configure SNMP for network management. Because only the corporation's
! management stations will manage this router, the trap source is set to
! use Ethernet 1/0.
!
snmp-server community public RO 75
snmp-server community private RW 75
snmp-server trap-source Ethernet1/0
snmp-server packetsize 4096
snmp-server enable traps snmp
snmp-server enable traps frame-relay
snmp-server enable traps syslog
```

snmp-server host 172.20.18.5 traps public

snmp-server tftp-server-list 75

!

! Console and vty are secured using TACACS+

!

line con 0

exec-timeout 5 0

transport input none

login authentication default

line aux 0

!

! Only allow Telnet to this router if the source address is in access list 70

line vty 0 4

access-class 70 in

password 7 1511021F0725

transport input telnet

!

! Configure NTP so that all the routers have the same time in the network.

ntp clock-period 17179770

ntp server 172.26.71.2

end

ISP NAS Configuration:

Hostname isp-nas

```
!
aaa new-model
aaa authentication login default enable
aaa authentication login console none
aaa authentication enable default enable
aaa authentication ppp default tacacs+ local
aaa authorization exec default none
aaa accounting exec default start-stop tacacs+
!
enable secret 5 $1$2Ezj$2ygSyGTzphmQadmU854aL1
enable password escape
!
ip domain-name isp1.net
!
! Enable VPDN on the NAS and make the source of tunnels to be the loopback.
vpdn enable
vpdn source-ip 201.1.1.1
!
! VPDN group 1, connection to the home gateway Denver-gw, use LT2P,
! and the ppp name isp1.
!
vpdn-group 1
request dialin l2tp ip 207.1.1.1 domain mkos.com
```

```
local name isp1
!
crypto isakmp policy 10
authentication rsa-encr
group 2
lifetime 240
!
! Define the IPsec transform policy set; because an L2TP is used, run IPsec in
! transport rather than tunnel mode.
crypto ipsec transform-set auth_cisco_dial ah-sha-hmac esp-des esp-sha-hmac
mode transport
!
crypto ca identity vpnnetwork
enrollment url http://mkosca
crl optional
cryto ca certificate chain vpnnetwork
certificate 44FC6C531FC3446927E4EE307A806B20
! Certificate is multiple lines of hex digits
quit
certificate ca 3051DF7169BEE31B821DFE4B3A338E5F
! Certificate is multiple lines of hex digits
quit
!
```

```
! Crypto map to encrypt traffic destined to Denver home gateway for mkos.com
!
crypto map VPDN_MKOS local-address Loopback0
crypto map VPDN_MKOS 1000 ipsec-isakmp
set peer 207.1.1.1
set transform-set auth_mkos_dial
match address VPDN_mkos_tunnel
!
! All L2TP traffic is sourced off the loopback, apply the crytpo map for IPsec.
!
interface Loopback0
ip address 201.1.1.1 255.255.255.255
no ip directed-broadcast
crypto map VPDN_MKOS
!
interface Ethernet1/2
ip address 207.7.31.1 255.255.255.252
no ip directed-broadcast
no ip mroute-cache
crypto map VPDN_MKOS
!
! ACL to determine what traffic IPsec should be applied to.
ip access-list extended VPDN_mkos_tunnel
```

permit ip host 201.1.1.1 host 207.1.1.1

!

Summary

This chapter described the implementation considerations for providing secure remote dial-in and virtual dial-in access. This includes establishing proper authentication and authorization for any telecommuters, mobile hosts, and remote branch offices attempting to gain access to resources in the main corporate network.

It is often necessary to restrict access to certain areas of the corporate network depending on who the remote user is and from where he or she is trying to obtain the connection. Also important is keeping track of connection details (such as who connected where and the duration of the connection) to keep accurate accounting statistics for an audit trail or billing purposes.

Lastly, virtual dial-in environments require some special considerations because the data is traveling over shared public networks. Usually, you will want to ensure authenticated and private (confidential) delivery of the data packets over these public networks. It is usually a good idea to incorporate firewall functionality into the dial-in access perimeters and to implement some kind of auditing and intrusion detection system to keep accurate connection and traffic statistics.

continues

continues

continues

continues

continues

continues

continues

Posted: Wed Jun 14 11:46:12 PDT 2000

Table of Contents

A

Sources of Technical Information

Cryptography and Network Security Books

Denning, Dorothy E. *Information Warfare and Security*. Reading, MA: Addison-Wesley, 1999.

Hughes, Larry J., Jr. *Actually Useful Internet Security Techniques.* Indianapolis, IN: New Riders Publishing, 1995.

Kaufman, C., R. Perlman, and M. Speciner. *Network Security: Private Communication in a Public World*. Upper Saddle River, NJ: Prentice-Hall, 1995.

McCarthy, Linda. *Intranet Security: Stories from the Trenches.* Palo Alto, CA: Sun Microsystems Press, 1998.

Schneier, Bruce. *Applied Cryptography,* Second Edition. New York, NY: John Wiley and Sons, 1996.

Stallings, William. *Network and Internetwork Security.* Upper Saddle River, NJ: Prentice-Hall, IEEE

Press, 1995.

Firewall Books

Chapman, D. Brent and Elizabeth D. Zwicky. *Building Internet Firewalls*. Cambridge, MA: O'Reilly and Associates, 1995.

Cheswick, William and Steven Bellovin. *Firewalls and Internet Security*. Reading, MA: Addison-Wesley, 1994.

IETF Working Groups and Sites for Standards and Drafts on Security Technologies Developed Through the IETF

Point-to-Point Protocol Extensions. Includes authentication and privacy technologies used with PPP:

http://www.ietf.org/html.charters/pppext-charter.html

Remote Authentication Dial-In User Service. Details the specifications of the RADIUS AAA protocol:

http://www.ietf.org/html.charters/radius-charter.html

Authenticated Firewall Traversal. Includes SOCKS specifications:

http://www.ietf.org/html.charters/aft-charter.html

Common Authentication Technology. Includes specifications for Kerberos:

http://www.ietf.org/html.charters/cat-charter.html

IP Security Protocol. Details specifications for IPsec:

http://www.ietf.org/html.charters/ipsec-charter.html

One-Time Password Authentication. Details standards for one-time password technologies:

http://www.ietf.org/html.charters/otp-charter.html

Public Key Infrastructure (X.509). Details Internet standards to support an X.509 PKI:

http://www.ietf.org/html.charters/pkix-charter.html

Secure Shell. Details SSH specifications:

http://www.ietf.org/html.charters/secsh-charter.html

Transport Layer Security. Specifies protocols providing security features at the Transport layer:

http://www.ietf.org/html.charters/tls-charter.html

Network Address Translation. Documents NAT requirements and limitations:

http://www.ietf.org/html.charters/nat-charter.html

Site Security Handbook. Handbook for users to create site-specific policies and procedures to deal with computer-security problems and their prevention:

http://www.ietf.org/html.charters/ssh-charter.html

Documents on the Scope and Content of Network Security Policies

RFC 2196: The Site Security Handbook. A guide created by the Internet Engineering Task Force (IETF) to develop computer security policies and procedures for sites that have systems on the Internet:

http://info.internet.isi.edu:80/in-notes/rfc/files/rfc2196.txt

A technical guide created by the National Institute of Standards and Technology (NIST) to help an organization create a coherent Internet-specific information security policy:

http://csrc.nist.gov/isptg/html/ISPTG-Contents.html

FIPS PUB-191. Created by NIST. Although it is written specifically for LANs, this publication is applicable to any computer network environment. The use of risk management is presented to help the reader determine LAN assets, to identify threats and vulnerabilities, to determine the risk of those threats to the LAN, and to determine the possible security services and mechanisms that may be used to help reduce the risk to the LAN.

http://www.itl.nist.gov/div897/pubs/fip191.htm

Note Federal Information Processing Standards Publications (FIPS PUBs) are issued by the NIST after approval by the Secretary of Commerce pursuant to Section 111(d) of the Federal Property and Administrative Services Act of 1949, as amended by the Computer Security Act of 1987, Public Law 100-235.

Incident Response Teams

NIST Special Publication (SP) 800-3, *Establishing a Computer Security Incident Response Capability (CSIRC).*

Computer Security Resource Clearinghouse (CSRC):

http://csrc.ncsl.nist.gov/topics/inchand.html

The Danish Computer Emergency Response Team provides a pointer to a number of different Computer Emergency Response Teams (CERTs) around the world:

http://www.cert.dk/other-irts/

Other Useful Sites for Security-Related Information

Electronic Privacy Information Center (EPIC):

http://epic.org/

Comprehensive archive of security-related links:

http://www.cs.purdue.edu/coast/hotlist/

Cisco Security Product Information

General information on Cisco security offerings:

http://www.cisco.com/go/security/

PIX Firewall, a standalone firewall product:

http://www.cisco.com/go/pix/

NetRanger, a network intrusion detection system:

http://www.cisco.com/go/netranger/

NetSonar, a vulnerability detection and reporting system:

http://www.cisco.com/go/netsonar/

Cisco IOS Firewall Feature Set, integrated firewall functionality for Cisco IOS software:

http://www.cisco.com/go/iosfirewall/

CiscoSecure, an access control server incorporating RADIUS and TACACS+ functionality:

http://www.cisco.com/go/ciscosecure/

Cisco IOS 12.0 Network Security. Indianapolis, IN: Cisco Press, 1999. Provides information about Cisco IOS security features.

Posted: Wed Jun 14 11:28:56 PDT 2000

Table of Contents

B

Reporting and Prevention Guidelines: Industrial Espionage and Network Intrusions

In today's high-technology environment, thefts of proprietary material and network intrusions are a major organizational threat. This appendix is designed to help organizations develop the ability to prevent such proprietary theft and network intrusion---and, when they do occur, to know how to respond to recover their property and stop further intrusions. I hope you can review this information quickly and easily, and that it will function as a check list as you review your organization's needs. If you have questions

regarding this appendix, please call or e-mail me at:

John C. Smith
Prevention and Recovery Consulting
Trade Secret Theft and Network Intrusions
Mountain View, CA 94040
(650) 964-1956
e-mail: John@JCSmithInv.com
Web site: http://www.JCSmithInv.com

The information in this appendix comes from my eight years of experience as the senior criminal investigator, High Technology Theft/Computer Crime Unit, Santa Clara County District Attorney's Office, working in high-technology crime in Silicon Valley. This appendix includes the insight I gained from investigating 50-plus trade secret/proprietary theft (industrial espionage) cases; recovering hundreds of millions of dollars' worth of stolen proprietary property; investigating more than 40 network intrusions; searching countless personal computers in various types of criminal cases; and interviewing many suspects, witnesses, victims, and other people involved in these crimes.

It has been my experience that, to determine the extent of your loss or the extent of a network intrusion, it is necessary to conduct an investigation and execute a search warrant on the suspect's workspace and/or personal computer system. We generally found more property than the victim thought had been taken. Such investigations allow investigators to search for the types of hacking tools and programs (such as backdoor logins) that may have been used on your systems.

For Immediate Problems

- *When a crime has been committed, do not confront or talk with the suspect*. If you do, you give the suspect the opportunity to hide or destroy evidence.
- *Know your options about talking with law enforcement*. Most agencies will not start an investigation unless the victim wants to do so. An official report must be filed before a search warrant can be issued.
- *Do not wait too long to call*. It is best to immediately consult with law enforcement to learn about your options. Evidence can be lost if you wait too long.

Reporting Options

- Call our office or your local law enforcement agency and make a police report. Request a search warrant to recover your property. You can use this information to file for an injunction.
- Make an official report to the federal authorities, probably the FBI.
- File a civil law suit and seek an injunction when appropriate.
- Take appropriate disciplinary action against any involved employees.
- Do nothing and hope that the problem stops before your organization suffers any substantial damage.

Conducting an Investigation

To conduct an investigation, think of Smith's Seven Step System, which consists of the following:

1. *SPEED.* The case should be handled quickly before evidence and property are destroyed.
2. *STEALTH.* The investigation must be done quietly or the suspect will learn of it.
3. *SYSTEM SECURITY.* No further damage should be allowed to your system.
4. *SECURE EVIDENCE.* Chain of possession to ensure it is admissible.
5. *SUSPICIOUS/SUSPECT EMPLOYEES.* Most thefts are done by employees.
6. *SHOW and TELL REPORTING.* Learn how to make a report understandable.
7. *SEARCH WARRANT.* Prepare and serve a warrant when necessary.

Workplace Philosophy

An organization is less likely to be victimized if it has the following characteristics:

- Has adopted security policies to protect its systems and data.
- Makes its security policies known to all who work in the organization.
- Has planned on how it will react to intrusions and losses.
- Encourages the reporting of suspicious incidents and has a method in place that makes reporting easy and confidential.
- Attempts to recover its stolen material.
- Makes it known that offenders will be criminally prosecuted.
- Has analyzed the major threats to the organization and has considered how to deal with them.
- Realizes that the major threat is probably a person authorized to be on the premises.

Organizations should continue to provide ongoing awareness training to remind everyone that the organization could be a target for the theft of proprietary data or a network intrusion.

Your plan and your working environment must be balanced. Your rules and operating instruc-tions cannot be so severe that work and creativity are restricted, yet rules and accepted security practices should convey the message that thefts, acts of vandalism, and computer misuse will not be condoned.

Management should take security seriously and allocate the resources needed to implement and inspect the correct policies. Training should be provided. Business goals (such as deadlines) should not be allowed to take precedence over security.

Most importantly, your company should develop an attitude and mind set that it is not willing to be a victim and that it will *not* tolerate people who steal from or attack its site. Law enforcement has long known that thieves and predators pick on easy and willing victims. Realize that incidents *do happen* and

can happen to your company. Your company management must also understand this fact.

Written Plan

Your written plan should be approved by corporate legal, corporate security, management, and the computer/network manager. The plan should be agreed on, be in writing, and be approved by the head of the organization.

Organizations should involve employees in developing a plan. Employees know organizational weaknesses and how to exploit them.

Identify the decision-maker who is authorized to call law enforcement. Identify who will be the day-to-day coordinator of an incident and who will work with law enforcement and attorneys. Provide for a response team that is trained to investigate network intrusions.

All managers, supervisors, and systems administrators should be very familiar with the plan and have a copy available. All employees should receive a copy of the plan or a briefing on the contents of the plan. Your plan should specify that any employee who learns of a theft or network intrusion *will not* discuss it with anyone except management, security, the legal department, or a designated person.

Remember that rumors fly at the speed of sound.

Law and the Legal Process

Know the appropriate state and federal laws. Include copies of state and federal laws with your plan. Determine your guidelines for prosecuting. Prosecution is necessary for a law enforce-ment investigation and if you want to use the search warrant process.

Know the appropriate local or federal law enforcement agency that has jurisdiction for any problems you might have. Establish the appropriate contacts. Keep names and phone numbers updated. Talk with law enforcement at least once a year. Offer tours or briefings. Know the capabilities of your law enforcement resources.

Know how long it will normally take local law enforcement and federal law enforcement to obtain a search warrant. Discuss what information or reports law enforcement will share with you. Know whether you will be able to obtain law enforcement reports for use in civil cases. Know whether you can you get reports from federal cases.

Plan for filing a civil injunction or temporary restraining order (TRO) as soon as law enforce-ment has completed the search warrant or covert investigation. Injunctions are frequently used by victims to prohibit suspects from using proprietary information that has been taken under questionable circumstances.

Computer and Network Systems

Make sure the audit or accounting functions are turned on.

Have servers in a physically secure location to prevent unauthorized access.

Control modem connections; use smart cards or a call-back system.

Make sure secure firewalls are set up and configured properly.

On a regular basis, run programs (for example, Crack, Tiger, COPS, and Satan) to check for system weaknesses.

Keep current on new programs designed to find system vulnerabilities.

Use a virus-checker program.

Have a password file in a hidden location (that is, a shadow password file).

Close holes in operating systems.

Do not allow the importation of software into the system.

Monitor the size of outgoing mail and notify the system administrator of large outgoing messages.

Track and audit company proprietary data when it is copied and printed.

Watch for the computer system behaving strangely or improperly.

Put names or hidden markers in source code---unusual code that would work only with something you have done or misspelled words.

Make timely system backups.

Keep one copy of backup tapes in a secure facility offsite.

Plan on how to handle various intrusions, such as broken accounts, system or root access, backdoor logins, sniffers, and Trojan horses.

Ensure that patches have been made to networks and that you apply the patch whenever a new one is made available. Watch CERT bulletins.

Employees

Several studies and my experience indicate that employees and other persons who are authorized to be on the company premises or who are in a trusted relationship commit most computer crimes.

Do complete background checks before hiring someone or allowing someone access to company resources.

In new employee indoctrination, stress the importance of proprietary data and that any compromise of

proprietary data will result in discipline, termination, or prosecution.

Warn against bringing in other companies' proprietary data.

Conduct thorough exit interviews.

Advise departing employees that it is against the law to take proprietary material, and that you will prosecute anyone caught taking any type of proprietary information.

Determine whether the employee who is leaving has worked on important-enough material that a letter should be sent to him or her or to the new employer reiterating the non-disclosure and confidentiality documents signed by the former employee. Letters are frequently used by companies to warn other companies when an employee has changed jobs and the former employer is concerned that the employee may divulge proprietary information.

Set up an easy-to-use system that allows employees to covertly or anonymously report suspicious behavior.

Set up a reward system for preventing loss of data or helping to recover data.

Develop a method to combat the belief by many employees that anyone who has worked on something has a right to take a copy. This feeling of ownership occurs regardless of the signing of non-disclosure agreements and ownership/invention agreements. One of the most common criminal defenses used is that the ex-employee just wanted a sample of their work.

Control and approve any articles written about the company by employees.

Educate current employees on the cost and impact to the organization---and to them personally---of the loss of proprietary information.

Do not give prospective or new employees an email account or access to their new work environment before they have officially terminated from their last employer.

Methods of Safeguarding Proprietary Material

For your proprietary material to be considered secret, you must be able to show that you took adequate steps to protect it.

In both civil and criminal cases, you must explain what steps or methods your company used to protect its property.

The following are measures that can be used to protect proprietary information:

- Require non-disclosure agreements from employees, contractors, and anyone with access to the protected material.
- Require non-employees to sign a contract describing their access to protected material before the non-employee is given any type of proprietary material.
- Conduct thorough exit interviews.
- Collect all documentation of terminating employees.

- Maintain secure and locked facilities.
- Require employees to wear badges; require visitors to wear badges and be accompanied by escorts.
- Maintain document control.
- Ensure that all documents are marked and numbered.
- Keep logs of who is issued what documents.
- Use a need-to-know policy to determine who can access proprietary material.
- Restrict on a need-to-know basis access to networks where proprietary data is kept.
- Password-protect computers and networks where important data is kept.

Document Control

Properly mark proprietary and confidential documents. The confidential markings can be minimized if they are seen on routine documents. Mark only proprietary documents, not everything.

Do not have more than two security classifications.

Have an easy-to-use accounting system in place to track who checks out and returns proprietary documents. Require that the document-control system be used and inspect its use. Have the document-control processes audited by management on a random basis.

Track printouts from the computer accounting system. Have confidential and proprietary markings automatically put on every printed proprietary document.

Track and audit downloads of computer files.

Set up a disposal method for documents when they are no longer needed.

Limit access to source code; limit physical access to documents.

Foreign/Competitor Contacts

Train employees in how to protect proprietary data when they are traveling. Discuss hazards and how employees can protect themselves or detect methods such as these:

- Microphones in hotels, meeting rooms, and transportation
- Searches of rooms and briefcases by unknown persons

Train employees in what to do when they are approached by representatives of a competitor, a foreign company, or a foreign country.

Require that employees report when they are asked to be a guest or a speaker, to serve on a committee of a foreign country, or are put in a situation of working with a person who may be collecting information. Debrief employees when they return from overseas trips.

Determine how to handle visitors who take photographs and notes while touring your facilities.

Determine how to handle employees who are asked to lunch or other social functions by competitors.

Managers and Supervisors

Managers and supervisors should be trained to recognize and report employees who manifest behavior that may lead to acts against an organization. Such behavior may include the following:

- Employees who are angry at the company or a supervisor for being passed over for promotion, for not receiving a raise, for a perceived lack of respect, and so on.
- Employees with an unusually high fixation on making large sums of money, getting promoted in a company, acquiring a lot of stock from a start-up company, and so on.
- Employees acting strangely or being spotted with suspicious people.

Management should continually reinforce that first-line managers and supervisors will often be the first to learn of unusual employee behavior and that most problems are caused by insiders.

Reporting Process---Rewards

Create an environment in which employees will report suspicious behavior or actions. Have in place an anonymous reporting or call-in process and ensure that management takes this seriously. Offer rewards for saving data in the face of thefts or attempts at theft.

Train managers, supervisors, and all staff on how to make reports and explain why it is important to react *quickly and quietly*.

Intelligence-Gathering Methods

There are many ways for people to get at confidential information:

- Dumpster diving
- Obtaining your data from other companies
- Hiring your key employees
- Sniffing data on networks
- Going through trash inside the building
- Monitoring unsecured faxes and telephones (particularly true in other countries)
- Voice gathering by using sound-directional equipment
- Foreign or competing representatives who visit or tour your facilities
- Interns or students assigned to your facilities

Look for Weak Links

Often, the employees who make the least money have the most access in a company: security personnel, maintenance personnel, and janitors. The following are possible weak links:

- Is the company contracting for services, and are those employees bonded or backgrounded?

- Don't overlook trash being put in unlocked dumpsters.
- Social engineering of unsophisticated employees who talk about passwords in front of others.
- Employees with gambling or drinking problems, or employees who hang around card clubs.
- Allowing non-employees and employees of contractors too much access to sensitive areas or documents.
- Allowing too many employees without the necessary need-to-know access to sensitive areas or documents.
- Allowing work to be done that is not understood by a supervisor or management.
- Unlimited access to copy machines or downloading of documents.
- Allowing computer data to be sent out of the company without some type of check or monitoring.
- Allowing employees to write papers or to give presentations about the company or its products without the information going through a review process.
- Not enforcing company policy.
- Allowing engineers or other technical employees to use their own equipment, computers, or notebooks.
- Not protecting customer information, strategic forecasts, or business plans.
- Not running Crack or other tools that check for network vulnerabilities.
- Not closing computer accounts of employees who have left the company.
- Proprietary documents that are not marked or that are printed from a computer without adequate proprietary notice.
- Allowing a proprietary document to be moved, downloaded, or printed from a computer network without a warning that the material is proprietary.

California State Laws

The following are the California state laws that are used in a majority of high-technology cases. They can be downloaded from this site:

http://www.leginfo.ca.gov.calaw.html

- 499c PC---Trade Secret Theft

  Trade secret means any information---including formula, pattern, compilation, program, device, method, technique, or process--- that derives independent economic value, actual or potential, from not being generally known to the public or to other persons who can obtain economic value from its disclosure or use. A felony. See the California Penal Code for complete wording.
- 502 PC---Computer (Network) Related Crimes, Illegal Intrusion

  Primarily a felony. See the California Penal Code for complete wording.

1 Accesses, alters, damages, deletes, destroys, or uses data to defraud or obtain something of value.

2 Knowingly accesses and without permission takes, copies, or makes use of any data from a computer system or a computer network.

3 Knowingly and without permission uses or causes to be used computer services. (Misdemeanor)

4 Knowingly accesses and alters, damages, deletes, or destroys any data on a computer or network.

5 Knowingly and without permission causes the disruption of computer services or denies or causes the denial of computer services to a computer, computer system, or computer network.

6 Knowingly and without permission accesses or causes to be accessed any computer, computer system, or computer network. (Misdemeanor)

7 Knowingly introduces any computer contaminant into any computer, computer system, or computer network. (Misdemeanor)

If the computer used by the suspect is located in Santa Clara County, we can prosecute even though the suspect broke into a system in another state.

- 641.3 PC---Commercial Bribery

  A felony. Any employee who solicits, accepts, or agrees to accept money or anything of value from a person other than his or her employer, other than in trust for the employer, corruptly and without the knowledge and consent of the employer, in return for using or agreeing to use his or her position for the benefit of that other person, and any person who offers or gives an employee money or anything of value under those circumstances is guilty of commercial bribery. The money or thing of value must exceed $100.

United States Code

Section 1832, Theft of Trade Secrets. Whoever, with intent to convert a trade secret that is related to or included in a product that is produced for or placed in interstate or foreign commerce, to the economic benefit of anyone other than the owner thereof, and intending or knowing that the offense will injure any owner of trade secret, knowingly (steals, copies, duplicates, sends, receives, buys, or possesses knowing it to be stolen).

Examples of Cases in Santa Clara County (Silicon Valley)

The following are some of the more serious cases of proprietary theft and network intrusions that the Santa Clara County District Attorney's Office has investigated:

- Kevin M. used the name of a victim company manager and obtained a modem account. He uploaded his own code and obtained superuser status on several systems. He then downloaded source code through cutouts and cellular phones.
- BV used cracking tools obtained on the Internet to gain system administration status at an Ivy League university. He then inserted a back-door login program into the operating system.
- RY, after leaving a company, gained access to the network through a security hole. On two occasions, he erased the manufacturing database and made hidden changes in the system. He almost stopped company operations for two days.

- MI, who wanted to make more money, gave notice and then compressed the victim company's source code. He emailed it to his account on a public provider and then to his home.
- CVD was the manager of the computer center. He used his employees to rewrite the company's source code and then sold it. He formed a company with the profit and was trying to sell the program overseas. The code was moved using modem and tape.
- Marc G. was caught trying to get on a flight back to France after working in a local software development company. He had taken enough papers to replicate that company's program. Five tar (copy) commands were found on the company's system.
- WBS, an angry employee in the defense industry, took a few papers at a time concerning a non-classified part of a proprietary project. By the time he was fired, he had an 18-inch-thick stack of papers. He also took a copy of the company's business plan. He was offering these to the victim company's competitors to get a job.
- INT wanted schematics and manufacturing/process information to help start up a new competing company. He hired a victim employee as a consultant who brought the information he needed to the new company. During a search warrant in a case over disputed source code, we found a proprietary document that would allow the replication of the victim's product. The engineer with the document said it had been given to him when he was a scientist in the Soviet Union, within six months of the publication date. He was able to retrieve it after the fall of the Iron Curtain.
- JW is an engineer who took processing data for a product and used it to obtain consulting fees and to get a job in another country. We arrested him two days before he was to leave for his new job in South America. This information may have been used as the basis of a partnership with a business in Europe.
- T & G took documents and source code. We found that T was, at the same time, also serving as the vice president of a company in Beijing. Further investigation revealed that T was sending documents to a company in Beijing.
- HT, while visiting a company with whom he had a business association, downloaded their customer database into his laptop computer and sent it to his company in Europe.
- F was employed as an engineer to develop computer instructions for manufacturing. He became angry and erased all the programs on the company computers. We recovered the programs at his home.
- AK acquired proprietary documents on his employer's new technology. He quit and obtained several jobs where it appeared he was using the documents to make himself look good and to advance in the new company.
- RC broke passwords on a network; using those accounts, he sent messages to the president of the institution trying to get the system administrators fired.
- A software engineer left the company where he developed the nucleus of a software program. In an extremely short time, he produced a similar competing product. Many lines of code are the same.
- A technician took prototype circuit boards out of new computers and sold them.
- Raj, an Indian electrical engineer, was working as a security guard in an R&D facility for one company while working in several other companies that had similar products. He had not listed his EE degree on his application for the security guard position. Raj was stopped trying to get back

into the R&D facility six months after he had walked off that job.

- A local manufacturing company, trying to do business with a Pacific Rim company, entered into a working agreement. When the local company stopped visitors from the other company from taking notes and photos of their equipment, a representative of the foreign company tried bribery to get manufacturing details. The victim did not prosecute for fear of not being able to do business in that country. A second local company discov-ered that a company from the same Pacific Rim country hired away a manager. That manager put together a team of former employees from the victim company. The team developed a duplicate product to put on the competing market in an extremely short time.

HOME CONTENTS PREVIOUS NEXT GLOSSARY FEEDBACK SEARCH HELP

Posted: Wed Jun 14 11:29:00 PDT 2000

Table of Contents

Basic Cryptography

This chapter details the basic building blocks and fundamental issues you need to understand before moving on to more complex security technologies. *Cryptography* is the basis for all secure communications; it is, therefore, important that you understand three basic cryptographic functions:

symmetric encryption, asymmetric encryption, and one-way hash functions. Most current authentication, integrity, and confidentiality technologies are derived from these three cryptographic functions. This chapter also introduces digital signatures as a practical example of how you can combine asymmetric encryption with one-way hash algorithms to provide data authentication and integrity.

Authentication, authorization, and key management issues are critical for you to understand because the compromise of either identity or secret keys is the most common form of security compromise. Authentication technologies are introduced in Chapter 2, "Security Technologies," but this chapter explores the methods of authentication, the establishment of trust domains for defining authorization boundaries, and the importance of the uniqueness of namespace.

A *key* is a digital code that can be used to encrypt, decrypt, and sign information. Some keys are kept private while others are shared and must be distributed in a secure manner. The area of key management has seen much progress in the past years; this is mainly because it makes key distribution secure and scaleable in an automated fashion. Important issues with key management are creating and distributing the keys securely. This chapter introduces some common mechanisms that are used to securely create and distribute secret and public keys. The controversial area of *key escrow* is explored to raise your awareness of what the controversy is all about and what role key escrow may play in a secure enterprise infrastructure.

Cryptography

Cryptography is the science of writing or reading coded messages; it is the basic building block that enables the mechanisms of authentication, integrity, and confidentiality. *Authentication* establishes the identity of both the sender and the receiver of information. *Integrity* ensures that the data has not been altered, and *confidentiality* ensures that no one except the sender and receiver of the data can actually understand the data.

Usually, cryptographic mechanisms use both an *algorithm* (a mathematical function) and a secret value known as a *key*. Most algorithms undergo years of scrutiny by the world's best cryptographers who validate the strength of the algorithm. The algorithms are widely known and available; it is the key that is kept secret and provides the required security. The key is analogous to the combination to a lock. Although the concept of a combination lock is well known, you can't open a combination lock easily without knowing the combination. In addition, the more numbers a given combination has, the more work must be done to guess the combination---the same is true for cryptographic keys. The more bits that are in a key, the less susceptible a key is to being compromised by a third party.

The number of bits required in a key to ensure secure encryption in a given environment can be controversial. The longer the *keyspace*---the range of possible values of the key---the more difficult it is to break the key in a brute-force attack. In a *brute-force attack*, you apply all combinations of a key to the algorithm until you succeed in deciphering the message.
Table 1-1 shows the number of keys that must be tried to exhaust all possibilities, given a specified key length.

Table 1-1: Brute Force Attack Combinations

Key Length (in bits)	Number of Combinations
40	2^{40}= 1,099,511,627,776
56	2^{56}= 7.205759403793 x 10^{16}
64	2^{64}= 1.844674407371 x 10^{19}
112	2^{112}= 5.192296858535 x 10^{33}
128	2^{128}= 3.402823669209 x 10^{38}

A natural inclination is to use the longest key available, which makes the key more difficult to break. However, the longer the key, the more computationally expensive the encryption and decryption process can be. The goal is to make breaking a key "cost" more than the worth of the information the key is protecting.

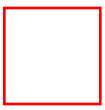
Note If confidential messages are to be exchanged on an international level, you must understand the current government policies and regulations. Many countries have controversial import and/or export regulations for encryption products based on the length of the key. This is discussed in more detail in Chapter 3, "Export Controls on Cryptography."

Three types of cryptographic functions enable authentication, integrity, and confidentiality: symmetric key encryption, asymmetric key encryption, and one-way hash functions.

Symmetric Key Encryption

Symmetric encryption, often referred to as *secret key encryption*, uses a common key and the same cryptographic algorithm to scramble and unscramble a message. Figure 1-1 shows two users, Alice and Bob, who want to communicate securely with each other. Both Alice and Bob have to agree on the same cryptographic algorithm to use for encrypting and decrypting data. They also have to agree on a common key---the secret key---to use with their chosen encryption/decryption algorithm.

Figure 1-1: Secret Key Encryption



A simplistic secret key algorithm is the Caesar Cipher. The *Caesar Cipher* replaces each letter in the original message with the letter of the alphabet *n* places further down the alphabet. The algorithm shifts the letters to the right or left (depending on whether you are encrypting or decrypting). Figure 1-2 shows Alice and Bob communicating with a Caesar Cipher where the key, *n*, is three letters. For example, the letter A is replaced with the letter D (the letter of the alphabet three places away). The steps of the Caesar Cipher are as follows:

1. Alice and Bob agree to use the Caesar Cipher to communicate and pick n=3 as the secret key.
2. Alice uses the Caesar Cipher to *encrypt* a confidential message to Bob and mails the message.
3. When he receives Alice's mail, Bob *decrypts* the message and reads the confidential message.

Figure 1-2: Encryption and Decryption Using the Caesar Cipher Algorithm

Anyone intercepting the message without knowing the secret key is unable to read it. However, you can see that if anyone intercepts the encrypted message and knows the algorithm (for example, shift letters to the right or left), it is fairly easy to succeed in a brute-force attack. Assuming the use of a 26-letter alphabet, the interceptor has to try at most 25 keys to determine the correct key.

Some secret key algorithms operate on 64-bit message blocks. Therefore, it is necessary to break up larger messages into 64-bit blocks and somehow chain them together. The following chaining mechanisms can also offer additional protection from tampering with the transmitted data.

Four common modes exist in which each mode defines a method of combining the plaintext (the message that is not encrypted), the secret key, and the ciphertext (the encrypted text) to generate the stream of ciphertext that is actually transmitted to the recipient. These four modes are:

- Electronic CodeBook (ECB)
- Cipher Block Chaining (CBC)
- Cipher FeedBack (CFB)
- Output FeedBack (OFB)

The ECB chaining mechanism encodes each 64-bit block independently---but uses the same key. This weakness can easily be exploited by an avid snooper who is interested only in changes in information and not the exact content. For example, consider someone snooping a certain employee's automatic payroll transactions to a bank. Assuming that the amount is the same for each paycheck, each ECB-encoded ciphertext message would appear the same. However, if the ciphertext changes, the snooper could conclude that the payroll recipient received a raise and perhaps was promoted.

The remaining three algorithms (CBC, CFB, and OFB) have inherent properties that add an element of

randomness to the encrypted messages. If you send the same plaintext block through one of these three algorithms, you get back different ciphertext blocks each time. This is accomplished by using different encryption keys or an *initialization vector* (IV). An IV is an encrypted block of random data used as the first 64-bit block to begin the chaining process. The IV is implementation specific but can be taken from a timestamp or some other random bit of data. If a snooper were listening to the encrypted traffic on the wire, and you sent the same message ten times using a different key or IV to encrypt the data, it would look like a different message each time. The snooper would gain virtually no information.

Most secret key algorithms will use one of these four modes to provide additional security for the transmitted data. Here are some of the more common secret key algorithms used today:

- Data Encryption Standard (DES)
- 3DES (read "triple DES")
- Rivest Cipher 4 (RC-4)
- International Data Encryption Algorithm (IDEA)

DES is the most widely used encryption scheme today. It operates on 64-bit message blocks. The algorithm uses a series of steps to transform 64-input bits into 64-output bits. In its standard form, the algorithm uses 64-bit keys---of which 56-bits are chosen randomly. The remaining 8 bits are parity bits (one for each 7-bit block of the 56-bit random value). DES is widely employed in many commercial applications today and can be used in all four modes: ECB, CBC, CFB, and OFB. Generally, however, DES operates in either the CBC mode or the CFB mode.

Note 40-bit DES is standard DES with all but 40 bits disclosed by the implementation of the communications mechanism. For example, you can implement 40-bit DES by prefacing each message with the same 24 bits of the DES key used to encrypt the data. 40-bit DES exists solely as an artifact of U.S. government export controls; there is no technical reason you should not use standard DES at all times.

3DES is an alternative to DES that preserves the existing investment in software but makes a brute-force attack more difficult. 3DES takes a 64-bit block of data and performs the operations of encrypt, decrypt, and encrypt. 3DES can use one, two, or three different keys. The advantage of using one key is that, with the exception of the additional processing time required, 3DES with one key is the same as standard DES (for backward compatibility). 3DES is defined only in ECB mode mainly for performance reasons: It compromises speed for the sake of a more secure algorithm. Both the DES and 3DES algorithms are in the public domain and freely available.

RC-4 is a proprietory algorithm invented by Ron Rivest and marketed by RSA Data Security. It is used often with a 128-bit key although its key size can vary. It is unpatented but is protected as a trade secret---although it was leaked to the Internet in September 1994. Because the U.S. government allows it to be exported when using secret key lengths of 40 bits or less, some implementations use a very short key length.

IDEA was developed to replace DES. It also operates on 64-bit message blocks but uses a 128-bit key. As with DES, IDEA can operate in all four modes: ECB, CBC, CFB, and OFB. IDEA was designed to be efficient in both hardware and software implementations. It is a patented algorithm and requires a license for commercial use.

Note References to specific algorithms are given to get you familiar with which algorithms pertain to which basic encryption concepts. Because most of the cryptanalytical and performance comparisons are useful more for implementers of the technology, they are not deeply explored here. References for more in-depth studies are given in Appendix A, "Sources of Technical Information."

Secret key encryption is most often used for data confidentiality because most symmetric key algorithms have been designed to be implemented in hardware and have been optimized for encrypting large amounts of data at one time. Challenges with secret key encryption include the following:

- Changing the secret keys frequently to avoid the risk of compromising the keys
- Securely generating the secret keys
- Securely distributing the secret keys

A commonly used mechanism to derive and exchange secret keys securely is the Diffie-Hellman algorithm. This algorithm is explained later in this chapter in the "Key Management" section.

Asymmetric Encryption

Asymmetric encryption is often referred to as *public key encryption*. It can use either the same algorithm, or different but complementary algorithms to scramble and unscramble data. Two different, but related, key values are required: a public key and a private key. If Alice and Bob want to communicate using public key encryption, both need a public key and private key pair (see Figure 1-3). Alice has to create her public key/private key pair, and Bob has to create his own public key/private key pair. When communicating with each other securely, Alice and Bob use different keys to encrypt and decrypt data.

Figure 1-3: Public Key Encryption

Some of the more common uses of public key algorithms are listed here:

- Data integrity
- Data confidentiality
- Sender nonrepudiation
- Sender authentication

Data confidentiality and sender authentication can be achieved using the public key algorithm. Figure 1-4 shows how data integrity and confidentiality is provided using public key encryption.

The following steps have to take place if Alice and Bob are to have *confidential data exchange*:

1. Both Alice and Bob create their individual public/private key pairs.

2. Alice and Bob exchange their public keys.

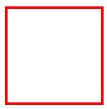
3. Alice writes a message to Bob and uses *Bob's public key* to encrypt her message. Then, she sends the encrypted data to Bob over the Internet.

4. Bob uses *his private key* to decrypt the message.

5. Bob writes a reply, encrypts the reply with *Alice's public key*, and sends the encrypted reply over the Internet to Alice.

6. Alice uses *her private key* to decrypt the reply.

Figure 1-4: Ensuring Data Integrity and Confidentiality with Public Key Encryption



Data confidentiality is ensured when Alice sends the initial message because only Bob can decrypt the message with his private key. Data integrity is also preserved because, to modify the message, a malicious attacker would need Bob's private key again. Data integrity and confidentiality is also ensured for the reply because only Alice has access to her private key and is the only one who can modify or decrypt the reply with her private key.

However, this exchange is not very reassuring because it is easy for a third party to pretend to be Alice and send a message to Bob encrypted with Bob's public key. The public key is, after all, widely available. Verification that it was Alice who sent the initial message is important. Figure 1-5 shows how public key cryptography resolves this problem and provides for sender authentication and non-repudiation.

Figure 1-5: Sender Authentication and Nonrepudiation Using Public Key Encryption

The following steps have to take place if Alice and Bob are to have an *authenticated data exchange*:

1. Both Alice and Bob create their public/private key pairs.

2. Alice and Bob exchange their public keys.

3. Alice writes a message for Bob, uses *her private key* to encrypt the message, and then sends the encrypted data over the Internet to Bob.

4. Bob uses *Alice's public key* to decrypt the message.

5. Bob writes a reply, encrypts the reply with *his private key*, and sends the encrypted reply over the Internet to Alice.

6. Alice uses *Bob's public key* to decrypt the reply.

An authenticated exchange is ensured because only Bob and Alice have access to their respective private keys. Bob and Alice meet the requirement of non-repudiation---they cannot later deny sending the given message if their keys have not been compromised. This, of course, lends itself to a hot debate on how honest Bob and Alice are; they can deny sending messages by simply stating that their private keys have been compromised.

If we want to use public key cryptography to perform an authenticated exchange as well as ensure data integrity and confidentiality, double encryption needs to occur. Alice would first encrypt her confidential message to Bob with Bob's public key and then encrypt again with her private key. Anyone would be able to decrypt the first message to get the embedded ciphertext but only Bob would be able to decrypt the ciphertext with his private key.

Note A crucial aspect of asymmetric encryption is that the private key *must* be kept private. If the private key is compromised, an evil attacker can impersonate you and send and receive messages *as* you.

The mechanisms used to generate these public/private key pairs are complex, but they result in the generation of two very large random numbers, one of which becomes the public key and the other becomes the private key. Because these numbers *as well as their product* must adhere to stringent mathematical criteria to preserve the uniqueness of each public/private key pair, generating these numbers is fairly processor intensive.

Note Key pairs are not guaranteed to be unique by any mathematical criteria. However, the math ensures that no *weak keys* are generated.

Public key encryption algorithms are rarely used for data confidentiality because of their performance constraints. Instead, public key encryption algorithms are typically used in applications involving authentication using digital signatures and key management.

Some of the more common public key algorithms are the Ron Rivest, Adi Shamir, and Leonard Adleman (RSA) algorithm and the El Gamal algorithm.

Hash Functions

A *hash function* takes an input message of arbitrary length and outputs fixed-length code. The fixed-length output is called the *hash*, or the *message digest*, of the original input message. If an algorithm is to be considered cryptographically suitable (that is, secure) for a hash function, it must exhibit the following properties:

- It must be consistent; that is, the same input must always create the same output.
- It must be random---or give the appearance of randomness---to prevent guessing of the original message.

- It must be unique; that is, it should be nearly impossible to find two messages that produce the same message digest.
- It must be one way; that is, if you are given the output, it must be extremely difficult, if not impossible, to ascertain the input message.

One-way hash functions are typically used to provide a *fingerprint* of a message or file. Much like a human fingerprint, a hash fingerprint is unique and thereby proves the integrity and authenticity of the message.

Consider the example shown in Figure 1-6; Alice and Bob are using a one-way hash function to verify that no one has tampered with the contents of the message during transit.

Figure 1-6: Using a One-Way Hash Function for Data Integrity

The following steps have to take place if Alice and Bob are to keep the integrity of their data:

1. Alice writes a message and uses the message as input to a one-way hash function.
2. The result of the hash function is appended as the fingerprint to the message that is sent to Bob.
3. Bob separates the message and the appended fingerprint and uses the message as input to the same one-way hash function that Alice used.
4. If the hashes match, Bob can be assured that the message was not tampered with.

The problem with this simplistic approach is that the fingerprint itself could be tampered with and is subject to the man-in-the-middle attack. The *man-in-the-middle attack* refers to an entity listening to a believed secure communication and impersonating either the sender or receiver. To effectively use hash functions as fingerprints, you can combine them with public key technology to provide digital signatures, which are discussed in the next section, "Digital Signatures."

Common hash functions include:

- Message Digest 4 (MD4) algorithm
- Message Digest 5 (MD5) algorithm
- Secure Hash Algorithm (SHA)

MD4 and MD5 were designed by Ron Rivest of MIT. SHA was developed by the National Institute of Standards and Technology (NIST). MD5 and SHA are the hash functions used most often in current security product implementations---both are based on MD4. MD5 processes its input in 512-bit blocks and produces a 128-bit message digest. SHA also processes its input in 512-bit blocks but produces a 160-bit message digest. SHA is more processor intensive and may run a little more slowly than MD5.

Digital Signatures

A *digital signature* is an encrypted message digest that is appended to a document. It can be used to confirm the identity of the sender and the integrity of the document. Digital signatures are based on a combination of public key encryption and one-way secure hash function algorithms. Figure 1-7 shows an example of how to create a digital signature.

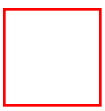
Figure 1-7: Creating a Digital Signature

The following steps must be followed for Bob to create a digital signature:

1. Bob creates a public/private key pair.
2. Bob gives his public key to Alice.
3. Bob writes a message for Alice and uses the document as input to a one-way hash function.
4. Bob encrypts the output of the hash algorithm, the message digest, with his private key, resulting in the digital signature.

The combination of the document and the digital signature is the message that Bob sends to Alice. Figure 1-8 shows the verification of the digital signature.

Figure 1-8: Verifying a Digital Signature



On the receiving side, these are the steps that Alice follows to verify that the message is indeed from Bob---that is, to verify the digital signature:

1. Alice separates received message into the original document and the digital signature.
2. Alice uses Bob's public key to decrypt the digital signature, which results in the original message digest.
3. Alice takes the original document and uses it as input to the same hash function Bob used, which results in a message digest.

4. Alice compares both of the message digests to see whether they match.

If Alice's calculation of the message digest matches Bob's decrypted message digest, the integrity of the document as well as the authentication of the sender are proven.

Note The initial public key exchange must be performed in a trusted manner to preserve security. This is critical and is the fundamental reason for the need for digital certificates. A digital certificate is a message that is digitally signed with the private key of a trusted third party stating that a specific public key belongs to someone or something with a specified name and set of attributes. If the initial public key exchange wasn't performed in a trusted manner, someone could easily impersonate a given entity.

Digital signatures do not provide confidentiality of the message contents. However, it is frequently more imperative to produce proof of the originator of a message than to conceal the contents of a message. It is plausible that you could want authentication and integrity of messages without confidentiality, as in the case where routing updates are passed in a core network. The routing contents may not be confidential, but it's important to verify that the originator of the routing update is a trusted source. An additional example of the importance of authenticating the originator of a message is in online commerce and banking transactions, where proof of origin is imperative before acting on any transactions.

Some of the more common public key digital signature algorithms are the Ron Rivest, Adi Shamir, and Leonard Adleman (RSA) algorithm and the Digital Signature Standard (DSS) algorithm. DSS was proposed by NIST and is based on the El Gamal public key algorithm. Compared to RSA, DSS is faster for key generation and has about the same performance for generating signatures but is much slower for signature verification.

Authentication and Authorization

Because authentication and authorization are critical parts of secure communications, they must be emphasized. *Authentication* establishes the identity of the sender and/or the receiver of information. Any integrity check or confidential information is often meaningless if the identity of the sending or receiving party is not properly established.

Authorization is usually tightly coupled to authentication in most network resource access requirements. Authorization establishes what you are allowed to do once you've identified yourself (it is also called *access control*, *capabilities*, and *permissions*). It can be argued that authorization does not always require *a priori* authentication, but in this book, authentication and authorization are tightly coupled; authorization usually follows any authentication procedure.

Issues related to authentication and authorization include the robustness of the methods used in verifying an entity's identity, the establishment of trusted domains to define authorization boundaries, and the requirement of uniqueness in namespace.

Methods of Authentication

All methods of authentication require you to specify who or what you are and to relay appropriate credentials to *prove* that you are who you say you are. These credentials generally take the form of

something you know, something you have, or something you are. *What you know* may be a password. *What you have* could be a smart card. *What you are* pertains to the field of biometrics, in which sophisticated equipment is used to scan a person's fingerprint or eye or to recognize a person's voice to provide authentication.

Authentication technologies are discussed in detail in Chapter 2, "Security Technologies." Here, the important element is to recognize that different mechanisms provide authentication services with varying degrees of certainty. Choosing the proper authentication technology largely depends on the location of the entities being authenticated and the degree of trust placed in the particular facets of the network.

Trust Models

Trust is the firm belief or confidence in the honesty, integrity, reliability, justice, and so on of another person or thing. *Authorization* is what you are allowed to do once your identity is established. All secure systems must have a framework for an organizational policy for authorization---this framework is called a *trust model*.

If something is difficult to obtain in a dishonest manner or is difficult to forge, we have inherent trust in that system. An example is the title to a car: This document is used as proof of ownership of a car because it is difficult to forge. It is this proof that authorizes a person to resell his or her car with the relative certainty that the car is actually his or hers to sell.

In the network world, trust models can be very complex. Suppose that we have a large corporation with a number of different affiliated departments---the research department, the marketing department, and the payroll department. These individual departments could structure their networks autonomously but with a spirit of cooperation. Each department sets up a trusted intermediary, which is the entity that keeps all the authentication and authorization information for the employees in that department (see Figure 1-9).

Figure 1-9: Trusted Intermediaries for Individual Corporate Departments

When an executive member of the research department wants to access a document off one of the research servers, he or she is authenticated by the research department's authentication/authorization server. Now if that same executive wants to access salary information for his or her employees, there must be a mechanism for authenticated and authorized access to the payroll department. Instead of each department server having separate account information for every user (this arrangement could become an administrative nightmare with large numbers of users), it may be necessary to create *groups* with *inherited trust*. For example, you can create an executive group on the payroll server that permits any executive member of the company to access payroll information.

Delegation of trust refers to giving someone or something permission to act on your behalf. If the

executive from the research department went on vacation and left someone else in charge, this individual could have permission to act on the executive's behalf to carry out a salary modification. When the executive returns, the authorization must be revoked because it was granted only on a temporary basis.

The difficulty in many trust models is deciding who to trust. Weighting risk factors (the amount of damage that can be done if trust is inappropriately placed) and having adequate mechanisms to deal with misplaced trust should be a part of every corporate security policy. Creating a security policy is discussed in more detail in Chapter 6, "Design and Implementation of the Corporation Security Policy."

Note It is important to recognize that *trust* does not mean *implicit* trust. You should have a trust model that works in high probability, but you must verify the trust relationships and put in place checks to verify that information has not been compromised. As Ronald Reagan once said, "Trust, but verify."

Name Space

In every trust domain, where we define authorization boundaries, it is important that a unique identifier to identify what is being acted on exists. At first, creating unique identifiers may seem trivial, but as the size and numbers of trust domains grow, the problem can get very complex.

Take the simple scenario of a typical enterprise network. Company A is a small start-up enterprise, and employees use their first names as login IDs. Company B decides to acquire Company A. Now there are a number of employees who have the same login ID. As you know, the IDs must be unique to preserve authentication and authorization rights. Typically, companies have a naming convention for this situation: Login IDs consist of the user's first initial and last name; any duplicates use the first two initials and last name, or the first three initials and last name, and so on.

Many large corporations create standard naming conventions for all entities that may require authentication (for example, employees and any and all network infrastructure devices). The concept of an object identifier has been used in the industry to veer away from the common notion that an entity has to have a specific name. The object identifier also can be an employee badge number, an employee social security number, a device IP address, a MAC address, or a telephone number. These object identifiers must be unique within given trust domains.

Key Management

Key management is a difficult problem in secure communications, mainly because of social rather than technical factors. Cryptographically secure ways of creating and distributing keys have been developed and are fairly robust. However, the weakest link in any secure system is that humans are responsible for keeping secret and private keys confidential. Keeping these keys in a secure place and not writing them down or telling other people what they are is a socially difficult task---especially in the face of greed and anger. Some people find it quite difficult not to divulge a secret in exchange for a million dollars or to get back at a seemingly unfair employer. Other people do not take secure procedures seriously---sometimes considering them just a nuisance---and are careless in keeping keys private. The human factor will always be an issue that necessitates sufficient checks to ensure that keys have not been compromised.

Creating and Distributing Secret Keys

For a small number of communicating entities, it is not unreasonable to create a key and manually deliver it. However, in most wide-scale corporations, this mechanism is awkward and outdated. Because secret key encryption is often used in applications requiring confidentiality, it is reasonable to assume that there may exist a secret key per session, a *session* being any single communication data transfer between two entities. For a large network with hundreds of communication hosts, each holding numerous sessions per hour, assigning and transferring secret keys is a large problem. Distributing the keys is often performed through centralized key distribution centers or through public-key algorithms that establish secret keys in a secure distributed fashion.

The centralized key distribution model relies on a trusted third party, the *Key Distribution Center (KDC)*, which issues the session keys to the communicating entities (see Figure 1-10).

Figure 1-10: Distributing Keys Through a Key Distribution Center

The centralized distribution model requires that all communicating entities have a shared secret key with which they can communicate with the KDC confidentially. The problem of how to distribute this shared secret key to each of the communicating nodes still exists, but it is much more scaleable. The KDC is manually configured with every shared key, and each communi-cating node has its corresponding shared key configured. Keys can be distributed physically to employees when they get an employee badge or to devices from the IS department as part of the initial system setup.

Note If a device is given the key, then anyone using that device may be authorized for accessing certain network resources. In this age of mobile hosts, it is good practice to authenticate a device as well as the user using the device to access network resources.

A common method used to create secret session keys in a distributed manner is the *Diffie-Hellman algorithm*. The Diffie-Hellman algorithm provides a way for two parties to establish a shared secret key that only those two parties know---even though they are communicating over an insecure channel. This secret key is then used to encrypt data using their favorite secret key encryption algorithm. Figure 1-11 shows how the Diffie-Hellman algorithm works.

Figure 1-11: Establishing Secret Keys Using the Diffie-Hellman Algorithm

The following steps are used in the Diffie-Hellman algorithm:

1. Alice initiates the exchange and transmits two large numbers (p and q) to Bob.
2. Alice chooses a random large integer X_A and computes the following equation:

$Y_A = (q^{XA}) \bmod p$

3. Bob chooses a random large integer X_B and computes this equation:

$Y_B = (q^{XB}) \bmod p$

4. Alice sends Y_A to Bob. Bob sends Y_B to Alice.
5. Alice computes the following equation:

$Z = (Y_B)^{XA} \bmod p$

6. Bob computes this equation:

$Z' = (Y_A)^{XB} \bmod p$

The resulting shared secret key is as follows:

$Z = Z' = q^{(XAXB)} \bmod p$

The security of Diffie-Hellman relies on two very difficult mathematical problems:

- Any eavesdropper has to compute a discrete logarithm to recover X_A and X_B (that is, the eavesdropper has to figure out X_A from seeing q^{XA} or figure out X_B from seeing q^{XB}).
- Any eavesdropper has to factor large prime numbers---numbers on the order of 100 to 200 digits can be considered *large*. Both p and q should be large prime numbers and (p-1)/2 should be prime.

For the reader interested in a more detailed mathematical explanation, see the following sidebar.

The Diffie-Hellman exchange is subject to a man-in-the-middle attack because the exchanges themselves are not authenticated. To circumvent this problem, the Diffie-Hellman exchange is used with a public-key algorithm to ensure authentication and integrity in the exchange.

More About the Diffie-Hellman Algorithm

Two basic mathematical transforms are necessary to understand the Diffie-Hellman exchange. The first transform to understand is a property of exponents:

$(n^x)^y = n^{xy} = n^{yx} = (n^y)^x$

If you substitute n=2, x=2, and y=3 to get the following for $(n^x)^y$:

$(2^2)^3 = 4^3 = 4x4x4 = 64$

You can swap the exponents x and y to get the same results:

$(2^2)^3 = (2^3)^2 = 2^6 = 2x2x2x2x2x2 = 64$

Finally, you can swap the exponents x and y in the original expression $(n^y)^x$ and get the same results:

$(2^3)^2 = 8^2 = 8x8 = 64$

This transform is important to the Diffie-Hellman algorithm because the two endpoints of the encrypted transmission exchange use this concept to scramble their private keys and exchange the keys as exponents. If Alice and Bob are the two endpoints, and Alice chooses the private key A while Bob chooses the private key B, and they both know a number q, they can do the following:

1. Alice sends her public key q^A to Bob.
2. Bob sends his public key q^B to Alice.
3. Alice computes $(q^B)^A$.
4. Bob computes $(q^A)^B$.

Now Bob and Alice both know the secret number q^{AB} and can use it as a session key.

The trouble is that anyone else who knows q can calculate A and B, and will know q^{AB} as well, and they can decode the ciphertext encrypted with the key. That's where the next mathematical operation comes in.

The operator *modulus*, or *mod*, is used quite regularly in number theory and some other branches of mathematics. Remember from your arithmetic classes that any integer can be divided by any other integer, to produce a quotient and a remainder. This is usually expressed as follows:

x ÷ y = (quotient,remainder)

Here's an example:

5 ÷ 3 = (1,2)

Usually, we are more interested in the quotient than in the remainder when we perform a division operation. In modular arithmetic, however, the remainder is the important part:

x mod y =r

Here's an example:

5 mod 3 = 2

One of the interesting things to notice is that there are an infinite number of numbers x such that x mod 3 equals 2. Substitute these values for x to prove that there are many xs to solve the equation x mod 3 = 2: 2, 5, 8, 11, 14, 17, 20, and so on.

How does modular arithmetic help Bob and Alice? If they share another number, p, then instead of using q^A and q^B as their public keys, they can exchange (q^A mod p) and (q^B mod p) as public keys without risk

of exposing their private keys, even if q and p are well-known numbers, because this statement is true:

$(q^A \bmod p)^B = (q^B \bmod p)^A$

Because many numbers equal $q^X \bmod p$, many computations must be performed to find x. To complicate the process of finding x even further, q and p should be large *prime* numbers and (p-1)/2 should be prime.

A *prime number* is a number that has only 1 and itself as factors. The number of integers over 100 digits long and known with absolute certainty to be prime is small and, among the mathematical community, very well known. With the choice between using well-known, proven, large prime numbers and the immense task of proving for yourself that the numbers p and q you are choosing are, in fact, primes, you have to compromise: You choose numbers that are relatively prime. A *relatively prime* number is one that is *fairly likely* to be prime. This modification of the rule allows you to choose numbers that are still computationally difficult but that can be generated in much less time than what is needed to prove that the numbers are absolute primes.

Creating and Distributing Public Keys

For public key algorithms, creating the public/private key pairs is complex. The pairs adhere to stringent rules as defined by varying public-key algorithms to ensure the uniqueness of each public/private key pair. Uniqueness is "statistically" guaranteed, that is, the odds of two identical keys being generated independently are astronomical. The complexity associated with generating public/private key pairs is the creation of sets of parameters that meet the needs of the algorithm (for example, primality for RSA and many other algorithms).

Note It is ideal for the end user (the person or thing being identified by the key) to generate the key-pair themselves. The private key should *never* leave the end user's possession. In corporate environments where this may not be practical or where key escrow is required, different rules apply. But all technical solutions should attempt self-generation as the first goal of a design architecture so that the private key is known only to the entity creating the key pair.

The problem is how you can distribute the public keys in a secure manner and how you can trust the entity that gives you the key. For a small number of communicating parties, it may be manageable to call each other or to meet face to face and give out your public key.

It should *never* be taken on faith that a public key belongs to someone. Many organizations today have so-called *key-signing parties*, a time in which people get together in the same room and exchange respective public keys. Someone in the room will know for sure that a person is who they say they are; however, it may be necessary to provide proof of identity with a drivers license or passport.

The key-signing parties are necessary when there is a lack of a trusted third party. A more scaleable approach is to use digital certificates to distribute public keys. Digital certificates require the use of a trusted third party---the *certificate authority*.

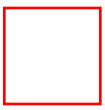
Note There is no need for complex key distribution methods of a public key cryptosystem to ensure data confidentiality. The security of the data encrypted with the public key doesn't depend on the authenticity

of the person advertising the public key. A public key cryptosystem encrypts a message just as strongly with a public key that is widely known by broadcasting it as it does with a public key obtained from a trusted certificate authority. Key distribution in public key systems is a problem only if you want true authentication of the person claiming to be associated with the public key.

Digital Certificates

A *digital certificate* is a digitally signed message that is typically used to attest to the validity of a public key of an entity. Certificates require a common format and are largely based on the ITU-T X.509 standard today. Figure 1-12 shows an example of a digital certificate format using the X.509 standard.

Figure 1-12: The X.509 Certificate Format



The general format of an X.509 certificate includes the following elements:

- Version number
- Serial number of certificate
- Issuer algorithm information
- Issuer of certificate
- Valid to/from date
- Public key algorithm information of the subject of the certificate
- Digital signature of the issuing authority

Digital certificates are a way to prove the validity of an entity's public key and may well be the future mechanism to provide single login capabilities in today's corporate networks. However, this technology is still in its infancy as far as deployment is concerned. Much of the format of certificates has been defined, but there is still the need to ensure that certificates are valid, manageable, and have consistent semantic interpretation.

Certificate Authorities

The *certificate authority (CA)* is the trusted third party that vouches for the validity of the certificate. It is up to the CA to enroll certificates, distribute certificates, and finally to remove (revoke) certificates when the information they contain becomes invalid. Figure 1-13 shows how Bob can obtain Alice's public key in a trusted manner using a CA.

Figure 1-13: Obtaining a Digital Certificate Through a Certificate Authority

Assume that Alice has a valid certificate stored in the CA and that Bob has securely obtained the CA's public key. The steps that Bob follows to obtain Alice's public key in a reliable manner are as follows:

1. Bob requests Alice's digital certificate from the CA.
2. The CA sends Alice's certificate, which is signed by the CA's private key.
3. Bob receives the certificate and verifies the CA's signature.
4. Because Alice's certificate contains her public key, Bob now has a "notarized" version of Alice's public key.

This scheme relies on the CA's public key to be distributed to users in a secure way. Most likely, this occurs using an out-of-band mechanism. There is still much debate over who should maintain CAs on the Internet. Many organizations (including financial institutions, government agencies, and application vendors) have expressed interest in offering certificate services. In all cases, it's a decision based on trust. Some corporations may want to control their own certificate infrastructure, and others may choose to outsource the control to a trusted third party.

The issues still being finalized in the industry include how to perform efficient certificate enrollment, how to revoke certificates, and how to handle cross-certifications in CA hierarchies. These issues are discussed in more detail in Chapter 2 in the section "Public Key Infrastructure and Distribution Models."

Key Escrow

Key escrow is the notion of putting a confidential secret key or private key in the care of a third party until certain conditions are fulfilled. This, in itself, is not a bad idea because it is easy to forget a private key, or the key may become garbled if the system it is stored on goes berserk. The controversy revolves around which keys should be in escrow and who becomes the trusted third party who has access to confidential keys while still protecting the privacy of the owners of the keys.

By far the most controversial key escrow issue surrounds whether cryptosystems should be developed to have a back door for wire-tapping purposes. The U.S. government for one would like secret keys and private keys to be made available to law and government officials for wire-tapping purposes. Many leading security and cryptography experts have found flaws in cryptographic systems that support key recovery. All the current algorithms operate on the premise that the private and secret keys cannot be compromised (unless they are written down or conveyed). Key recovery goes against all these assumptions.

The Business Case

In a corporate environment, many business needs for key escrow exist. It would not seem unreasonable for a corporation to keep in escrow keys used to encrypt and decrypt corporate secrets. The corporation must make a business decision about which kinds of traffic requires encryption and which information is critical to be able to retrieve. Typically, the encryption/decryption is performed at the application level; the keys used can be offered to trusted key escrow personnel. In all cases, the business keeps all parts of a key and the cryptosystem private within the business. No external escrow agent is needed.

The Political Angle

Government policy is still being defined for key escrow. A technical solution initially proposed by NIST and the NSA during the Bush administration was a new tamper-proof encryption chip called the *Clipper chip*. The algorithm it used contained a superkey---essentially a law enforcement agency field in the key. Each Clipper chip is unique and has a key field tied to the chip's serial number. The FBI, supposedly only with a court-ordered warrant, could use the superkey to open up your message. Matt Blaze, Principal Research Scientist at AT&T Laboratories, and others showed that the Clipper chip is not secure; the Clipper proposals have mostly been cast aside.

Now the government is back to brute-force escrow: You give your private key or keys to the escrow agency. The government has "compromised" by allowing in its proposal that the escrow agency can be a private business that has been "certified" by the U.S. government.

The Clinton administration continues to pursue a policy of key-recovery both inside the United States and abroad. An extensive study on the risks of key recovery mechanisms has recently been conducted by a group of leading computer scientists and cryptographers. This report attempts to outline the technical risks, costs, and implications of deploying systems that provide government access to encryption keys. You can find this report at http://www.crypto.com/key_study.

Updates on the U.S. government's position on key escrow can be found at http://www.cdt.org/crypto/ and http://www.epic.org.

The Human Element

Aside from the political and business problems with government key escrow (who wants to buy a cryptosystem for which you know someone else has the keys?), there is the critical human element to key escrow. Assume that there is a government key escrow system in which all keys are escrowed with a very few "trusted" agents. Further assume that a large amount of commerce, trade, banking, currency transfer, and so on is performed on these escrowed cryptosystems.

The equivalent of a huge pot of gold is now concentrated in a few, well-known places: the escrow agencies. All you have to do is get a few escrowed keys, tap in to some secure banking or currency transfer sessions, and you can quickly become a very wealthy thief. There is no need to spoof an encrypted session or spoof a wire transaction to put a lot of money in an off-shore bank account. In the world of finance and banking, having prior knowledge of significant events coupled with fully legitimate investments or trading moves in the open market can make you extremely wealthy without having to resort to anything more than mere eavesdropping on what are thought to be "secure" channels.

Greed and anger are the issues that most severely weaken a cryptosystem. If a large amount of wealth is tied up in one place (the key escrow system), a foreign government or economic terrorist would conceivably offer a large price to escrow agency employees. In the example of a compromised escrowed key being used to get rich in open markets with insider knowledge, an unscrupulous person could offer an escrow agent a million dollars as well as a percentage of the gains. In this way, the more keys the employee reveals, the more money he or she makes. Greed can be a major factor in causing the entire escrowed key system to crumble. It is more because of human reasons than technical or legal ones that escrowed encryption is largely not workable.

Summary

This chapter has explored many fundamental security concepts. The intent was to provide you with a precursory understanding of three basic cryptographic functions: symmetric encryption, asymmetric encryption, and one-way hash functions. You also learned how these cryptographic functions can be used to enable security services such as authentication, integrity, and confidentiality.

In many systems today, end-user or device authentication can be established using public key technology, where the public keys are distributed in some secure manner, possibly through the use of digital certificates. Digital signatures are created through the use of hash functions and ensure the integrity of the data within the certificate. Data confidentiality is typically achieved through the use of some secret key algorithm where the Diffie-Hellman exchange is used to derive the shared secret key used between the two communicating parties.

Authentication and authorization issues were discussed. It is important to recognize that different mechanisms provide authentication services with varying degrees of certainty. Choosing the proper authentication technology largely depends on the location of the entities being authenticated and the degree of trust placed in the particular facets of the network.

Also, the issues of key management systems were explored. How to effectively generate and distribute cryptographic keys will probably change over time and will require a periodic assessment. The human factor will always be an issue that necessitates sufficient checks to ensure that keys have not been compromised.

Posted: Wed Jun 14 11:30:30 PDT 2000

Table of Contents

Copyright and License Information

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

Portions of online documents can be copied and pasted to your electronic mail or word-processing applications for your personal use only, but cannot be distributed to third parties. In no event may you copy or use this information for any commercial purposes except the operation of Cisco products and you may not transmit this information to third parties without Cisco's consent.

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

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.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED 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.

AtmDirector, AutoConnect, AutoRoute, AXIS, BPX, Catalyst, CD-PAC, CiscoAdvantage, CiscoFusion, Cisco IOS, the Cisco IOS logo, CiscoLink, CiscoPro, the CiscoPro logo, CiscoRemote, the CiscoRemote logo, CiscoSecure, Cisco Systems, CiscoView, CiscoVision, CiscoWorks, ClickStart, ControlStream, EdgeConnect, EtherChannel, FairShare, FastCell, FastForward, FastManager, FastMate, FastPADlmp, FastPADmicro, FastPADmp, FragmentFree, FrameClass, Fulcrum INS, IGX, Impact, Internet Junction, JumpStart, LAN^{2}LAN Enterprise, LAN^{2}LAN Remote Office, LightSwitch, MICA, NetBeyond, NetFlow, Newport Systems Solutions, Packet, PIX, Point and Click Internetworking, RouteStream, Secure/IP, SMARTnet, StrataSphere, StrataSphere BILLder, StrataSphere Connection Manager, StrataSphere Modeler, StrataSphere Optimizer, Stratm, StrataView Plus, StreamView, SwitchProbe, SwitchVision, SwitchWare, SynchroniCD, The Cell, The FastPacket Company, TokenSwitch, TrafficDirector, Virtual EtherSwitch, VirtualStream, VlanDirector, Web Clusters, WNIC, Workgroup Director, Workgroup Stack, and XCI are trademarks; Access by Cisco, Bringing the Power of Internetworking to Everyone, Enter the Net with MultiNet, and The Network Works. No Excuses. are service marks; and Cisco, the Cisco Systems logo, CollisionFree, Combinet, EtherSwitch, FastHub, FastLink, FastNIC, FastPacket, FastPAD, FastSwitch, ForeSight, Grand, Grand Junction, Grand Junction Networks, the Grand Junction Networks logo, HSSI, IGRP, IPX, Kalpana, the Kalpana logo, LightStream, MultiNet, MultiWare, OptiClass, Personal Ethernet, Phase/IP, RPS, StrataCom, TGV, the TGV logo, and UniverCD are registered trademarks of Cisco Systems, Inc. All other trademarks, service marks, registered trademarks, or registered service marks mentioned in this document are the property of their respective owners.

9611R

Software License

READ THIS SOFTWARE LICENSE AGREEMENT CAREFULLY BEFORE USING THE SOFTWARE.

PLEASE READ THESE TERMS AND CONDITIONS CAREFULLY BEFORE USING THE SOFTWARE. BY USING THE SOFTWARE OF CISCO SYSTEMS, INC. AND ITS SUPPLIERS FROM TIME TO TIME, 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 SOFTWARE, MANUAL, AND RELATED EQUIPMENT (WITH PROOF OF PAYMENT) TO THE PLACE OF PURCHASE FOR A FULL REFUND.

Cisco Systems, Inc. ("Cisco") grants to Customer ("Customer") a nonexclusive and nontransferable license to use the Cisco software ("Software") in object code form solely on a single central processing unit owned or leased by Customer or otherwise embedded in equipment provided by Cisco. Customer may make one (1) archival copy of the software provided Customer affixes to such copy all copyright, confidentiality, and proprietary notices that appear on the original.

EXCEPT AS EXPRESSLY AUTHORIZED ABOVE, CUSTOMER SHALL NOT: COPY, IN WHOLE OR IN PART, SOFTWARE OR DOCUMENTATION; 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 SOFTWARE.

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

LIMITED WARRANTY. Cisco warrants that the Software will substantially conform to the published specifications for such Software, if used properly in accordance with the Documentation, for a period of ninety (90) days from the date of shipment. To be eligible for a remedy, Customer must report all warranted problems within the warranty period to the party that supplied the Product to Customer or to the Cisco Service Partner if the Software was exported under the multinational uplift program. Cisco's sole and exclusive obligation and Customer's exclusive remedy with respect to nonconforming Software upon contact will be, at Cisco's option and potentially through the Sales or Service Partner, either (i) to provide a correction or a workaround for any reproducible errors, or (ii) to refund to Customer the license fee for the Software in the event that a license fee was paid and the other remedy is not available, or, if the license fee was zero, refund the price of the hardware less depreciation calculated on a straight-line basis. Customer agrees to cooperate with Cisco or its Sales or Service Partner in creating the environment in which the error occurred. Further, Customer agrees to supply any necessary equipment for such tests.

This Limited Warranty does not apply to Software which (1) has been altered, except as authorized by Cisco, (2) has not been installed, operated, repaired, or maintained in accordance with any installation, handling, maintenance, or operating instructions supplied by Cisco, (3) has been subjected to unusual

physical or electrical stress, misuse, negligence, or accident, (4) is used in ultrahazardous activities, (5) has been used in such a way that Cisco or its Sales Partner cannot reasonably reproduce the Software error, (6) has been exported from the original country of destination without payment of an uplift, or (7) has been misapplied. In no event does Cisco warrant that the Software is error free or that Customer will be able to operate its networks without problems or interruptions.

DISCLAIMER. THIS WARRANTY IS IN LIEU OF AND CISCO DISCLAIMS ALL OTHER WARRANTIES AND CONDITIONS, EXPRESSED OR IMPLIED, INCLUDING THOSE OF MERCHANTABILITY, NONINFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

CONSEQUENTIAL DAMAGES. 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 CISCO SOFTWARE, EVEN IF CISCO HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME STATES DO NOT ALLOW LIMITATION OR EXCLUSION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES.

Customer will comply with all applicable export laws and regulations if it exports the products. This restriction shall survive termination of this Agreement.

This License is effective until terminated. Customer may terminate this License at any time by destroying the software together with all copies thereof. Cisco may immediately terminate this License if the Customer fails to comply with any term or condition hereof. Upon any termination of this License, Customer shall discontinue use of the Software and shall destroy all copies of the software.

This License shall be governed by and construed in accordance with the laws of the State of California. 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 Software.

Restricted Rights - Cisco's software and supporting documentation are provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the 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 and subparagraph (c)(1)(ii) of The Rights in Technical Data and Computer Software clause at DFARS §52.227-7013.

Hardware Warranty

Performance Warranty. Cisco warrants to Customer, for a period of ninety (90) days from the shipping date, that Hardware purchased from Cisco will be free from hardware defects in material and workmanship. To be eligible for a remedy, Customer must report all warranted problems within the warranty period to the party that supplied the Product to Customer or to the Cisco Service Partner if the Hardware was exported under the multinational uplift program.

Hardware Remedies. In the event of a warranted problem with respect to the Hardware, Customer must

contact the place it acquired the Hardware or the Cisco Service Partner if the Hardware was exported pursuant to the multinational uplift program as soon as possible after Customer becomes aware of the defect. Cisco or the Sales or Service Partner (as appropriate) will supply replacement parts for the products listed in Cisco's recommended spares list. Replacement parts will be shipped within five (5) working days after receipt of Customer's request. Cisco or its Sales or Service Partner will bear the cost for shipment of advance replacements to Customer. Customer must return all defective boards and assemblies prior to installation of the replacement boards and assemblies to Cisco or the Sales or Service Partner in accordance with the then-current return material authorization (RMA) procedures. Cisco's sole and exclusive obligation with respect to defective Hardware will be, at Cisco's option and through a Sales or Service Partner if necessary, to either (i) provide advance replacement service as described above, (ii) replace the Product with a Product that does not contain the defect, or (iii) refund the price paid for the Hardware less depreciation calculated on a straight-line basis.

Exclusions. The above warranty does not apply to any Product which (1) has been altered, except as authorized by Cisco, (2) has not been installed, operated, repaired, or maintained in accordance with any installation, handling, maintenance, or operating instructions supplied by Cisco, (3) has been subjected to unusual physical or electrical stress, misuse, negligence, or accident, (4) is used in ultrahazardous activities, (5) has been used in such a way that Cisco cannot reasonably reproduce the Software error, or (6) has been exported from the original country of destination without payment of an uplift. In no event does Cisco warrant that Customer will be able to operate its networks without problems or interruptions.

DISCLAIMER. THIS WARRANTY IS IN LIEU OF AND CISCO DISCLAIMS ALL OTHER WARRANTIES AND CONDITIONS, EXPRESSED OR IMPLIED, INCLUDING THOSE OF MERCHANTABILITY, NONINFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

CONSEQUENTIAL DAMAGES. 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 CISCO SOFTWARE, EVEN IF CISCO HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME STATES DO NOT ALLOW LIMITATION OR EXCLUSION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES.

HOME CONTENTS PREVIOUS NEXT GLOSSARY SEARCH HELP