Protected Internet, Intranet, & Virtual Private Networks

Alexander Moldovyan
Nick Moldovyan
Doug Summerville
Vladimir Zima

A-List

A-LIST, LLC
295 East Swedesford Rd.
PMB #285
Wayne, PA 19087
702-977-5377 (FAX)
mail@alistpublishing.com
http://www.alistpublishing.com

Protected Internet, Intranet, & Virtual Private Networks

By Alexander Moldovyan, Nick Moldovyan, Doug Summerville, Vladimir Zima

1-931769-14-1

03 04 7 6 5 4 3 2 1

A-LIST, LLC titles are distributed by Independent Publishers Group and are available for site license or bulk purchase by institutions, user groups, corporations, etc.

Book Editors: Rizwati Freeman, Peter Morley

LIMITED WARRANTY AND DISCLAIMER OF LIABILITY

Foreword

Currently, computer network technologies are dynamically evolving. They are employed in nearly all aspects of human activity. There are more and more users on the scene whose activities are influenced by e-commerce, and the same is true for most technical systems created by humans. Therefore, it is obvious that security violations during the transmission and processing of electronic data result in serious damages, the extent of which directly depends on the value of electronic information. Some-times such damage can even be compared to global disasters.

Despite the efforts of global information security developers, contemporary computer networks are not becoming less vulnerable. The main reasons for this are as follows:

- Inefficient data processing technologies and protocols or a lack thereof
- Software bugs
- Complexities of managing contemporary software

In terms of security, technologies and protocols that serve as the basis for contemporary computer networks are not bug-free. Firstly, this is due to the fact that most computer technologies and protocols were developed quite a long time ago, when computer security was not a matter of primary importance. For example, most UNIX clones have inherent vulnerabilities related to network services. As well, the MS-DOS operating system and its successors such as Windows 95 and Windows 98 do not provide any information security at all. The same is also true for network protocols (TCP/IP, SPX/IPX, NetBIOS/NetBEUI). On the other hand, currently, new computer technologies are evolving which are improving efficiency but which, at the same time, degrade data processing security. Web technology and its related mechanisms of distributed data processing on the basis of mobile programs, such as Java from Sun Microsystems and ActiveX from Microsoft, serve as examples of this fact. Java applets and ActiveX controls transmitted from the server to the workstation for execution might contain unauthorized functions or a malignant code that could cause computer to malfunction.

Since contemporary software products represent quite complicated systems, they aren't quite free from bugs upon their release. Competition for market positions even strengthens this trend. Developers are physically unable to carefully debug all functions that they have included into software products. These bugs and drawbacks result in accidental and intentional security violations. For example, most data losses are caused by software malfunctions and hardware failures, while most attacks on computer systems are based on detected software bugs and vulnerabilities. As practice has shown, Microsoft Corporation is the leader in software bugs. Although Microsoft has been gradually releasing hotfixes and Service Packs that eliminate the detected bugs, by the time they are released, lots of users might have already suffered from security breaches caused by the software bugs.

New versions of most software products are created as superficial improvements of the current version and its predecessors. Besides this, since developers are short of time, not all development stages required by the technology in use are followed. For example, in most cases not enough attention is paid to the development of initial requirements and specifications. Methods of structural and object-oriented design are often neglected. Quite frequently, ergonomic requirements, such as ease of use and convenience of operation and maintenance, are totally ignored. As a result, the design of software products becomes more and more sophisticated, while the products themselves become harder to configure and maintain. At the same time, it is only possible to prove the quality and accuracy (formally or informally) of a simple security tool. Only an easily managed system can be efficiently administered and checked for consistency. Thus, a lack of simplicity and ease of use results in configuration errors, which intruders can exploit in order to implement attacks. Such errors can also cause accidental security violations, such as data losses or failures of specific network components. As a result, the computer network remains vulnerable, despite its use of security tools. This happens because the reliability and security level of any computer system is determined by the reliability of its weakest link.

Under these conditions, it is impossible to achieve a required level of information security without a sound understanding of modern technologies, standards, protocols and security tools widely used in computer networks. Recently, these products, standards and tools have become rather numerous, and they are addressed quite extensively in the technical literature available. However, some publications are too general (thus preventing the reader from acquiring a proper understanding of the operating principles of a specific security system), while other ones cover security topics in too much detail. The second approach isn't sufficiently structured, and often omits conceptual aspects, which also complicates a proper understanding of the problem under discussion. The vast amount of often inconsistent materials (that sometimes consist of nothing more than promotional literature), coupled with an excessive numberof security standards, which sometimes aren't logically related, confuse and put off readers. Most of these readers find it hard to get past this, which then hinders them from giving the security system a careful look. However, this doesn't need to be the case. Actually, all sophisticated systems comprise simpler components and you only need to divide a complex system into simpler components and detect the relationships between them. Understanding will then follow. In this book, the authors have made an attempt to apply a systematic approach to describing contemporary technologies, standards, protocols and security tools widely used to protect information in computer networks. The material is presented in such a way as to enable the reader to understand the basic principles of network security on his or her own. The book comprises three chapters.

The first chapter discusses the basics of network technologies that influence information security. It covers the basic features of Web technology, as well as the principles of distributed data processing, which are based on mobile software. Further, the chapter includes explanations of the operating principles of various network services, including directory services. The scalability of computer networks is considered in particular detail.

The second chapter provides a systematic classification of attacks on computer networks, and considers methods and tools for securing local area networks against unauthorized activities and attacs via public networks. Various functions of firewall protection on different layers of the OSI reference model are examined in detail. Also, this chapter addresses the various types of firewalls, and specific features of their installation and usage, depending on the required security level for the protected LAN.

The third chapter discusses Virtual Private Networks (VPNs). It considers the technological basics and principles of forming cryptographic tunnels via public communication networks. After that, the discussion proceeds to the basic protocols for building VPNs on various layers of the OSI reference model. A detailed explanation is provided for cryptographic key distribution protocols along with the coordination of protected tunnels' parameters. The aspects of ensuring security of remote access to the LAN resources are also discussed in explicit detail. Finally, various types of VPN tools are covered as well.

Chapter 1: Introduction to Internet and Intranet

1.1. The Logical Architecture of Computer Networks

1.1.1. The Concept of Logical Network Architecture

The rapid growth of both the scale and heterogeneity of modern computer networks has resulted in a substantial degree of complexity in their operational technology, an understanding of which is essential in enabling the organization of effective protection of informational computer resources. The operational technology of any system depends on the architecture representing its components, their functions and relationships.

It is difficult to understand and analyze the architecture of such a complex system in its entirety. To reduce the complexity of the analysis, such an architecture should be considered at different levels of detail [3, 10, 12]. It is necessary to view each level as an abstraction of its functionality that hides the details of the implementation of its components. These details are only unveiled at the underlying layers, upon closer examination.

When studying the architecture of computer networks, it is useful to distinguish between their physical and logical architectures. The physical architecture describes the structure, function and intermediate relationships between the implementations of the protocols at the lower and intermediate layers of the standard layered model of network interaction (OSI Reference model, specifically the protocols of the physical, data-link, network, transport, and session layers (Fig. 1.1). The physical architecture, therefore, depends not only on the structure, function, and interrelationships of the network hardware, but also on software implementations of the protocols at the lower and intermediate layers of the standard OSI model. For a complete analysis of the physical architecture, one must consider it on all levels, corresponding to the physical, data-link, network, transport, and session layers of the standard layered model of network interaction.

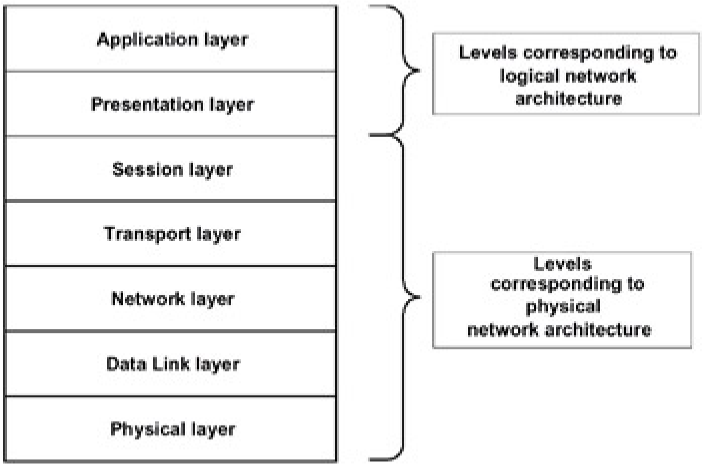

Application layer
Presentation layer
Session layer
Transport layer
Network layer
Data Link layer
Physical layer
Levels corresponding to logical network architecture
Levels corresponding to physical network architecture

Fig. 1.1: The relationship between standard OSI layers and types of network architecture

The logical architecture of computer networks describes the structure, assignment and relationships of its software which implement the protocols of the top layers of the standard layered network model, specifically the protocols of the presentation and application layers. This architecture reflects the unified and integrated technology of the computer network and can build upon different levels of abstraction of the physical architecture.

Currently, the following types of logical network architecture are the most common:

- Peer-to-peer architecture
- Classic client/server architecture
- Web-based client/server architecture

The arrival of each of these models is connected with different stages in the evolution of computing systems. A correctly selected model for the logical architecture of a computer network allows the network designer to meet the requirements of total productivity, the reliability of the protection of network resources, the flexibility of the network setup, and also the minimization of expenses incurred for its construction and administration.

1.1.2. The First Computing Systems and Peer-to-Peer Architecture

The first stage in the evolution of computing systems took place from the 1940s to the 1970s, and, in fact, goes back to right after the invention of the first computer (the ENIAC system placed in operation at the Moore School, component by component, in June 1944).

As a rule, each computing system of that time was based on the shared use of one multi-user computer, as personal computers had not yet appeared. The architecture of such computing systems, operating in an autonomous mode, was centralized, with character terminals connected to one central computer (Fig. 1.2).

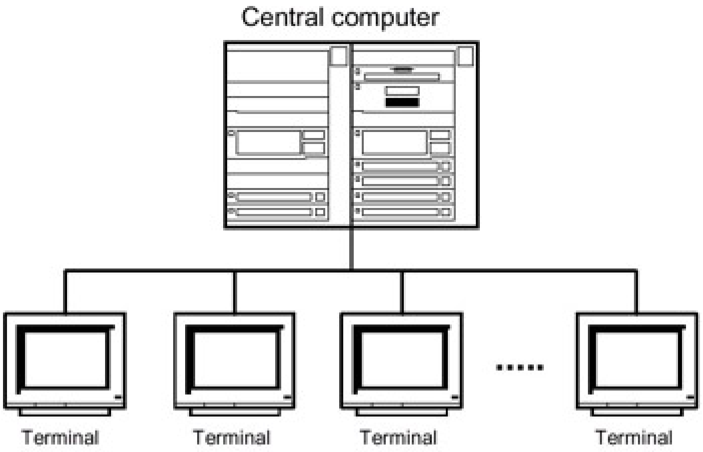


Fig. 1.2: The architecture of the first computing systems

However, if the computers were connected by communication links to form a network, such a network had a peer-to-peer architecture, in which there were no dedicated computers submitting their resources entirely for common use by the other computers of the network.

Thus, a centralized architecture falls into the category of an autonomous computing system based on the shared use of one multi-user computer, while a peer-to-peer architecture falls into the category of a computer network consisting of computers of equal rank, with no dedicated computers submitting their resources for common use.

With a centralized architecture, all the resources of the computing system, including the information, were concentrated on the central computer, also known as mainframe (the mainframe is the central component of the computer system). Character terminals which were connected to the central computer by a cable were used as the primary means of access to information resources. Since a terminal is a relatively unsophisticated device, it did not require any special operations in order to be set up, nor did it require configuration of the software to be performed by the end user, as no software was contained on the terminal. The control of the terminals was carried out centrally from the mainframe, and all of the terminals were of the same type. Therefore, it was guaranteed that a program run on one terminal would work on all of the terminals in the same way.

In terms of data storage and data processing security, the main advantage of a centralized architecture is the relative simplicity of the construction and administration of the information security system. This is a result of the resource centralization, since it is much easier to protect many objects if they are located in one place than if they are distributed to other locations.

Despite these advantages, the first computing systems had a number of drawbacks, including a lack of system flexibility, the inconvenience of use by the end users, and a high cost.

As the use of computing systems with a centralized architecture began to wane, peer-to-peer networks became increasingly more common, because of their relatively low cost. However, currently peer-to-peer networks interconnect PCs rather that multi-user computers. Thus, a major property of a peer-to-peer network is the absence of centralized computers submitting their resources for common use.

Among the significant disadvantages of peer-to-peer networks are their low level of safety, security and performance, and the complexity of their administration. In addition, these disadvantages increase sharply with the number of computers in the network. Therefore, peer-to-peer network architecture is best suited for interconnecting a small number of computers that have low-level requirements in terms of safety and data-processing capability.

1.1.3 Classic Client/Server Architecture

The disadvantages characteristic for centralized computer systems and more recent peer-to-peer computer networks have been eliminated with the construction of computer systems based on a client/server architecture. This architecture, which appeared during the 1980s, marks the second stage in the evolution of computing technologies. The features of this stage include the decentralization of the architecture of autonomous computer systems and their interconnection into global computer networks.

The decentralization of the architecture associated with the first computing systems became possible as a result of the appearance of personal computers, which, unlike simple terminals, can take over many of the functions that were previously performed by central computers. As a result of this decentralization, it became possible to create distributed local and global computer systems, joining both personal computers and computers that completely submitted their resources for common use by other computers in the network. The computers submitting their resources are called servers, and the computers using the shared resources are called clients. The architecture of such distributed computer systems accordingly came to be known as client/server architecture (Fig. 1.3). Personal computers, acting as clients, are also called network workstations.

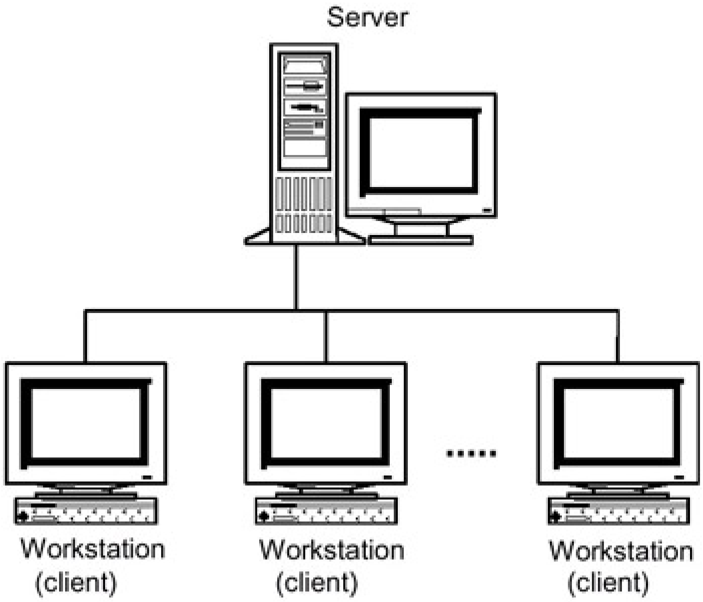


Fig. 1.3: Standard client/server architecture

A specific server is primarily characterized by the kind of resource it maintains. For example, if the resource is a database, then the server is referred to as a database server, the principal purpose of which is servicing client queries for data processing. If the resource is a file system, the server is known as a file server, and its primary purpose is to deliver files to client computers. Generally, servers are now capable of providing a variety of resources for common use, including databases, file systems, and various services, by executing a number of server programs. In addition, servers can provide access to peripherals (for example, a print server can provide shared access to a printer).

We distinguish between several models of client/server architecture, each of them reflecting the appropriate distribution of the software architecture's components among network computers. The distributed software components are distinguished by the functional capabilities they provide.

The functions of any software application can be divided into three groups:

- Functions related to input and output
- Applied functions, specific to the knowledge domain of the application
- Data mining data management functions (e.g. databases, files)

Any software application can accordingly be presented as a structure consisting of three components:

- Presentation components, which implement the user interface
- Applied components, which execute the application functions
- Components providing access to information resources (resource managers), accumulate information and manage the data

The following models of client/server architecture are identified, corresponding to the distribution methods of the three primary software components between the workstations and the network server:

- Only data is stored on the server (Fig. 1.4).

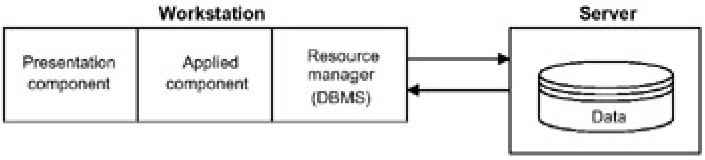


Fig. 1.4: Model of access to remote data

- In addition to the data, the resource manager is located on the server; for example, a database management system (DBMS) (Fig. 1.5).

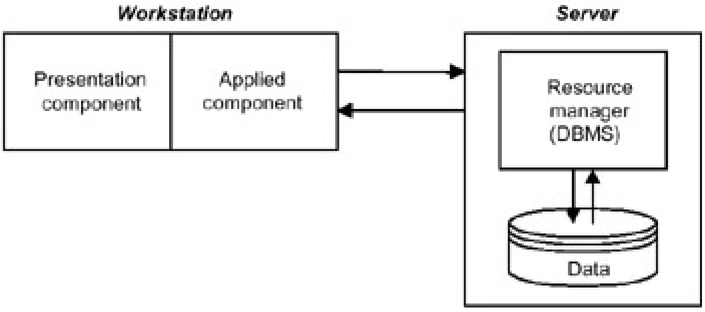


Fig. 1.5: Data control server model

- The data, resource manager, and applied components are concentrated on the server (Fig. 1.6).

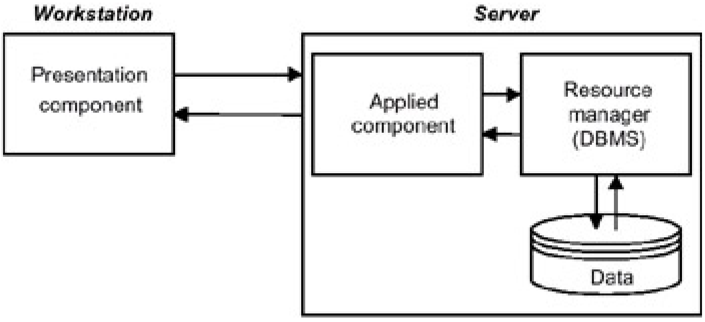


Fig. 1.6: Two-tier client/server model

- The applied components are located on one server, while the data and resource manager are located on another one (Fig. 1.7).

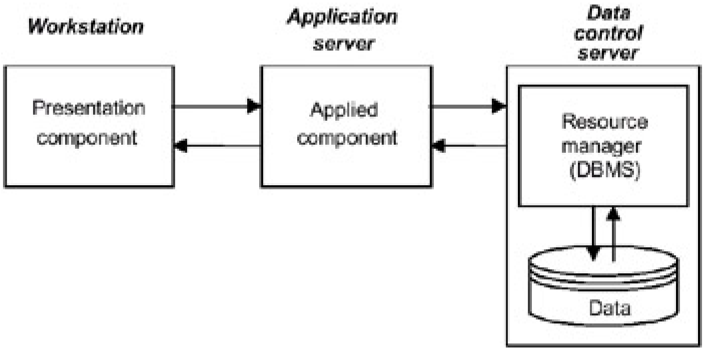


Fig. 1.7: Three-tier client/server architecture

The first model of client/server architecture, where only the data is located on the server, does not provide high efficiency, since the information is processed on the workstations, and the files containing this information must be transferred for processing from the server over the network. The transfer of large volumes of data over the network results in slow rate of information exchange. In turn, this can overload the network. For these reasons, this model of access to remote data can only be used for relatively small networks, which process small amounts of data.

Fig. 1.5 shows a second model of client/server architecture in which, in addition to the information being stored on the server, there is a resource manager (for example, DBMS). This is the model of a data control server. The presentation and application components are combined and executed on the client computer, which supports the functions of data input and display, as well as the applied functions. As a rule, access to information resources is provided with operators of a special language (for example, SQL, in the case of databases) or calls to functions contained in specialized program libraries. The queries made to information resources are routed to the resource manager, for example, to a database server on the network. The resource manager handles the queries and returns the data blocks back to the client.

The main advantage of the data control server model, compared with the remotedata access model, is that the network transfers less information. This is mainly due to the fact that the selection of the required information units from files is not performed on the workstations, but on the server. In addition, at the present time, there are numerous development tools, providing rapid application development, using a standard interface, operating with SQL-oriented DBMS. This provides unification, interoperability, and a wide choice of software development tools.

The main disadvantage of the data control server model is that there is no strict line of demarcation between the presentation component and the applied component. This, in turn, hampers the further improvement of the computing system that has architecture built on the basis of the given model. Furthermore, changes to one component require changes to the entire system.

Taking into account the advantages and disadvantages of the data control server model listed above, it is possible to conclude that the given model is best suited to the construction of computing systems oriented towards processing moderate information content, which is not expected to increase significantly over time. Thus, complexity of the application's applied component should not be high.

In comparison with the data control server model, the two-tier client/server model is more convenient to operate. It has been developed under the assumption that processing performed on the client computer is limited to presentation functions, while applied and data access functions are performed by the server. Applied functions can be implemented in separate programs or in stored procedures, also know as database procedures. These procedures are stored in the database and are performed on the server, where the component controlling data access, i.e. the core of DBMS, also operates.

In contrast to the data control server model, the advantages of the integrated server are obvious: higher efficiency and simpler, centralized administration, and, accordingly, a reduction in the utilization of network resources. Having taken the indicated advantages into account, it is possible to come to the conclusion that the integrated server model is optimal for large networks oriented towards the processing of large amount of information, or those that are expected to increase with time.

When applied components become increasingly complicated and exhibit an increase in resource consumption, a separate server (the application server) can be used. This produces a three-tier model of client/server architecture. The first tier is the client computer, the second is the application server, and the third is the data-control server. The client/server architecture is two-tiered when the applied components are located on the workstations with the presentation component (Fig. 1.4, Fig. 1.5), or on the server with the resource manager and data (Fig. 1.6).

Within the framework of the application server, several applied functions can be implemented, each of them taking the form of a separate service, submitting some functions to any program that can and would like to use them. There may be several application servers, each of them oriented towards the specific set of services. Any program that uses one of them is considered to be a client. The implementation details of applied functions on the application server are completely hidden from the client. Queries received from the application's clients are placed in a queue associated with the application-processing server, which extracts and transfers them for processing according to a particular set of priorities.

The client can be more than just a presentation component. It can support the interface with the end user (at which time, it acts as a presentation component); it can provide for the arrival of data from devices (for example, sensors); and it can also be the application server. The latter allows the realization of application systems, including application servers, of several levels. The architecture of such a system can be viewed as a core surrounded by concentric rings. The core consists of the application servers, in which the basic applied functions are implemented. The rings symbolize sets of application servers, that act as clients in relation to servers of an inner level. There can be an unlimited number of levels of application servers.

The presence of a strict demarcation line between the components of software applications in the client/server architecture, and the balanced distribution of these components among computers of the network allows such a level of flexibility that is not available in peer-to-peer architecture. As a result, computer resources achieve a high level of performance and the potential for expanding and improving the computing system's functionality can be implemented.

The client/server network architecture, which appeared at the second stage of the evolution of computer technologies, is called classic client/server architecture. The following features characterize it:

- The server does not generate the final information but the data, which is subject to interpretation by the client computers.
- The components of the application system are distributed among the computers of the network.

- Proprietary protocols, which are incompatible with the TCP/IP open standard used in the Internet, can be used for data exchange between the clients and the server.
- Each of the network computers is oriented towards execution of only local programs.

This last feature promotes the rise of information security. When only local programs are executed on each computer, the migration of programs on the network during the processing of queries made by clients to servers does not occur. The probability of the execution of harmful programs or infection by computer viruses, is accordingly reduced.

In terms of data storage and processing security, the client/server architecture also has a number of potential disadvantages:

- The physical distribution of the components of the application programs and the irregularity and heterogeneity of the computing system significantly complicates the construction and administration of a security system.
- Part of the protected information resources located on the personal computers is characterized by and subject to increased vulnerability.
- The utilization of proprietary protocols for data exchange between computers requires the development of a unique security tools and, as a result, increased expenditures.
- Upon losing settings of the software of any of the client computers, the execution of complex procedures for reconnecting and coordinating this computer with the remaining part of the system is necessary, resulting in an increase in time required recovery from failures.

1.1.4. Web-Based Client/Server Architecture

Many of the disadvantages of computer networks with classic client/server architecture can be eliminated by constructing computer systems with architecture combining the best properties of centralized systems and classic client/server systems. This new architecture of computer networks is called Intranet architecture. It is also often called Web architecture or client/server architecture based on Web technology. This architecture is the result of long-term research and development in the field of applying Internet technologies to local-area networks. The appearance of Intranet architecture in 1993 is considered to be the beginning of the third stage in the evolution of computer systems.

The main distinctive feature of Intranet architecture is the return to the server of a number of the functions that were removed from the central computer during the second stage in the evolution of the computing systems. The basis of the new architecture is the Web technology that has come from the Internet.

The basis for Web technology are so-called "Web documents", which are stored on the server and interpreted and visualized by a navigating program operating on the workstations (Fig. 1.8). The navigating program is also called a Web browser or Web browser.

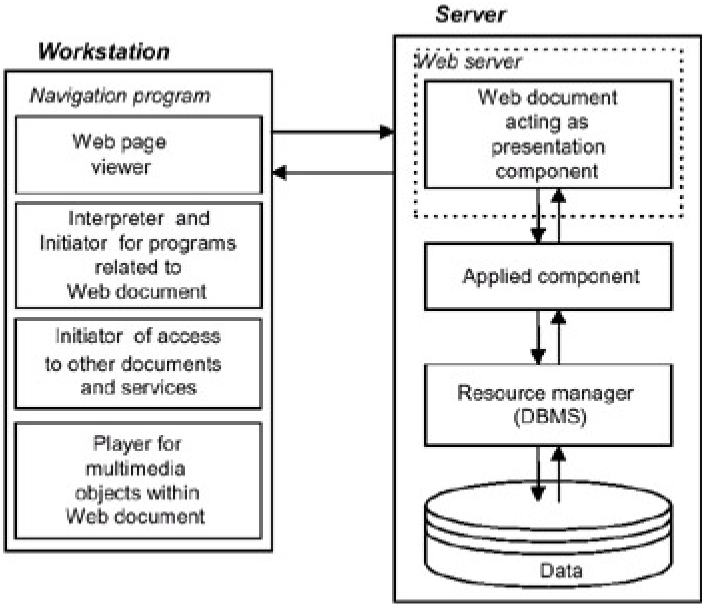

Workstation
Navigation program
Web page viewer
Interpreter and Initiator for programs related to Web document
Initiator of access to other documents and services
Player for multimedia objects within Web document
Server
Web server
Web document acting as presentation component
Applied component
Resource manager (DBMS)
Data

Fig. 1.8: Client/server architecture based on Web technology

Logically, a Web document represents a hypermedia document, consisting of different Web pages joined by links. Each Web page can contain objects and links to other Web pages. Physically, a Web document is represented by a text file, with a special format that specifies links to other objects and Web documents, potentially located on different network hosts. Actually, a Web document substantially contains only one Web page, but, logically, it can combine any quantity of such pages, belonging to different Web documents.

A Web page, the informational analogue of a page from a "hard copy", can contain both text and pictures. But, unlike a paper document, a Web page can be interconnected to computer programs and can contain links to other objects. The execution of a program connected to a Web page starts automatically upon transition to the appropriate link, or upon opening a Web page. Any links included in a Web page are marked with distinct colors and/or by underlining. It is simply enough to click on the link with the mouse to initiate a transition under this link.

The system of hyperlinks obtained here is based on the fact that some selected areas of one document, which may be represented as text or images, act as links to other objects that are logically connected with them. Thus, the objects the links lead to can be located on any computer on the network. A Web page may contain links to the following types of objects:

- Other parts of the Web document
- Other Web documents or documents of other formats (for example, wordprocessing or spreadsheet documents), which can be located on any computer of the network
- Multimedia objects, including pictures, sound, and video
- A program that will be executed on the server after transition to it by the link
- A program that will be transferred for interpretation or execution from the server to the workstation by browser
- Any other services, including e-mail, copying of files via the network, information searches, etc.

From this definition of the concept of a Web document, it becomes clear that the navigating program, executed on any workstation, is not restricted only to visualizing Web pages and performing transitions to other objects. It can also both actuate programs on the server and interpret or launch executable modules related to the Web document on the workstation.

The transfer of documents and other objects from the server to the workstation after requests from the browser is carried out by a program, operating on the server, which is called the Web server. When it is necessary for the browser to receive documents or other objects from the Web server, it sends the appropriate query to the server. If access rights are sufficient, a logical connection between the client and the server will be established. Then, the server handles the query, transfers the results of processing (for example, the requested Web document) to the Web browser, and terminates the established connection.

The Web server acts as an information concentrator that delivers information from different sources and presents it to the user in a homogeneous way. The browser, with a universal and natural human interface, allows the user to view information easily without dependence on its format.

Thus, within the framework of a Web document, the integration of data and program objects of different types, located on different hosts of the computer network, can be performed. It allows to distribute information according to the natural order of its creation and consumption, and also implements uniform access to it. The prefix "Web" here also, in terms of characterizing a specific technology, reflects that fact that the user operation is performed on the basis of transition, with the help of links that, like filaments of a web, link varied objects distributed between the sites of the computer network.

In addition to the connection of physically distributed and wide-ranging types of data, Web documents allow for the consideration of the information with the required amount of detail, which significantly simplifies the analysis of large information contents. It is possible to focus attention on the most important aspects of the data, and then to study the selected material in full detail. Also, it is possible to effectively implement a multi-model approach to the presentation of information, while creating different views of the requested data domain, reflecting the needs and points of view of different members of an organization.

The client computer, on which the navigating program is executed, can be completely standardized. It is enough that such computer, in addition to a processor, RAM and monitor, has a small amount of storage space necessary for the permanent storage and operation of the navigating program, and also an interface unit with a communication line. In addition, the navigating program can, in general, be permanented in the hardware of a specialized processor or embedded system.

The following distinctive features of Intranet architecture should be emphasized:

- The final information intended for presentation to the user by the navigating program is created on the server, not an intermediate form of information, as in systems with classic client/server architecture.

- All the information resources and the application system are concentrated on the server.
- The protocols of the TCP/IP open standard, used on the Internet, will be used for data exchange between clients and server.
- The centralized control of not only the server but also of the client computers is facilitated, as they are standardized in terms of software (on each workstation, it is enough to have a standard navigating program).
- Workstations may execute programs from other computers on the network, not just programs stored locally.

All of these features, except for the last, helped to solve the problem of informational computer safety.

The concentration of all of the informational resources and application systems on the server significantly simplifies the construction and administration of a security system, and the protection of many objects located in one place will be realized much more easily than in the case of their physical distribution.

The use of the TCP/IP open standard for data exchange among computers on the network results in the unification of all interaction methods between workstations and servers. It is not necessary to solve the problem of providing safe information interchange for each set of applications on every computer. The solution for the safety of interaction developed for one computer will be applicable to all of them. In addition, the enormous body of research and development, and the myriad technologies and information available on informational safety, are more accessible, and the choice of protective mechanisms is greater in relation to the TCP/IP open standard.

The facilitation of centralized control of client computers from the server reduces the probability of inadvertent errors made by users, operators, and managers. Such errors are one of the main threats to informational computer safety, and result in direct damage. For example, there may be an incorrect data input, or accidentally introduced program errors, which may cause outages, or corruption of the data in the system. These errors also create vulnerabilities that may be exploited.

In Intranet architecture, the distributed information processing assumes that the programs obtained from the server may be executed on the workstations. Such a system of distributed processing allows the concentration of all application systems on the server. However, the possibility of executing programs from the server on workstations generates new threats to informational computer safety (for example, the threat of substitution of the program transferred from the server [1, 14]). Accordingly, the possibility of the migration of programs adds more requirements to maintaining security in network interaction.

1.2. The Fundamentals of Modern Network Technologies

1.2.1. Interaction with the Web Server

Generalized Description

Today, the most prevalent method of interaction with the Web server is that of client/server architecture, based on Web technology. The process of information exchange used in Web technologies does not differ from the process implemented by standard client/server architecture, where the server program handles the processing of queries received from client programs.

In the process of information exchange used in Web technologies, the client programs are performed in Web-navigation programs, which are commonly found on networked workstations, as well as auxiliary applications that act as clients. Web browsers are used for immediate visualization and interpretation of Web documents stored on the server, as well as for access to other special services (Fig. 1.9), such as:

- Copying files from the server to the client (FTP)
- Providing a virtual session on the server (Telnet)
- Multilevel menu access to computer resources (Gopher)

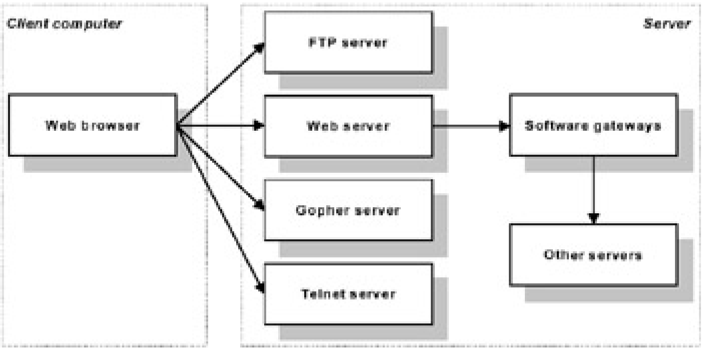


Fig. 1.9: Generalized scheme of interaction between a Web browser and a server

Access to other special services is possible, on account of the fact that, from the very beginning, Web-navigation programs were designed as multi-protocol programs that could provide a common interface for access to many network resources. Currently, the most common Web-navigation programs are Netscape Navigator, by Netscape, and Internet Explorer, by Microsoft. Auxiliary Web applications are primarily used for obtaining some statistical data from the Web server for indexing the information contained on the Web server, or for the purpose of updating the databases of search engines.

In the interaction scheme with Web technologies, the Web server acts as the main server program. This is launched on the server and implements the processing of queries that come from Web clients. The interaction between Web clients and Web servers is carried out according to a set of rules defined as the Hypertext Transfer Protocol (HTTP). When executed, the Web server controls a logical port on the network, which by default is assigned port number 80, and presumes that any messages sent to this port are intended for the Web server.

Upon receiving a query from a Web client, the Web server establishes communication using TCP/IP and exchanges information with the Web client according to HTTP. If the Web client requests access to protected information contained on the Web server, the Web server can demand that a user identifier and password be entered. These protected Web documents are only presented to users with appropriate access rights.

Web documents received by the browser from the Web server represent text files written in a special language, called Hypertext Markup Language (HTML). This language consists of a set of agreements that define text formatting (in any language) and how it appears in the windows of the Web browser. Markups, which define the formatting, also control how links to any objects and graphics are to be displayed. In addition to the markup language, programs written in the JavaScript (Java scripting) and VBScript (Visual Basic scripting) languages can be inserted into the Web document and interpreted by the Web browser when the Web document is loaded and displayed.

For access to information that cannot be processed directly by the server (access to database, for example), a system of software gateways is used. The software gateways, having received a query from the Web server, process the query, or act as a proxy between the Web server and another server (a DBMS server, for example). The software gateways are developed according to special standards that define methods in which the server can call applications or functions from dynamic libraries, and also the means of information exchange with the program object. The Common Gateway Interface (CGI) is one of the most common standards of this type.

Processing Query from a Web Client

Let us consider a complete sequence of steps implemented by the Web server to process a query received from a Web client:

1. The Web browser or other Web client, sends a query to the Web server, requesting of some information resource. This query is sent in HTTP format, while the requested resource's address is specified in the Uniform Resource Locator (URL) format.
2. After receiving the query from the client, the Web server determines the existence of the requested resource among local resources, i.e. among resources that the given server controls.
3. If the requested resource is available, the Web server checks this resource's access rights and, if the rights have not been violated, returns the resource's contents to the Web client.
4. If the Web client's request violates a resource's access rights, the Web server rejects the query and returns the appropriate warning to the client.
5. If the requested resource is not among the local resources on the Web server, the server determines information about the resource's location from its configuration files, including its possible relocation within the network. If the resource has been allocated on the server, but has been redirected to another location for the moment, the server informs the client (Fig. 1.10).

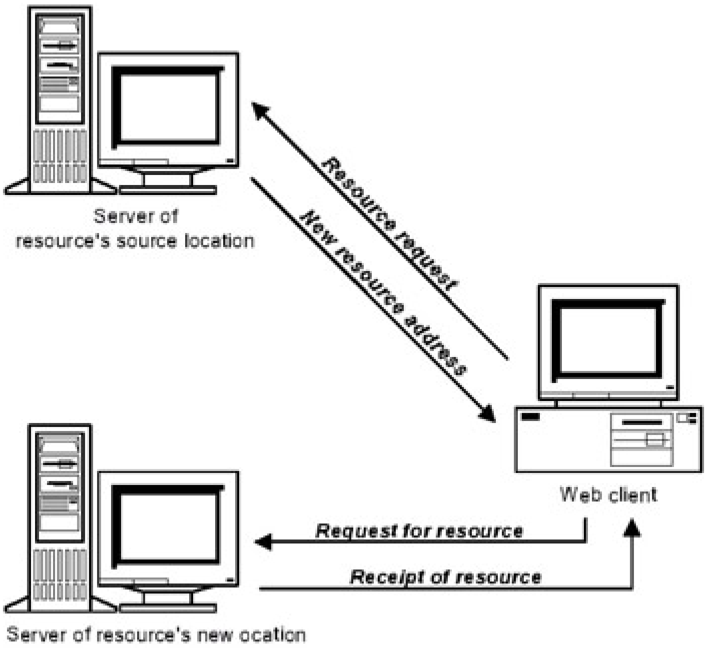


Fig. 1.10: Query redirection

6. If the Web server supports a virtual tree of other Web servers, the search will be redirected to the necessary resource.
7. If the Web server is used as a proxy server, it acts, on the one hand, as the Web server for the client that has sent the query, and, on the other hand, as a Web client, which sends the query to another Web server (Fig. 1.11).

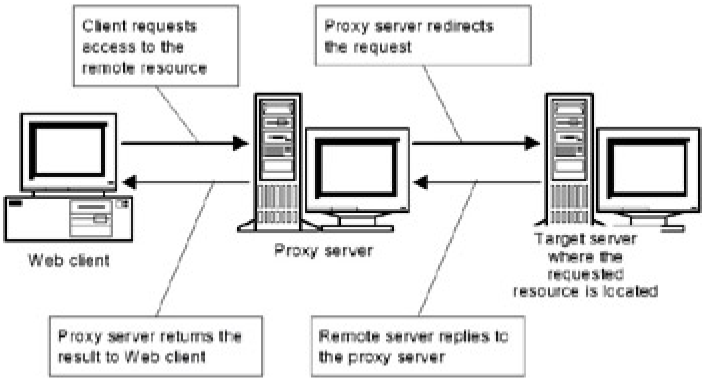

Client requests access to the remote resource
Proxy server redirects the request
Web client
Proxy server
Target server where the requested resource is located
Proxy server returns the result to Web client
Remote server replies to the proxy server

Fig. 1.11: Usage of the Web server as a proxy server

8. After returning the information to the client, the server terminates the connection.

The Web server can be used for solving a large class of problems. Typically, the following functions are supported on modern servers:

- Support for a hierarchical document database, query processing, and access control to information from client-side programs
- Pre-processing of data before answering a query
- Interaction with external programs and other servers, for example, search engines

The majority of modern Web servers, such as Netscape's Enterprise Server and Microsoft's Internet Information Server, implement the Secure Sockets Layer (SSL) cryptography protocol, maintaining the privacy, integrity and authenticity of data transferred by the network. The given protocol is also implemented in modern Web browsers, such as Netscape Navigator and Internet Explorer. This feature provides a safe environment for the use of Web technology in computer networks.

1.2.2. Distributed Information Processing on the Basis of Mobile Programs

One of the key features of Intranet architecture is distributed information processing on the basis of mobile programs. The Web-navigation program executed on a workstation cannot only visualize Web pages and perform transitions to other resources, but it can also activate programs on the server, and interpret and launch them, as the Web document being executed demands. These programs are transferred, together with the document from the server. This type of distributed information processing ensures concentration of the entire application system on the server.

There are three main types of programs that can be associated with a Web document and transferred to a workstation for execution:

- Java applets prepared and executed by Java technology
- Programs written in the scripting languages of JavaScript, VBScript or VRML
- ActiveX components, related to ActiveX technologies

The fact that several varieties of mobile programs exist can be explained by their different functional capabilities, and also by competition among leading corporations in the field of program and network technologies-Sun Microsystems, Netscape and Microsoft corporations, and others.

Java Technology

Java was designed by Sun Microsystems at the beginning of the 1990s, in response to an acute need for computer programs oriented towards usage in network environments and integration with Web technology. The driving force behind Java technology lay in the combined requirements of mobility, independence from hardware and operational platforms, and safety and reliability of information processing. As a result, the Java language was developed, and the integrated technology, involving the mobile programs' creation and usage, came to be known as Java technology.

Java is a simple, object-oriented programming language built on the basis of the C++ language, from which all unnecessary features have been removed, while new functional capabilities, for providing safety and reliability of distributed computations, have been added. Many of these features were borrowed from the Objective C and Smalltalk languages. In order to reduce programming complexity and the number of errors in the final code, rigid object orientation and the strict typing of data were introduced into Java. All data elements are included in the objects, and all functions embody the methods of some object. The strict typing of information units allows for error determination connected with-data type incompatibility at the compilation stage.

The modular approach to the program design, implemented in the language and the simplicity of the language itself, enables not only rapid development of new programs, but also the upgrading of applications previously written and tested in Java. Furthermore, Java offers effective upgrading of older applications written in other programming languages. In addition to standard language elements, Java includes a number of useful libraries, from which one can build computing systems of any complexity. The standard set of libraries can be constantly supplemented with new important functions.

During the development of the virtual Java machine, which executes Java programs through interpretation, independence from hardware and operating system specific platforms have been achieved. Safety and reliability of information processing are also apart of it. Its architecture, presentation of data items, and command systems are all specified. The virtual Java processor provides a complete environment required for Javaprogram execution. Thus, every Java program must meet the specifications of this abstract processor, which determines its machine-independent instruction set, the types of data allowed, and the registers that can be used. The source text of a Java program is compiled into machine-independent codes, called byte codes, which are then interpreted by an abstract processor, to be executed by the virtual Java processor.

Compiled Java programs intended for execution on a workstation within the environment of a Web browser are called Java applets, or, simply, applets. According to its very nature, every applet represents a small program, in which some mandatory functions should be specified. The applet is loaded from the server via the network and executed in the Web browser environment (Fig. 1.12). Links to applets are allocated in Web documents, but applets are not included in Web-document composition. Instead, they are stored in separate files on the server.

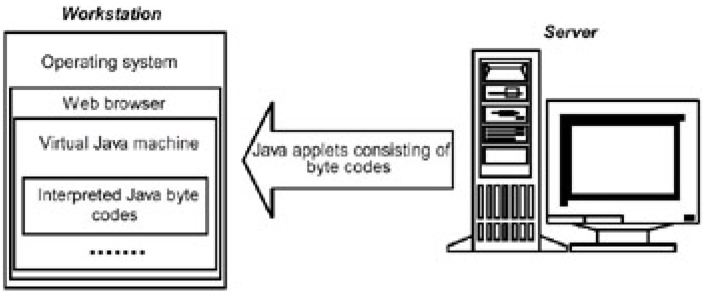


Fig. 1.12: The transfer and execution of machine-independent Java programs

The independence of Java byte codes from hardware and specific operating-system platforms is made possible by program implementation of a virtual Java processor. This is intended for the interpretation of applets for each of these platforms.

The byte codes of Java programs have the following features:

- They can be easily interpreted, and can also be effectively compiled "on the fly" directly into machine code for any modern hardware platform.

- The average length of the command in byte codes is shortened to a minimum, there by reducing the complexity and size of Java applets as compared with customary executed programs.
- Each program's byte codes contain complete program information, which allows them to be tested for safety of execution.

Compilation "on the fly", otherwise known as dynamic compilation, refers to the conversion of Java applets into the workstation's native machine code just prior to execution. After conversion, these programs can be executed as native programs. Dynamic compilation uses a specialized compiler, instead of a virtual machine. The dynamic compilation of byte codes into a native machine code accelerates the execution speed of Java applets. Instead of executing at the much slower speeds customary with program interpretation, dynamically compiled programs execute at speeds closer to those of customary applications. However, additional measures towards the safety of information processing may be lost when dynamic compilation is used. Therefore, dynamic compilation of Java applets in a Web browser is not currently used.

Byte codes have been developed to reduce the average length of a program as much as possible. The virtual Java processor has a small number of registers and stack architecture, and often uses indirect addressing. Therefore, the majority of commands occupy only a single byte, to which a number of operations can be added, if necessary. In addition, the Java processor has an instruction set for processing each data type. As a result, the average length of a Java command is only one to eight bytes. The average length of a command for a classical RISC processor is equal to approximately four bytes.

Two important functions are stipulated for high rates of execution reliability and safety of Java applets:

- Checking of byte codes to ensure integrity and regularity of instructions before their execution
- Control and blocking of dangerous operations during byte code interpretation

The loader and verifier of byte codes implement the first function, and the security manager of the virtual Java processor performs the second function. The security manager accesses files and peripheral units with applets, and also performs critical system functions, such as memory allocation.

The virtual Java processor provides execution and other functions influencing information-processing reliability, for example, "garbage collection", i.e. the release of unused memory. In addition, Java contains the necessary features for correct operations with all objects and resources in cases of exceptional situations.

Java applets' software cycle is the same as it is for programs in other programming languages. The only difference is that, while editing external communications, the requested components can be delivered by the network. The process of applet execution differs essentially from the process for customary programs (Fig. 1.13).

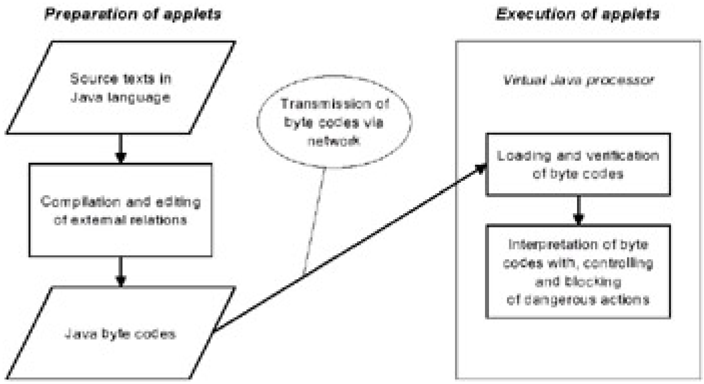

Preparation of applets
Execution of applets
Source texts in Java language
Transmission of byte codes via network
Virtual Java processor
Loading and verification of byte codes
Compilation and editing of external relations
Interpretation of byte codes with, controlling and blocking of dangerous actions
Java byte codes

Fig. 1.13: Preparation and execution of Java applets

As applets and other parts of the application system are stored on the server, system support and administration are facilitated at the expense of centralization. This, in turn, guarantees the permanent use of the latest and most up-to-date versions of programs.

Not only applets representing mobile programs can be written in the Java language-static application programs can be written as well. However, to achieve a high performance level, the source code of such programs should be compiled, not into byte codes, but into machine-dependent codes that allow direct execution by the workstation processor.

Currently, there are a lots of tools for the development of both Java applets and Java applications. They include Microsoft Visual J++, Symantec Cafe, Borland Jbuilder, Sun Microsystems Java Workshop, and a number of others.

Technologies Based on Using Script Languages

Technologies for the development and application of mobile programs based on the use of script languages have appeared in parallel with Java technology. The most important difference between script technologies and Java is the command-by-command interpretation of the source text of script programs, which makes compilation into byte codes before execution unnecessary. (To be executed, mobile Java programs must be compiled into byte codes.) The function of mobile program interpretation written in script languages is assigned to the Web browser.

The nature of script languages (also called macro languages) facilitates the debugging and development of programs written in them. The main script languages currently intended for writing mobile programs include:

- The JavaScript language, developed jointly by Netscape and Sun Microsystems, and also the Microsoft VBScript language (Visual Basic Scripting) which is similar to it.
- The Virtual Reality Modeling Language (VRML), developed by Silicon Graphics. Initially, JavaScript appeared in the Netscape Navigator 2.0 Web browser under the title of LiveScript. Later, Netscape refused to use this title, after having begun to work with Sun Microsystems and, thus, coming under the influence of Java. JavaScript does not represent a derivative language

from Java at all. Though these languages have some common attributes, it is possible to call them distant relatives (Table 1.1).

Table 1.1: The Comparative Characteristics of Java and JavaScript Languages

Java	JavaScript
The program must be compiled into byte codes for execution on the client side.	The program is interpreted on the client side in its initial, text-based form.
Object-oriented. Applets comprise objects described using classes with inheritance.	Object-based. There are no classes for mechanisms of inheritance.
Applets are called from Web pages, but stored separately from the Web documents in individual files.	Programs are called from Web pages, but are built into Web documents directly.
All data types and variables should be declared beforehand.	Data types and variables are not declared.
Static binding. The object links should exist at the compilation stage.	Dynamic binding. The object links are checked during execution.
Cannot write to disk or perform system functions.	Cannot write to disk or perform system functions.

JavaScript is a simplified, interpreted language, based upon object-oriented functions. Its simplicity is due to the absence of a rigid architecture of types and semantics. Its object-oriented nature is manifested by its ability to operate with browser windows, a status bar, and other units of the Web-browser interface and network objects in the hierarchy. JavaScript is not as rich as the Java language, but it is much more convenient, and more effective, for a number of tasks related to the processing of Web documents, and for interacting with the user when viewing them. It has a large number of built-in functions and commands. Programs written in JavaScript can output dialog boxes to the screen, perform mathematical calculations, play a variety of audio and video files, produce new documents, handle button objects in forms, and much more. With the help of JavaScript, it is possible to set the attributes and properties of Java binary libraries and, also, of program modules (plug-ins) plugged into the Web browser.

JavaScript commands are written directly onto the Web page, and are performed by the Web browser while this page is being loaded, or during specific operations that the user performs when interacting with a Web page (for example, when clicking one of the page objects with the mouse, positioning the mouse pointer at a link location, or upon entering data into the fields of HTML forms). As with any other technology or language used in a computer network, information-processing safety is a top priority. JavaScript, although it may not be considered to be a language with a high level of security, meets the majority of these requirements. Some features that lead to vulnerability are not included in the language, thereby providing security in an indirect way. Programs written in JavaScript, as in Java, are forbidden to perform operations with local files. Therefore, a program is not capable of changing or gaining access to a user's data. In addition, JavaScript does not support network functions. A program cannot, for example, open up a TCP/IP port, and is capable only of loading objects to a given address and forming the data to be transferred to Web servers. Modern Web browsers allow users to set safety levels and to control them, so that programs written in JavaScript can only focus on a very specific range of information.

JavaScript has gained a great deal of popularity, thanks to its rapid application development (RAD) capabilities, small program modules, convenient access to all internal functions of the Web browser, and safety.

The primary disadvantage of JavaScript is its relatively low execution rate, owing to the interpretive nature of the language. This disadvantage is characteristic of interpreted programming languages.

It is important to note that Netscape and Microsoft implement JavaScript differently. These discrepancies can result in incompatibilities in the usage of Netscape Navigator and Internet Explorer. Therefore, when writing applications in JavaScript, it is necessary to check the service capability in the environment of different navigating programs.

The Microsoft VBScript (Visual Basic Scripting) is similar to JavaScript in many respects. It is a subset of Visual Basic, and is also intended for programming Web pages. It is possible to develop miscellaneous objects, including program components of other languages, in Web pages while using VBScript. For example, Java applets and ActiveX Controls can be included in a Web page, to facilitate user interaction.

Unlike JavaScript and VBScript, the VRML language, designed by Silicon Graphics, is specifically intended to help with creating interpreted programs simulating three-dimensional virtual worlds. The VRML interpreters are plugged into the Web browser, typically as separate program modules (plug-ins). The source codes of programs in the VRML language are included as a separate VRML file and are called by the link from the Web document when it is viewed in the Web browser. Clicking on such a link results in the opening of a separate window, allowing the user to "walk along" a fragment of a three-dimensional reality.

ActiveX Technologies

ActiveX is a set of technologies from Microsoft that focuses on the addition, integration and unification of current methods of representing and processing information in computer networks built in accordance with Web architecture. The main idea of ActiveX technologies lies in its identical means of access to all information resources of the network (Fig. 1.14). Web technology was selected as the basis for such a unified means of access.

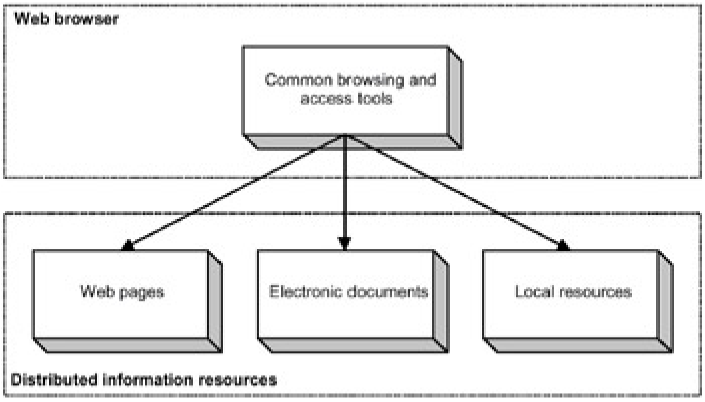

Web browser
Common browsing and access tools
Web pages
Electronic documents
Local resources
Distributed information resources

Fig. 1.14: Uniform access to network-information resources

According to ActiveX philosophy, the Web browser should become an integral part of the operating system. Moreover, methods for gaining access to any information on the user's computer, on the server of a local-area network or on the Internet should be absolutely identical and transparent to the user. This concept is already partially implemented in the Microsoft Internet Explorer 4.0 Web browser.

In terms of the development of mobile programs, on one hand, the set of ActiveX technologies is an alternative to Java and JavaScript technologies, while, on the other hand, it is an essential supplement to them. ActiveX provides not only the development and execution of mobile programs, but also the implementation of a number of additional possibilities, including for example, a call to functions for viewing and editing Word, Excel and PowerPoint documents from the Web browser environment. ActiveX makes a set of Application Programming Interface functions (API) (implemented both for the client and for the server) available for programmers and authors of Web documents.

ActiveX supports the following types of mobile programs, which can be attached to a Web document and transmitted to a workstation for execution (Fig. 1.15):

- ActiveX Controls
- Java applets
- Programs written in the script languages such as JavaScript, VBScript and VRML

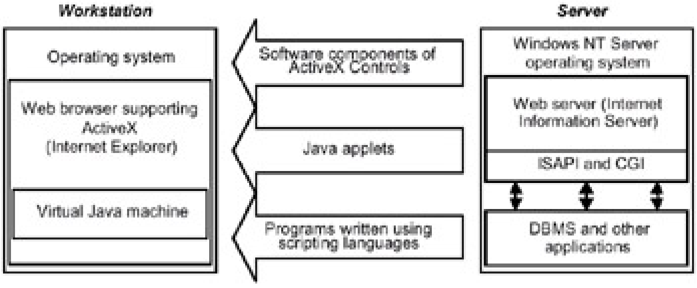

Workstation
Operating system
Web browser supporting ActiveX (Internet Explorer)
Virtual Java machine
Software components of ActiveX Controls
Java applets
Programs written using scripting languages
Server
Windows NT Server operating system
Web server (Internet Information Server)
ISAPI and CGI
DBMS and other applications

Fig. 1.15: Migration of programs while using ActiveX technology

The technologies of the development and usage of ActiveX Controls, and also programs written in the VBScript macro language, are all Microsoft developments.

System-wide software from Microsoft should operate on the server to ensure the efficiency of ActiveX technologies, specifically on the Windows NT Server operating system and IIS (Internet Information Server). The interaction of the IIS Web server with other applications, for example, with a database-management system (DBMS), is provided via the ISAPI (Internet Server API) and CGI (Common Gateway Interface) interfaces implemented in it.

ActiveX Controls represent customary executable programs, which can be loaded from the server for execution on a workstation. As with the usage of Java applets, links to these programs are allocated in Web documents. ActiveX Controls are not included in a composition of Web documents. They are stored in separate files on the server.

The ActiveX Controls components differ from Java applets in the following ways:

- The ActiveX Controls programs include an executable code, depending on the "hardwar-operating system" platform. The byte codes of Java applets are machine-independent.
- The loaded units of ActiveX Controls remain in the client system, whereas it is necessary to load Java applets each time they are required.
- As the ActiveX Controls programs do not work in a similar way to Java applets under the control of a security manager, they can gain access to disks and perform other functions typical for the conventional applications.

Since ActiveX control programs are customary programmed applications, they can be developed, and can perform operations with the help of any programming language. For this purpose, development tools, such as Visual C++, Visual Basic, Delphi, Visual J ++ and a number of others, may be used. Microsoft has developed a complex package of tools, called Microsoft ActiveX Development Kit (MADK).

ActiveX components, as well as programs written in the JavaScript and VBScript macro languages, can include calls to ActiveX API functions for rendering a number of services, such as:

- The creation of high-quality multimedia effects
- Opening and editing electronic documents, by calling applications supporting the Object Linking and Embedding standard (OLE), for example, to the Microsoft Office applications
- Access to the operating system for optimizing of the execution parameters of programs obtained from the server

Programs written in the JavaScript and VBScript macro languages can automate interaction between many objects, including Java applets, ActiveX components and other programs on the client computer, allowing them to work together as a part of an integrated, active Web space. It is possible to create a custom macro language, and to add its interpreter to the Internet Explorer Web browser, with the help of the dynamically loaded library (DLL).

Compared with Java, ActiveX technology has both advantages and disadvantages. Its disadvantages are primarily due to its lower level of security in terms of distributed information processing. The ActiveX components loaded on the client system can access any part of the system, similar to customary applications. In the ActiveX framework, Microsoft has implemented confidence protection on the basis of digital certificates. These provide authenticity of the confirmation of the program components loaded from the network. However, confirmation of authenticity does not mean confirmation of security. Aside from that, the confidence protection scheme of ActiveX can turn out to be ineffective when a user loads ActiveX components from the Internet, especially if they are from unknown sources.

At the same time, unlike Java applets, ActiveX components allow for the implementation of functions appropriate for full-scale application-program development. This feature is an essential advantage for a corporate network, on the condition that appropriate security measures are taken (for example, granting permission to load ActiveX Controls only from the corporate servers).

In terms of performance, since Java is an interpreted language, Java applets are executed on a virtual machine of the client system at a slower rate than that of compiled ActiveX Controls. But, on the other hand, Java applets are very compact, and, therefore are loaded quickly. Loading ActiveX Controls requires more time. It is also necessary to take into account that the loaded ActiveX Controls programs remain in the client system, whereas it is necessary to load all Java applets anew each time. With regard to safety, this property is a disadvantage, as it distrubs the application system's centralization. But, with regard to performance, it is advantageous, as compared to Java applets.

As for independence from hardware and OS, ActiveX is second to Java technologies. Despite Microsoft's statement that ActiveX provides open multi-platform support for Macintosh, Windows and Unix operating systems, ActiveX technologies work on Microsoft Windows platforms better, as they were developed principally for using functions built into these operating systems. Accordingly, ActiveX can be used to a full extent in networks operating under the control of Microsoft Windows operating systems.

1.2.3. Access to Relational Databases

In client/server architecture based on Web technology, the Web server represents itself as the information concentrator, supplying the information from miscellaneous sources that it then submits to the user in a uniform way, with the help of the Web browser. The immediate integration of heterogeneous information is performed during the visualization and interpretation of Web documents. These procedures are executed by the Web browser during interaction with the Web server, as well as with other servers, for submitting information resources.

The interaction of the Web browser with the server of a database-management system (DBMS server) can be carried out in two primary ways:

- Access to the DBMS server through the Web server
- Direct access to the DBMS server

Access to the DBMS Server through the Web Server

In order for the Web browser to access the DBMS server through the Web server, the program-gateways system is used (Fig. 1.16). The program gateways, having received a query from the Web server, act as the proxy between the Web and DBMS servers. The program gateways have been developed according to special standards that determine ways of calling applications or DLL functions by the Web server, and, also, of information exchange with these program objects. One of the most common standards of the given type is the Common Gateway Interface (CGI), in addition to its improved specification, FastCGI.

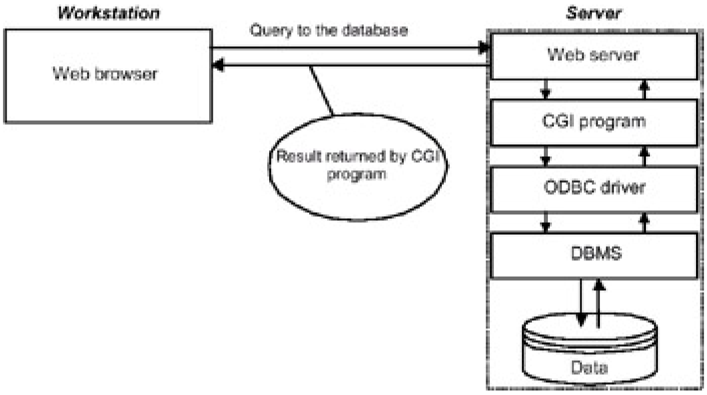


Fig. 1.16: Access to DBMS through CGI program

The CGI Interface

An appropriate CGI program, serving as a program gateway between the Web server and DBMS server, is necessary for the Web browser's access to the DBMS server, through the Web server, according to CGI standards.

CGI applications operate independently from the Web server, and are launched upon calls from a Web document's processing by the Web browser. The CGI program interacts with the Web server by means of a double-sided exchange of environmental variables through standard input/output channels of the given application.

Since CGI programs work independently from the Web server and have a simple, common interface, Web-document developers are able to create CGI programs in any language supporting standard file input/output operations. Furthermore, with independent development it is possible to create applications that can be easily transferred from one Web server to another. There are also standard CGI programs specially designed for the interaction of Web servers with different DBMS (for example, the WebDBC program).

HTML forms, which allow users to formulate queries to the database, are used as the interface between the Web browser and the DBMS server included in Web documents. The CGI program obtains the information from the Web server through environment variables or standard input. This depends on the access method used at the time of data exchange between the Web browser and Web server. Then, the CGI program accesses the DBMS server through the Open DataBase Connectivity

(ODBC) driver and returns a response to the Web server query, through standard output.

The ODBC driver provides a unified method of accessing different DBMS, by means of a standard query language (SQL). Due to the ODBC standard applications, it can use only SQL dialect and interact with miscellaneous DBMS. It is possible to do without an ODBC driver, but, in this case, the CGI program should be oriented towards specific DBMS operating on the server.

Thus, the developer of the CGI application does not need to know anything about how the Web server is organized. Moreover, it is not necessary to use complex languages such as C++. The CGI program can also be written in a command language, for example, Perl. The main requirement is an adherence to all agreements imposed by a standard CGI. Such an approach essentially simplifies the development of application software for the Web in general and, in particular, for the interface to databases and Web server.

One of the other disadvantages of the CGI standard is its inability to slow down query processing while their rate of arrival intensifies. Each time the CGI program is called, it has to be loaded from a disk (i.e. it must be launched anew). After program termination, it is necessary to release the resources that have been used by it. Such operations create a noticeable, additional load on the server that affects its performance. In addition, starting a new procedure at each query reduces the efficiency of continuous processes and data availability. The information generated during the processing of one query is not reusable.

API and FastCGI Interfaces

To bypass problems connected with the speed of CGI, many suppliers of Web servers, including Microsoft and Netscape, have elaborated appropriate application programming interfaces (API). Microsoft has developed the ISAPI (Internet Server API) interface and Netscape has released the NSAPI interface (Netscape Server API).

These interfaces are closely integrated with the Web server, allowing the user to keep access to constantly used processes and data. The programs with the ISAPI interface are compiled in DU files. They are loaded into memory during the first call to them and, consequently, it is unnecessary to generate a new process for the next call of these programs. The NSAPI-interface functions are loaded in the server's process space. Accordingly, upon calls to these functions, additional processes are not generated. Due to API interface, the program using it can leave a DBMS connection open, so the following query to the database does not have to waste time on opening and closing the connection.

However, using API interfaces of Web servers is quite a good, though nonstandard, solution. The majority of applications cannot be ported from one API to another, and it is seldom possible to port the applications to other platforms. Furthermore, the majority of Web-server applications are still developed for the CGI interface, so a transition to the applications based on API is a financially justified decision.

Therefore, some ways to construct a few intermediate variants have come onto the scene that, on one hand, would meet the requirements of mobility, independence and simplicity of programming, and, on the other hand, would be sufficiently effective. One of these solutions is the FastCGI specification. This specification stems from the fact that an application uses a mode of parameter and data transfer that is used in CGI, but not deleted from memory, and remains resident while processing incoming searches.

Thus, the applications based on FastCGI, like CGI programs, work irrespective of the Web server, and are launched through standard links in Web documents. But, like programs based on API, the programs for FastCGI operate continuously. When the program completes the processing of the next query, it remains open and waits for a new query.

When the Web browser gains access to the relational database through the FastCGI interface, a scheme is created in which three servers are used: the Web server, the FastCGI program and the database server. The Web server receives a Web browser query and transfers it to its FastCGI program, which in turn accesses a database server. The result is returned via a return path.

Direct Access to the DBMS Server

For direct access from the Web browser to the DBMS server, you can use both Java applets (Fig. 1.17) and ActiveX controls (Fig. 1.18), and specialized program modules connected to the browser (plug-ins).

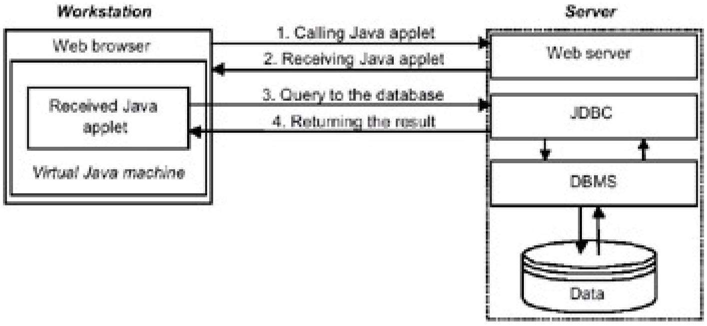


Fig. 1.17: Access to a DBMS with the help of a Java applet

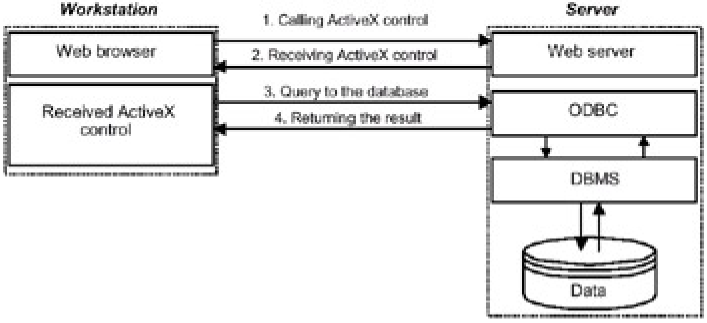


Fig. 1.18: Access to a DBMS with the help of an ActiveX control

The standard Java DataBase Connectivity interface (JDBC) was developed in order to use Java applets to access different servers of a DBMS. The given interface is oriented to support interaction with the DBMS server, not only of Java applets executed on client servers, but also of Java programs launched on the server.

Access from the Web browser to the DBMS server with the help of ActiveX control implies, as is the case with Java applets, the query and transfer of the appropriate program on a workstation, and also its further execution on a workstation. In this case, interaction with the DBMS server should be performed through the ODBC interface. You should bear in mind that Java is executed by the Web browser in an interpretive mode of the mobile code, and, consequently, the requirements for instrumentation of a workstation in terms of productivity and RAM size will significantly increase.

In order to use the specialized program modules connected to the browser (plugins) for access of the Web browser to the DBMS server, the installation of an appropriate program addition on the workstation is required beforehand. After that, the interaction with the DBMS server will realize the installed software that obtains control from the Web browser, while processing the appropriate call in the Web document. In order to avoid incompatibility, the interaction of connected program modules with the DBMS server, as well with ActiveX components, should be performed through the ODBC interface.

1.2.4. Managing Information on Network Resources and Users

The scale and heterogeneity of modern computer networks not only make administration and protection of computer resources more difficult, but also lower operability of distributed network services by end users. Under these conditions, the required level of control, safety and ease of management of a computer network can be provided only with efficient control of information on its resources and users. Otherwise, a manager will not be able to control computer resources in an appropriate way, and users will not have transparent access to any service of the network independent of its location.

The efficient control of resource and network-user information implies dynamic information accumulation and upgrading, and also the output of the necessary data on users' and programs' queries, according to their authorities. There are two sorts of controlled information on network resources and users:

- Administrative information, including information on users and network resources. This does not imply explicit descriptions of information resources at a separate file level, for example, document files (i.e. Web documents, text documents, Word, Excel documents, etc.)
- Detailed information on information resources of the network at a separate file level, reflecting their contents and addresses

Methods to control administrative information on the network and detailed information on information-network resources were developed before Web technology was created. However, at the time of the transition to Web architecture, as a result of which availability, popularity, and accordingly scale of computer networks, have grown, these methods have undergone some changes.

Management of Administrative Information

For management of administrative information on the network, the subsystem known as the directory service is included into modern network operating systems. This service supports names and descriptions of both resources and network addresses, whereby essentially simplifying the installation of network operation connections and control. Due to the directory service, a uniform unitized network space for all users and network services is created at the expense of allocation of uniform access points and uniform resource and user management.

The Tasks of the Directory Service

The directory service ensures that the following important tasks are carried out:

- Automatic search of network resources and registered users, and also transparent access to network resources
- Administrative control of both registration of computer resources and users
- Support of a convenient naming system of network resources and users
- Unified registration of the network's users and resources

Searches of network resources and registered users are transparently performed with the help of the directory service. Like the Yellow Pages it allows the location of necessary services by name within a known environment (for example, in any city), or searching by particular categories, for example "museums". The directory service performs search functions in a similar way, but uses the computer network as an environment. Also, unlike the Yellow Pages, the search is performed automatically. It provides direct access to any services in the network, independent of their location.

The directory service allows the user to embrace all the computer network's resources and users of any scale, via centralized administrative control, eliminating outdated management methods of configuring each server separately. This service helps managers to gather and view resource information that has been distributed on sites of the network, and provides a uniform presentation of this information.

One of the major functions of the directory service is mapping user or resource names to network addresses or, in other words, translating user or resource names to network addresses. This function, known as the name service, provides the capability of working with user-friendly aliases, translating them to machine addresses and vice versa. The most up-to-date directory services support all standard naming systems, enabling the user to manage different namespaces in heterogeneous computer networks in a unified manner.

Due to the directory service, users and resources are registered in the network only once, and all servers can access the same directories at the same time. When there was no common network directory service, each network server had to be controlled personally. The user who required any of the server's resources had to have an account on it. This resulted in multiple logons of the same user at different network servers within an organization or company. The users had to memorize this information, as well as remember the locations of all resources.

A large number of accounts posed great difficulty for both users and administrators. But, what is more important, it exposed security systems to risk. Because it was necessary to remember a number of names and passwords, users either wrote them down, perhaps storing them in unsafe places, used some easily guessed passwords or, even worse, set the same password for all accounts.

While using the directory service, the presence of one identifier and password for access to the network allows the user to reduce security risks for the system and, besides that, administrators can obtain greater user access control over specific network resources. In cases of single logon, the authentication of a specific user is performed on the basis of one password or hardware key. Thus, the users gain access to any network resource to which they were given the rights by a manager.

Principles of the Directory Service's Construction

The directory service provides a uniform, consistent presentation of the network, and unified access to administrative information on network resources and users. All network users and services require it. Any user's or service's access to the directory service is implemented according to their authorities.

The usage scheme and organizational structure of the directory service is represented in Fig. 1.19. Access to any administrative information on the network is gained by a specialized DBMS through the interface subsystem, which provides a uniform mode of providing this information. The directory service DBMS during query processing interacts with a subsystem of identification intended for mapping the network names of users to resources to their actual addresses in the network.

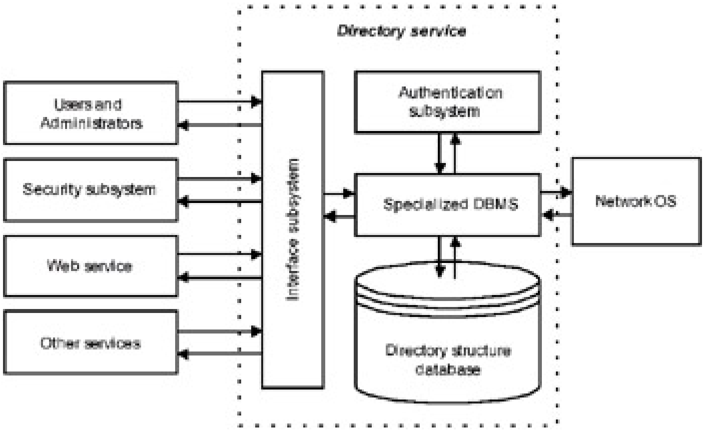


Fig. 1.19: The directory service's usage and organization structure

The directory service database is organized in the form of hierarchies of the directories, similar to the structure of a file system's directories. The directories' hierarchies provide a systematization of object stored in them by their allocation on the directories, according to given criteria.

Objects in the directory service's database are the portions of information describing resources and network users. The objects are grouped in the directories by particular criteria, for example, according to the department or division of an organization to which they belong. Each directory can contain other directories and objects (Fig.

1.20). A directory that is not included into any other directories is called a root directory.

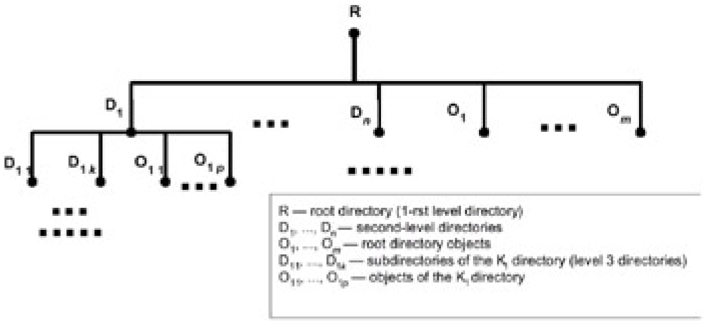

R
D$_1$
D$_n$
O$_1$
O$_m$
D$_{1\,1}$
D$_{1\,k}$
O$_{1\,1}$
O$_{1\,p}$
R — root directory (1-rst level directory)
D$_1$, ..., D$_n$ — second-level directories
O$_1$, ..., O$_m$ — root directory objects
D$_{11}$, ..., D$_{1k}$ — subdirectories of the K$_1$ directory (level 3 directories)
O$_{11}$, ..., O$_{1p}$ — objects of the K$_1$ directory

Fig. 1.20: A root-directory tree structure in the directory service's database

Thus, in a tree structure of any root directory, "non-leaf" points (having outgoing branches) are the directories, and "leaf" points (without outgoing branches) are the objects. For separate operating systems, for example, for Windows NT, the directory service's database can include some roots joined by particular types of relations. In this case, the root's complete tree is named the domain.

The object in the database's directory represents its record relative to an actual object or the network's subject, for example, a printer or a user. Each directory object contains information in the form of a set of properties (attributes) and their values. For example, the network printer is characterized in the database by the object Printer, for which such properties as the name, description, location and network address are defined. The same types of objects have identical properties, while properties can differ for different types of objects. For each type of object, the mandatory properties are defined. If the object's values are not indicated, it cannot be created. For example, a mandatory property of all types of objects is that they must all have names.

The following types of objects of the directory service database are distinguished:

- The *workstation* object, which contains a description of the workstation of the network
- The *server* object, describing the network server
- The object of *volume*, describing the logical volume on the disk
- The *printer* object, which contains a description of the printer
- The *queue* object, describing the printing jobs waiting to be carried out
- The *user* object, containing user accounts (the identifier, first and last names, password, authorities, e-mail address, script of registration, etc.)
- The *group* object, circumscribing a group of users
- The *profile* object, describing the configuration's parameters, assigned for users
- The *directory schema* object, describing the directory
- Other objects, which depend on the particular directory service in use

The association of the descriptions of actual network objects and subjects, with the help of directory hierarchies, substantially simplifies network administration, and also the process of searching through its resources and users.

The population of the directory service database is performed with the appropriate components of a network operating system, network services and also by network administrators.

For high reliability of the computer network, current directory services support the functions of replication and synchronization of their databases. Replication assumes the creation of several copies of the database, distributed on different servers. Synchronization provides upgrades of distributed copies of the database on a timely basis to maintain their actual status.

Current directory services meet X.500 and Lightweight Directory Access Protocol (LDAP) standards. The most common is the LDAP standard, which represents a subset of the Directory Access Protocol (DAP) used for building X.500 directories. However, DAP works only in stacks of protocols of the Open System Interconnection (OSI) model, and demands powerful computational facilities. The LDAP protocol, as well as DAP, is intended for the retrieval of information from the hierarchical directories, but is limited by the number of replies to the X.500 directory it can query, which reduces the network's load. The main advantage of LDAP is that its programming interface is more user-friendly than X.500 or DAP interfaces. Additionally, LDAP can be implemented more easily than the above-mentioned protocols, since usual text lines without additional formatting are used for coding.

Novell Directory Services (NDS) is used in the Novell NetWare network operating systems. Microsoft Windows NT Server 4.0 includes the Windows NT Directory Service (NTDS). The new Active Directory (AD) directory service was intorduced with the next version of this operating system (Windows 2000). The NDS and AD directory services correspond to the LDAP protocol.

Management of Detailed Information on Information Resources

General Information

Specialized search engines are used for the management of detailed information on a network's information resources at a level of separate files. These systems are oriented to the performance of the following functions:

- Periodic scanning of files stored in the computer network's hosts, with the purpose of defining their contents
- Systematization of information obtained at scanning, and recording it in the database that contains information on the network's information resources
- Search and output of necessary information on user and program queries, according to their authorities

The search and output functions of the necessary information on user and program queries are employed with a specialized DBMS, on the basis of information from network resources stored in the search engine database (Figs. 1.21 and 1.22). The result of the search is a list of file indexes corresponding to queries, together with their descriptions.

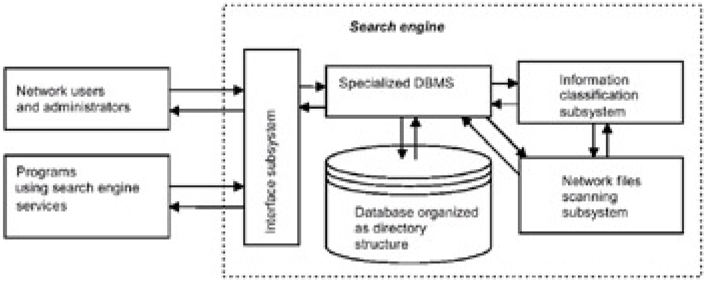

Search engine
Network users and administrators
Programs using search engine services
Interface subsystem
Specialized DBMS
Information classification subsystem
Network files scanning subsystem
Database organized as directory structure

Fig. 1.21: The usage and organization of an information-retrieval system based on directory construction

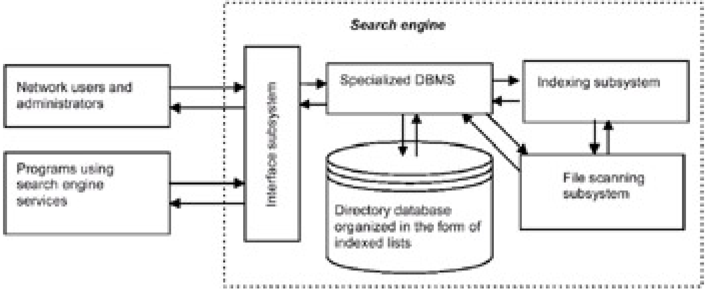

Search engine
Network users and administrators
Programs using search engine services
Interface subsystem
Specialized DBMS
Indexing subsystem
File scanning subsystem
Directory database organized in the form of indexed lists

Fig. 1.22: The usage and organization of a search engine, based on the construction of indexes

Depending on the automation of methods of information accumulation from information resources in the database, and also its structures, two types of search engine are distinguished:

- Systems based on the construction of directories, which provide searches both by navigation in subject directories and by keyword
- Systems based on the construction of indexes, which only provide keyword searches

There are also search engine of a combined type.

A common feature of the aforementioned types of search engine is their scanning of files stored in the network hosts, a procedure meant to defining their contents.

Network file scanning for subject directory and index construction is performed automatically. The main problem of file scanning is with the creation of their descirptions. The file's description becomes its retrieval image, as it replaces this file and is then used during the search, instead of the actual file.

The most popular model of a file's retrieval image is the vector model, in which a list of the terms adequately reflecting its description is assigned to each file. To be more precise, the file is assigned the vector of a dimension equal to the number of the terms that may be used during a search. With a Boolean-vector model, the unit of a vector is equal to 1 or 0, depending on whether or not the term of a retrieval image is present. In more complex models, the terms are weighed. A unit of a vector is equal not to 1 or 0 but to some number (weight) reflecting correspondence with given term in the document. The later model has become the most popular.

The user can create file-retrieval images by including pertinent keywords in the retrieval image of each file. This procedure is often referred to as indexing. This name, however, is not absolutely correct, as indexing is understood as a compilation of an inverted list, in which each term corresponds to the index of the list of file retrieval images to which this term relates.

For scanning files in the network and creating their retrieval images, special scanning programs, often called robots, are used. The robot program is launched on a computer connected to the network, and automatically downloads files from network hosts for analysis. The development of such scanning programs is not a trivial task as a network's computer file contents vary, and do not correspond to one another's data formats: different types of electronic documents, text in miscellaneous encoding (ASCII, ANSI, UNICODE), graphics, audio, video data, and other programs. The robot should know how to extract the information on these files and to form their retrieval images by adding the appropriate keywords.

The sources of information in analyzed documents are the headers, summaries, and lists of keywords, links and complete texts of documents. To create retrieval images of files with non-textual information, links to this information (URL) are mainly used, in addition to users' and administrators' messages located in specialized files. The description of news Usenet and mail lists is implemented on the basis of Subject and Keywords fields.

It is necessary to keep in mind that, during file-scanning, not all terms from analyzed sources of information get into retrieval images. The assignment of a retrieval image to a file or document is performed on the basis of the dictionary from which the keywords placed in a retrieval image are selected. Systems with a controlled dictionary are not the same as those with a free dictionary.

A controlled dictionary assumes support of some lexical database, to which terms are added by the system administrator. In this case, retrieval images can be composed only from the terms of the lexical database.

A free dictionary is extended automatically, in accordance with the appearance of new terms. Accordingly, retrieval images can be composed from new terms, which are automatically recorded in the lexical database. In this case, lists of forbidden words (stop-words), which cannot be used to create new terms and construction of retrieval images (for example, prepositions, conjunctions, etc.) are applied. In order to prevent overloading the dictionaries in use, considerations such as the term's weight are taken into account. The dictionary is enlarged only in the event when the added word occurs no less than the number of times specified, for example, 30.

Search Engines Based on Categories

In search engines, the database system is organized as a structure of categories. Apart from the interface subsystem, which provides uniform directory performance, database and DBMS hierarchies, this retrieval includes a subsystem of file scanning in the network and, also, a subsystem of information classification (Fig. 1.21).

The classification subsystem aims to systematize information obtained as a result of scanning. Information classification and directory creation are more often performed manually by a division that the subject directories' support.

The result of manual information classification on network information resources is found in permanently updated hierarchical directories, at the top level of which categories pertaining to areas of separate divisions' activity of organization are assembled (for example, the most common information categories). The directory objects that are "non-leaf" points of a hierarchical tree represent links to files, for example, to files of electronic documents (Web, Word, Excel, etc.), together with a brief description of their content.

The key advantage of the category-based directories is the concise nature of their information selection, which surpasses the capabilities of any computer so far. But, as the category-based directories are filled in manually, there is no guarantee that they are complete. Moreover, manual information classification requires quite a substantial degree of human labor that not every company can afford.

On the Internet, such international category-based directories as Yahoo (http://www.yahoo.com) and Infoseek (http://www.infoseek.com) enjoy the most popularity.

Search Engines Based on the Construction of Indexes

The inadequacies intrinsic to the services of category-based directories are not an issue in search engines based on the construction of indexes. In these search engines, the indexing subsystem is used instead of classification (Fig. 1.22), and the database is organized as lists assigning the indexes of file description to keywords.

After scanning files in the network and creating their retrieval images, the obtained information items are systematized automatically by their indexing. At indexing, each keyword is assigned an index that serves as a pointer to the list of the file retrieval images, to which the keyword relates. The lack of indexing would result in a prolonged search of images (descriptions) of documents by given keywords.

The structure and composition of indexed lists of different systems can differ and depend on many factors: the size of the array of retrieval images, the information retrieval language, the allocation of different components of the system, etc. Let us consider the structure of an indexed list, with an example of a system in which it is possible to realize not only a primitive Boolean, but also a context-sensitive, a weighted search of Web pages, and a number of other possibilities.

The indexed list of such a system should include tables of: Web-page identifiers (page-ID), keywords (keyword-ID), modification of Web pages, headers, hypertext links, inverted list (IL), and direct lists (FL).

Page-ID maps Web-page identifiers in their addresses (URL); Keyword-ID maps each keyword in the unique identifier of this word; the table of headers maps the Web-page identifier in its header; and the table of links maps the Web-page identifier in a link in this page. The inverted list assigns a list of the Web-page identifier, and the position of the word on a page to each keyword of the document. The direct list is an array of Web-page-retrieval images.

All these files are used during a search in any event, but the main one is the file of an inverted list. The result of a search in this file is an association and/or an intersection of Web-page-identifier lists. The resulting list, which is converted into a list of headers provided with hypertext links, is returned to the user in its Web browser. In order to provide a quick search of recordings in the inverted list, some more files, for example, a file of pairs of letters with a specification of recordings of an inverted list starting from these couples, are added. In addition, the mechanism of direct access to data hashing is applied.

For an upgrade of an index, a combination of two approaches is used. The first one can be called a correction of an index "on the move", with the help of the table of the pages' modifications. The essence of such a solution is quite simple: the old recording of an index refers to a new one, which is used during the search. When the number of such links becomes sufficient to affect the search, the complete upgrade of an index takes place (i.e. its reloading). The efficiency of a search with each concrete retrieval system is determined by an index's architecture. As a rule, organizing these arrays is the "company secret".

As file scanning in the network and indexing of the information obtained at scanning are performed automatically, the search engines based on the construction of indexes operate fully automatically. This makes them accessible to any organization's computer networks.

The key way for a user to retrieve information in a search engine based on the construction of indexes is through a keyword search. This method is more powerful than a similar search method conducted through systems based on the directories' construction. The information-retrieval language allows the user to formulate a search in a simple and understandable way. During query processing, the content of the information retrieved is divided into lexical units, from which forbidden and common words are deleted. Sometimes the normalization of a lexicon is performed, and then all words are connected by logical operations that are indicated by the user or which operate by default.

Besides a normal set of the logical operations AND, OR, NOT, the most advanced search engines allow the use of the NEAR operation, providing a contextual search. In a query, it is also possible to specify parts of the document for the search: the link, title, summary, etc. The user can also set a field of output ranking and criterion of document proximity into the search.

In a number of search engines, searches are checked according to their relevance. Relevance is a measure of the correspondence of the document retrieved by the system, tailored to the user's needs. Relevance is either formal or actual. The former is calculated by the system, and, on that basis, the sampling of the retrieved documents is ranked. With the latter, the user issues a rating of the retrieved documents. Some systems have a special field for this purpose, where the user can mark the document as relevant. Upon retrieval iteration, the search is expanded by this document's terms, and the result is ranked again. The iterations take place until stabilization occurs, meaning that nothing better than the sample obtained will be achieved from the system.

A number of search engines for corporate computer networks have recently been released. Within the framework of the Netscape Web server, Enterprise, a search engine based on the construction of indexes is implemented. Netscape has also issued a separate retrieval server, Catalog Server, combining functions of construction of the indexed lists and directories. The Index Server search engine, based on the construction of indexes, includes a composition of a Microsoft Windows NT Server operating system beginning from the fourth version.

1.2.5. E-Mail and News Systems

Apart from Web services, two of the most popular network services are electronic mail (e-mail) and news systems (Network News). These network services operate at the application level of the OSI model and are intended for the processing and delivery of electronic messages in a distributed-network environment. However, in contrast to e-mail, where the recipient address is specified, news systems specify only the message subject.

The e-mail message is delivered to the recipient whose address is specified in the message header. Anyone who has subscribed to the mail-list also indicated in the message header can receive the messages or "articles" of the news system. Therefore, e-mail messages are similar to normal letters and the news system is similar to newspapers and magazines delivered by subscription.

Modern e-mail programs and news systems allow users not only to create messages in HTML format used in Web documents, but also to attach any files to transmitted messages, for example, audio or video data.

Exchange of Electronic Messages

The Guidelines of the X.400 Standard

Using e-mail for on-line information exchange between people, both inside a particular organization or company and beyond its limits, essentially boosts cooperation efficiency.

In general, the structure and principles of the operation of any modern mailing system correspond to X.400 standard guidelines, which stem from the activity of the International Telecommunication Union (CITT in the French transcription, or ITU in English). These guidelines embrace all aspects of the environment's construction of message management: the scheme and components of their interaction; control protocols and transmission protocols; message formats; and the rules of their conversion.

The exchange system of electronic messages within the framework of a large computer network represents a set of post offices integrated into the network environment (Fig. 1.23). The local-area network can contain only one post office, serving all users.

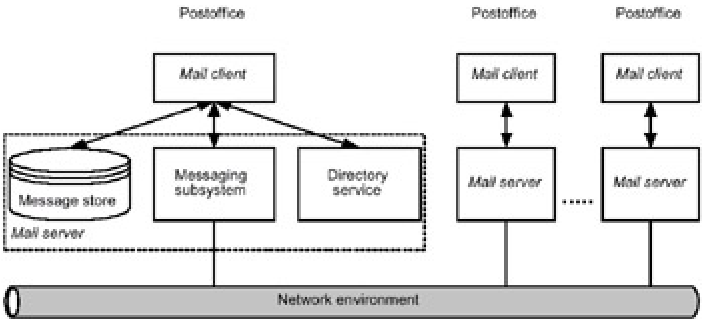


Fig. 1.23: The messaging system

The operation scheme of each post office is implemented according to client/server technology, where the mail server implements the query processing received from mail clients. E-mail programs installed on workstations act as mail clients. Programs that act as mail servers are most often installed on network servers.

The principal components of a mail server are:

- A message-delivery subsystem performing transfer of messages
- Message store, intended for intermediate storage before transfer to the recipient or transmission to mail clients

Depending on the scale of the network, message transfer by a message-transmission subsystem can be performed either directly to the recipient's mail server, or through intermediate mail servers, according to the routing rules defined by the employed protocol of electronic-message exchange.

Each mail server's message store consists of users' mailboxes, and also of intermediate mailboxes used for the storage of messages in transit. Storage allows for messages to be sent in the most convenient way possible. In addition, storage does not require continuous connection to the Internet for the user to access his or her mail server. In this case, mail clients can retrieve their messages upon connection to the server.

Thus, the main ways to a send mail messages involve the following:

- Sending the message by the mail client through storage, when the user, with the help of an e-mail program, puts the message to be transferred directly into message store. It is then selected from there and sent by a subsystem of message transmission.
- Sending the message by the mail client through a message-transmission subsystem, when the message is transferred directly to a given subsystem and delivered further by its resources.

The main ways to retrieve messages include:

- Receiving the message from store by the mail client, in which case the message-transmission subsystem delivers the message to the recipient's mail box for further processing by the user's e-mail program

- Receiving the message from a transmission subsystem by the mail client, when the given subsystem sends the message directly to the recipient's e-mail program

The first of the above-mentioned methods to send and receive electronic messages is used when computers are not permanently connected to a mail server, and the second one is used when there is a permanent connection to the network.

The directory service, supporting names, descriptions and network addresses, is an additional, though very important, component of a mail server. Mail clients must include an address book, also intended for storage of names, descriptions and network addresses. However, in contrast to the directory service, which stores the whole reference information, the address book is filled in by the user. Actually, the address book is a user-supported directory service, and should provide interaction with the network's general directory service.

The address book and the network's directory service should be able to create, save and retrieve mailing lists. The mailing list represents a group of electronic addresses that messages can be sent to simultaneously. After it is sent to a mailing list address, the message will be delivered to all recipients included in it.

For a description of an e-mail message format, X.400 guidelines adopted a paradigm of an envelope and its contents, something that is used in conventional mailing systems.

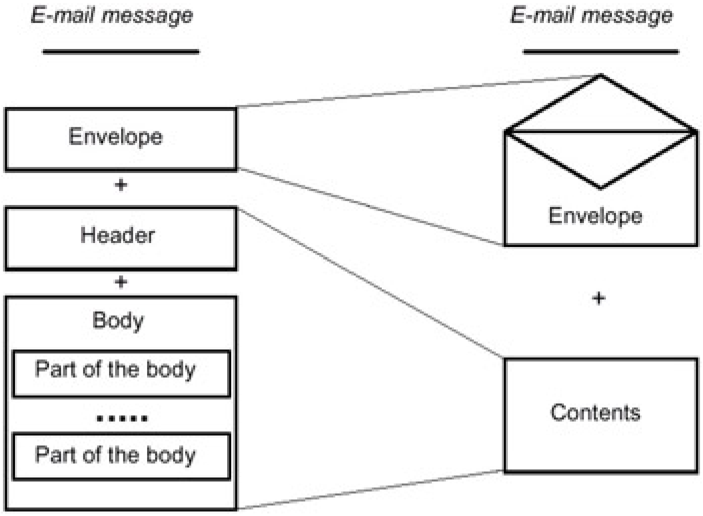


Fig. 1.24: The electronic message's structure

As is customary, the envelope contains comprehensive information regarding where and to whom the letter should be delivered, including the sender, return address, and a checkmark indicating urgency of delivery. Thus, the system does not need to know anything about the letter's contents. On the basis of the information specified on the envelope, the system performs the necessary delivery routing and transmission, with probable intermediate storing. Intermediate points and transport resources are carried out by transit mail servers.

The envelope can have a special checkmark if necessary, in the form of an electronic "stamp" by each mail server through which it passes on its way to the recipient. This, in particular, allows the system to eliminate originating of loops automatically.

The envelope's contents consist of a header and the body. The header usually includes a copy of the information indicated on the envelope, and additional fields defining extended message properties. The body can be composite and include different types of information, such as text, graphics, documents of different formats, attached files, etc.

X.400 guidelines also offer automatic notification messages to the sender, confirming that the message sent has been delivered and/or read.

The SMTP Protocol

Despite its powerful theoretical base and practically flawless architectural design, X.400 isn't widely employed, outside state and financial institutions. Its flaws lie in its excessively complex implementation and costly system installation and maintenance. However, common management principles of the messages originating from this standard have become a basis of modern mail services.

The most common e-mail protocol is the Simple Mail Transfer Protocol (SMTP), which has become the message-exchange standard of the Internet and Intranet. This protocol's popularity can be explained by the relative simplicity of its implementation and its wide range of expansion possibilities. Also, there is no threat of damage resulting from incompatibility with existing versions of mail systems. A particularly important factor is the fact that its specifications are widely available and they can be obtained free of charge.

The SMTP protocol uses TCP as the transport protocol and it is used for the implementation of two functions (Fig. 1.25):

- The transfer of transmitted messages from mail clients to the mail servers of these clients
- The transfer of messages between mail servers

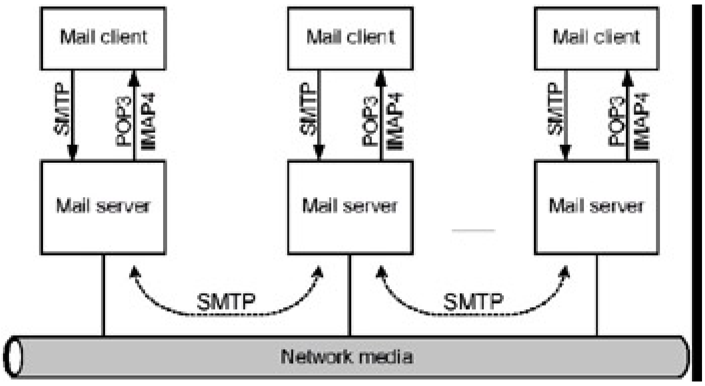


Fig. 1.25: Message exchange by the SMTP protocol

The initial version of the SMTP protocol supported the restricted instruction set of both commands and services for receiving and sending messages. Its extended variant (Extended SMTP, or ESMTP) provides the standard possibility of further extension, and support of such functions as confirmation of delivery (Delivery Notification Request, or DNR). An agreement of the maximum size of messages transmitted between servers, and a forced initialization of the transmission of accumulated mail were also recently developed.

However, upon autonomous application, the SMTP protocol isn't free from drawbacks, the most important of which are:

- An inability to authenticate incoming connections
- The orientation to transmission of only text information
- An inability to encrypt the transmitted messages

To overcome these problems, SMTP can be used together with complementary protocols and standards.

The absence of authentication resources for incoming connections precludes SMTP from using the client-access service. The classic SMTP mailing system requires that the client have file access to the mailbox for message retrieval and operation. To implement operations in client/server mode, Post Office Protocol, or POP, was created. The version POP3 has been the most successful to date, and is widely used in modern SMTP systems. The POP3 protocol allows the user to retrieve incoming messages from a mailbox on a mail server, with the help of an e-mail program.

The most advanced implementations of POP3 support authentication with name and password encryption, and also traffic encryption according to the Secure Socket Layer (SSL) protocol. However, use of the POP3 protocol does not allow viewing of the message's characteristics, without first loading it onto the client's server. The IMAP4 protocol was developed for solving the problem of viewing and manipulating the mail message's properties directly on the server. It also aims to deal with a number of other functional constraints that have been designed. Unlike POP3, the IMAP4 protocol is equipped to do the following:

- View the message headers to determine which of them should be read (loaded from a mail server on a workstation)
- Selective loading of the messages' parts from the server in the MIME format
- Search the messages on the server
- Create both standard message attributes and those defined by the user, for example, for identification of workgroups, projects and so on
- Organize a hierarchy of folders outside an inbox on the server
- Allocate the mail messages into folders that have been created on the server, and ensure their upgrade and long-time centralized storage
- Centralized backup and restore of the mail messages stored on the server

Modern e-mail programs, for example, Microsoft Outlook Express, support both POP3 and IMAP4 as the client mail protocols.

Even despite the fact that while, for receiving messages, the POP3 or IMAP4 protocol is used by the mail client, sending messages from the mail client on the server is still implemented according to the SMTP protocol (Fig. 1.25).

Initially, SMTP systems were designed for the transmission of exclusively textual information (text only), and were not oriented towards transmission of national characters, i.e. they used a 7-bit character set.

The UUENCODE standard, allowing the insertion of arbitrary data converted from binary into text mode directly into the text of the message, helps to solve the problem of binary file transmission. However, it is difficult to call this approach universal, as, in general, the recipient does not have any information on the type of transmitted data or the application which has generated it.

In accordance with the Internet's growth, software becoming complicated and the active introduction of multimedia, it became necessary to create a universal format for typing and presenting binary information and text containing national characters. Multifunction extensions of Internet mail (Multipurpose Internet Mail Extensions, or MIME) have become this universal format. The MIME format has turned out to be extremely successful, as the unlimited extension of both supported data types and national code tables have been included into it.

Using MIME allows a user to include audio or binary information or a digitized video signal into an electronic letter, and also to attach any files to the transmitted message. With the help of MIME, it is possible to create and read electronic letters containing information in RTF and HTML formats, in particular, those with different text fonts, scanned images and spreadsheets.

An important problem, in terms of data transfer through SMTP systems, is the issue of privacy. In order to solve problems of information protection, the standard for message-body encryption, the so-called secret multifunction extensions of mail (Secure MIME or S/MIME) were created. However, this protocol is not able to prevent the interception of message headers.

According to the guidelines of the X.400 standard, SMTP messages consist of anevelope and message content. The content, in turn, has a header and body. They function in absolutely identical ways. The fields in which headers are composed are determined by the message body's format (UUENCODE or MIME). No field is mandatory, but, as a rule, such fields as To:, From: and Subject: are specified. While using the MIME format, the MIME-Version: field, in which the version number of the standard MIME is specified, should be necessarily specified in the header.

Addressing and Routing

In order to avoid contradictions during message exchanges, each user of the mail system should have his own, unique mail address. This address should identify the user, not the computer. The assignment scheme of unique addresses for the users in any mailing system is called *addressing*.

With a particularly excessive amount of links in the network, message delivery to the recipient should be performed through the best route - from the sender's mail server to the recipient's mail server. The process of the selecting next point on the message's route to a receiving mail server, known as *routing*, is performed on the basis of special tables. The mail message routing is implemented over the packet routing at the network level. After selection of the next transit mail server, according to mail messages' routing rules, routing at the network level of OSI model is performed, to provide optimum delivery to the selected transit server.

The addressing and routing schemes in a specific e-mail system are determined by the used protocol of mail-message exchange. The administrator responsible for ensuring support of interaction between several heterogeneous messaging systems should be aware of addressing and routing methods used in each of them. Let's consider the addressing and routing schemes on the example of the SMTP protocol.

Addressing in SMTP

For SMTP-based systems, an intuitively clear, easy and, at the same time, very powerful hierarchical system of addressing, similar to the one used in the Internte name service (Domain Name Services or DNS) is used. This system can provide unique addresses to an unlimited number of user. The SMTP mail address is written as follows:

mailbox@domain

Here *mailbox* is a symbolic name of the user's mailbox (with a length of up to 63 characters), and *domain* is the unique name of the mail domain in which the mentioned user is registered (with a length of up to 255 characters). The combination of the names of the mailbox and mail domain creates a unique user identifier.

The mail domain stores complete information on the system's position in the hierarchy of an organization's mail space (Fig. 1.26). In the domain name, the name of each subsequent hierarchy level is separated from the previous one by a dot. The analysis of a domain name is performed from right to left. The uppermost level, the root domain, corresponds to the type of organization (for example, com-commercial, gov-governmental, org-public), or geographical location (for example, ru-Russia, and fr-France). First-level domains, usually representing the organization name, go next.

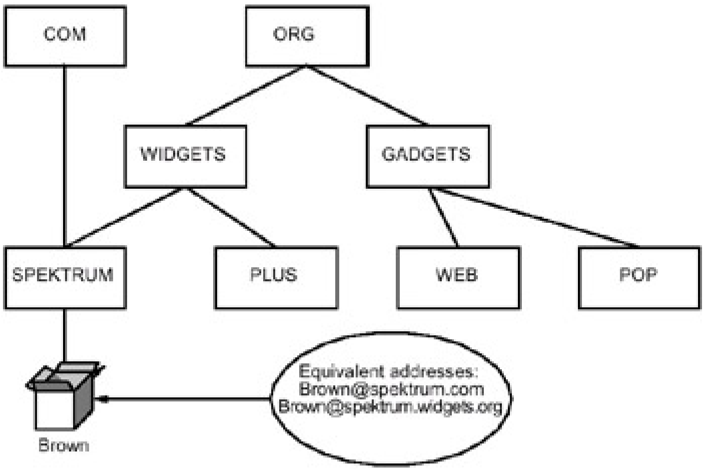


Fig. 1.26: An example of a hierarchical diagram of addressing in SMTP

Registration of domain names of the first level is performed by the international Internet center (Internet Network Information Center or InterNIC). Companies are responsible more often for domain name assignment of a lower level. As organizations are not forbidden to register some domains at the same time for their own needs (for example, spektr.com), the user can have more than one SMTP address. Moreover, modern SMTP systems often allow one to assign aliases to a mailbox.

In the example given, spektr.spb.ru is the subdomain of spb.ru, which is the subdomain of ru. The company *Spektr* has two registered names, and each user can have two mail addresses.

Routing in SMTP

In order to a make SMTP server deliver mail addressed to the recipient Brown@spektrum.com, it needs to know the IP address of a mail server serving the domain SPEKTRUM.COM beforehand, having applied with the appropriate query to the DNS server. In the DNS name service, a special type of resource record servicing such types of queries is provided-MX or Mail Exchanger. The given recording has the following format:

domain MX [cost] hostname

Here *domain* is the name of the mail domain that the recipient belongs to; *hostname* is a symbolic name of a mail server which has information on how to implement the further delivery of the message; and *cost* is the relative cost of delivery through this computer. In order to obtain the IP address of the computer with a *hostname*, a search of the address-resource recording in DNS is performed. When there are several MX recordings for the same domain, an attempt will be made to make a connection to that mail server, whose delivery cost will be lower. If such a computer is unavailable or overloaded, the computers with greater cost values will be used.

Thus, in order to deliver the message addressed to the recipient Brown@spektrum.com, first of all, the query to the DNS server to obtain the resource records' list with the MX type will be executed. If the list is not empty, the address of the computer name with the lowest delivery-cost value (again through DNS) will be obtained, the connection will be made and the mail will be sent. If there is no MX record for the domain spektrum.com, the domain will be treated as the computer name. An attempt will be made to get its IP address and to deliver the message directly.

In connection with that, the DNS service of names is supposed to be a source of static information. Therefore, the routing scheme of SMTP messages is static. According to the SMTP protocol, in networks with no direct connection to the Internet and which do not use features of MX records of DNS, separate, static routing tables of mail messages can be used.

Depending on routing possibilities, the SMTP server can play one or more of the following roles:

- Mail exchanger: on a computer directly connected to the Internet, which performs message delivery directly to recipients inside an organization. In an organization, some computers can have different or identical delivery cost values.

- Relay: the computer can accept incoming mail traffic on behalf of other domains that do not have direct and/or continuous connections to the Internet and that, as a rule, do not belong to organizations whose domains it serves. Each separate domain should not have more than one relay server.
- Smart host: the computer is able to realize the transfer of messages on the basis of its own static routing table. One of Smart Host's functions is the ability to rewrite the recipient's address on an envelope. It can also do this for the sender prior to any further message transfer.

The majority of modern implementations of SMTP servers allows users to combine all above-mentioned functions onto one computer.

News Transmission

The task of data exchange is a pertinent issue in computer networks, along with that of replicating electronic messages for shared use of information of different subjects. News systems intended for dealing with this task distribute electronic articles for every subject sent to all network users who have subscribed to this subject.

The following two main methods of replicating are used within news systems:

- Sending articles to members of mailing lists
- Usage of a distributed news database, from which users can get any articles as needed

In the first case, replicating servers keep lists of subscribers' addresses for articles on different subjects. If it is necessary to duplicate any articles on particular subjects, the user sends the given article as an e-mail message to the appropriate server's address. That one, in turn, immediately delivers copies of the given article to electronic addresses of users who have subscribed for the subject specified in the article. Replicating papers through mailing lists becomes inefficient when there is a significant increase in the amount of subscribers. This has to do with the complexity of maintaining up-to-date mailing lists and, also, with additional requirements to reliability of those servers doing the mailing.

This problem can be eliminated, however, with replicating by using a distributed base of news articles, which the users get themselves. The given method of replicating forms the basis of the USENET service, which is the most popular on the Internet network and has become the conventional standard for news systems.

The USENET service was initially oriented to operating in client/server architecture and allows for the support of bases of news distributed between several servers with the ability to automatically replicate incoming messages. The NNTP protocol (Network News Transport Protocol) was developed for the interaction of news servers with each other, as well as of clients with servers.

USENET uses the message format and addressing mode concurrent with those for SMTP-systems. The information specific for the news service is specified in the extended fields of the message header. It allows the development of client programs for reading mail and news, on the basis of a uniform code. It also gives one the opportunity to use existing SMTP networks to obtain news in places where direct access to the news server is impossible for various reasons. Besides that, the service messages intended for the exchange of control information between news servers are used. Due to service messages, the process of automatic creation, deleting of subject newsgroups and the liquidation of outdated articles is simplified.

A uniform hierarchical tree organized by subject criterion represents the information kept in USENET. In this context, USENET plays the role of a subject directory containing people's opinions on different subjects. Articles on the same subject are placed in subject groups called newsgroups. Newsgroups, in their turn, can be contained inside other groups, thus making subject hierarchies. Each level of a hierarchy is called a category. Within the framework of a category, each group has a unique name. The complete reference name of a group is gained by the consecutive addition of category names from left to right, moving downwards from the top to the root of the hierarchical tree. The names of categories are separated by dots. For example, the name relcom.comp.security corresponds to a newsgroup on computer security of the Russian Relcom network.

Hierarchies, or their special branches, are replicated between news servers forming USENET space. A single article acts as a unit of replication. The publisher/subscriber scheme is used during replication. Each USENET server can be subscribed for some groups submitted by other servers. Simultaneously, it can publish some groups located directly on this server including groups received by subscription. In USENET terms, replication is called feeding. Depending on which server poineers this process, two types of feeding are distinguished:

- Pull feeding-when a server, waiting for new articles to arrive, accesses the publisher
- Push feeding-when a server that the has new articles makes an attempt to transmit them to a subscriber

One more important point of the USENET service is its ability to create moderated newsgroups. In moderated groups, each new message is automatically redirected to the person acting as the censor or moderator. If the message does not contradict the conference charter and is approved by the moderator, it becomes publicly accessible for reading. Otherwise, it is simply deleted.

Since the news service was initially formed as a means of maintaining publicly accessible information, the assignment functions and verification of rights of access to different branches of the directory were not provided in it. Most existing news services are capable of performing only a single check of the user's name and password while establishing a connection with the server, after which all articles become accessible to the client. Furthermore, it is neither possible to authorize servers nor to provide service messages. As a result, the large-scale utilization of USENET is justified only for organizing public newsgroups with an anonymous access mode.

E-mail and news systems specially designed for corporate networks. For example, the Microsoft Exchange system, which is compatible with the USENET news service, possesses capabilities of reliable checking of users' authentication and the differentiation of their access to articles by subjects.

1.3. Secure Scaling of Computer Networks

1.3.1. General Information

One of the main requirements of a modern computer network is scalability, which allows the network to be large and complex enough without hindering its performance level, security or manageability.

The scalability of computer networks is made possible by their expansion and internetworking, in the presence of effective support resources of computer security and network control.

Depending on their scale, geographic location and ownership, computer networks are categorized as local, corporate, regional or global.

A local-area network represents a group of computers connected to each other by information-transmission links, and disposed within closely located buildings.

A local-area network can be divided into separate segments. Each network segment is a part of a local-area network and, beyond these bounds, the only message packets propagated are those addressed to computers not belonging to this segment. The method of multiple access is implemented within the bounds of a local-area network's segment when a message packet sent by the segment computer is delivered to all remaining computers in this segment and is only received by the computer it is addressed to. The remaining computers ignore this message packet.

The division of a local-area network into segments improves its performance by reducing traffic. This ties into the fact that packets of separate computers of a segment, whose recipient is located within the same segment, are not spread through the whole network. However, it is necessary to take into consideration that the improvement of a local-area network's performance by its division into segments will only be provided in cases where those segments correspond to workgroups where the intensive information exchange takes place. If a local-area network is not divided into segments, the given network will consist of one network segment.

Corporate, regional and also global networks unite distributed local-area networks with the help of communication links. The main feature of a corporate network is that it belongs to one organization; the main feature of a regional one is that it covers some area, for example, one city; and the main feature of a global one is its coverage of global areas, for example, countries and continents. A large regional network can combine smaller regional and corporate networks, and a global one can combine different types of networks. As a rule, unlike corporate networks, regional and global networks do not belong to any particular organization. A vivid example of a global network is the Internet, which has no owner, although the computer networks involved in it belong to different organizations.

For expanding and integrating computer networks, the following different types of hardware and software devices are used:

- Repeaters, providing amplification and, if necessary, splitting of an electrical signal to increase the size of a local-area network's segment
- Bridges and switches, intended for the dividing of a local-area network into segments, and also for the interconnection of obtained segments and small local-area networks
- Routers, intended for connection to global networks and for the interconnection of local-area networks and their large parts
- Gateways, performing routing functions, which are intended for the interconnection of computer networks using incompatible protocols.

The above-mentioned devices operate at different layers of a standard model of network interaction (the OSI model, or Open System Interconnection).

Repeaters that permit a network segment's size to be enlarged at the expense of amplifying and splitting an electrical signal operate on the physical layer of the OSI model (Fig. 1.27). Notice that, at the moment, repeaters expanding their area of influence to the data-link layer are being brought into use in order to provide safe usage of network addresses at the physical layer. We shall discuss these below.

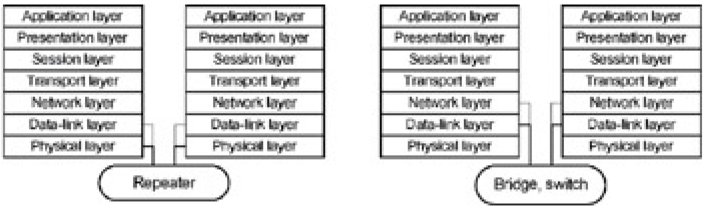


Fig. 1.27: The layers of the OSI model with repeaters and bridges

Bridges and switches used for interconnecting segments of a local-area network and, also, of small local-area networks-operate at the data-link layer of the OSI model (Fig. 1.27). Those segments and local-area networks should operate under the same protocols of the intermediate and upper layers of the standard model (from the network to the application layer). The protocols of the data-link and physical layers can differ. Accordingly, the bridges and switches provide an interconnection of segments and local-area networks with different topology, for example, Ethernet and Token Ring.

Some bridges and switches, for example, the routing bridge, apart from their functions, support some functions of the network layer of the standard model for optimization of data transfer. Contemporary switches allow the interconnection of segments and local-area networks operating under different protocols of the physical and data-link layers and, also, of the network layer, for example, operating under protocols IP and IPX.

Routers used for the combining of local-area networks and their large parts and, also, for connection of local-area networks to global ones, operate at the network layer of the OSI model (Fig. 1.28). However, some intellectual routers that support improved functions of filtering message packets can handle packets of the transport layer of the OSI model.

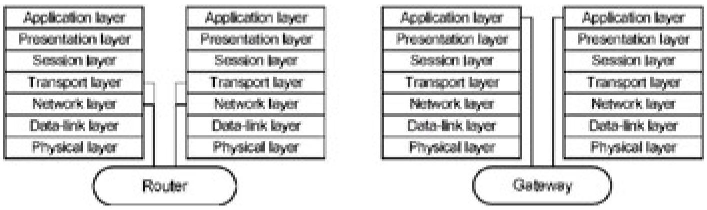


Fig. 1.28: The layers of the OSI model with routers and gateways

Unlike bridges and switches, routers search for the best route when transmitting message packets between segments of the network or local-area networks. Bridges and switches do not implement a function of selection of the best route, but only transfer message packets from one segment of a local-area network to another, or from one local-area network to another.

For the interconnection of segments and local-area networks with different protocols of the network layer. For example, with the protocols IP and IPX, it is necessary to use multi-protocol routers providing the exact analysis of IP and IPX message packets and their transmission to appropriate segments or local-area networks.

There are also combined network devices such as the bridge/router (or brouter) that operate at a normal mode like multi-protocol routers and, on obtaining the packet of an unknown network protocol, handles it as a bridge.

If local-area networks are built on protocols distinguished, not only at the physical, data-link and network layers of the OSI model, but also at the upper layers, then, for the interconnection of such networks or segments of the network, specialized computers called gateways should be used. Gateways, which usually operate at the application layer of the OSI model, route the transmitted information and provide protocol conversion for all layers of a standard model of network interaction. For example, gateways are necessary for the connection of large, obsolete computers (mainframes) that have a centralized architecture, to modern local-area networks. Usually, gateways allow users of integrated systems to take advantage of such tools as e-mail, transfer of files and database access.

1.3.2. Usage of Repeaters

Assignment and Types of Repeaters

Repeaters are hardware or hardware/software devices of the physical layer of a standard model that provide amplification and, if necessary, split electrical signals to expand to a size of a local-area network's segment. In modern network topology, such as Fast Ethernet or Token Ring, the segments of a local-area network are connected to a bridge, switch or router only through repeaters (Fig. 1.29).

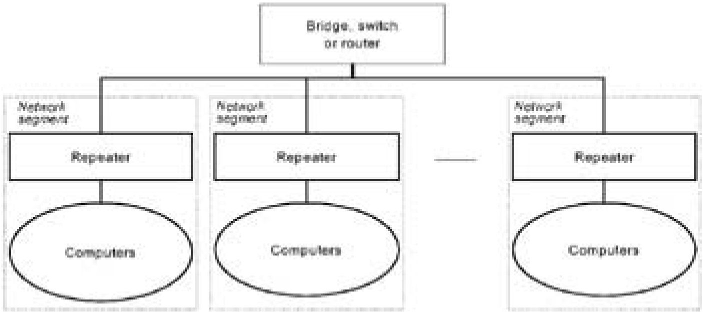


Fig. 1.29: The connection of network segments

With ring and star topology, for example, in Token Ring and Fast Ethernet networks, the role of repeaters is played by active concentrators, or hubs, as they are also called, by uniting groups of computers in a network segment. Modern switches also perform the functions of repeaters.

Let's recall that active concentrators implement functions of both switching and signal amplification. Passive concentrators only implement functions of signal dividing. As a result, many computers, for example, 16 or 32 hosts, can be connected to an active concentrator, and the cable connections can be longer, for example, from 45 to 200 meters, depending on the type of cable. A passive concentrator is used in addition to an active one, and provides connection to only a few computers. Thus, the largest allowed length of a cable connecting the computer to a passive concentrator should not exceed several dozen meters.

For example, with "common-bus" topology in Ethernet networks such as 10-Base5, repeaters are utilized to connect separate segments with a common bus and, also, to increase the length of one segment. In the latter case, segments of a common bus with the greatest possible length are connected to each other through repeaters.

The repeaters for "common-bus" topology are referred to as linear repeaters, as they contain one input and one output port. In active concentrators, all available ports are both input and output ports.

Repeaters are transparent network devices without addressing, operating with the same speed as the segments of the network connected by them. Apart from amplification and division of electrical signals, repeaters provide signal-restoring and error-handling functions. If the signals are distorted and/or noisy, but still visible for the repeater, then, prior to forwarding to other ports, the repeater restores them. In cases where the error signals are received at any repeater port, the repeater blocks their further distribution.

At the present time, repeaters operating not only at a physical, but also at the data-link layer of the OSI model with the purpose of supporting the safe usage of network addresses of the physical layer called MAC addresses (MAC, or Media Access Control), are coming into use. A MAC address usually consists of 48 bits, is unique for each site of a local-area network and is assigned to a network adapter during its manufacturing. All network adapters have different MAC addresses. The only exception is on the ArcNet local-area network which has adapters with 8-bit addresses assigned by the network administrator.

With the purpose of supporting the secure usage of MAC addresses, repeaters can perform functions of selective encryption of packets of the data-link layer and, also, the filtering of MAC addresses.

Selective Encryption of Packets of the Data-Link Layer

The function of packet encryption of the data-link layer facilitates protection against network traffic monitoring (network sniffing).

In the usual mode, the site of a local-area network ignores message packets that are not addressed to it. It becomes possible to monitor network traffic when changing over a network adapter into the monitor mode of operation, when it accepts all incoming message packets. Changing over a network adapter into the monitor mode and analysis of network traffic are implemented by means of a special program: the network analyzer. Such programs are widespread, and are used to search for errors and the analysis of network perfomance. For monitoring network traffic in any segment of the network, it is enough to start the network analyzer on any computer of the segment.

In order to prevent the monitoring of network traffic by encryption of packets of the data-link layer, the repeater should know the MAC addresses of the computers connected to it. These MAC addresses can be set by the administrator with the help of special software or can be found by the repeater using the bridge operation algorithm.

If the MAC addresses of hosts connected to the repeater are known, the repeater acts as follows. When forwarding message packets to output ports, it performs encryption of those point-to-point packets for which the recipient's MAC address does not coincide wiht the MAC address of the computer connected to the port. Point-to-point message packets are understood as packets addressed to one host. If the packets are addressed to all hosts, or group of hosts, of a network segment, they are known as broadcast or group packets. The packet forwarded by a port to which the recipient of this packet is connected is not encrypted. Therefore, the computer connected to the encoding repeater receives all packets, but only the broadcasting and group packets and the packets intended for this computer will not be encrypted.

Fig. 1.30 shows an example of encryption of a transmitted message packet. Computers from A to F are connected to the appropriate ports of the repeater. A packet transferred by computer A for computer E will be sent to all computers, except for E, as encoded by the repeater.

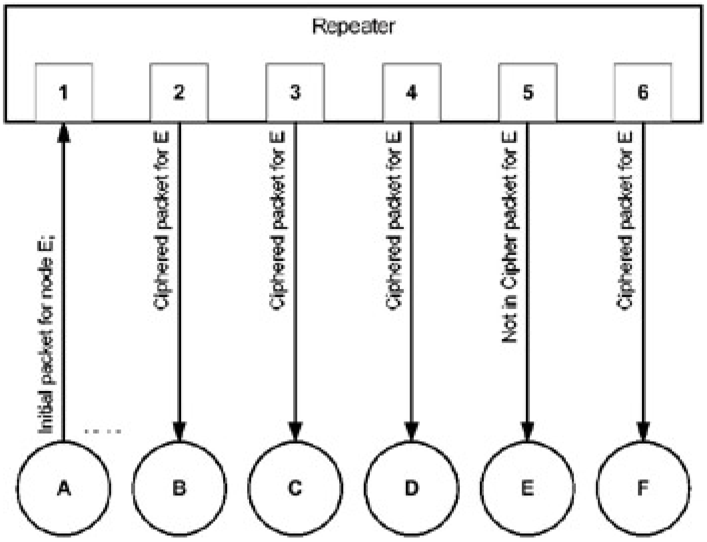


Fig. 1.30: The transmission of a packet from computer A to computer E

Broadcasting packets are not subject to encryption, as they are intended for all hosts of a network segment. Group packets are also not encrypted, as the repeater is unable to find out for which hosts they are intended. Such information is only available to recipients at the network layer. But this is not a problem as group packets seldom contain secret data.

Filtering of MAC Addresses

Filtering of MAC addresses by repeaters is oriented towards support of authenticity of message packets and protection against MAC addresses spoofing.

To obtain access to the resources of a local-area network, the hacker can connect his own computer to it. This might be necessary in cases where the hacker does not have physical access to network computers. Besides that, most network adapters allow users to program and/or to change their MAC addresses dynamically. For that reason, it is not difficult to create a program that sends packets with different MAC addresses of the sender by performing substitution of MAC addresses. The purpose of such an attack is to deceive a network operating system and another software bound with the data-link layer to make them do things that they would not usually do. The HACK.EXE program, which performs substitution of MAC addresses (address spoofing) to receive supervisor privileges on any NetWare 3.11 server, is an example of this. Address spoofing can also be used to deny an authorized user service. The filtering of MAC addresses affords protection from similar attacks.

To enable filtering, the MAC addresses of all network computers from which message packets may be received, are set to the repeater. With knowledge of authorized senders' MAC addresses, the repeater filters out the packets that have non-registered MAC addresses when receiving message packets. Therefore, message packets with unknown MAC addresses are ignored for further transmission, and are not delivered to the repeater's output ports. Upon detection of the host with a non-registered MAC address, the repeater completely disables the port to which this host is connected.

To increase security, repeaters with security functions should be installed in a locked distribution cabinet.

1.3.3. Segmentation of the Network with the Help of Bridges

General Information

Bridges, which operate at the data-link layer of the OSI model, are used to divide a local-area network into segments, and for the interconnection of obtained segments and small local-area networks.

Within the boundaries of a local-area network's segment, multiple access is implemented when a message packet transmitted by a segment's computer is delivered to all remaining computers in this segment and received by the computer it is addressed to. The remaining computers ignore this message packet. Hence, if a large local-area network consists only of one segment, network performance is degraded, because of an increase in traffic. When dividing a local-area network into segments with the help of bridges, switches or routers, only those message packets addressed to a computer not included in it are spread beyond the boundaries of each segment. As a result, network performance is enhanced, on account of less traffic.

An optimal ratio of traffic intensity within segments to traffic between segments is believed to be 80:20. To adhere to this rule, two requirements need to be met:

- Dedicated segments of the network correspond to workgroups, within whose boundaries intensive information exchange is carried out.
- Each server is residing in the same segment where its user is located.

Bridges and switches are simpler devices than routers, which do not implement a selection of the best route, but only transfer message packets from one segment of a local-area network to another, or from one local-area network to another. In addition, unlike routers, bridges and switches are transparent network devices that do not require the special setup of other net hosts.

Keeping in mind the complexity of administration, the considerable expense of routers and the fact that, at message-packet transmission between sparse segments of a local-area network, it is not necessary to search for the best route, the effectiveness and high-quality performance of using bridges and switches to connect local-area network segments becomes obvious.

The bridges and switches that have been developed for dividing a local-area network into segments and interconnecting obtained segments are similar in their assignment but different in terms of operation. Generally, bridges have fewer functional capabilities, and their internal organization is less complex than that which exists in switches.

The Technology of Bridge Operations

The interconnected segments of a local-area network are connected to network adapters acting as bridge ports (Fig. 1.31). Each connected network segment is connected to a network adapter of the same type as the segment. Most often, the bridge has from two to four ports.

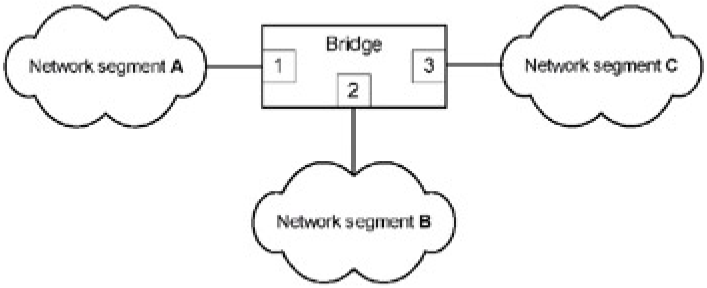


Fig. 1.31: The connection of segments of a local-area network with a bridge

Any packet sent by the computer of any of the network's segments arrives at a bridge port to which this segment is connected. If the recipient of the given packet is in another segment of the network, the bridge routes this packet to a port to which the segment and the recipient are connected. This process is called forwarding. The packet is said to be forwarded if it is received by one port of the bridge and is transferred through another.

In cases where a packet received by a bridge port has the address of the recipient falling within the boundaries of the segment connected to this port, the packet is not forwarded. This process is called filtering. The frame is filtered if it is received by one bridge port and is not forwarded by another.

Decision-making by the bridge, i.e. deciding whether to forward or fiter the received packet, is based on the storing of MAC addresses of the computers belonging to segments connected to bridge ports. These addresses are stored in the bridge's memory as a table, by assigning the port number that connects the segment and the computer to the MAC addresses of each computer.

As a part of its operations, the bridge files in the address table. After the first connection to the network, the bridge, over a period of several seconds, stores the addresses of all the active hosts contained in segments connected to it in its memory. If packets arrive with a MAC address of a recipient unknown to the bridge at any port, the bridge forwards this packet to all other ports.

For each MAC address, the address table has three fields: the address, the number of the port where the address was noticed the last time, and the age of the MAC address.

When obtaining each packet with an unknown sender's MAC address, the bridge registers the address, the number of the port that has received the packet, and it sets the value of the age of the registered address to 0. The age of addresses in the table is incremented by 1 every second. When a certain value, called the age limit, is reached, the information on this MAC address is deleted from the table. This process is called aging. The field of the age of each MAC address is reset when the bridge receives the packet with the same MAC address of the sender. As this takes place, the port corresponding to the MAC address is also updated.

The given technology of filling and updating the address table guarantees the maintenance of the information stored there. With the addition of computers to network segments and, also, when computers are moved from one segment to another, the information in the address table will be updated at the appropriate time. Also, when computers are unlinked from network segments, unnecessary information will be removed from the address table.

Upon obtaining each packet, the bridge, first of all, updates the address table in accordance with the described technique, and only after that makes a decision as to whether to forward the packet it has received or to filter it.

The broadcasting and group packets are forwarded to all bridge ports, with the exception of the port that has received this packet. Such forwarding of group packets is performed so that the bridge can find out for which nodes they are intended. This information is only accessible to receiving computers at the network layer. In relation to the fact that bridges and switches are transparent to permit the passage of broadcasting packets in all directions, the local-area network, or part of it, made by the segments' interconnection by bridges and switches, is called the broadcasting area.

If a point-to-point packet is received at a bridge, the appropriate line of the address table of the MAC address of this packet's recipient will be searched and analyzed. Table 1.2 presents the decision, determined by the results of the search and analysis.

Table 1.2: Decision on Forwarding

Result of search and analysis	Decision
The address is not detected in the table	The packet is forwarded to all ports, except for the port at which it arrived
The address is found in the table, and the number of the port appropriate for it does not coincide with the one that has received the packet	The packet is forwarded only to a port whose number is indicated in the address table
The address which exists in the table is present, and the number of the port appropriate for it coincides with the one which has received the packet	The packet is filtered. It is not forwarded to any port.

All bridges work in accordance with this technology. However, some of them can make the decision to forward on the basis of more complex rules. Many bridges allow the manual setup of separate units of the address table, so-called static units, which are never deleted from the table. Such a setup is required for passive devices, for example, for the network printer, which can be "silent" during a longer period of time than the valid maximal age of its MAC address. When it is necessary to print something, the printer will be sent a packet, and when information about the MAC address of the printer from the address table is deleted, the bridge should forward this packet to all ports.

The Architecture of the Bridge

Bridges have a rather straightforward architecture, and consist of a specialized computer with two or more network adapters acting as bridge ports (Fig. 1.32). Each port receives all incoming message packets. Its specialized software checks each received packet, fills and updates the address table, and makes a decision about forwarding.

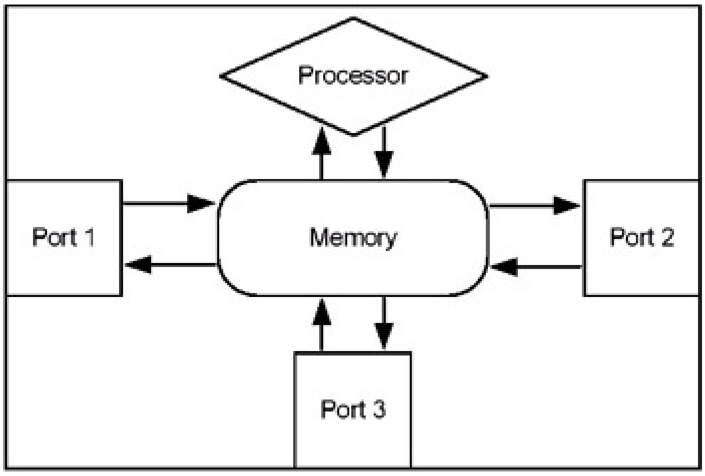


Fig. 1.32: The simplified architecture of a bridge with three ports

Bridges with one processor can simultaneously handle only one packet. Accordingly, such bridges, as a rule, have no more than four ports, to support a high speed of processing of incoming packets. Multiprocessor bridges have more ports, but they are much more expensive.

The functions of the bridge can be performed by a standard computer that is connected to the network, with special software and several network adapters, each intended for one of the network's connected segments. If the server is used as a bridge, such a bridge is referred to as internal, and if a workstation is used as a bridge, that bridge is an external one. When using a workstation as a bridge, the workstation should not be used for any other functions, as any failure of the user's program run on it can result in an interruption of information exchange in the network.

Segmentation of Complex Local-Area Networks

If the local-area network is large enough, in most cases using one bridge to divide it into segments and for the interconnection of these segments is not enough. The transparency of bridges enables the creation of rather complex networks, in which it is possible to provide backup routes between network segments in cases of failure of intermediate segments of the network. For example, Fig. 1.33 shows the scheme of a network with several bridges and a backup route between segments A and D. Upon the failure of segment C, a connection between groups of segments {A, B} and {D, E, F} will not be disrupted, as an alternative route, via the bridge with number 3, will operate.

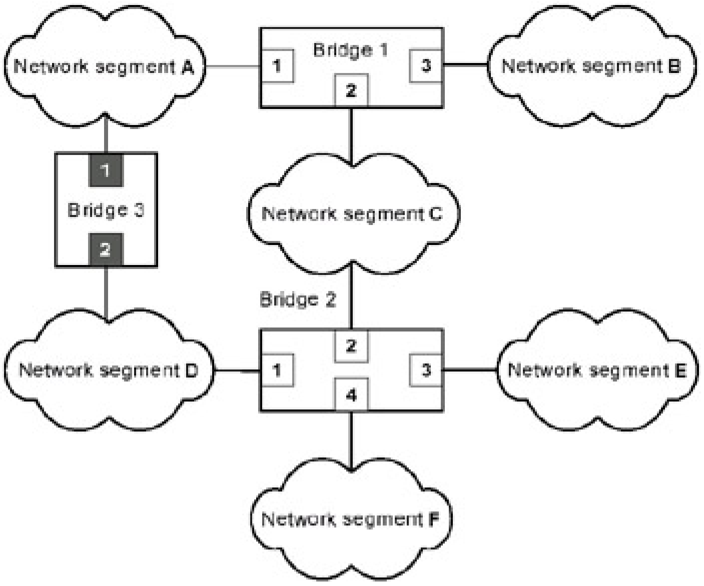


Fig. 1.33: A network with several bridges and a backup route

However, in a local-area network, just one path between any two hosts should be present. Otherwise, with a number of paths between two hosts, called a loop, the following undesirable consequences may occur:

- Broadcasting storms
- Reproduction of point-to-point packets
- Storage problems

Without taking the appropriate measures, broadcasting storms result in complete network failure. The given effect arises at the presence of closed loops in a network's architecture when the bridge forwards a broadcasting packet. As a result, this packet, multiplied by alternative paths by bridges, returns back through the loop to the initial bridge which forwards the received identical broadcasting packets again.

Computers' network protocols are designed so that each packet is only received once. The receipt of a multiplied packet is taken as an error, and a query on repeat retransmission is then sent to the multiplied packet's sender. As a result, repeatedly transmitted and multiplied packets will occupy transmission bands of all segments. Point-to-point packets will also be multiplied, but with less intensity.

During the transmission of a broadcasting packet along the network, each bridge of a loop will observe the same address of the sender, at several receiving ports. This will result in the storage of false information about the port number in the address table to which the sender of the broadcasting packet is connected. This will lead to confusion, as point-to-point packets will not be routed to the ports they should be routed to.

In order to prevent such problems resulting in malfunctions of local-area networks, modern bridges allow the operator to find and disable these effects in due time and, also, automatically to convert a network with loops to one with a proper structure, i.e., path between any two hosts. Such automatic conversion is performed through a number of special controlling messages, by means of which the bridges interact with each other and study the network's topology. Upon detecting loops, the bridges start jointly to disable some of the ports. As a result, the loops are deleted, and the tree's topology, connecting any two hosts by the only path available, is obtained.

When any connection that has a backup fails, the bridges automatically change the network's topology by actuating the backup path. For example, if all segments of the network shown in Fig. 1.33, are in a state of operability, the ports of the bridge with the number 3 will be disabled. If segment C fails, a bridge with the number 3 will automatically enable its ports, and the connection between groups of segments {A, B} and {D, E, F} will be restored.

If the available bridges do not support the function of automatic conversion of a network with loops into a network with a proper structure, it will initially be necessary to configure the network as a tree connecting any two network hosts by the only path available.

1.3.4. Using Switches

Features of Network Switching

Like bridges, switches operate at the data-link layer of the OSI model, and are used for dividing a local-area network into segments and for the internetworking of received segments and small local-area networks. Unlike bridges, however, switches contain a large number of ports, perform at a higher level, and cannot only divide a network into segments, but also separate message flows between different nodes of a network from each other.

The first switches had only 6 or 8 ports. With technological development, switches with 16 or 24 ports, and sometimes more, have appeared. A larger amount of ports allows the segmentation of a local-area network into smaller ones, thus significantly enhancing its throughput rate. Additionally, a large number of ports makes the connection of network segments to ports possible and, also, of separate computers of a local-area network. With the use of switches, connected computers can be put together in segments of the network, and easily regrouped when necessary.

By its design, a switch is like the network concentrator. In spite of the fact that switches are similar to high-speed, multi-port bridges, their internal architecture essentially differs from that of bridges. A switch can handle many packets simultaneously. It checks packets, controls the address table and makes decisions about simultaneous or parallel forwarding for all its ports.

Each port of the switch, similar to that of a network adapter, has receiving and transmitting parts. Each part of a port is logically connected to a cross-point switch matrix, implemented by the hardware via ASIC chips (Application-Specific Integrated Circuit). The receiving part of each port is connected to a horizontal line of a cross-point switch matrix, and the transmitting part is connected to a vertical one (Fig. 1.34). When a horizontal line is connected with a vertical one, the path from the receiving part of one port to the transmitting part of another one is established. It is possible to connect any pair of ports in such a way. For example, if the pairs of ports 2 and 4, as well as 3 and 5, are connected, two packets can be simultaneously received at ports 2 and 3, and transferred to ports 4 and 5.

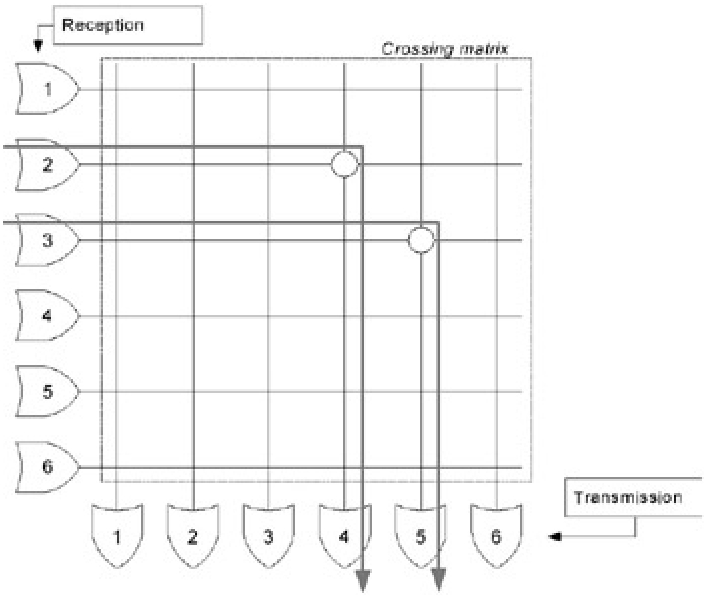


Fig. 1.34: The logical scheme of a switch

When a port receives a packet, the switch reads the recipient's address and makes a decision about forwarding on the basis of the address table. If the packet requires forwarding, a connection with the necessary port is established. As there may be many logical paths between ports of the switch, it is possible to forward many frames simultaneously. Point-to-point packets with unknown addresses, as well as multipoint and broadcasting packets, are forwarded to all ports, except for those to which these packets have arrived by switches and bridges.

This technology is named packet, or frame, switching, and is used in Ethernet and Token Ring local-area networks. Each packet is handled by the nearest switch and is transmitted further along the network directly to the recipient. As a result, the network becomes a collection of high-speed, direct connections operating in parallel. The switching of cells in ATM networks is performed in a similar way. A special feature of cells, unlike that of usual frames, is their fixed size. The usage of small cells of a fixed length simplifies creation of low-budget, high-speed switching structures at the hardware layer.

Virtual Networks

Switches with a large number of ports can enable the connection of separate computers to their ports, to distribute connected computers among network segments (Fig. 1.35) and to regroup them easily when necessary.

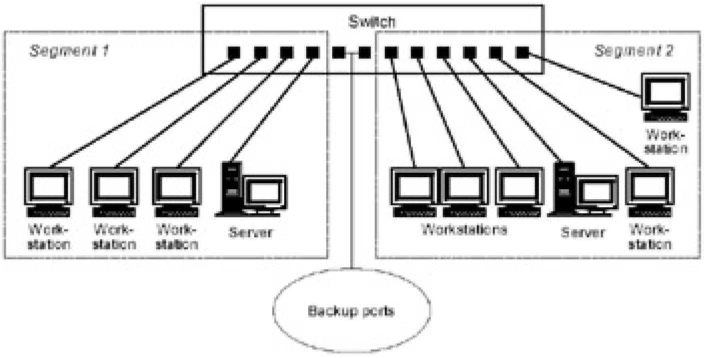


Fig. 1.35: An example of switching

As a result, it becomes possible to move network computers while keeping their places in a logical network structure.

A logical network architecture ensures a dividing of the network into workgroups formed by network segments. Within the boundaries of a network segment, multiple access is implemented when a message packet transmitted by the computer of a segment is supplied to all remaining computers in this segment and is received only by the computer to which it is addressed. Segments of a local-area network, whose firmware provides a separation of its logical and physical structures, are called virtual networks (VLAN, or Virtual Local Area Network).

Modern switches support some types of virtual networks (network segments), configured programmatically:

- Virtual networks, in which the physical ports of switches determine belonging
- Virtual networks, in which MAC addresses of devices connected to switches, not physical ports of switches, determine belonging
- Virtual networks, in which not only MAC addresses but also addresses of the network layer of devices connected to switches, for example, addresses of the IP or IPX protocols, determine belonging

In the first case, segments of a local-area network making virtual networks are formed by means of a logical connection of ports inside the switch's physical infrastructure. For example, some of the switch's ports are assigned to the first segment of the network, and others to the second. The disadvantage of such a method of virtual-network organization is that all servers connected to the same port have to belong to the same segment of the network.

Another way to create virtual networks can be realized through the MAC addresses of connected devices. With such an approach of virtual-network organization, an employee can connect a computer, for example, a laptop, to any of the switch's ports, and the switch will automatically determine to which network segment this computer belongs to on the basis of its MAC address. Such a method also permits computers connected to the same port of the switch to belong to different segments of the network. It makes sense when the same server or network printer services several virtual networks. However, it will then be necessary to "pay" for one device to belong to two or more network segments, with an additional load on the switch to which it is connected.

With the purpose of maintaining network security, each of the switch's ports can be configured so as to permit only connections with specific MAC addresses. The most complex form of controlling the assignment of ports is found through using one or more configuration servers permitting the connection to a given port. Such mobility is one of the main purposes of the development of virtual networks.

Belonging to different virtual networks can be determined not only on the basis of MAC addresses, but also on the basis of network-layer addresses. For example, one virtual network can be oriented to the IP protocol, and another can be oriented to the IPX protocol. The management system of virtual networks gives users of both virtual networks the possibility to organize access to servers supporting appropriate protocols (IP or IPX). In this case, the system separates servers by separate network segments. According to this method, each device connected to a switch, for example, a server or network printer, can also be a part of some virtual networks.

The organization of virtual networks on the basis of network-layer addresses is only possible when using routed protocols, for example, IP or IPX protocols. Otherwise, for example, while using the NetBIOS protocol, virtual networks only need to be organized on the basis of ports and MAC addresses.

Hierarchical Switching and Perspectives

The described methods of constructing virtual networks are especially effective when using a hierarchy of distributed switches controlled in a centralized way (Fig. 1.36). The switches of hierarchy can be distributed, for example, between separate floors of a building. To provide resistance to network failure, additional connections between switches on floors, prohibited at normal network operation, are introduced. In case of the failure of any working communication line, however, they will be automatically activated. If it is necessary to use one server for the whole local-area network, then it will need to be connected to the root switch.

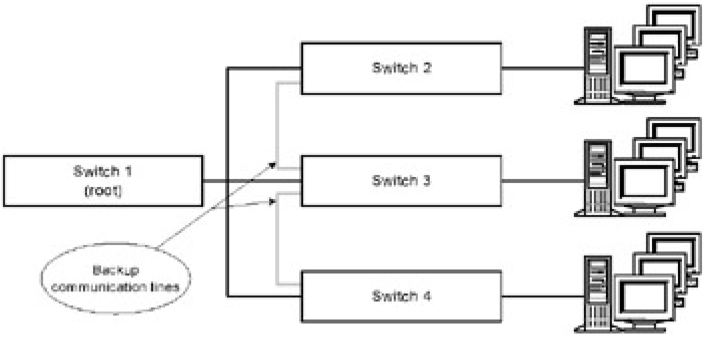


Fig. 1.36: Construction of a local-area network through a hierarchy of switches with backup communication lines

In spite of all their advantages, switches, like bridges, have one essential flaw: they are unable to protect the network from a flood of broadcasting packets, and this leads to inefficient network loading of the network. Routers can inspect and filter unnecessary broadcasting traffic but they work more slowly. Routers typically function at about 10,000 packets per second, and this cannot be compared with attribute of the switch of about 600,000 packets per second. Therefore, many manufacturers have begun to add a routing function to switches, which in addition, allows the organization of virtual networks through analysis of the network layer's addresses.

Currently, network-switching technologies are under intensive development. In parallel with completing the development of a universal standard of virtual networks, the manufacturers of switches are actively working to increase their automatic reconfiguration possibilities. The main goal in the near future is to have completely dynamic virtual networks, where the computer's connection to a switch's port is automatically traced by the central controlling application, transmitting all information necessary for network configuration to a switch without the administrator's intervention.

The final goal in manufacturer's strategic plans is the creation of virtual networks with an internal policy. As the level of automation of virtual-network control will be very high, managers should be able to set the network operation's internal policy. The policy, in this context, is understood to be a set of particular parameters, defining the rules of access to network resources and the rules of their usage. At the manufacturers' request, such virtual networks will be responsible for many functions of network operating systems, leaving them only to render the required services to users.

The creation of virtual networks with an internal policy will allow the removal of a number of control functions from the server. It will also essentially increase the degree of information and computer security, thus making the intruder overcome protection levels set by switches in order to get to network resources. If the infruder achieves this goal, many functions of network management and resource protection of network operating systems could be put in jeopardy.

1.3.5. Construction of Routed Networks

General Information on Routers

The bridges and switches intended for the dividing of a local-area network into segments, and for the interconnection of received segments and small local-area networks, do not select the best path, but instead transfer message packets from one segment of a local-area network to another, or from one local-area network to another. Moreover, these devices are unable to protect the network from flooding of broadcasting packets that results in inefficient network loading. For these reasons, bridges and switches can neither be used effectively for connection to global networks, nor for interconnection of local-area networks and their large parts. These functions are assigned to routers operating on the network layer of the standard OSI model.

Routers are especially important for integrated and global networks in which remote-communication lines are used. They provide an optimal level of traffic along complex routes in extensive integrated networks with an abundance of connections. In addition to that, they filter the flow of unnecessary broadcasting messages, and increase the throughput rate of communication links. This last function is particularly relevant upon connection of local-area networks with slow links.

The router, like the bridge, is a specialized computer, with two or more network adapters acting as the router's ports. The ports accept incoming message packets to the router. Specialized software provides checks of every received packet. For each forwarded packet, the best path for its further transmission is determined, and then the packet is forwarded to the router's appropriate port. An ordinary computer, with special software and several network adapters to which each of the appropriate data links are connected, can also perform router functions.

Unlike bridges and switches, routers possess the following features:

- For making decisions about forwarding, routers do not analyze the MAC address of the packet's recipient, but the recipient's address, corresponding to the network layer of the OSI model, for example, an IP or IPX address. This address, unlike MAC addresses, includes the number of the network to which computer is connected, and the number of the computer within the boundaries of the given network.
- Routers have their network address and are not transparent devices. If it is necessary to send a message through the router, the router needs to be applied to.
- While using routers, routed protocols such as IP or IPX should be applied. In cases where non-routed protocols are used, for example, NetBIOS, in which packets do not include any information on the network address, packets transmitted through the router should be encapsulated in the routed protocol's packets, for example, IP or IPX.

Let's consider the first two features in more detail.

Repeaters work with packets of the physical layer of the OSI model (Fig. 1.37) and do not analyze the data field of these packets. The exceptions to this are the repeaters oriented to execute functions of selective encryption of data-link layer packets and, also, MAC addresse filtering. In this case, the auxiliary data, corresponding to packets of the data-link layer, is handled.

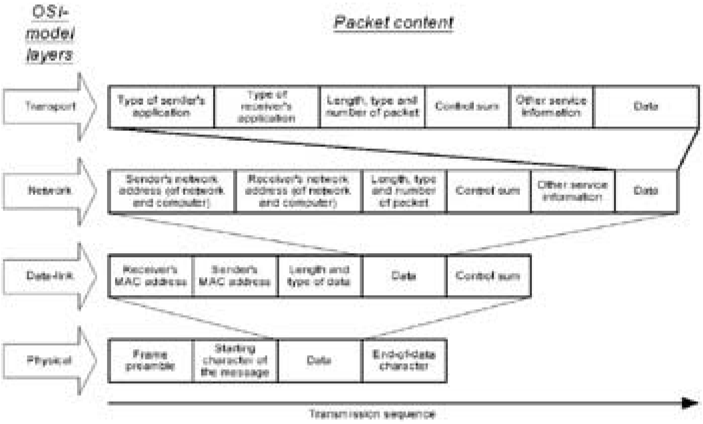

OSI-model layers
Packet content
Transport
Type of sender's application
Type of receiver's application
Length, type and number of packet
Control sum
Other service information
Data
Network
Sender's network address (of network and computer)
Receiver's network address (of network and computer)
Length, type and number of packet
Control sum
Other service information
Data
Data-link
Receiver's MAC address
Sender's MAC address
Length and type of data
Data
Control sum
Physical
Frame preamble
Starting character of the message
Data
End-of-data character
Transmission sequence

Fig. 1.37: The scheme of encapsulation of a transmitted packet

Bridges and switches work with data-link layer packets of the standard model (Fig. 1.37), analyzing them for decision-making about forwarding their sender and recipient's MAC address. Separate bridges and switches can also handle network-layer packets, in order to optimize the transfer of data. For example, many manufacturers have begun to provide switches with routing functions that, in addition, allow virtual networks to be organized through the analysis of network-layer addresses.

Routers work with network-layer packets of the OSI model, analyzing not MAC addresses but recipients' network addresses, each of which, unlike MAC addresses, includes the number of the network, the computer is connected and the number of the computer within the boundaries of a given network. Routers can also support improved filtering functions by handling packets of the standard model's transport layer. For example, filtering is possible, not only by the sender and recipient's address from headers of network-layer packets, but also by types of source and recipient application from headers of transport-layer packets.

When sending a message, the network host puts the sender and recipient's addresses (each consisting of corresponding numbers of the network and computer) in headers of the network layer's packets. While using the router in the network, the node should, prior to forming data-link-layer packets, determine whether it can transmit these packets directly to the recipient, or if they need to be transferred to the router.

Packets can be sent directly to the recipient if the number of the sender's network coincides with that of the recipient's network. This correspondence means that the sender and recipient are in the same broadcasting area.

Recall that the broadcasting area is a local-area network, or a part of it, made up by network segments, interconnected by bridges and switches. Broadcasting packets within the boundaries of such an area are delivered to all network nodes.

For forwarding packets directly to the recipient, the user needs to specify the receiving computer's actual MAC address at the data-link layer as MAC addresses of the recipient of these packets.

If the number of the sender's network does not coincide with that of the recipient, the transmitting host should route these packets to the router. For this purpose, the MAC address of the router must be specified as the recipient's MAC addresses of these packets at the data-link layer. After receiving the packets, the router will deliver them to the recipient. For the router, the recipient will be identified by the network-layer address, including the number of the network to which the receiving computeris connected and the number of the receiving computer within the boundaries of the given network.

The Role of Routers in Network Scaling

Bridges and switches, since they are simpler devices than routers, provide greater efficiency at a smaller price. But these devices solve problems of computer network scaling to a limited extent. There is a practical limit to which a network constructed on the basis of bridges and switches can expanded. The main reasons for this are increasing flow of broadcasting packets and an inability to provide active backup paths and possibility overloads. Routers and routing protocols were specially designed for solving the problem of scaling.

Any autonomous and integrated networks built on the basis of routers are called routed networks. Such networks are well-scaled and can become extremely large. A good example of a completely routed network is the Internet, which connects thousandss of networks and millions of computers from all around the world.

In terms of network scaling, routers have following important features:

- They provide improved filtering. As this takes place, broadcasting packets are filtered out and are not forwarded to any ports. Also, routers can filter packets according to information contained in packet headers of the network and transport layers: sender and recipient's addresses, information on the protocol, kinds of source applications and the recipient.
- Routers support networks with abundant active links, by means of which a number of active paths of data transfer between any pair of hosts is provided. On the contrary, one active path of transmission should be present in networks with bridges and switches, i.e., such a network should be configured as a tree.
- Routers determine the best route of transmission for forwarded packets. The route providing the minimum delivery time at a maximal level of reliability is usually selected, and such a route may be represented by one possessing a minimum number of transit hosts that are able to bypass the over loaded network hosts.

Bridges and switches are well suited to dividing a local-area network into segments that separate areas of multiple access from each other. An area of multiple access is understood here to be a segment of the network beyond the boundaries of which only message packets addressed to computers not belonging to this segment are spread. Within boundaries of an area of multiple access, message packets are delivered to all the computers, but are only received by those to which they are addressed. The areas of multiple access do not limit distribution of broadcasting packets, which are delivered to all network hosts within the boundaries of the broadcasting area.

Just as bridges and switches divide networks into areas of multiple access, so the routers divide networks into broadcasting areas and support multiple active links between them (Fig. 1.38). As this takes place, the broadcasting area also limits the area of multiple access that is typical for cases where the broadcasting area consists of one segment. Any host of the routed network can interact with any other one. However, broadcasting packets never leave the broadcasting area, and never use the transmission bands of data links disposed outside of it.

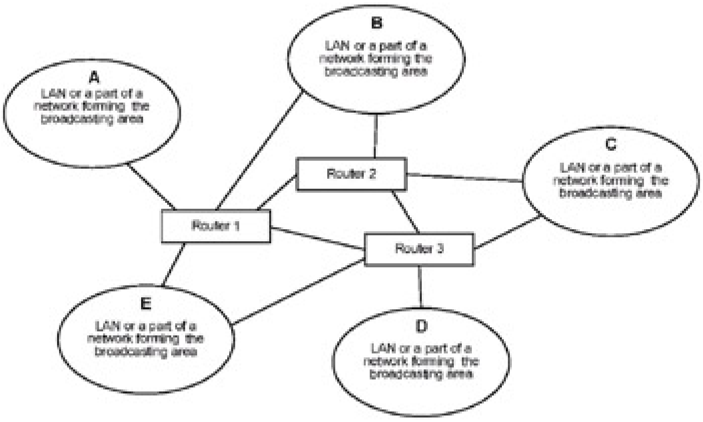

A
LAN or a part of a network forming the broadcasting area
B
LAN or a part of a network forming the broadcasting area
C
LAN or a part of a network forming the broadcasting area
D
LAN or a part of a network forming the broadcasting area
E
LAN or a part of a network forming the broadcasting area
Router 1
Router 2
Router 3

Fig. 1.38: Interconnection of broadcasting areas with the help of routers

The multiple active connections between broadcasting areas increase security and performance of computer networks due to:

- The presence of backup paths for cases of overload or failure of the main route
- The ability of bandwidth expansion between broadcasting areas by distribution of traffic over existing paths

For example, there are some alternate paths between broadcasting areas B, C and E in Fig. 1.38. These provide backup connections and the capability of uniform distribution of traffic over alternate routes, with due regard to current loading of data links.

Packets are always transferred along the best possible paths by routers. The best paths are determined by protocols on the basis of routing tables. Routing tables can initially be filled both manually and automatically. For maintaining the actual status of these tables, the routers interact with each other by transmitting different messages and controlling the status of the data links connected to them. Each router updates the detailed information on the topology of the networks stored in its table with the information received.

The presence of multiple active links in a routed network enables the possibility of transmitting packets to the recipient by different routes. If the first packet is transmitted by slow or overloaded routes, the next one can be directed by faster routes and it can reach the recipient before the first one does. The inconsistency between received packets and the order of their forwarding is eliminated at the transport layer of the OSI model.

Target Schemes in Using Routers

Interconnection of Parts of Local-Area Networks and Protection of Slow Devices against Overloads

Using routers instead of bridges and switches to interconnect large parts and segments of a local-area network is expedient when the broadcasting area of a local-area network becomes too large, or the amount of network segments and complexity of the network is significantly increased. In such a case, parts of local-area networks divided and interconnected by routers are easier to provide service to, since, on the network layer of the OSI model, each of these logical subnets has its separate number that is also included in the address of the network nodes.

Interconnected parts of a local-area network are directly connected to the router's ports (Fig. 1.39). Each of the interconnected parts can be a separate, large segment of the network, or a set of network segments interconnected by bridges and/or by switches.

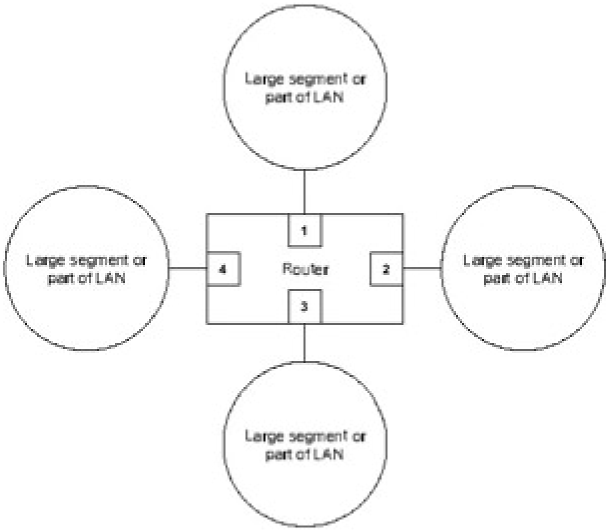

Large segment or part of LAN
Large segment or part of LAN
1
4
Router
2
Large segment or part of LAN
3
Large segment or part of LAN

Fig. 1.39: The interconnection of parts of a large network

In a very large local-area network, for internetworking of its parts, it is reasonable to use some routers with the formation of a large number of active links between broadcasting areas, for example, as shown in Fig. 1.38.

For internetworking of large parts and segments of a local-area network, combined devices, called bridges/routers (brouter) maybe used. Like many multi-protocol routers, they support operation under several protocols. These devices handle message packets of one type as routers and of another type as bridges. To determine which function it is necessary to perform, the filtering mask, which is adjusted by the administrator, is used. For example, in normal mode, these devices work as multiprotocol routers and, when they receive a packet with an unknown network protocol, they treat it like a bridge.

It is work taking into consideration that the routers for interconnection of parts of a local-area network should be quite productive, unlike those used for interconnection of networks and connection to backbones. This stems from a higher throughput rate of data links within boundaries of a local-area network.

In a high-speed local-area network, for example, in the Fast Ethernet network, routers can also be used to protect slow devices against overloads. In such a situation, all slow devices (printers, old workstations and servers, and routers of global networks) are isolated from the high-speed part of the network by a router and special repeater (Fig. 1.40). The port of a repeater to which a router is connected should be oriented to the rate of information exchange in a local-area network, for example, 100 Mbit/sec, and the ports to which slow devices are connected should be oriented to a rate available for them, for example, 10 Mbit/sec.

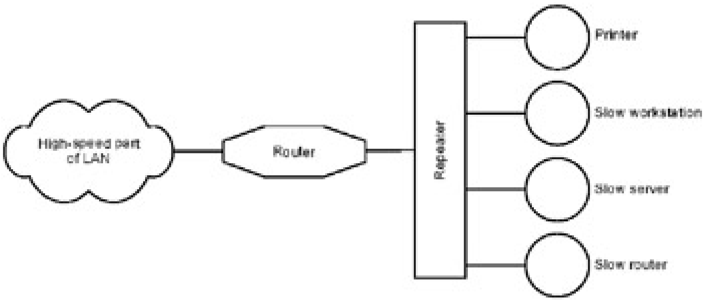


Fig. 1.40: The protection scheme for slow devices against overloading

A router that isolates slow devices from the high-speed part of a network creates a separate broadcasting area for them that is also a separate area of multiple access. It then follows that slow devices will not handle the flow of broadcasting packets that are not addressed to them and that will allow these devices to operate in normal mode.

LAN Internetworking and Connection to Global Networks

It is always expedient to interconnect local-area networks with the help of routers, as the application of bridges and switches for this purpose will not provide an adequate performance level, scalability or network security. The reasons for this include: intensive flows of broadcasting packets, resulting in overloading; an absence of optimal routing; an inability to provide a large number of active links for backup paths and bandwidth extension; and the absence of improved packet filtering, which plays an important role in providing internetworking security.

LAN internetworking is performed by the LAN connection to backbones represented by corporate, regional or global network through routers (see Fig. 1.41a).

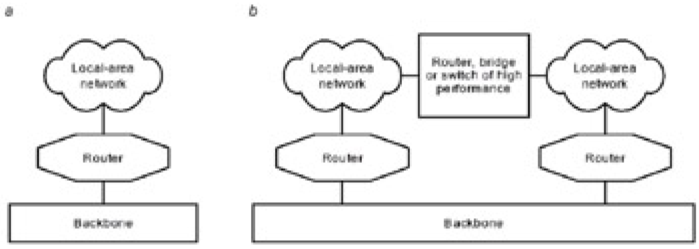


Fig. 1.41: Connection to a backbone

Backbones, unlike segments of local-area networks, realize selective, rather than multiple, access to devices directly connected to these networks. Routers play the role of these devices. Upon information exchange between two computers through a backbone, routers establish point-to-point connections with each other. As a result, the message packets within boundaries of a backbone are transmitted along routes determined by the routers. When the message packets get into a segment of a local-area network of the recipient, multiple access (in which the received packets are delivered to all nodes of a segment, but received only by the nodes to which they are addressed) is implemented within the framework of this segment.

Many bridges and switches also support links of backbones, but, in this case, broadcasting packets of the messages that are not filtered by these devices reduces the throughput rate of a backbone.

With LAN internetworking, it is necessary to take the 80:20 rule into consideration, which holds that the intensity of traffic inside a local-area network in relation to that between networks should not be any lower than a ratio of 80:20. Otherwise, it is necessary to interconnect LANs with intensive traffic between networks by data links with a high throughput rate directly through the high-performance router, bridge or switch, having left a connection through a backbone as a backup (Fig. 1.41b). Such a method of internetworking lowers the load on a backbone and increases the rate of information exchange between directly connected local-area networks.

An FDDI ring (Fiber Distributed Data Interface), known as a routed ring (Fig. 1.42), is often used as a corporate backbone. Routers used in a similar way are called edge routers.

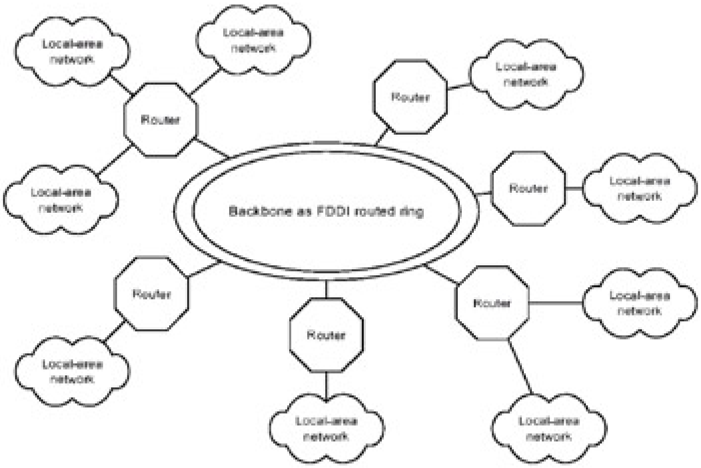


Fig. 1.42: A backbone as a routed ring

Regional and global backbones often have a different structure (Fig. 1.43), called a routed cloud. Edge routers interconnect local-area networks into the routed network and internal routers provide connection at the level of a backbone. To connect routers with each other, any data transfer links may be used, beginning from slow analogue and digital phone lines and ending with high-speed links of fiber optic and satellite communication.

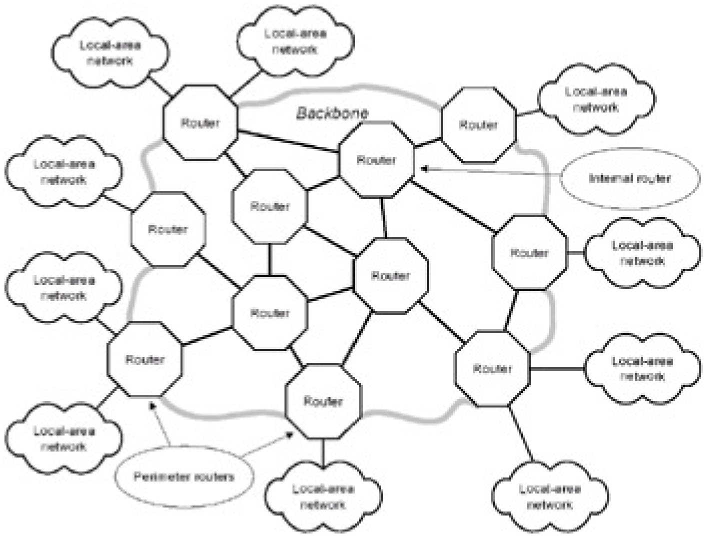


Fig. 1.43: A backbone as a routed cloud

In using different links, the principle of routing does not vary, but the types of network adapters do. The types of network adapters of each router should correspond to those of the data-transfer links connected to these adapters. For example, if a router links an Ethernet local-area network with an ISDN digital phone line of a backbone, one port of this router should represent an Ethernet network adapter, and another should represent an ISDN adapter. An Internet service provider can have routers supporting hundreds of modems for data transfer through analogue phone lines.

The routers coordinate slow and fast data links, connecting them with each other, for example, analogue phone lines with ISDN links. Upon transfer of information from fast links to slow ones, routers perform data buffering, which prevents possible overloading of slow data links.

The routers allow the interconnection of local-area networks and network segments differing from each other, by protocols of physical and data-link layers of the OSI model, for example, the Ethernet, Token Ring, FDDI and ArcNet networks. It is known that the maximal size of a frame in a Token Ring network is 4 kilobytes, and, in FDDI, it is 64 kilobytes. In an Ethernet network, the length of a frame does not exceed 1500 bytes. When using the IP protocol, the router can fragment large IP packets received from a local-area Token Ring or FDDI network into several packets of an Ethernet network. This is called IP fragmentation.

Multi-protocol routers provide an interconnection of segments and networks also operating under different protocols of the network layer, for example, under IP and IPX protocols. They perform independent processing of IP and IPX packets, and then transmit them to the appropriate segments or local-area networks.

Upon connection of a local-area network to the Internet as well as to other public backbones, threats of unauthorized invasion into a local-area network from a global one, and the threat of unauthorized access from a local-area network to resources of a global one, are both possible. Protection against these threats can be secured by using software and hardware complexes called internetwork screens, or firewalls, that support a routing function. The internetwork screen, as well as the router, is installed at a meeting place of local and backbone networks. It supports local-area-network security through filtering double-sided message traffic and authentication of the participants of information exchange. Firewalls can replace the router's functions, and can be also used together with the router to provide more reliable protection (Fig. 1.44).

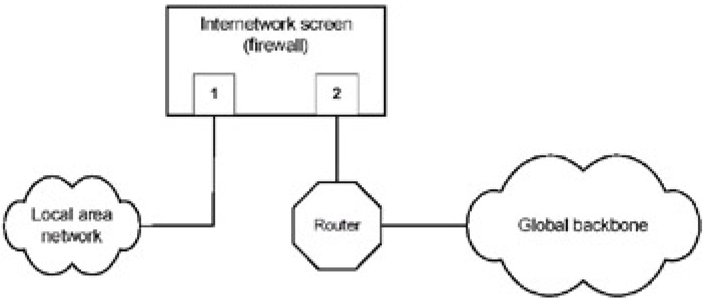


Fig. 1.44: Secure connection to a global network

The role of an elementary firewall can also be played by a modern router, which filters packets of network and transport levels of the OSI model. More perfect firewalls, like a gateway, operate at the application layer of the OSI model. They provide filtering of information, not only at network and transport layers, but also at higher layers of the standard model.

1.3.6. Routing Algorithms and Protocols

General Description

The main functions of each router realized in accordance with routing protocols are:

- Determination of the best routes up to a possible destination point, and the saving of the information obtained in a routing table
- Packet transmission by optimal routes, selected from a routing table on the basis of the recipients' addresses.

Modern protocols of routing are able to automatically create routing tables and maintain their actual status on the basis of interaction of routers with each other. Interrogating and listening programs operate on each router, which helps with the exchange of information with other routers. The received information will be used for construction and upgrade of a routing table.

The routing table, sometimes called the routing database, includes a set of optimal routes used by the router at the moment of packet transmission. Each line of this table contains, at least, the following information:

- The recipient's network address
- The address of the next router transfer corresponding to an optimal path up to a destination point
- A characteristic of the path, for example, the throughput rate of a data link and a time stamp denoting when this characteristic was determined
- Information on the method of transfer, for example, the number of an output port

The information on the next several possible transit routers that specifies different criteria of path optimization can be stored in one line of the table. The way a transit router is selected depends on the scheme of the routing protocol.

Path optimization upon the creation and upgrade of a routing table can be defined according to such criteria or their combinations as:

- The length of the route measured by an amount of routers through which it is necessary to get to a destination point
- The throughput rate of a data link
- The predicted total transfer time
- The cost of a data link

When a routing table is present, packet transmission according to optimal paths is implemented simply enough by the router. To send the packet through the router, the host of a local-area network puts the address of the real recipient in the packet's header on the network layer of the OSI model and the MAC address of the router on the data-link layer. When it receives the next packet, the router performs the following operations:

1. Reads the address of the destination from the packet's header corresponding to the network layer of the OSI model, i.e., the recipient's network address
2. Determines the address of the following transit router, which transfer by an optimal path to the destination point is represented by a routing table
3. Replaces the MAC address with that of the selected transit router in the packet's header, corresponding to the data-link layer of the OSI model
4. Sends the packet to the selected transit router

As the packet advances through the network, its recipient's physical address (MAC address) is changed, but the logical destination address, corresponding to the network layer of the OSI model, remains unchanged.

The Requirements of Routing Algorithms

The algorithms that make up the basis of the forming and upgrading of a routing table are called routing algorithms. The best routes to possible points of destination are determined by these algorithms. The algorithms of a packet's transmission along optimal routes selected from a routing table are called switching algorithms.

From the description given above, it is clear that the switching algorithms that specify the order of packets moving through the network along the best-known routes are simple enough. Routing algorithms, which form the basis of routing protocols, are complex and very important. These algorithms should meet the following functional requirements:

1. For optimization of determined routes-the ability to determine the best route depending on given values and their weighting coefficients
2. For flexibility-the capability of fast and precise adaptation to changes in structures and conditions of network operations

3. For convergence-the ability to establish fast agreements on the best routes between all routers of the network

In routing protocols, the value of route optimization is often called metrics. The optimal route is the shortest one. Thus the metrics, i.e., the measure of the route's length, are set by a special formula, where any of the route's parameters may act as variables, for example, the total number of transit routers and the total transfer time.

The requirements of routing algorithms for flexibility and convergence are interconnected with each other. When any changes influencing the selection of the best routes occur in the network, for examples, an overload of any node of the network or the appearance of a new data link, the routers that first learned about these changes should re-establish their best routes to adapt to the changes. Besides that, they should deliver messages about the changes to other routers. These messages are transfusing networks that stimulate recalculation of the best routes. In the end, all routers should come to a general consesus about the best routes.

The routing algorithms that do not possess a high degree of flexibility or fast convergence lead to routing loops and even to network failure.

Classification of Routing Algorithms and Protocols

In most cases, the ranking features of routing algorithms and protocols coincide with each other. The most relevant ranking features are:

- The degree of dynamic properties, reflecting the presence or absence of flexibility and convergence
- The amount of simulataneously supported routes to the same destination point
- A method of organizing routers
- The area of influence
- A method of obtaining route information

By the degree of flexibiltiy and convergence, static and dynamic-routing algorithms are distingushed.

Static algorithms represent the set of rules on filling and using static routing table which do not change in an automatic mode. These table are formed and updated by the administrator who should follow all changes in the network. Static algorithms do not provide flexibility and convergence. It is expedient to use them only in simple and small networks where traffic can be predicted.

Dynamic routing algorithms automatically create and upgrade routing table in real time mode. Message exchange between routers is carried out according to these algorithms. Without any route information, the routers interrogate each other. In cases of changes in the network, the routers notify each other. The messages obtained from each other stimulate the recalculation of the best routes and the upgrade of routing tables in real-time mode. Without dynamic routing algorithms, the admini-stration of large and complex networks becomes significantly more difficult. All routing protocols listed below are based on dynamic algorithms.

In terms of the number of the simultaneously supported routes to one destination point, routing algorithms can be single-path or multi-path. Multi-path algorithms allow a multiplex transmission of traffic by several routes at the same time. Such a possibility accelerates transmission and increases data links' throughput rate. Routing protocols based on multi-path algorithms are the Open Shortest Route First (OSPF) and Intermediate System to Intermediate System (IS-IS) protocols.

In respect of organizational methods of routers, flat and hierarchical algorithms are distinguished.

Flat algorithms provide equality of all routers with each other. The Routing Information Protocol (RIP) protocol is an example of a protocol using a flat algorithm.

While using hierarchical algorithms, routers are divided into levels. As a rule, two levels of routing, top and bottom, are introduced. The routers of separate network areas belong to the lower level, while the routers providing links between areas belong to the upper level. Packet transmission within the boundaries of the same network area is provided, with routers of the lower layer belonging to this area. During packet transmission to other areas of the network, these packets are transmitted from routers of the lower layer to routers of the upper level that deliver packets to a necessary area. For delivery of these packets directly to the recipient, they are transferred from an upper level to routers of a lower level belonging to this area.

The primary advantage of hierarchical organization of routers is the reflection of an inner structure of large, corporate networks. Algorithms of a hierarchical organization form the basis of such routing protocols as OSPF, IS-IS, NLSP (NetWare Link Services Protocol).

With regard to area of influence, routing algorithms can be intradomain and interdomain.

In this context, a domain is an autonomous system, representing a group of integrated networks controlled by the same authorized body, for example, one organization. The connection of domains makes up a more expanded network.

The routings algorithms used within the framework of autonomous systems are called intradomain algorithms. These algorithms can differ from each other, and each of them may be represented by an algorithm of a single level and hierarchical routing. However, for interaction between autonomous systems, only an interdomain algorithm should be used. This algorithm provides signals between routers specially allocated in each domain. As this takes place, intradomain routing algorithms should agree with interdomain algorithms.

Intradomain algorithms are used in most modern routing protocols, for example, in RIP, OSPF, IS-IS protocols.

Interdomain routing algorithms from the basis of such protocols as Exterior Gateway Protocol (EGP) and Border Gateway Protocol (BGP).

In terms of a method of obtaining route information, distance-vector and link-state algorithms are distinguished.

According to distance-vector algorithms, each router periodically delivers a copy of the routing table to adjacent routers. The adjacent routers compare the obtained data with their own routing tables and introduce the necessary changes. These algorithms are simple to implement and do not take up a lot of the computer's resources, but work badly in large networks. A key reason for this is the slow distribution of information about changes in the network, for example, information inaccessibility of a specific line or failure of a specific router. Besides that, the number of delivered messages is quite numerous, thus reducing the throughput rate of data links.

Distance-vector algorithms are used in such protocols as RIP, IGRP (Interior Gateway Routing Protocol), etc.

In case of link-state algorithms, the router gathers information on direct neighbors by determining the current state of data links connecting them, for example, the throughput rate. Instead of sending neighbors the complete contents of its routing tables, each router performs a broadcasting delivery of lists of directly connected routers and local-area networks, and information on the state of its data links. This information, except for periodic broadcasting about its presence on the network, is delivered by the router only when any changes in the data links on queries of other routers are detected and upon expiration of a given time period.

Differing by faster convergence, link-state algorithms are less inclined to make routing loops and consequently, are optimal for large networks. Additionally, the overhead expense for data transfer with the change in the topology of these algorithms is smaller: the routing table as a whole is not subject to delivery but only the information about chagnes. The weak points of link-status algorithms are complexity of implementation, strict requirements of processor performance, and RAM size.

The algorithms of a link status form the basis of such routing protocols as OSPF, IS-IS, NLSP, EIGRP (Enhanced Interior Gateway Routing Protocol).

Chapter 2: Attacks on Networks and Protection against Unauthorized Access

2.1. Methods of Unauthorized Access to Information in Computer Networks

2.1.1. Classification of Methods of Unauthorized Access and Life Cycle of Attacks

Nowadays, information, as a result of automated data processing, every year determines not just the operations of an increasing number of people, but also of an increasing number of man-made technical systems. Hence, the relevance of information protection in computer systems with the purpose of preventing it from being used to the detriment of people and the government became obvious. A careful analysis of all possible methods of unauthorized access to information in computer systems is necessary to solve the given problems effectively, and allows one to take measures to counteract possible threats as quickly as possible. Unauthorized access to information in this situation is access that breaks the rules for using information resources of computer systems that have been established for its users. Unauthorized access involves the implementation of intentional threats to information and computer-security, and is often referred to as an attack on the computer system.

Modern computing systems are geographically distributed computer networks connecting different computers and local-area networks with the help of data links. The vulnerability of distributed computing systems essentially exceeds that of standalone computers. It is connected, first of all, with the openness, the large scale and the heterogeneity of computer networks. Accordingly, there are many different kinds of attacks that can be made on modern computer networks. In this case, the number of threats to information and computer security and their methods of implementation are constantly increasing. The main reasons for this can be found in the drawbacks of modern information technologies, and also the steady growth of software/hardware resources' complexity.

The classification given below does not consider attacks on information in fully open and unprotected computer systems, as their implementation methods are obvious and do not differ from ordinary methods of access to information resources. All possible ways to gain unauthorized access to information in protected computer systems (Fig. 2.1) can be grouped by the following criteria.

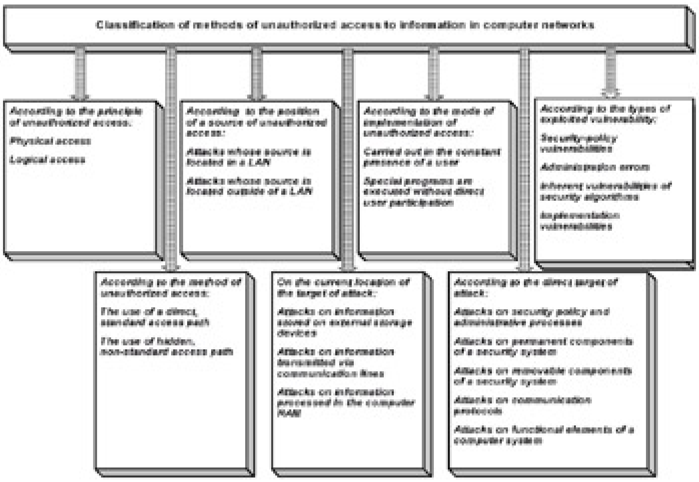

Classification of methods of unauthorized access to information in computer networks
According to the principle of unauthorized access:
Physical access
Logical access
According to the position of a source of unauthorized access:
Attacks whose source is located in a LAN
Attacks whose source is located outside of a LAN
According to the mode of implementation of unauthorized access:
Carried out in the constant presence of a user
Special programs are executed without direct user participation
According to the types of exploited vulnerability:
Security-policy vulnerabilities
Administration errors
Inherent vulnerabilities of security algorithms
Implementation vulnerabilities
According to the method of unauthorized access:
The use of a direct, standard access path
The use of hidden, non-standard access path
On the current location of the target of attack:
Attacks on information stored on external storage devices
Attacks on information transmitted via communication lines
Attacks on information processed in the computer RAM
According to the direct target of attack:
Attacks on security policy and administrative processes
Attacks on permanent components of a security system
Attacks on removable components of a security system
Attacks on communication protocols
Attacks on functional elements of a computer system

Fig. 2.1: Classification of methods of attacks on computer networks

1. By principles of unauthorized access:
    - Physical unauthorized access
    - Logical unauthorized access

    Physical unauthorized access can be implemented by one of the following methods:

- Overcoming perimeter-protection tools and accessing unprotected information resources
- Theft of documents and media
- Visual interception of the information displayed on screens of monitors and printers, and bugging
- Eavesdropping, or unauthorized interception of electromagnetic emission

  Logical intrusion is a logical penetration into a protected computer network. Having taken into consideration that the overwhelming majority of threats to information can only be implemented during the computing system's operation, as well as that logical unauthorized access is especially favorable for the intruder; this will be the main subject for analysis. The methods of unauthorized physical access will not be considered further.

2. By location of the source of intrusion:
   - Attacks whose source is located within the protected LAN
   - Attacks whose source is located outside the protected LAN

     In the first case, attacks are performed directly from any point of a local-area network. The initiator of such an attack is most often an authorized user.

     The possibilities of intrusion into a protected network from a public one are becoming a very significant problem upon the connection of any protected computer network to public networks, for example, to the Internet. These kinds of attacks are also typical in cases where separate networks, oriented to the processing of confidential information of absolutely different levels of secrecy or different categories, are connected. Threats of violation of established limitations exist when these networks are given restricted access to each other.

3. By mode of implementation:
   - Attacks performed with continuous participation of the user
   - Attacks performed by specially designed programs without human involvement

     In the first case, standard software can also be used for affecting the computer system. In the second case, specially designed programs, based on virus technology, are always used.

4. By types of OS or security vulnerability exploited:
   - Attacks based on the weaknesses of the established security policy
   - Attacks based on errors of network administration
   - Attacks based on the drawbacks of protection algorithms in facilities providing information and computer security
   - Attacks based on errors of implementation of a protection system's project

Flaws in the security policy entail that the security policy designed for a specific computer network does not meet the actual aspects of information processing necessary, so that it becomes possible to use this mismatch to execute unauthorized operations. Errors of administration are understood to be a part of incorrect organizational performance or poor administrative support of the security policy that has been established for the computer network. For example, according to the security policy, user access to a particular directory should be prohibited, but, actually, due to the carelessness of the administrator, this directory is accessible to all users. Effective methods of attacks can also be based on the drawbacks of the algorithms of protection and implementation errors of the information and computer-security system.

5. By methods of unauthorized access:
   - Attacks oriented to using direct, standard methods of access to computer resources
   - Attacks oriented to using hidden, nonstandard methods of access to computer resources

   Attacks of the first type are most often achieved through using vulnerabilities of the established security policy, as well as the weaknesses of the computer network's administration. For example, without control of secure passwords, an intruder can "mask" as an authorized user. Attacks of the second type are most often carried out by using undocumented features of the information system and computer security.

6. By the current location of the target of attack:
   - Attacks on information stored on external storage devices
   - Attacks on information transmitted by communication channels
   - Attacks on information processed in the computer's RAM

   The first two types of attacks are the most common.

7. By the object of attack:
   - Attacks on the security policy and the process of administration management
   - Attacks on the permanent components of a security system
   - Attacks on the changeable components of a security system
   - Attacks on Internet protocols
   - Attacks on the functional components of the computer system

   The target of an attack is always the protected information. An object of direct attack is the object whose analysis or usage will allow successful unauthorized access to protected information. For example, a cryptosystem, permitting the intruder to forecast a value of a generated secret key, may be the object of a direct attack. The classification criteria of unauthorized access methods to an attack object is especially relevant, as it enables the differentiation of methods of attacks in a more precise way. All attack methods listed in accordance with these classification criteria will be studied further.

The given system of attack-method classification draws the conclusion that effective unauthorized access to information is only carried out on the basis of vulnerabilities of a security system of an attacked computer network. Therefore, the generalized algorithm of attack preparation and implementation, as a rule, includes the following stages (Fig. 2.2):

1. A careful analysis of the structure and principles of operation of an attacked computer network, to search for security-system vulnerabilities and points of weakness.
2. An analysis of vulnerabilities that have been found, and development of the most effective methods of penetration into the information and computer security system.
3. Execution of prepared attacks and evaluation of obtained results.
4. In instances of discrepancy between obtained and demanded results, a careful analysis of the execution of attacks, and transition to the first step for improvement of their implementation methods.

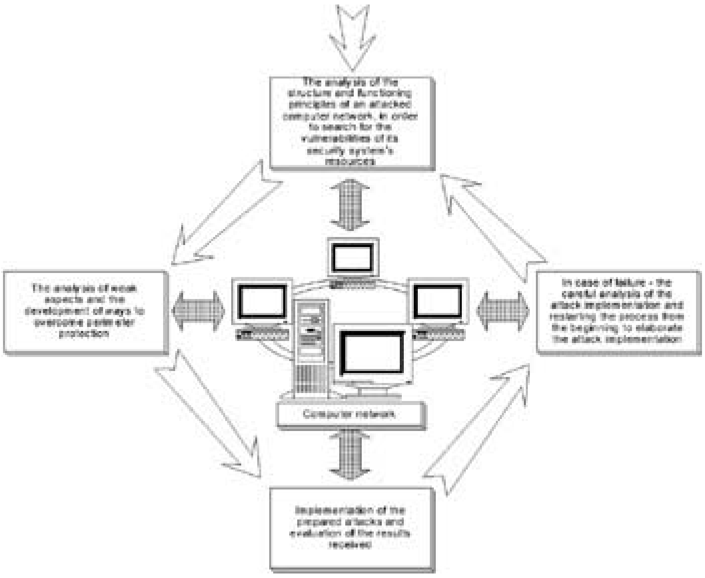

The analysis of the structure and functioning principles of an attacked computer network, in order to search for the vulnerabilities of its security system's resources
The analysis of weak aspects and the development of ways to overcome perimeter protection
In case of failure - the careful analysis of the attack implementation and restarting the process from the beginning to elaborate the attack implementation
Computer network
Implementation of the prepared attacks and evaluation of the results received

Fig. 2.2: A typical scheme of attack preparation and implementation

The presented algorithm involves a step-by-step process of maximizing effects on the attacked computer system. Security is like a chain, and so it is necessary to locate its weak link in order to implement an attack. Such a link can be found in any aspect of information and computer security; in the security policy, protection facilities, implementation of software and hardware, and system control. Finally, hidden defects, which at first glance are not security-related, such as, for example, software bugs, can also be exploited.

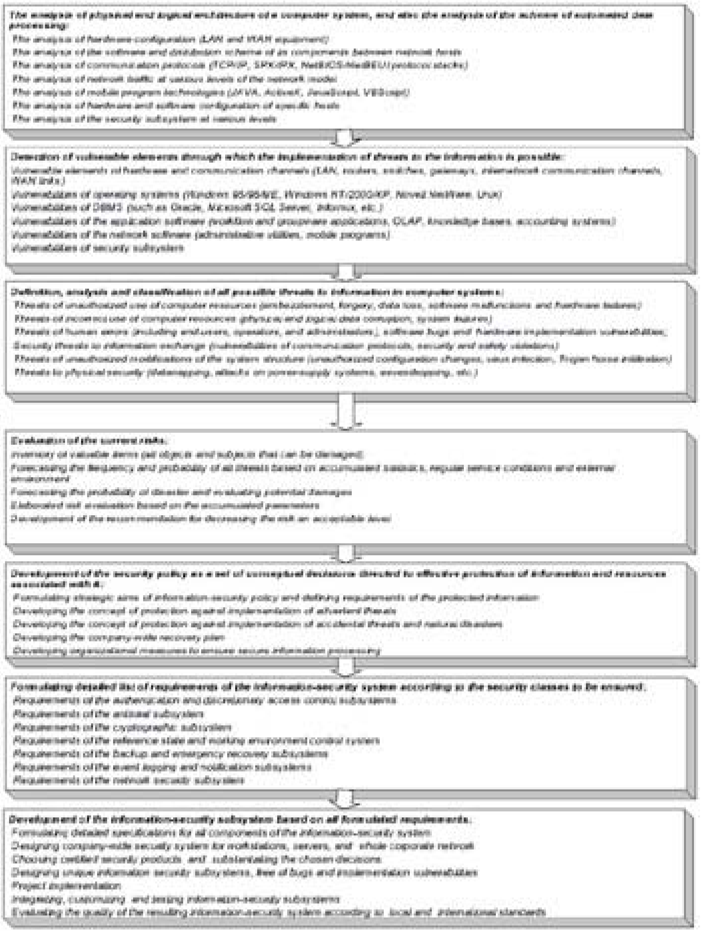
2.1.2. Attacks on the Security Policy and Administration Procedures

The development of security policy should precede effective protection of a computer network, just as any operation should start with careful planning. The security policy is a collection of documented conceptual solutions aimed at protecting information and the resources associated with it. The given policy is elaborated by those responsible for maintaining information and computer security, and is approved by the organization's administration.

The security policy should include the following sections:

- Strategic purposes of providing information and computer security, and the requirements of information protection
- A global concept of information protection in a computer network
- Collection of organizational measures aimed at protecting information, and the resources associated with it
- Responsibilities and official duties of information protection for an organization's employees

Fig. 2.3 outlines the developmental stages of the security policy involved in building an information protection system. It follows from the given figure that, if an information protection system is not properly executed during the building stages, prior to the development of the security policy, the latter may contain essential defects, which may be successfully used for attacks on the computer network. The defects can be contained in the prevention and timely detection of unauthorized effects on computer resources, as well as in the recovery of the computing system's security (Fig. 2.4).


The analysis of physical and logical architecture of a computer system, and also the analysis of the scheme of automated data processing:
The analysis of hardware configuration (LAN and WAN equipment)
The analysis of the software and distribution scheme of its components between network hosts
The analysis of communication protocols (TCP/IP, SPX/IPX, NetBIOS/NetBEUI protocol stacks)
The analysis of network traffic at various levels of the network model
The analysis of mobile program technologies (JAVA, ActiveX, JavaScript, VBScript)
The analysis of hardware and software configuration of specific hosts
The analysis of the security subsystem at various levels

Detection of vulnerable elements through which the implementation of threats to the information is possible:
Vulnerable elements of hardware and communication channels (LAN, routers, switches, gateways, internetwork communication channels, WAN links)
Vulnerabilities of operating systems (Windows 95/98/ME, Windows NT/2000/XP, Novell NetWare, Unix)
Vulnerabilities of DBMS (such as Oracle, Microsoft SQL Server, Informix, etc.)
Vulnerabilities of the application software (workflow and groupware applications, OLAP, knowledge bases, accounting systems)
Vulnerabilities of the network software (administrative utilities, mobile programs)
Vulnerabilities of security subsystem

Definition, analysis and classification of all possible threats to information in computer systems:
Threats of unauthorized use of computer resources (embezzlement, forgery, data loss, software malfunctions and hardware failures)
Threats of incorrect use of computer resources (physical and logical data corruption, system failures)
Threats of human errors (including end-users, operators, and administrators), software bugs and hardware implementation vulnerabilities
Security threats to information exchange (vulnerabilities of communication protocols, security and safety violations)
Threats of unauthorized modifications of the system structure (unauthorized configuration changes, virus infection, Trojan horse infiltration)
Threats to physical security (wiretapping, attacks on power-supply systems, eavesdropping, etc.)

Evaluation of the current risks:
Inventory of valuable items (all objects and subjects that can be damaged)
Forecasting the frequency and probability of all threats based on accumulated statistics, regular service conditions and external environment
Forecasting the probability of disaster and evaluating potential damages
Elaborated risk evaluation based on the accumulated parameters
Development of the recommendation for decreasing the risk on acceptable level

Development of the security policy as a set of conceptual decisions directed to effective protection of information and resources associated with it:
Formulating strategic aims of information-security policy and defining requirements of the protected information
Developing the concept of protection against implementation of advertent threats
Developing the concept of protection against implementation of accidental threats and natural disasters
Developing the company-wide recovery plan
Developing organizational measures to ensure secure information processing

Formulating detailed list of requirements of the information-security system according to the security classes to be ensured:
Requirements of the authentication and discretionary access control subsystems
Requirements of the antiviral subsystem
Requirements of the cryptographic subsystem
Requirements of the reference state and working environment control system
Requirements of the backup and emergency recovery subsystems
Requirements of the event logging and notification subsystems
Requirements of the network security subsystem

Development of the information-security subsystem based on all formulated requirements:
Formulating detailed specifications for all components of the information-security system
Designing company-wide security system for workstations, servers, and whole corporate network
Choosing certified security products and substantiating the chosen decisions
Designing unique information security subsystems, free of bugs and implementation vulnerabilities
Project implementation
Integrating, customizing and testing information-security subsystems
Evaluating the quality of the resulting information-security system according to local and international standards

Fig. 2.3: The stages of building an information-security system

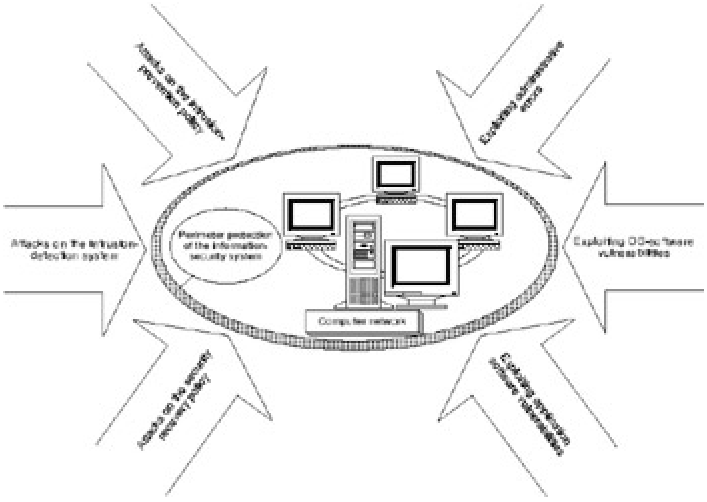

Attacks on the intrusion-prevention policy
Exploiting administrative errors
Attacks on the intrusion-detection system
Perimeter protection of the information security system
Exploiting OS-software vulnerabilities
Computer network
Attacks on the security-recovery policy
Exploiting application software vulnerabilities

Fig. 2.4: Attacks on the security policy and the process of administration management

Security system functions that need to be analyzed carefully for completeness and accuracy during searches for vulnerabilities are represented in Fig. 2.5.

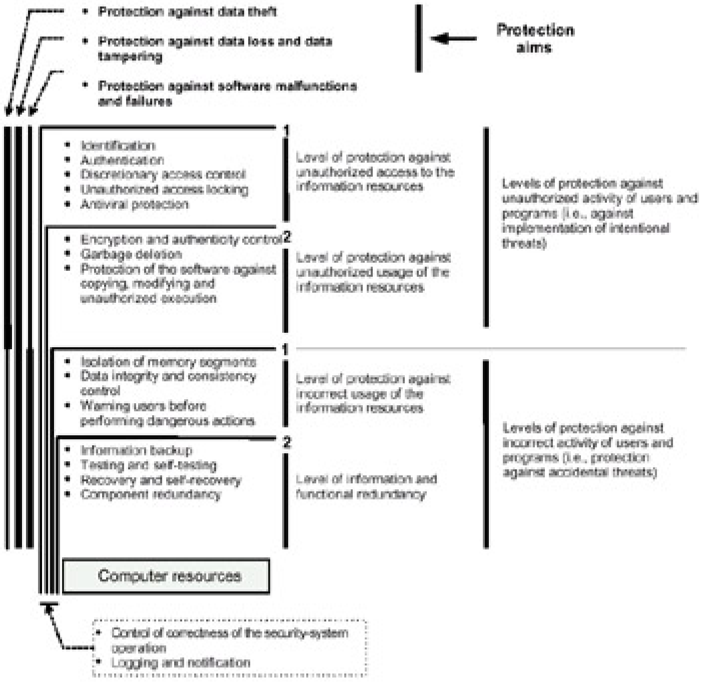

• Protection against data theft
• Protection against data loss and data tampering
• Protection against software malfunctions and failures
Protection aims
1
• Identification
• Authentication
• Discretionary access control
• Unauthorized access locking
• Antiviral protection
Level of protection against unauthorized access to the information resources
2
• Encryption and authenticity control
• Garbage deletion
• Protection of the software against copying, modifying and unauthorized execution
Level of protection against unauthorized usage of the information resources
Levels of protection against unauthorized activity of users and programs (i.e., against implementation of intentional threats)
1
• Isolation of memory segments
• Data integrity and consistency control
• Warning users before performing dangerous actions
Level of protection against incorrect usage of the information resources
2
• Information backup
• Testing and self-testing
• Recovery and self-recovery
• Component redundancy
Level of information and functional redundancy
Levels of protection against incorrect activity of users and programs (i.e., protection against accidental threats)
Computer resources
• Control of correctness of the security-system operation
• Logging and notification

Fig. 2.5: Functions of a security system that should be analyzed during searches for vulnerabilities

Drawbacks of Protection against Unauthorized Operations of Users and Programs

User-authentication. In most cases, a flexible system of user-authentication upon logging on to the computer system does not exist. The administrator does not have the option to select a method of authentication, as most often a protection system only includes functions of authenticity confirmation through a simple password. The dynamic check of the quality of assigned passwords is not provided and, quite often, passwords are transmitted publicly via data links. Authentication cannot be achieved simply by independently entering several different passwords.

It is not the case in all systems that, after successful user-authentication, the statistical data, for example, the date and time of the previous connection and termination of the session, allowing the detection of an unauthorized logon using the name of the given user are reported.

Restricting access to computer resources. In many systems, the access-control system is not reliable, due to the lack of effective methods of cryptographic protection. In addition, most systems do not provide mandatory access control, and, accordingly, they do not allow restrictions to be applied to access of computer resources by security levels and categories. In some cases, setting passwords for access to some of the most important computer resources is not even an option. Restriction of user access to floppy disks is also problematic.

Protection against running unauthorized programs. Many protection systems cannot prevent unauthorized launching of executable files. For example, executable files can be renamed for such launching. In many cases, indirect program launching and macros at the opening and processing of HLP-files, as well as of document files, spreadsheets and databases that can contain this call for program launching, are not controlled. It is also necessary to take into consideration that the remote-procedures call (RPC) for access to resources of other computers is used in network operating systems. For example, the remote execution of procedures allows one to change the configuration files on a remote computer. In view of that, it is possible to make the RPC-server determine login name or set access rights incorrectly.

Protection against computer viruses. In most systems, there is no built-in protection against computer viruses, which decreases the security level of data processing and data storage. Infection of the computer or local-area network with a virus may lead to a weakened performance of the computer system, corruption of data and loss of privacy of information stored in it.

Cryptographic protection of information. Cryptographic protection is a basis of information protection against data theft, forgery or abuse. The main drawback of modern security systems is an insufficient rate of cryptographic transformation, which forces users to refuse to use functions of encryption. Cryptographic resources currently available on the market do not provide the security level declared in advertising materials. Often, program implementations of cryptographic tools contain errors or use cut-down encryption algorithms not on a par with cryptographic standards. Most products are neither developed nor applied in cooperation with professional cryptographers. Instead, they are developed by engineers, for whom cryptography is simply one more program component. But cryptography is not just a component. It is impossible to provide system security simply by "introducing" cryptography after the system has already been developed. At each stage, starting from the mere idea to installation itself, one needs to realize exactly what the point of it is and why it should be carried out.

Control of data and program integrity. In a number of systems, there are no tools for detection of unauthorized or unintentional changes of data and programs. All confidential information, as well as that which is stored in the computing system, including programs, is subject to periodical monitoring for integrity.

The periodic check of the integrity of confidential information enables the timely detection of falsification attempts and loss of data. Also, these periodic checks allow the system to detect Trojan horses and computer viruses.

Control of correctness of the security-system operation and alarm subsystem. Irrespective of the capabilities of a particular protection system, it is impossible to achieve timely detection of illegal operations and high information security, without effectively implementing functions that can check security system for correctness of operation. In the overwhelming majority of protection systems, such functions are not provided. Many systems do not even provide execution of notification and alarm functions.

Safety of information exchange. The data between server and client workstations (except for functions of authentication) is most often tranmitted publicly and without checks for authentication. Subsequently, messages circulating within a local-area network can be forged and intercepted by an intruder. Quite often, there are no effective functions of key distribution between network hosts. As a result, privacy, authenticity and integrity of data circulating within the network and security of a local-area network are not provided upon connection to Internet.

Drawbacks of Protection from Loss of Information and Malfunction of the Computer System

Backing up information. Many systems do not provide effective functions that periodically back up information (scheduling of backup copying, backup of active files, etc.). Background backup, which, despite the option of automatic recovery of information, provides data recovery only after incidental hard-disk failures and after physical defects in memory become evident. Background backup does not provide recovery of information lost, owing to incorrect operation of software/hardware resources, or to unauthorized user and program operations. Periodic backup, provided that backup media is protected from unauthorized access, provides recovery of any lost data after the implementation of both incidental and intentional threats of distortion, or after information is erased [5, 6].

Automated system recovery. Most often, there are no tools providing recovery of a damaged system after booting from an operating-system boot diskette or bootable CD, without using the contents of the hard disk. Such a recovery mode can prove to be the only one available when it is impossible to boot from the hard disk. It is particularly effective in cases of total infection by a computer virus.

Safe installation of software. In operating systems, the function of safe installation of the software, i.e. a special, dynamic mode of registration and backup of all changes introduced by the program to be installed into the computer system, is not provided. Such backup, which is put into effect by specialized utilities, provides the ability for correct uninstallation, if necessary, for example, in cases of incorrect operation of the installed program.

Testing of hardware and protection against defects of computer memory. There are no tools that allow comprehensive testing of computer hardware in order to prevent failures. In spite of the fact that many operating systems have tools for diagnosing and eliminating defects of disk memory, through using resident tools that monitor the integrity of disks providing operation, this is not provided transparently for users. Also, built-in functions of defragmenting of disk memory do not always exist. The recovery functions of the diskette format in cases of data-read errors are not always provided.

Drawbacks of Network Administration

For planning attacks, besides flaws in the security policy, the drawbacks of the network's administration procedures can also be successfully used. The following control functions can be implemented with errors:

- Configuration control, intended for obtaining comprehensive information on hardware configuration and software of the network, and also for automated configuration of its components
- Performance control, permitting the system to receive data on the usage of network resources, and to adjust network components for maximum efficiency
- Control of access to shared network resources, for protection against any unauthorized operations performed by users
- Control of redundant network components, in order to achieve high reliability of operation
- Control of preparation for recovery, that is, development of working disaster-recovery plan, including properly implemented scheme for information backup
- Control of recovery, oriented to the timely detection of data losses, network component failures, and on-the-fly data and system recovery
- Check of observance of all regulations for providing information and computer security, and control over correctness of the security system operation

The last function is especially relevant. The administrator may also skip the following functions, which should be performed only periodically:

- Checking protection systems for correspondence to guiding and normative documents in the field of information and computer security
- Testing protection components for correctness of reacting to threat simulations
- Checking network-interaction safety
- Complex control of security-system performance at simulation of computer system components's malfunction
- The policy analysis of the creation and usage of reference information (keys, passwords, etc.)

2.1.3. Attacks on Permanent Components of a Security System

All components of information-protection systems are subdivided into two categories: permanent (long-term) and changeable. Components that were formed when the protection system was developed, and that require operations performed by experts or developers for their changes, fall under the of category of permanent components. Those system components intended for arbitrary updating or updating in accordance with a rule stated beforehand (probably on the basis of initial parameters selected in a random way), such as keys, passwords, identifiers, etc., are changeable components.

The most critical component of any information-security system is the subsystem of cryptographic protection [2]. Cryptography is used as the basis of operation of tools providing protection against the implementation of intentional threats to information. Therefore, a common method of attack on permanent components of a protection system is cryptanalysis (Fig. 2.6), intended to bypass a protection system of cryptographically protected information. Apart from classic cryptanalysis, the problems in implementing cryptographic algorithms can be used for attacks on cryptosystems. The search for vulnerabilities of cryptosystems, and also, of other permanent components of protection systems, is based on their research by special technical methods [9, 11].

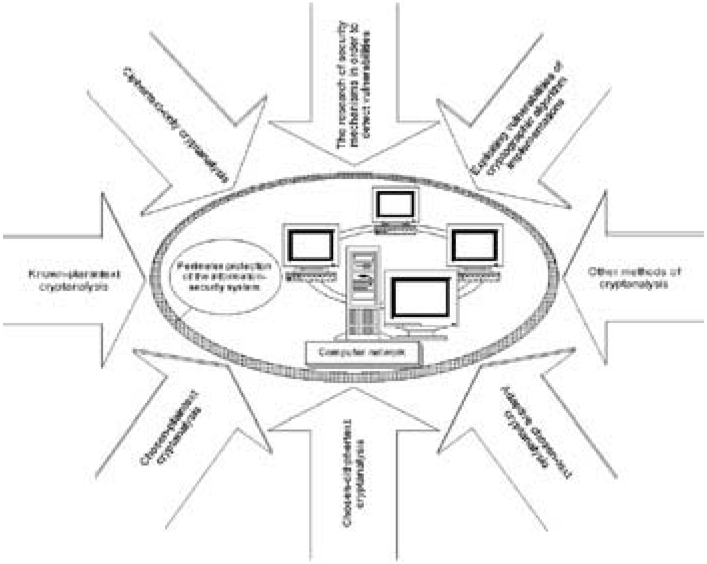

Ciphertext-only cryptanalysis
The research of security mechanisms in order to detect vulnerabilities
Exploiting vulnerabilities of cryptographic algorithm implementations
Known-plaintext cryptanalysis
Perimeter protection of the information-security system
Other methods of cryptanalysis
Computer network
Chosen-plaintext cryptanalysis
Chosen-ciphertext cryptanalysis
Adaptive chosen-text cryptanalysis

Fig. 2.6: Attacks on the permanent components of the security system

Cryptanalysis is used to solve the following problems:

- Recovery of the source data by its ciphertext (also known as cryptogram)
- Calculation of the secret key, using a known public key
- Creating a digital signature for specific message without knowing the secret key
- Creating a forged electronic document, corresponding to a known digital signature

The following types of attacks are considered in modern cryptanalysis:

- Ciphertext-only cryptanalysis
- Cryptanalysis based on a known plaintext and ciphertext corresponding to it
- Chosen-plaintext cryptanalysis
- Chosen-ciphertext cryptanalysis
- Adaptive-plaintext cryptanalysis
- Adaptive-ciphertext cryptanalysist

Data methods of cryptanalysis are intended first of all for the execution of attacks on cryptosystems oriented to data encryption, in order to protect them against unauthorized reading. Particular types of cryptanalysis are used for other types of cryptosystems, which it is expedient to consider in relation to a specific cryptographic algorithm.

In the case of cryptanalysis via a ciphered text, it is assumed that the opponent knows the encryption mechanism and only the ciphered text is available for him. It corresponds to a model of an external intruder who has physical access to a communication channel, but not to access to the means of encryption.

In the case of cryptanalysis through a plain text, it is supposed that the cryptoanalyst knows the ciphered text and some part of the initial information, and, in some cases, the correspondence between the ciphered text and the initial text. The possibility of such an attack appears upon encrypting standard documents, prepared on the basis of standard templates, when specific data fragments are repeated and known. In a number of modern approaches of information protection in computer systems, the mode of total encryption, in which all information on the built-in magnetic media is written as a ciphered text including boot records, system programs, etc., is used. Upon theft of this media (or computer), it is easy to determine what part of the cryptogram corresponds to standard system information and to obtain a great volume of the known initial text for conducting cryptanalysis.

In attacks through a chosen plain text, it is assumed that the opponent's cryptanalyst can enter a specially chosen text into the encipherer and obtain a cryptogram formed under the control of a secret key. This corresponds to the internal-intruder model. In pratice, such a situation can arise when persons, who do not know the secret key, but, according to official credentials, are able to use the encipherer to protect the transmitted messages, are involved in attacks on the cipher. To carry out such an attack, technical personnel, preparing forms of documents, electronic forms, etc. can also be used.

Cryptanalysis through the chosen ciphertext assumes that the opponent is able to input fictitious ciphertexts for deciphering. These are chosen in a special way so that they may make computations using the encryption key with minimal effort, by means of texts obtained at the decryptor output.

Attacks via adaptive texts correspond to those cases where an attacker performs input of the texts to be encrypted several times, and each new portion of the data is selected depending on the result of the previous portion's conversion. This type of attack is the most favorable for the attacker.

Attacks on the hash function relate to other methods of cryptanalysis, which are applied for obtaining and checking the reference characteristics of information objects. Hashing refers to the process of creating a message's characteristic through any hash function.

The hash function is the cryptographic function of a message of an arbitrary length whose value depends on every bit of the message in a sophisticated way. As a rule, the hash function is implemented as an iterated procedure that allows the computation of the so-called $H(M)$ hash-code of a fixed size m (128 or 160 bits) for the M message of an arbitrary length. This code is a standard characteristic of the M message. The standard hash function is calculated by sequential encryption of M_i binary blocks of the M message, according to the following iterated expression: $H_i = E(H_{i-1}, M_i)$, where E is the base function of encryption, and H_0 is a specified initial value of a hash function.

In some practical cases, it may become possible to perform attacks on a hash function by modifying its given initial value H_0 in order to achieve the following:

- For given H_0 and M − find H'_0 and $M' \neq M$ meeting the following condition: $H(H'_0, M') = H(H_0, M)$
- Find H_0, M and $M' \neq M$, satisfying the following equation: $H(H_0, M) = H(H_0, M')$
- Find H_0, $H'_0 \neq \mathrm{H}_0$, M and $M' \neq M$, satisfying the following equation: $H(H_0, M) = H(H'_0, M')$

It is rather uncommon to encounter conditions in which such attacks can be implemented. For example, in cases when the user may arbitrarily choose an H_0 value, and the intruder is able to substitute a true H_0 value for a false one or if they are able to weave in a false value H_0. These attacks are studied by the developers of hashing algorithms for a more integrated analysis of hash functions' properties and how they fare in conditions quite favorable for attacks on hash functions, in comparison with the actual position of an attacker. If the algorithm of a hash function's calculation is resistant to the attacks mentioned, it also meets the key requirements mentioned above.

Cryptanalysis is usually an extremely complex procedure. Therefore, initially, intruders attempt to use the weaknesses present in cryptographic algorithms' implementation and cryptographic programs themselves for attacks. The most common drawbacks include the following:

- A low level of cryptographic security at key generation, when the cryptosystem either truncates the user's password employed for key generation or generates a key from it of a length which is less than acceptable
- The lack of a monitoring system that checks for short and trivial passwords that a intruder could easily guess (it is rather typical for users to fail to take passwords seriously)
- No test for "weak" keys, which fail to provide a due level of cryptographic security to cryptographic algorithms (this particularly concerns asymmetric cryptographic systems)
- Insufficient level of protection against software bugs, such as Trojan horses, for example, consistency check with the standard characteristics of a working environment, is often lacking
- The intentional implementation of a secret back-door entrance function that can be initiated by the intruder, which enables the bypassing of cryptographic protection
- The generation of encryption keys through parameters which are not of a random nature
- Design and implementation errors when the algorithm is implemented with technical programming errors

In order to assess the drawbacks of cryptographic algorithms, disassemblers and debuggers help to enable their research. Disassemblers enable conversion of an executable program's machine code into the source text in the assembly language. Also, source-text analysis allows one to find the program operation's algorithm. It is possible to find the program's algorithm with a debugger, on account of its command-by-command execution, or execution with breakpoints. One can also find the algorithm through the content analysis of the processor's registers and other memory areas of the computer during program execution.

2.1.4. Attacks on Changeable Components of a Security System

The following data items are classified as changeable components of a security system:

- User information (identifiers, privileges, rights, restrictions, etc.)
- Keys and password information
- Security system settings

Each of the above-listed changeable components can be abused, and thus used as an attack targets (Fig. 2.7) [22, 25, 27].

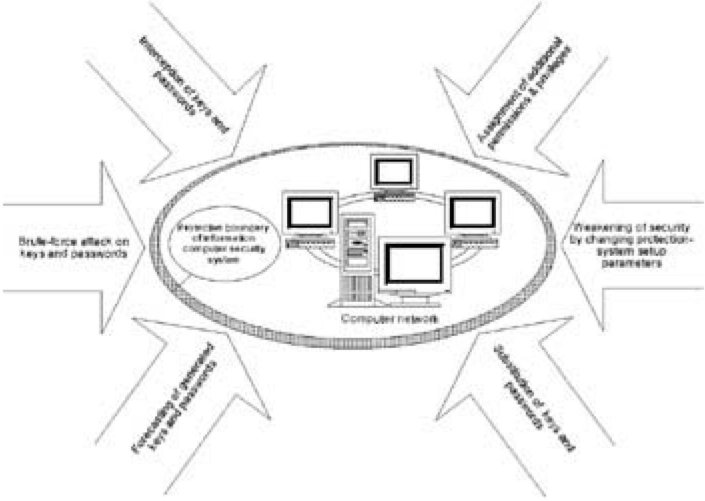

Interception of keys and passwords
Assignment of additional permissions & privileges
Brute-force attack on keys and passwords
Protective boundary of information computer security system
Weakening of security by changing protection-system setup parameters
Computer network
Forecasting of generated keys and passwords
Substitution of keys and passwords

Fig. 2.7: Attacks on changeable components of a security system

Attacks on key information are carried out in order to obtain the secret keys that would give way to a total violation of privacy and authenticity of protected messages. Therefore, it is of the utmost importance to watch out for this type of attack.

The secret keys can be obtained by different methods: interception, an exhaustive search, or by the forecasting of values of these keys at their generation by a cryptosystem. In addition, it is possible to replace the users' public keys with those of the attacker at their distribution. As a result of having the appropriate secret keys, it is possible to decipher the messages enciphered by false public keys, and to send false messages with fictitious digital signatures.

Conducting an exhaustive search and calculating a secret key are especially effective methods for short and trivial keys. Until recently, the smallest valid length of keys for symmetric encryption was considered to be equal to 56 bits. However, as a result of advances in computer technology, the minimum safe length of such keys is now considered to be 80, or even 128, bits. Keys of asymmetric encryption are rather easily calculated with sizes not exceeding 512 bits.

For example, the cryptographic security of the RSA cryptosystem is based on the factoring of the large *n* modulus, which is a product of two large prime numbers. In terms of calculation, with the correct selection, the *n* number's factorization is an unrealizable problem today. Having found the factorization method of an n number, it is possible to calculate the modulus' Euler function, and then the secret key, by means of a known public key. With a small value of the modulus (i.e. small key size), secret parameters can be computed with an exhaustive search on the basis of the following relations:

- $n = pq$
- $ed \bmod [(p - 1) (q - 1)] = 1$

In using a cryptographic system of complex protection, the cryptosystem generates the asymmetric and symmetric encryption keys on the basis of random parameters supplied by the user. Therefore, the prediction of generated key values is possible with algorithms of generation as well as the parameters by which they are generated.

Public-key substitution is carried out through the flaws in the procedures' authentication. The authentication of public keys is usually performed by references endorsed by certification centers, at users' personal meetings or by the endorsement of new users' public keys by specially appointed agents (persons authorized to act for a certification center). Public-key verification is most often performed according to their standard characteristics.

While a public key is being checked for authenticity by a reference directory that has been endorsed by certification centers, it should be kept in mind that, in the course of time, the reference directory of public keys can be updated by adding new users' public keys, or by replacing old public keys with new ones. This feature can also be used for attacks.

Let's consider some methods of an exhaustive search of the passwords for Windows NT 4.0.

All user information of Windows NT and its relevant passwords is stored in the special system database, known as the registry. Such information is located in the registry's SAM hive, located in the *%SystemRoot%*\SYSTEM32\CONFIG\ directory. The given file is readable by default, as it is used by other components of the system, but unaccessible for unprivileged user applications. A copy of the SAM file, contained in the *%SystemRoot %*\REPAIR\ directory, is created after an Emergency Repair Disk (ERD) has been created by the administrator. Notice that this file can be easily copied. However, passwords are not stored openly in this file. For each password, two 16-byte values located in the SAM database are created, using different hash functions. The first 16-byte value intended for Windows NT is calculated with the MD4 hash algorithm. The second 16-byte value designed for compatibility with the LAN Manager operating system, is calculated using a DES algorithm.

The hashed representation of the password in the LAN Manager system is less secure against breaking than the hash value of the password in the Windows NT system. This is due to the fact that, to create a hash value of a password in the LAN Manager, standard characters of the initial string of the password are converted to upper-case letters and, accordingly, the difference between lower- and upper-cases characters is not considered. Moreover, only 14 characters of the password are allowed. If the length of the password is less than 14 characters, it is supplemented with zeros. Each half of 14-character password is processed separately for the creation of an 8-byte value, which also decreases cryptographic security. The obtained 8-byte values are combined into a 16-byte one.

To obtain a password in Windows NT, the intruder must select a login name and a 16-byte hash-value corresponding to it in the LAN Manager from the security database (SAM-registry hive). The given procedure can be performed by using the PWDUMP freeware utility, which can be easily downloaded from the Internet. In order to be able to read the appropriate values of the registry, one must have administrator privileges. But the given program can also be used in cases where it is possible to get a copy of a SAM file from the attacked system. Such a copy can be obtained with the NTFSDOS.EXE freeware utility, allowing the user to access NTFS partitions from DOS, Windows 3.1*x* and Windows 9*x*. In this case, access rights are ignored. The authors of this driver promised to release a version with the option of data writing. In order to enable this driver, only the DOS boot diskette is needed.

After selecting passwords and hash-values corresponding to them, one can use any program enabling brute force password attack. The rate of an exhaustive search is approximately 2,500 passwords per second on computers of the Pentium class. Thus, unlike Unix, random numbers are not used for hashing the passwords in Windows NT, whereas the dictionary can be hashed beforehand, and thus the rate can be accelerated significantly. Furthermore, in Windows NT, the identical passwords being hashed do not differ from each other – a fact that also reduces their cryptographic security.

The random number applied in Unix to the hashing of the user's password is saved in the accounts' file with the hash value of this password. That is why preliminary hashing is hampered and the hash values of identical passwords of miscellaneous user accounts stored in the /etc/passwd file of the Unix system differ from each other. In early versions of Unix, 6-bit random numbers were used for password hashing. Now, 24-bit random numbers are most often applied.

During installation of Service Pack 3 for Windows NT, hash values are ciphered by a DES algorithm. A key of hash-value encryption of each password is the Relative Identifier (RID) of the corresponding user. The RID is an ordinal number of a user's account in the security database (file SAM). Accordingly, at hash-value encryption of the passwords by the DES algorithm, the cryptographic security does not increase as the keys of encryption are public.

The safe operation of any cryptosystem demands the employment of a protected software/hardware environment, which ensures that the working environment possesses the standard, requisite characteristics. Without such a monitoring system, some dangers could exist, including: the interception of keys and passwords, the assignment of additional rights, and system vulnerability with changing of a protection system's parameters.

2.1.5. Attacks on Network Communication Protocols

The most common types of attacks on computer networks are attacks on network communication protocols (Fig. 2.8).

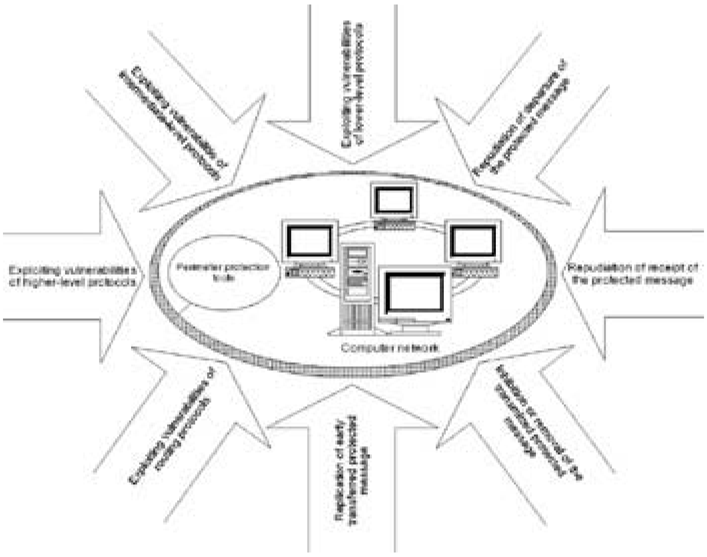

Exploiting vulnerabilities of lower-level protocols
Repudiation of departure of the protected message
Exploiting vulnerabilities of higher-level protocols
Perimeter protection tools
Repudiation of receipt of the protected message
Computer network
Exploiting vulnerabilities of routing protocols
Replication of early transferred protected message

Fig. 2.8: Attacks on network communication protocols

Generally, the protocol is understood to be a collection of functional and operational requirements for any component of network software/hardware that are adhered to by the component's manufacturers. The protocol is the standard in the field of network information exchange. And so, it is important to understand that the standard 7-layer model of network interaction reflects standard requirements for network software/hardware, as outlined by the International Standards Organization (ISO). The protocol meets the requirements of specific network software/hardware components that are upheld by the components' manufacturers. The protocol requirements may differ from those of a standard OSI model.

The protocol sets the collection of operations (instructions, commands, and calculation algorithms) performed in a specific order by two or more components of the computer network in order to achieve a specific result. The accuracy of the protocol's execution depends on operations performed by each program or hardware component. The components involved in protocols operate in accordance with the predefined algorithms, i.e., the algorithm acts as an internal component of the protocol. The protocol should have the following properties in order to reach the desired goal:

- *Correctness:* the collection of operations provided by the protocol should lead to the necessary result.
- *Completeness:* the protocol should specify operations of each of the protocol's participant for all possible situations.

- *Consistency and lack of ambiguity:* results obtained by different participants of the protocol, and the operations performed by them, should not be contradictory.

Unfortunately, the communication protocols currently in use very seldom satisfy all above-listed requirements. Any drawback of the protocol can be used for unauthorized access to information in the computer network [8, 14].

All possible attacks on communication protocols can be subdivided into two groups:

- Exploiting vulnerabilities characteristic for normal communication protocols
- Exploiting vulnerabilities specific for network cryptography protocols

Attacks on Normal Communication Protocols

The methods of attacks based on vulnerabilities of normal communication protocols can correlate with OSI model layers. The following types of attacks are distinguished:

- Attacks through vulnerabilities of lower-layer protocols (Ethernet, Token Ring, FDDI, ATM etc.)
- Attacks through vulnerabilities of intermediate-layer protocols (TCP/IP, SPX/IPX, NetBIOS and NetBEUI)
- Attacks through vulnerabilities of upper-layer protocols (SMB, NCP, SNMP, NFS, RPC, FTP, HTTP, SMTP, etc.)

A separate group of attacks is constituted by attacks based on vulnerabilities of routing protocols.

Generally, an attacker can implement the following threats:

- Interception of transmitted data with the purpose of datanapping, data modification or forwarding
- Unauthorized forwarding of data on behalf of another user
- Denial of data authenticity as well as repudiation of the facts of receiving or sending information

The interception of messages transmitted on the network can be performed by different methods:

- Direct connection to a communication channel
- Access to the network computer receiving the messages or performing routing functions
- Implantation of an unauthorized router in the network to redirect communications traffic through it

One can select the necessary information from intercepted message packets by using specialized analyzing programs. Similar programs can also be used for modifying the intercepted packets. Address information modification in their headers is performed for forwarding of the message packets.

In order to replace initial message packets with modified ones, it is necessary to implement a mode of interception providing a dynamic conversion of the flow of transmitted messages. In large networks, such a mode is implemented most successfully by gaining unauthorized access to a network computer that performs routing functions. Another way is by implanting an unauthorized router to redirect communications traffic through it. In local-area networks, implementing the mode of dynamic conversion of communications traffic depends on the lower-layer data-transfer protocol that has been used.

The unauthorized sending of data on behalf of another user is performed with the assistance of appropriate software after connecting to a computer network. This type of threat enables one to impersonate as an authorized user, thereby obtaining access to secret information, or being able to deceive the data recipient in order to cause harm the system.

Users' denial of data authenticity, as well as repudiation of the facts of obtaining or sending the messages, allows, in particular, one of the parties technically break the concluded agreements (financial, trading, etc.), by officially following through with them for the sake of unfair profit earning or of harming the other party.

Attacks on communication protocols often seem technically difficult, in terms of implementation. However, it is not difficult for a good programmer to implement corresponding service programs. Such programs are available to the general public on the Internet.

In case of attacks on lower-layer protocols (Ethernet, Token Ring, etc.), the intruder can watch the traffic, substitute the contents of message packets, and also perform substitution of MAC addresses. Without having physical access to network computers, the attacker can connect his or her own computer to a local-area network. In local-area networks, packets are transmitted with frames whose headers contain MAC addresses of the recipient's and sender's servers. The sender's address is not checked after it has been sent, as it is not an easy task. Even in cases where the sender's address is analyzed by any means (for example, by switches), it is not difficult to substitute a MAC address with another one.

The majority of network adapters allow one to program and/or change the MAC address dynamically. For example, in NetWare, the ODIPKT driver allows formation and sending of an entire packet independently. Accordingly, it is not difficult to write a program for spoofing sender's MAC addresses. The purpose of such an attack is to deceive a network-operating system and other software connected with a data-link layer and to make them behave in a way they normally would not. An example of this is the HACK.EXE program, which performs substitution of MAC addresses in order to obtain a supervisor's rights on any NetWare 3.11 server. Address substitution can also be used to implement threats, such as the denial of service. Filtering of MAC addresses executed by specialized concentrators allows users to be protected against such attacks.

One of the main drawbacks of intermediate-level protocols (TCP/IP, SPX/IPX, NetBIOS and NetBEUI) is the fact that they do not contain built-in security functions. There are no provisions for the cryptographic encoding of message packets' contents, control of their authenticity, and authentication of the exchange participants. Consequently, message packets can be accessed for analysis and may be forged.

The protocols of the TCP/IP family, initially designed for a global Internet network, are now serving as the basis of the construction of local-area networks. These protocols have always been base protocols for Unix-systems. Now, these protocols are supported by all modern operating systems. In spite of the fact that the development of TCP/IP was sponsored by the U.S. Department of Defense, these protocols have no security functions, and allow both passive and active attacks.

In case of passive attacks on TCP/IP protocols, as well as attacks on other communication protocols, intruders do not reveal themselves in any way, and all unauthorized operations are reduced to monitoring message packets. With active attacks on TCP/IP, the intruder modifies and/or filters the contents of the message packets, to trick the recipient, or to hinder the performance of the computer system of the receiving party. Having the necessary permissions, or simply using DOS or Windows, which don't provide discretionary access control to computer resources, the attacker can manually create IP packets and send them through the network. Naturally, the header fields of each packet can be created arbitrarily. Having received such a packet, it is impossible to find out from where it was actually received, as the packets do not contain their route. Certainly, at specification of a return address not coinciding with the current IP-address, the attacker never receives an answer to the sent packet. However, this is not often required. The key element in the execution of active attacks lies in the ability to create arbitrary IP packets.

The SPX/IPX protocols used in a NetWare network operating system are also unprotected. An intruder can create an IPX packet completely identical to that of the administrator. Therefore, practically any network user can violate system security. If somebody creates the IPX packet independently, he can specify the address of another sender, for example, of the network administrator. Such a packet will necessarily reach a recipient (server), and will be processed by it in the same way as all other authorized packets.

The complexity of attacks on unprotected upper-layer protocols and routing protocols are determined only by protocol complexity, and not in any way by reliable functions of protection.

Exploiting Vulnerabilities of Network Cryptographic Protocols

The cryptographic protection of sent messages does not guarantee their security. Attacks based on using vulnerabilities of network cryptographic protocols include such actions (performed by the intruder) as repeating messages that have been sent before, delaying or deleting sent messages and also the denial of the fact of having received or sent messages. In this case, the attacker has to take possible methods of counteraction into consideration.

For protection against repetition, deleting or delaying of the messages, additional information is added into each message before cryptographic protection. Ordinal numbers, random numbers and also timestamps can be used as such additional information. Inserting numbers into the initial messages provides protection of these messages against repetition and deleting. For implementation of such protection, the ordinal numbers of the messages should be related to counters, whose status should be observed by both the sender and recipient security system. Each network object should have an individual counter for each of the objects interacting with it. While the numbers of the messages of a specified sender are being checked, the deletion of any message will be discovered immediately upon the message's arrival with a number whose value differs from the number of the previous message by more than 1. The repeated messages can be found by detecting repeated message numbers.

The numbers of the message packets created on a data-link layer cannot be used for detection of repetitions and deletions of messages, as these numbers are set to zero for each message in the case of the datagram protocol or for each connection with a session protocol.

The insertion of random numbers into initial messages also provides protection of these messages against repetition and deleting. But, with the use of random numbers, the presumed recipient should send the enciphered and signed random number to the sender, prior to the message's transmission. The sender should put this random number in the initial message before encryption. The recipient can check the random number sent when this message is obtained. If the presumed recipient knows the maximal time of the expected message's transmission, then, along with repeated messages, deleted and delayed messages can be easily detected.

The insertion of timestamps in the initial messages denoting the sending time provides protection of these messages against repetitions and delays. These timestamps, and also a standard time period for detection of delayed messages, should be selected so that it is possible, on the one hand, to find repeated messages, and, on the other hand, to take natural delays, appropriate to data channels, into account.

To ensure the confirmation that a message has been received, the messaging protocol should send a notification message. The person who receives the message should then sign it.

Making sure that a sent message has been acknowledged, i.e. protection against non-recognition of a digital signature, is only provided according to legal and organizational measures that deal with legalizing a digital signature. To make sure public keys and a digital signature are acknowledged, the exchange of public keys should be backed up with a legal procedure.

2.1.6. Attacks on Functional Components of Computer Networks

There are two types of attacks that can be executed on functional components of computer networks:

- Attacks, which cause system malfunction (denial-of-service attacks)
- Implantation of Trojan horses

The first type of attack involves disruption of components' performance of a protection system, with the aim of gaining unauthorized access. The second type allows the attacker to insert Trojan horses that automatically carry out unauthorized operations.

In the case of malfunction of functional network elements, the rate of functions' execution provided for servicing of queries sent for processing significantly drops. A component whose operation is often disrupted is called a server. In this context, a server is understood to be any program or network operating system that processes queries for access to any service or resource.

The malfunction of security system components can caused by the following (Fig. 2.9):

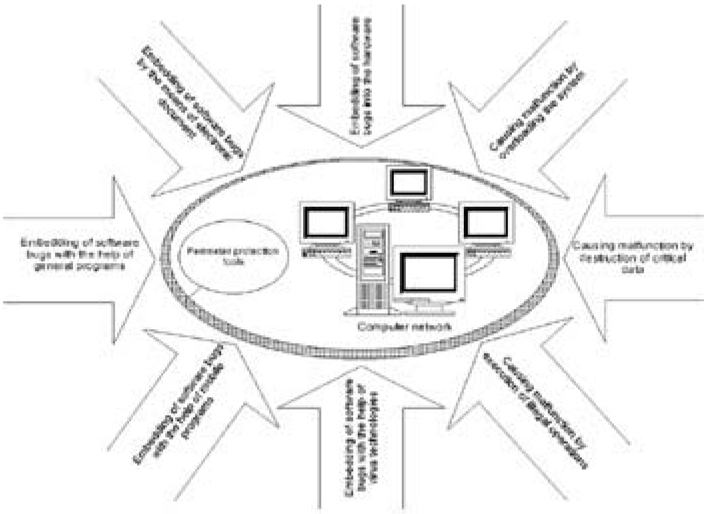

Embedding of software bugs into the hardware
Causing malfunction by overloading the system
Embedding of software bugs with the help of general programs
Perimeter protection tools
Causing malfunction by destruction of critical data
Computer network
Embedding of software bugs with the help of virus technologies
Causing malfunction by execution of illegal operations

Fig. 2.9: Attacks on functional elements of computer networks

- Overload of functional components of the computer network
- Deletion of critical data
- Violation of communication protocols and execution of incorrect operations

The server can only respond to a limited number of queries being processed. These limitations depend on different parameters of the computer system: performance, size of RAM, bandwidths of data links, etc. If the authentication of the senders of queries is not provided and the server allows an invalid number of anonymous queries for processing, performance will be significantly degraded.

For execution of an attack through deletion of critical data, it is necessary to locate information units stored in a computer's memory, whose corruption will cause the security system to fail. For example, corrution of the data stored in a Flash BIOS chip of the server responsible for security functions will cause total failure.

Currently, it is common to see attacks on computer networks based on intentional communication protocol violation. If the attacked computer system does not take into consideration all possible violations of communication protocols, deliberate violations can cause an inadequate response and even total failure in service. Malfunctions can be also caused by the execution of incorrect operations. For example, there are executable commands to which the processor reacts inadequately.

Implantation of Trojan horses provides the broadest capabilities for implementing attacks. Such a program has been specially designed for the independent execution of unauthrorized operations, and is understood to be any instruction sequence subject to execution by the processor or another program. For example, macros included in document files of the Word text processor are also essentially programs, as they are instruction sequences executed by the text processor for automation of user operations. From this, it becomes clear that all methods accumulated in the field of computer virus technologies can be used for the development of Trojan horses.

The first, and one of the most relevant, stages of the life cycle of a Trojan horse after its development is its implantation into the computer system, also known as infecting. Infecting is only possible at the launch of an infected or virus-like program for execution. After the activation of a Trojan program, the executable programs as well as programs stored in external storage devices can be infected. As a rule, a copy of a Trojan horse may be inserted into the program to be infected so that, as the infected program is launched, the Trojan horse is the first that gains control. A computer system is said to be ‘infected’ if any of its programs have been tainted. The Trojan horse does not necessarily possess the ability to self-reproduce.

A Trojan horse can be implanted into a computer system with the help of any type of infected program existent under virus technology, and also by means of hardware. At implantation of a Trojan horse by way of virus technology, it should have the property of self-reproduction intrinsic to a customary virus.

With the help of hardware, the implantation of a Trojan horse involves the infection of programs contained in hardware devices, for example, programs of a BIOS chip.

A wide array of possibilities for the implantation of Trojan programs is provided by increasingly popular Web-technology, based on mobile calculations. A navigating program executed on a workstation is able not only to visualize Web pages and perform transitions to other resources, but also to actuate programs on the server. It can also interpret and launch programs for execution that relate to Web documents that are transmitted together with this document from the server. This kind of distributed processing has allowed a concentration of all application systems on the server. However, being able to execute programs from the server on workstations leads to effective methods of Trojan-horse implantation. Implantation can be put into effect by substituting the program transmitted from the server as well as by initial placement of a mobile Trojan horse on the server.

In order to start to perform its functions, the Trojan horse must gain control, i.e., the processor or the interpreter has to begin to execute commands relating to the Trojan program.

The Trojan horse goes into effect according to the external conditions programmed into it. The processing of a general Trojan program, relative to the operations most often represented by interrupts or specified events, accomplishes an analysis of external conditions. Such interrupts include the ones from the system timer, peripherals, disk I/O, as well as OS interrupts such as file I/O and starting executable modules. The interpretable Trojan horse should be designed as macro and automatically executed when specific events occur, such as: opening or closing of documents, program launching, termination of operation of the shell, etc.

For highly effective attacks, Trojan programs can hide the fact that they are present in a computer system. In order to protect the program from being examined, the file, with its executable code stored on the external media, is protected, along with its executable code loaded into the RAM for program execution.

In the first case, protection against examination is based on the encryption of a secret part of the program, and in the second case, on the blocking of access by debuggers to the program's executable code in the RAM. In addition, before the termination of the protected program, its code in the RAM should be updated. This will prevent the possibility of unauthorized copying of a decrypted executable code after a protected program's execution from the RAM.

The program protected from examination should include the following components:

- An initializer
- An enciphered, secret part
- A destructor (de-initializer)

The initializer should provide execution of the following functions:

- Saving of parameters of user's operating environment (interrupt vectors, content of the processor registers, etc.)
- The prohibition of all internal and external interrupts whose processing cannot be controlled by the protected program
- Loading of a secret part of the program into RAM and deciphering of the code
- Transfer of control to a secret part of the program

The requirement of an initializer for disabling of all interrupts whose processing cannot be controlled is not excessive, as, instead of the standard handler of any interrupt, the handler intended for tracing can be used.

The secret part is oriented towards execution of all goal functions of the protected program.

The destructor, after executing a secret part of the program, should do the following:

- Delete the secret code of the program from the RAM
- Restore the parameters of the user's operating environment (interrupt vectors, content of the processor's registers, etc.) that were set prior to disabling of unmonitored interrupts
- Execute operations that were impossible to perform upon disabling of unmonitored interrupts
- Release all involved resources of the computer and terminate the program

For greater reliability, the initializer may be partially enciphered, and it may decrypt itself during execution. Moreover, the destructor can be enciphered, too. The secret part of the program can also decrypt itself during execution.

The deciphering of any part of the program at run-time is called dynamic deciphering of the executable code. In this case, the parts of the program that need to be executed are decrypted immediately before their usage, and are subject to destruction right after execution. The current, decrypted part of the program for execution can be copied to antoher location of the RAM, for example, at the location of a part of the program that has already been executed.

In order to increase efficiency of program protection from examination, along with the measures mentioned above, it is necessary to enter additional security functions oriented to protection against tracing in a secret part of the program. Such functions include the following:

- Periodic calculation of the control sum of the area of the RAM occupied by a protected initial code, and comparison between the current control sum and standard sum calculated previously. In case of mismatch appropriate measures must be taken.
- Checking the amount of RAM occupied by the protected program; comparison with the size of the program that is being adapted, and taking of necessary measures in case of mismatch.
- Time supervision of execution of individual parts of the program.
- Keyboard lock for the period of execution of particularly secret algorithms.

For program protection against examination using disassemblers, it is also possible to make the structure of the program more complicated in order to confuse the intruder. For example, it is possible to utilize different segment addresses for calls to the same memory area. In this case, it would be difficult for the intruder to guess that the program actually deals with the same memory area.

Upon implementation of self-reproduction functions, Trojan programs can have a structure similar to that of mutant viruses. The mutant virus consists of two main bodies:

- The decipherer, intended for decryption of the executable virus code prior to its execution
- An enciphered executable virus code

The executable code of a virus, besides the components intrinsic to a customary virus, also contains an encryptor intended for encryption of a virus' master code during self-reproduction. After activation of a virus, the first component that gains control is the decipherer that decrypts the main body of a virus and transfers control to it. During self-reproduction, the decipherer is implanted in each copy of a virus, as well as in the enciphered master code. What is particularly significant is that for each new copy, the master code of a virus is enciphered with a new key. The key can depend on the characteristics of the infected file. Varying copies of a virus differ in their usage of different keys of encryption.

In order for the different copies of a mutant virus to contain different decipherers, a master code of a virus contains a generator of decipherers. The main and only function of the generator of decipherers is the creation of a decipherer for each new copy of a virus, which, though possessing a different form, has the same functions. Mutant viruses, including the generator of decipherers, are referred to as polymorphs.

2.2. Counteraction to Unauthorized Internetwork Access

2.2.1. General information

Upon connection of any private computer network to public networks, for example, to the Internet, threats of unauthorized invasion into the protected network from outside becomes quite a significant problem. Along with this, the threat of unauthorized access from a protected network to resources of a public one is a current issue. Similar types of threats are also typical for cases when it is necessary to connect different networks oriented towards completely different levels of information security. When these networks have limited access to each other, violation threats of set limitations exist.

Thus, if a public network, or any other potentially hostile one, is used as an external network, the following threats will exist:

- Threats of intrusion into an internal network from an external one
- Threats of unauthorized access to an external network from an internal one

An intrusion into the internal network from an external one can be performed both for the purpose of unauthorized usage of the internal network's resources, for example, stealing information, and in order to cause network failure. Without the proper means of protection, the probability that these various threats could be implemented is quite high. This is due to the inherent drawbacks of the set of TCP/IP protocols most commonly used for internetworking [7, 13]. This stack of protocols was initially designed for the Internet, which was created as an open system for free information exchange. In earlier versions of the IP protocol, no functions of protection from unauthorized operations were provided, and this remains the case in the current version (e.g., the IPv4 version).

Threats of unauthorized access to an external network from an internal one are possible in cases of restricted access to an external network. Such limitations, which are especially typical for interaction with public networks, can be necessary in the following cases:

- For preventing leakage of confidential data
- For restricting access, for example, in educational institutions, to obscene and undesirable information
- In order to prevent access to entertainment resources during business hours

It is impossible to thwart the threats mentioned above by means of general-purpose operating systems. A universal operating system is too large and too complex, which on one hand, can result in inherent errors and defects, and on the other hand does not always provide protection against administrators' and users; errors.

Modern programming technology fails to ensure the safety of such large programs. Operating systems contain both obvious development errors and essential drawbacks, related to conceptual flaws, as well as to the security system's detailed requirements. Besides this, the administrator dealing with a complex system is far from always being able to adjust and configure it effectively. Finally, there are often breaches committed in the security system of a universal multi-user system by users, for example, by trivial and seldom changed passwords.

It is also important to keep in mind the heterogeneity of modern computer networks. The network of any organization generally represents a heterogeneous set of different computers, controlled by different operating systems and connected by dissimilar network equipment. Computers of the same type and running the same operating system can, in accordance to their purposes, have absolutely different configurations. In such conditions, it is problematic to implement reliable protection of each computer individually from a hostile external network environment.

Therefore, problems of protection from unauthorized operations during interactions with external networks can be successfully resolved only with the help of specialized software and hardware complexes that provide integrated protection of the computer network from a hostile environment. Such complexes are called *internetwork screens* or *firewall systems.* A firewall is installed at the junction between internal and external networks, and undertakes functions involved in counteracting unauthorized internetwork access.

2.2.2. Functions of Internetwork Screening

For counteracting unauthorized internetwork access, a firewall should be placed between the protected network of an organization (an internal network), and a potentially hostile external network (Fig. 2.10). Thus, all interactions between these networks should only be carried out through the firewall. The firewall is incorporated into a protected network.

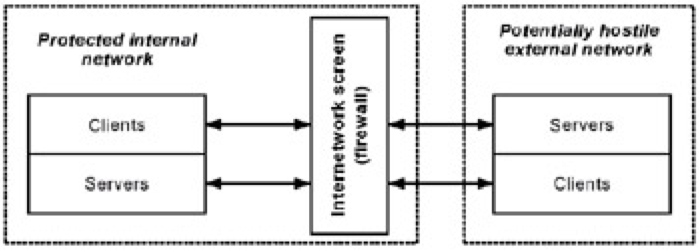

Protected internal network
Clients
Servers
Internetwork screen (firewall)
Potentially hostile external network
Servers
Clients

Fig. 2.10: The connection scheme of an internetwork screen (firewall)

The firewall should allow for communication protocols to be used as the basis for operation of internal and external networks. If these protocols differ, then a firewall should support a mode of operation with a multiple number of protocols, providing protocol conversion for protocols implemented at different levels of the OSI model. Most often, cooperative support of SPX/IPX and TCP/IP protocol stacks becomes necessary.

It so happens that firewalls are asymmetrical. Rules restricting access from an internal network to an external one and vice versa are specially set up. Generally, the operation of the firewall is based on the dynamic execution of two groups of functions:

- Filtering of information flows passing through the firewall
- Mediating at implementation of internetworking actions

Depending on the firewall type, these functions can be performed up to a different extent. Simple firewalls are oriented to execution of only one of these functions. More advanced ones provide the execution of all specified functions of protection. Firewalls can gain self-protection in the same way that protection of general-purpose systems is achieved [4, 23].

In order to provide network security effectively, a complex firewall is obliged to control the entire flow passing through it, and to monitor its status. For making decisions on tools to be used, the firewall should obtain, store, select and process information obtained from all communication levels and from other applications. It is not sufficient to just check packets individually. Information on connection status received from inspection of connections made in the past and from other applications is a major factor when making a controlling decision during new connection attempts. For decision-making, both the status of connection (received from the previous dataflow), and the status of the application (received from other applications) may be considered. Completed and accurate control requires that a complex firewall be able to analyze and use the following units.

- Information on connections: information from all seven layers in the packet
- History of connections: information obtained from previous connections. For example, the outgoing PORT command of an FTP session should be saved, so that it is possible in the future to test an incoming connection's FTP data
- Status of the application level: information collected from other applications. For example, the user authenticated for the current moment can be given the right of access through a firewall only for authorized sorts of tools
- Aggregating units: calculations of different expressions based on all abovementioned factors

The device, similar to a firewall, can also be used for protection of an individual computer. In this case, a firewall is installed on a protected computer. Such a firewall, called a personal firewall, or the system of network screening, inspects all outgoing and incoming traffic, irrespective of all other protective system resources. At the screening of the individual computer, the network service accessibility is supproted, but the load induced by external activity will be reduced or even eliminated. As a result, the internal service vulnerability of the computers protected by this method decreases as an intruder who was initially outside the system, must pass through the firewall where the protective resources are configured very carefully.

2.2.3. Traffic Filtering

The filtering of information flow consists of a selective passage through the firewall, most likely with the execution of some conversions and his data notification to the sender that has been denied admission [14]. Filtering is carried out according to a set of rules that have been loaded in the firewall beforehand which represent network aspects of conventional security policies. Therefore, it is convenient to present a firewall as a sequence of filters (Fig. 2.11), processing the information flow. Each filter is intended for interpretation of individual filtering rules by the execution of the following stages:

1. The analysis of the information by criteria set in interpreted rules, for example, by the recipient's and sender's address, or by the application type for which this information is intended.
2. Coming to one of the following solutions on the basis of interpreted rules:
    - Not to pass data
    - To process data on behalf of the recipient and to return the result to the sender
    - To transmit data to the next filter so the analysis can be continued
    - To let the data pass through by skipping the following filters

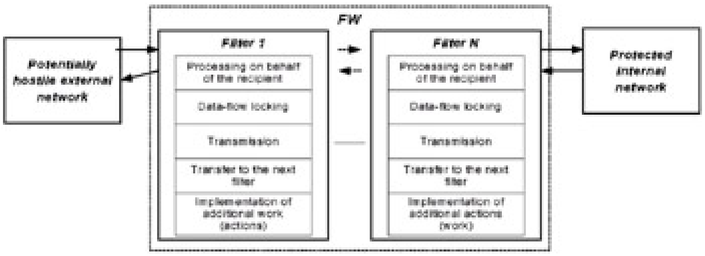

FW
Potentially hostile external network
Filter 1
Processing on behalf of the recipient
Data-flow locking
Transmission
Transfer to the next filter
Implementation of additional work (actions)
Filter N
Processing on behalf of the recipient
Data-flow locking
Transmission
Transfer to the next filter
Implementation of additional actions (work)
Protected internal network

Fig. 2.11: Structure of the firewall

The rules of filtering can also set additional operations that belong to proxy functions, for example, data conversion, event logging, etc. Accordingly, rules of filtering determine the list of conditions by which the following actions are executed by using the specified analysis criteria:

- Permission or prohibition of further data transfer
- Execution of additional protective functions

    The following parameters can be used as criteria of information-flow analysis:

- The service fields of message packets containing e-mail addresses, identifiers, interface addresses, port numbers and other significant data
- Content of the message packets that have been checked, for example, for computer viruses
- External characteristics of information flow; for example, time, frequency characteristics, data amount, etc.

The analysis criteria depend on layers of OSI model in which filtering is performed. In general, the higher the level of the OSI model at which the firewall filters packets, the higher the protection level provided by it.

2.2.4. Execution of Proxy Functions

The firewall performs proxy functions via special programs called *screening agents* or agent programs. They are resident programs that prohibit direct transfer of message packets between external and internal networks.

If access to the external network from the internal network or vice versa is necessary, a logical connection should first be established, with the agent program operating on the firewall computer. The agent program checks if the requested internetworking activity is allowed and, when granting permission, sets individual connection to the required computer. Further information exchange between computers of the internal and external networks is carried out through the agent program, which can perform traffic filtering and other protective functions.

It is necessary to understand that the firewall can perform filtering without using agent programs, by providing transparent interaction between the internal and external networks. At the same time, agent programs do not necessarily filter the traffic.

In general, screening agents can perform the following functions by locking transparent transmission of communication traffic:

- User identification and authentication
- Authentication of transmitted data
- Control of access to internal network resources
- Control of access to external network resources
- Filtering and conversion of communication traffic, for example, dynamic search of viruses and transparent encryption
- Translation of internal network addresses for outgoing message packets
- Event logging, reactions to assigned events, analysis of registered information, and report generation
- Caching of data requested from the external network

User identification and authentication is necessary for a high level of security, not only upon access to the internal network from the external network, but vice versa as well. The password should not be transmitted if it has not been encrypted through commonly used communication facilities. This will prevent the unauthorized access of network-packet interception that is possible, for example, in cases of standard services such as Telnet. Authentication can be successfully achieved by using one-time passwords. Using digital certificates issued by certification centers, for example, by key distribution centers, is reliable and convenient. Most agent programs are designed so that the user needs authentication only at the beginning of session with the firewall. Afterwards, authentication will not be needed for a specific period of time as designated by the administrator.

Agent programs can perform authentication of received and transmitted data. This is not only important for authentication of electronic messages, but also for mobile programs (Java, ActiveX controls), which can be vulnerable to fraud. Message and program authentication is accomplished through checking their digital signatures. Digital certificates can be also used for this purpose.

User identification and authentication enables access to resources of the internal or external network to be demarcated. The methods of differentiation applied to internal-network resources do not differ from differentiation methods of resources supported at an operating-system level [12]. With access differentiation to resources of the external network, one of the following approaches is most often used:

- Permission of access only to specified addresses of the external network
- Filtering of queries, according to updated lists of forbidden addresses, and blocking searches of information resources by using undesirable keywords
- Accumulation and upgrade of the external network's authorized information resources stored on the firewall's disk, and complete prohibition of access to the external network

On the basis of a given set of rules, an agent performs traffic filtering and conversion. It is necessary to distinguish between two kinds of agent programs:

- Screening agents oriented to traffic analysis for particular kinds of services, for example, FTP, HTTP, Telnet
- Universal screening agents processing all traffic, for example, agents oriented towards transparent data encryption or towards virus detection

The program agent analyzes any data packets received and, if any object does not meet the predefined criteria, it either locks it, thus preventing it from being forwarded, or performs appropriate conversions, for example, neutralizing detected viruses. During packet content analysis, it is important that the screening agent can automatically unpack file archives passing through it.

Firewalls, along with agents, enable Virtual Private Networks, or VPNs, to be organized. For example, it is safe to connect some local-area networks that are connected to the Internet as one virtual network. VPN provides connection of local-area networks that are transparent for users, providing privacy and integrity of the transmitted information through dynamic encryption. During transmission over the Internet, the encryption of not only users' data, but also of service information, e.g., source and destination network addresses, port numbers, etc., is possible.

Agent programs can also perform such relevant functions as translation of internal network addresses. The given function is implemented for all packets, following from an internal network to an external one. For these packets, the agent performs an automatic conversion of IP addresses of the computers, sending information into one reliable IP address associated with the firewall, from which all outgoing packets are transmitted. As a result, all packets coming from the internal network happen to be sent by the firewall, thus eliminating direct contact between an authorized internal network and a potentially dangerous external network. The firewall's IP address becomes the only active IP address that gets into the external network.

With such an approach, internal network topology is hidden from external users, making unauthorized access more difficult. In addition to improvement of security, address translation allows for an addressing system inside the network that does not match external network addressing, for example, in the Internet. It effectively solves the problems of address-space expansion of the internal network and the shortage of addresses in the external network.

The key functions of agent programs include event logging, reacting to specific events, registered-information analysis and report compilation. As a necessary response to detecting unauthorized operation attempts, the administrative notification, i.e. warning messages, should be specified. Any firewall that is not capable of sending warning signals upon attack detection is not an effective tool of internetwork security.

Many firewalls contain powerful logging systems for registration, collection and statistical analysis. Registration can be performed by the client's and server's addresses, user identifiers, session duration, connections time, amount of transferred and received data, and administrator and user operations. Logging systems can analyze statistics and provide detailed reports to administrators. Thanks to the use of special protocols, agents can perform remote warnings about particular events in a real-time mode.

With the help of special agents, the caching of data requested from the external network is also supported. When users of the internal network gain access to the external network's information resources, the information is accumulated in a firewall hard disk storage space (in this case the firewall is called a proxy server). Therefore, if the necessary information is found on the proxy server during a regular query, the agent presents it, without a call, to the external network, essentially making access quicker. The administrator needs only to be concerned with periodic upgrades of the proxy-server contents.

The caching function can be used successfully for restriction of access to external-network information resources. In this case, all authorized information resources on the external network are accumulated and updated by the administrator on the proxy server. Internal network users are authorized to have access only to information resources on the proxy server, and direct access to external-network resources is forbidden.

Screening agents are much more reliable than customary filters and provide a greater degree of protection. However, they reduce the performance of data exchange between internal and external networks, and do not have the degree of transparency for applications and end users that is typical for simple filters.

2.3. Features of Firewalls at Different Layers of the OSI Model

Firewalls support security of internetworking at different layers of the OSI model. As this takes place, the functions of protection executed at different layers of a standard model essentially differ from each other. Therefore, a complex firewalls can be conveniently presented as a collection of indivisible firewalls, each of which is oriented to an individual layer of the OSI model. Most often, complex firewalls operate at the network, session and application layers of a standard model. Accordingly, such indivisible firewalls are distinguished (Fig. 2.12) as the screening router, screening transport (session-layer gateway), and also the screening gateway (an application-layer gateway).

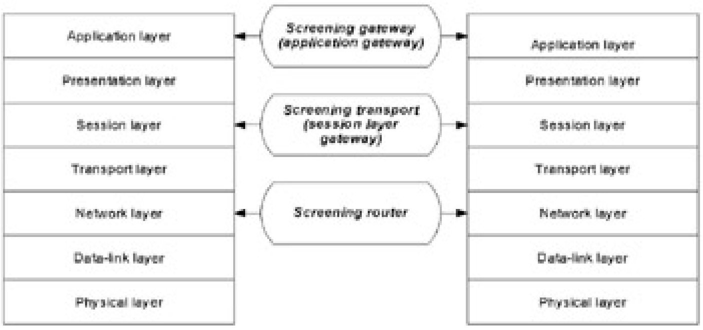


Fig. 2.12: Types of firewalls operating at separate layers of the OSI model

Taking into account that the protocols used in networks (TCP/IP, SPX/IPX) do not exactly correspond to the OSI model, the types of firewalls mentioned above at execution of the functions can also cover adjacent layers of a standard model. For example, the application screen can perform automatic enciphering of messages when they are transmitted to the external network, and also automatic deciphering of received data that is cryptographically protected. In this case, such a firewall operates not only on the application layer of the OSI model, but also on the presentation layer. The session-layer gateway in the operation covers the transport and network layers of the OSI model. At message-packet analysis, the screening router checks their network and transport-layer headers.

Each type of the firewall has its advantages and disadvantages. Many firewalls are either application gateways or screening routers, not supporting complete internetworking security. Reliable protection is provided only with complex firewalls, each of them combining the screening router, the session-layer gateway, and also the application gateway.

2.3.1. The Screening Router

The screening router, also called the packet filter, is intended for filtering message packets and for providing transparent interaction between internal and external networks. It operates on the network layer of the OSI model, but it can cover the transport layer of a standard model for some functions' execution. The decision of whether to let the data pass through or to filter it is made for each packet separately, on the basis of given filtering rules. For decision-making, packet headers of network and transport levels (Fig. 2.13) are analyzed. The following fields of IP and TCP (UDP) headers of each packet are subject to analysis:

- The sender's address
- The recipient's address
- The type of the packet
- The fragmentation flag of the packet
- The number of the source's port
- The number of the recipient's port

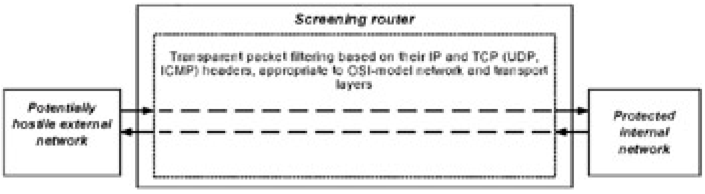

Screening router
Transparent packet filtering based on their IP and TCP (UDP, ICMP) headers, appropriate to OSI-model network and transport layers
Potentially hostile external network
Protected internal network

Fig. 2.13. he operation scheme of a packet filter

The first four parameters belong to the IP header of the packet, and the next parameters belong to the TCP or UDP header.

The addresses of the sender and recipient are IP addresses. These addresses are filled in when the packet is created, and remain unchanged during its transmission on the network.

The packet-type field contains the ICMP protocol code corresponding to the network layer, or the protocol code of the transport layer (TCP or UDP), to which the analyzed IP packet belongs.

The fragmentation flag of the packet specifies whether or not there is a fragmentation of IP packets. If the fragmentation flag for the analyzed packet has been set, the given packet will be a nested packet of the fragmented IP packet.

The source and recipient port numbers are added by the TCP or UDP driver to each transmitted message packet. This uniquely identifies the sending application, as well as the application for which this packet is intended. For example, while using the file-transfer protocol FTP, the implementation of the given protocol on the server by default assigns the number 21 as the number of a TCP port. Each Telnet server has a TCP-port number equal to 23. In order to be able to filter packets by the port numbers, the user must know the network-convention agreements concerning the allocation of port numbers to high-level protocols.

When each packet is processed, the screening router sequentially scans the given table of rules until it discovers a rule to which the complete association of the packet will correspond. Association is understood here as a collection of the parameters indicated in headers of the given packet. If the screening router has received the packet not corresponding to any table rule, it applies a rule specified by default. For safety reasons, this rule specifies that it is necessary to reject all packets that do not correspond to any other rules.

Both customary routers and programs operating on the server configured to filter incoming and outgoing packets can be used as packet filters. Modern routers, for example, routing devices manufactured by such companies as Bay Networks and Cisco, allow users to assign dozens of rules to each port and to filter packets both at input and output.

The advantages of screening routers include:

- Simple and straightforward configuration and installation procedures
- Transparency for software applications and minimal influence upon network performance

- The low cost provided with the fact that any router is able to filter packets to some extent

However, screening routers do not provide a high degree of safety, as they only check packet headers, and do not support many indispensable functions of protection, for example, authentication of source and destination hosts, encryption of message packets, and monitoring of their integrity and authenticity. Screening routers are vulnerable to such common network attacks as address spoofing and the unauthorized change of messagepacket content. It is not difficult to "deceive" such firewalls; one only needs to create packet headers that correspond to filtering rules.

2.3.2. The Session Layer Gateway

The session layer gateway, also called *the screening transport*, controls virtual connections and translation of IP-addresses during interaction with an external network. It operates on the session layer of the OSI model, also covering the transport and network levels of the standard model during the operation. The protective functions of a screening transport are proxy functions.

The control of virtual connections involves the control of connection acknowledgement and also of the information transfer via established virtual channels.

Upon control of an acknowledgement of the link, the session layer gateway watches the establishment of a virtual connection between the internal network's workstation and the external network's computer, determining whether the requested connection session is allowed. Such control is based on information contained in packet headers on the session layer of the TCP protocol. If, however, at TCP-header analysis, the packet filter only checks the port numbers of the source and recipient, the screening transport analyzes other fields related to the process of an acknowledgement of the link.

In order to determine whether the request for a connection is allowed, the session layer gateway performs the following operations. When the workstation (client) requests connection with the external network, the gateway accepts this request, and checks to see whether it corresponds to the basic filtering criteria, for example, whether the DNS server can determine the client's IP address and a name associated with it. Then, operating on behalf of the client, the gateway installs a connection to the external-network computer, and observes the procedures of the connection's acknowledgement by the TCP protocol being executed.

This procedure consists of an exchange of TCP-packets that have the flags SYN (synchronize) and ACK (acknowledge) (Fig. 2.14).

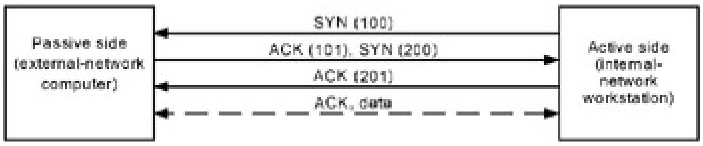

Passive side (external-network computer)
SYN (100)
ACK (101), SYN (200)
ACK (201)
ACK, data
Active side (internal-network workstation)

Fig. 2.14: The scheme of acknowledgement of connection by the TCP protocol

The first packet of a TCP session with a SYN flag containing an arbitrary number, for example 100, is the client's request to open the session. In reply to this, the external-network computer that obtained this packet sends the packet with the ACK flag containing a number exceeding that of the received packet by 1 (in this case, 101), thus confirming that the SYN packet has been received. Additionally, by implementing the return procedure, the external network's computer also sends the SYN packet to the client that will contain a serial number of the first byte of the transmitted data (for example, 200). The client confirms that it has been received by transmitting the ACK packet containing the number 201. The process of the connection's acknowledgement is now complete.

For a session-layer gateway, the requested session is only allowed in the cases where, at execution of an acknowledgement of connection, the SYN and ACK flags and the numbers contained in TCP-packet headers are logically bound with each other. After the gateway has determined that the internal network's workstation and the external-network computer are authorized participants of a TCP session, and has checked validity of this session, then it will allow connection. From this time on, the gateway copies and redirects packets back and forward, inspecting the information transfer via established virtual channel. It supports the table of established connections, skipping data concerning one of the connection sessions that has been registered in this table. When the session is completed, the gateway deletes an appropriate unit from the table and breaks the connection that was used in this session.

During a check of the information transfer by virtual circuits, packet filtering via the screening transport is not carried out. However, the session layer gateway is capable of watching an amount of the transmitted information and terminating connections after a particular limit has been exceeded. This precludes the unauthorized export of the information. Accumulation of the registration information on virtual connections is also possible.

Special programs, called pipe proxies, are used for controlling virtual connections in session-layer gateways. These proxies set virtual circuits between internal and external networks, and then inspect transmission packets generated by the TCP/IP applications on these channels.

Pipe proxies are oriented to specific TCP/IP services. Therefore, session-layer gateways can be used for the extension of possibilities of application-layer gateways, whose operation is based on the agent programs of specific applications.

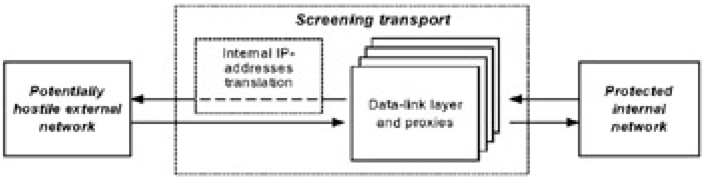

Screening transport
Internal IP-addresses translation
Potentially hostile external network
Data-link layer and proxies
Protected internal network

Fig. 2.15: The operation of a session-layer gateway

In practice, most session-layer gateways are not independent products, but are delivered in a packet with application-layer gateways. The Gauntlet Internet Firewall, put out by Trusted Information Systems, and Digital Equipment Corporation's Alta

Vista Firewall are examples of such gateways. The Alta Vista Firewall use pipe proxies with the proxies of the application layer for six TCP/IP services including, for example, FTP, HTTP (HyperText Transport Protocol) and Telnet.

The session-layer gateway also provides translation of internal addresses of a network layer (IP addresses) upon interaction with the external network. The translation of internal addresses is performed for all packets moving between the internal network and the external one. For these packets, IP addresses of the computers of the internal network sending information are automatically convered into one IP address, which is associated with the screening transport. In the end, the firewall sends all outgoing packets from the internal network, thus eliminating direct contact between the internal and external networks. The IP address of a session-layer gateway becomes the only active IP address that gets into the external network.

On one hand, address translation is caused by the need to increase protection of the secured internal network structure from external users by concealing it. Upon internal IP-address translation, the session-layer gateway screens or shields the internal network from an external world. At the same time, it seems to the internal network's subjects that they are directly communicating with the external-network computers. In addition to increasing safety, address translation provides an addressing system inside the network that does not correspond to addressing in the external network for example, on the Internet. It effectively solves the problem of address space expansion on the internal network and the shortage of addresses in the external one.

On the other hand, address translation occurs on account of the fact that pipe proxies create new connections each time they are activated. Pipe proxies accept requests from the internal network's workstation, and then issue a new request to the external-network computer. Therefore, the external-network computer perceives requests as outgoing from the pipe proxy, instead of from the actual client.

In terms of implementation, the session-layer gateway is a rather simple, reliable program. It supplements the screening router with functions of control of virtual connections and internal IP-address translation.

The drawbacks of a session-layer gateway are the same as those of the screening router: neither control nor protection of message packet content is provided, while user and host authentication, along with other protection functions of a local-area network, are not supported. Therefore, the session-layer gateway is used in addition to the application gateway.

2.3.3. The Application Gateway

The application gateway, also called a *screening gateway,* operates on the application layer of the OSI model, also covering the presentation layer, and provides the most reliable protection of internetworking. The protective functions of an application gateway and screening transport are proxy functions. However, unlike a session-layer gateway, an application gateway can perform many more protective functions. They include the following:

- User identification and authentication during attempts of establishing connections through firewalls
- Authentication of the information transmitted through a gateway
- Access demarcation of internal and external network resources
- Filtering and conversion of communications traffic, for example, dynamic searches for viruses and transparent encryption
- Event logging, reacting to assigned events, analysis of the logged information and generation of the reports
- Caching of data requested from the external network

Since functions of an application gateway are proxy functions, the gateway represents a universal computer, on which program agents (screening agents) operate specifically, one for each serviced application protocol (HTTP, FTP, SMTP, NNTP, etc.).

The agent of each TCP/IP service is oriented to message processing and carrying out protection functions pertaining to this service. The application gateway, like the sessionlayer gateway, intercepts incoming and outgoing packets with the help of appropriate screenig agents, and copies and redirects the information through a gateway. It also operates as the agent server by removing direct connections between the internal and external networks. However, agents used by an application gateway are significantly different from the pipe proxies of session-layer gateways. First, the agents of an application gateway are bound with specific applications (application servers), and, secondly, they can filter communications traffic on the application layer of the OSI model (Fig. 2.16).

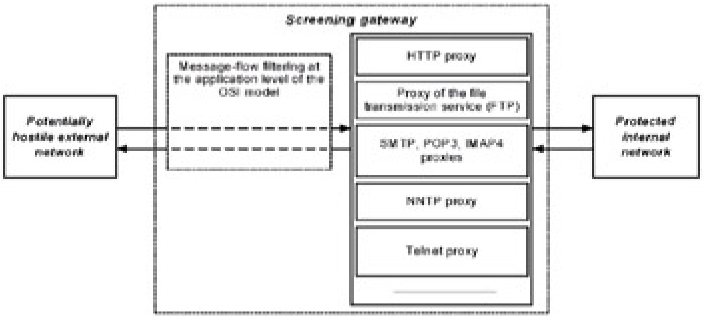

Screening gateway
Message-flow filtering at the application level of the OSI model
HTTP proxy
Proxy of the file transmission service (FTP)
SMTP, POP3, IMAP4 proxies
NNTP proxy
Telnet proxy
Potentially hostile external network
Protected internal network

Fig. 2.16: The operation of an application gateway

Application gateways use application servers as agents of specific TCP/IP services, using HTTP, FTP, SMTP, NNTP, etc. These servers operate on firewalls in the resident mode and implement functions of protection for appropriate TCP/IP services. UDP traffic is serviced by a special translator of UDP-packet content.

As in the case of a session-layer gateway, in order to gain connection between the internal network's workstation and the external-network computer, the appropriate agent of an application gateway sets two connections: from a workstation up to the firewall and from the firewall up to the point of destination. But, unlike pipe proxies, the agents of an application gateway only pass packets generated by those applications that they are authorized to service. For example, the agent program of the HTTP service can only process traffic generated by this service. If the application gateway operates in the network, the incoming and outgoing packets can only be transferred for those services for which appropriate agents exist. So, if the application gateway only uses agent programs such as HTTP, FTP and Telnet, it will only process packets pertaining to these services, and lock those of all remaining services.

Communication-traffic filtering is put into effect by application gateways on the application layer of the OSI model. Accordingly, agents of an application gateway, unlike the pipe proxies, check the contents of processed packets. They can filter specific types of commands or information in the protocol messages of the application layer which are authorized to be serviced by them. For example, for the FTP service, computer viruses in files copied from the external network can be dynamically neutralized. Furthermore, the agent of the given service can be configured to prevent usage of the PUT command, intended to record by clients on the FTP-server. Such a restriction reduces the risk that information will be accidentally damaged and prevets tons of unneeded information from being stored on the FTP server. It also makes it less likely that space will be taken up by gigabytes of waste data.

Upon adjustment of an application gateway and description of the rules of message filtering, the following parameters are used: the service name; the allowed time period of its usage; restriction of message content; the computers from which it is possible to use this service; user identifiers; and authentication schemes, among other things.

The application layer gateway has the following key advantages:

- Due to its ability to execute an overwhelming majority of proxy functions, it provides the highest level of protection of a local-area network.
- Protection at the application level ensures several extra monitoring opportunities, thus reducing the likelihood of successful attacks brought about by weaknesses in the software.
- Upon malfunction of an application gateway, packets between divided networks will be blocked from passing through, which does not reduce the safety of the protected network in cases of system failure.

In addition to its high cost, the drawbacks of an application gateway include:

- A highly complex firewall, including its installation procedures and configuration.
- The computer platform must guarantee a high level of performance and offer the proper resources.
- Lack of transparency for users and a reduced throughput.

Let us consider the latter in greater detail.

The operation technology of an application gateway is based on using agents that check the authenticity of clients who are trying to gain access to them. It is also geared toward setting necessary connection and protection and performing other functions of internetworking. Agents act as proxies during packet transmission between the server and client. First, the connection to the agent is set, then the agent makes a decision about whether or not to set connection with the recipient.

Accordingly, during its operation, an application gateway duplicates any connection that has been allowed. This results in a lack of transparency for users and the additional overhead expense for servicing connections.

In order to eliminate this issue, Check Point and ON Technology have created a new technology of packet filtering, which is sometimes called expert-level filtering, or filtering with connection-status control (stateful inspection). Filtering is carried out through special methods of a multilevel analysis of packet status (Stateful Multi-Layer Technique, or SMLT). This hybrid technology permits inspection of the network connection's status by intercepting packets on the network layer and extracting application-layer information from them. This is then used for connection monitoring. Fast comparison of packets which pass through with "friendly" packets considerably reduces processing time in comparison with, for example, application-level firewalls.

Firewalls, whose functioning is based on the filtering technology described above, are called expert-level firewalls. Such firewalls combine both the components of screening routers and application gateways. Like screening routers, they provide packet filtering by their header content on the OSI model's network and transport levels. Expert level firewalls also perform all functions of an application gateway in respect to packet filtering on the OSI model's application layer. They estimate the content of each packet according to a given security policy.

In addition to transparency for users and a higher rate of information processing, one of the advantages of expert-level firewalls is that firewalls do not change IP addresses of packets passing through them. This means that any protocol of an application layer using IP addresses will correctly operate with these firewalls without any changes or special programming. However, since these firewalls allow a point-to-point connection between an authorized client and a computer of the external network, they do not provide such a high level of protection. Therefore, in practice, expert-level filtering technology will be used to increase efficiency of operation of complex firewalls. Examples of complex firewalls that conduct expert-level filtering technology are FireWall-1 by Check Point and ON Technology's ON Guard.

Expert-level filtering is becoming one of the new routers' functions. For example, Bay Networks and Check Point have drawn up a partner agreement with the purpose of transferring the architecture of expert-level firewalls designed by Check Point to Bay routers. Cisco Systems has elaborated its own high-level firewall technology, and has thus created the Cisco PIX Firewall.

2.4. Installation and Configuration of Firewall Systems

For effective protection of internetworking, the firewall system should be correctly installed and configured. The given process is carried out by execution of the following steps:

- Developments of internetworking policies
- Evaluation of the interconnection scheme, as well as that of the firewall's direct connection
- Setting parameters of firewall operation

The steps mentioned above represent the system's approach to installation of any software/hardware resource meaning consequent detailing of the given task's solution.

2.4.1. Development of Internetworking Policy

The internetworking policy is that part of the security policy in an organization which determines the safety requirements of information exchange with the external world. These requirements should necessarily mirror two aspects:

- The policy of access to network services
- The policy of operation of the firewall

The policy of access to network services determines rules of rendering, and usage of all possible services to a protected computer network. Accordingly, within the framework of a given policy, all services provided through the firewall, and also permissible client addresses for each service, should be set. In addition, the rules designating which users may use a service, which service that is, when it can be used and on which computer should all be specified. User and computer-authentication rules, as well as the operation conditions of users outside a local-area network of an organization are specifically determined.

The operation policy of a firewall sets a base principle of control of internetworking forming the basis for firewall operation. One of two such principles can be selected:

- Everything that is not explicitly allowed is forbidden
- Everything that is not explicitly forbidden is allowed

Depending on the choice, a solution can be accepted for the benefit of safety and to the detriment of operability of network services, and vice versa. In the first case, the firewall should be configured so as to lock any internetworking communications not allowed in an explicit form. Bearing in mind that such an approach allows for a significant reduction in privileges, it is the best one in terms of security. With this approach, the administrator cannot accidentally leave any privileges open, as this is forbidden by default. The accessible, extra services can be used to the detriment of security, which is especially typical for closed and complex software that may contain different errors. The principle "Everything that is not explicitly allowed is forbidden" is, in fact, the admission of the fact that ignorance can cause harm.

In choosing the principle "Everything that is not explicitly forbidden is allowed", the firewall is adjusted so as to lock only internetwork communications forbidden in an explicit form. Operability of network services for users is increased in this case, but internetworking security is reduced. The administrator may not take into account all operations prohibited to the users, and should work in the response mode by foreseeing and prohibiting those internetwork communications that can have negative effects on network security.

2.4.2. Elaboration of the Farewall Connection Scheme

Different schemes that depend on operation conditions, and also on the amount of network interfaces of a firewall can be used for connection of firewalls.

Firewalls with one network interface (Fig. 2.17) are not effective enough both with respect to safety, and convenience of configuration. They do not differentiate internal and external networks physically and, accordingly, they cannot provide reliable protection of internetworking. The adjustment of such firewalls, and of the routers bound with them, is a rather complex task, whose solution cost exceeds the cost of replacing a firewall with one network interface with a firewall with two or three network interfaces. Therefore, we shall only consider the connection scheme of firewalls with two or three network interfaces. In this case, we shall consider a protected local-area network as a collection of protected and public subnets. A public subnet in this context is understood as a subnet to which access from a potentially hostile external network can be provided partially or completely. A public subnet can, for example, include shared WWW, FTP and SMTP servers, and also a terminal server with a modem pool.

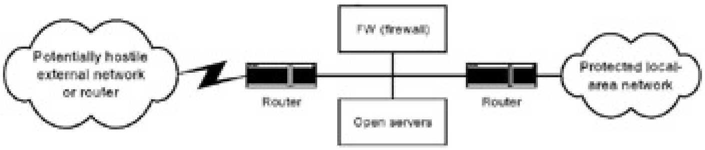

Potentially hostile external network or router
FW (firewall)
Router
Router
Open servers
Protected local-area network

Fig. 2.17: Protection of a local-area network with a firewall with one network interface

Among all sets of the possible connection schemes of firewalls, the following are among the standard ones:

- The comprehensive protection scheme of a local-area network
- The scheme with a protected subnet and an unprotected public subnet
- The scheme with separate protection of internal and public subnets

The comprehensive protection scheme of a local-area network is the simplest solution (Fig. 2.18), in which one firewall completely screens a local-area network from a potentially hostile external network. There is only one path between the router and firewall, along which all traffic passes. Usually, a router is adjusted in such a manner that the firewall is the only machine visible outside the computer. The firewall will also protect the public servers that are included in a local-area network. However, the association of servers accessible from the external network with other resources of a protected local-area network significantly reduces the security of internetwork communications. Therefore, the given connection scheme of a firewall can only be used if public servers are absent in a local-area network, or when available public servers are made accessible from the external network for only a limited number of users.

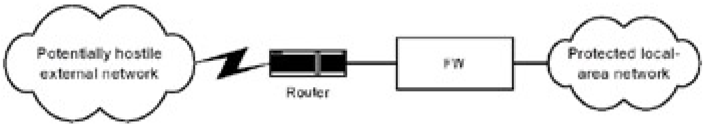

Potentially hostile external network
Router
FW
Protected local-area network

Fig. 2.18: The comprehensive protection scheme of a local-area network

When shared public servers are present in a local-area network, it makes sense to take them out as a public subnet from a local-area network outside the firewall (Fig. 2.19). The given method provides a higher security level for the protected part of a local-area network, but a lower security level for public servers located outside the firewall. Some firewalls allow users to locate these servers by themselves. But this is not the best solution with regard to computer workload and firewall security. Taking these factors into consideration, it is possible to draw the conclusion that it is a good idea to use the connection scheme of a firewall with a protected subnet, and not a protected public subnet only with low-level requirements for public subnet security.

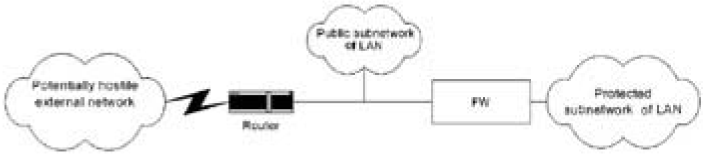

Public subnetwork of LAN
Potentially hostile external network
Router
FW
Protected subnetwork of LAN

Fig. 2.19: A scheme with protected and unprotected public subnets

In cases where the safety of public servers needs to meet more stringent requirements, it is necessary to use a scheme with separate protection of protected and public subnets. Such a scheme can be built with one firewall that has three network interfaces (Fig. 2.20) or on the basis of two firewalls with two network interfaces (Fig. 2.21). In both cases, access to the public and protected subnets to those of a local-area network is only possible through the firewall. Thus, the access to a public subnet does not allow access to the protected subnet.

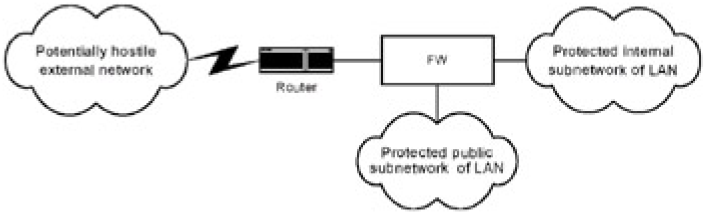

Potentially hostile external network
Router
FW
Protected internal subnetwork of LAN
Protected public subnetwork of LAN

Fig. 2.20: A scheme with separate protection of internal and public subnets on the basis of one firewall with three network interfaces

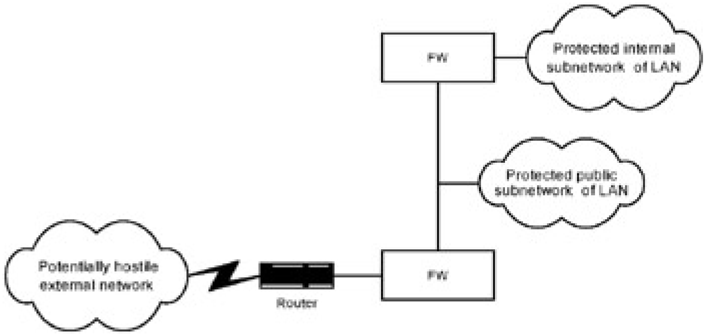

FW
Protected internal subnetwork of LAN
Protected public subnetwork of LAN
Potentially hostile external network
Router
FW

Fig. 2.21: A scheme with separate protection of internal and public subnets on the basis of two firewalls with two network interfaces

Of the last two schemes, a greater degree of internetworking safety is provided with a scheme with two firewalls, each of them giving a separate protection level of the protected internal subnet. The protected public subnet here acts as a screening subnet. Usually, a screening subnet is configured in order to provide access to computers of a subnet, both from a potentially hostile external network and the protected internal subnet of a local-area network. However, it is impossible to have a direct exchange of information packets between the external network and a protected internal subnet. During system attacks with a screening subnet, it is necessary to get through at least two independent lines of protection, which is a rather difficult task. Monitoring facilities of firewall status will necessarily detect such an attempt, and the system administrator will undertake the necessary measures to prevent unauthorized access.

One should pay attention to the fact that remote user operation, connected through switched communication lines, should also be inspected, in accordance with conventional security policies. The standard solution of this task can be found in the installation of a remote-access server (a terminal server) which has the necessary functional capabilities, for example, the Bay Networks Annex terminal server. The terminal server is a system with several asynchronous ports and one interface of a local-area network. The information exchange between asynchronous ports and the local-area network is carried out only after the external user's appropriate authentication.

The terminal server's connection should be carried out so that its operation is only performed through the firewall. This will ensure the necessary degree of security during remote user operation with an organization's information resources. Such a connection is possible if the terminal server is included with the public subnet during use of the firewall's connection schemes with separate protection of public and protected subnets.

The terminal server's software should provide administration and control of connection sessions through switched channels. The control modules of modern terminal servers are advanced enough to provide security to the server and to demarcate client access, by doing the following:

- Using local passwords for access to a serial port, for remote access by the PPP protocol, and also for access to an administrative console
- Using requests for authentication from any computer of a local-area network
- Using external authentication tools
- Setting access control lists for terminal server ports
- Logging connections through the terminal server

2.4.3. Setting up Parameters for Firewall Operation

The firewall is the software/hardware protection complex consisting of the computer, the operating system and the special software run on it. Notice that this special software is often also called the firewall.

The computer on which the firewall is installed should be quite powerful and physically protected, for example, it should be located in a specially assigned and secure place. Additionally, it should have tools protecting against booting of an operating system from an unauthorized media.

The operating system of a firewall should also meet a number of requirements:

- Have the means to demarcate access to system resources

- Lock access to computer resources and to the provided program interface
- Prohibit privileged access to its resources from a local-area network
- Contain monitoring/auditing capabilities of any administrative operations

The given requirements are met by different clones of the Unix operating system, and also by Microsoft Windows NT/2000.

After installation of a selected operating system on the computer, firewall configuration, and installation of special software, it is possible to start adjusting operation parameters of the whole firewall. This process includes the following stages:

- Elaboration of operation rules of the internetwork screen, according to developed internetworking policies and description of rules in the firewall interface
- Check the given rules for consistency
- Check that firewall settings correspond to developed policies of internetworking

The base of operation rules of the firewall created at the first stage is the formalized reflection of developed internetworking policies. The rules' components are protected objects, users and services.

Protected objects may include customary computers with one network interface, gateways (computers with several network interfaces), routers, network and control areas. Protected objects can be combined into groups. Each object has a set of attributes, such as a network address, the subnet mask, etc. Users need to set some of these attributes manually, and the rest are extracted automatically from information bases, for example NIS/NIS+, SNMP MIB, DNS. Users should pay attention to complete object description, as it is possible to check the accuracy of given screening rules only when all network interfaces of gateways and routers are determined. Such information can be received automatically from the SNMP agent.

During the description of the firewall's operation rules, users are assigned entry names and combined in groups. The permissible initial and target network addresses, the range of the operation's date and time, and also the authentication scheme and order are specified for users.

The definition of the set of services to be used is performed through the database, and is built into the distribution kit of firewalls that have large sets of TCP/IP services. Non-standard services can be set manually with the help of special attributes. Prior to specifying the service while setting the rules, one must specify its properties. Modern firewalls contain definitions of all standard TCP/IP services prepared beforehand, and subdivided into four categories: TCP, UDP, RPC, ICMP.

The TCP services are completely inspected tools, as they are rendered and used on the basis of easily diagnosed virtual connections.

UDP services are traditionally difficult for filtering, as there is neither a phase for virtual-connection settings, nor a dialogue context between the client and server.

Firewalls can figure out this context by watching all UDP packets crossing the firewall in both directions, and by associating requests with their answers. As a result, there is a kind of virtual connection for datagram protocol, and all attempts of setting such connections illegally, as well as datagrams outside established connections, are processed according to the conventional policies of internetworking.

The RPC services are difficult for filtering, because of variable numbers of used ports. Firewalls watch RPC traffic, determine requests to the PORTMAPPER function and extract port numbers from the answers.

The ICMP protocol is used by the IP protocol itself for sending test messages, information on errors, and for testing of network integrity. For the ICMP protocol, the concept of ports is not used. It uses the numbers from 0 to 255 to indicate the type of service that, along with addresses, is considered during internetworking control.

After the base of rules is created, it is checked for consistency. This is a very important point, especially for developed, multi-component network configurations with complex internetworking policies. Without such a check, the administration of the firewall would inevitability result in numerous errors and the creation of vulnerabilities. Checking the created rules for consistency is performed automatically. Any ambiguities that have been detected should be eliminated, by editing the contradicting rules. After the final determination of rules and elimination of errors, the administrator may have to perform additional operations on the compilation and installation of filters and agents. After creating a base of rules, most firewalls automatically go through the final adjustment process.

The verification that firewall settings correspond to the designed internetworking policies can be fulfilled through operation-protocol analysis of the firewall. However, the greatest performance of such a test will be gained using specialized systems of the analysis of the network's security level. The Internet Scanner SAFEsuite software package, put out by Internet Security, is the most promising representative of such systems.

The Firewall Scanner subsystem, which is included in the given package, provides searches of weak places in firewall configuration and provides guidelines on how to correct them. This search is carried out via a response check of firewalls for different types of safety-violation attempts. As this takes place, all network services where access is carried out through the firewall, are scanned. For permanent maintenance of a high level of network security, it is recommended to include Firewall Scanner as a part of the firewall installation.

When the firewall is set up, it is necessary to remember that, like any other feature, it cannot protect the system from administrator and user incompetence. Unauthorized penetrations into protected networks can take place, for example, owing to the selection of a password that can be easily guessed. The screening system also does not protect itself from attacks on data links inspected by it. If there is an unmonitored channel between a potentially hostile external network and a protected internal network, a firewall will not be able to protect it from attacks that come through it. This also applies to telephone channels of data transfer. If a modem allows connection to a protected network passing through the firewall, protection will be reduced to nothing. In this case, one must recall the philosophy of protection: the system is as safe as its most unguarded link. Therefore, that the screening system needs to inspect all channels of information transmission between internal and external networks.

2.5. Firewall Evaluation Criteria

2.5.1. General Requirements

The qualifying standards in a computer network for any information-protection tool can be divided into the following categories:

- *Functional requirements*-the tool must provide the solution for a required group of protection tasks
- *Reliability requirements*-the tool must be able to fulfill all provided protection functions correctly and at the proper time
- *Adaptability requirements*-the tool must be able to adapt to structural changes of the organization, technological schemes and conditions of the network operation
- *Ergonomic requirements*-the tool must be convenient to administer and easy to use, and provide a minimal amount of difficulties to end users
- *Economic requirements*-the tool must minimize financial expenses and resource consumption

A firewall must correspond to the following groups of more detailed requirements:

Functionality-Firewalls must secure the protected internal network, and provide full control over external connections and communication sessions. They must be able to authorize user access through external connections. A typical situation is one in which personnel have to go away, for example, on a business trip (or a study tour) and, while working, they need access to some resources of the organization's internal computer network. The firewall must be able recognize them and give them all necessary means of access.

Controllability and flexibility-A firewall must provide a powerful and flexible set of control tools, in order fully to implement the organization's security policy. Furthermore, the firewall must be capable of easy reconfiguration in case the organization's network undergoes structural changes. If an organization has several external connections, including remote affiliations, the firewall-management system must provide the capabilities for centralized distribution of the unified internetwork-communication policy.

Performance and transparency-A firewall must be efficient, and able to process all incoming and outgoing traffic. This requirement must be observed in order to prevent the possibility of overloading the firewall with a large amount of requests, thus disrupting its operation. For local-network users, a firewall must operate smoothly and transparently, without complicating execution of legal operations. Otherwise, users will try to bypass all installed security limitations in any way possible.

Self-protection-A firewall must be able to protect itself against any unauthorized activities. Since the firewall is both the key and the door to the organization's confidential information, it must thwart all attempts at unauthorized modifications of its configuration parameters. It must also include advanced tools for self-testing and notifying the administrator of its status. The notification and alerting system must warn security personnel in a timely manner when the firewall detects any unauthorized activity, malfunctions or fails to operate.

Currently, the usage of a predefined set of qualitative requirements to security subsystems is a generally accepted approach to the development of security facilities evaluation criteria. The same approach is also applicable to firewalls, which are classified by their level of protection against unauthorized access. This classification is built on the basis of the list of security parameters and a set of requirements describing them.

Security parameters are applied to firewalls in order to evaluate the security level that they provide during internetwork communications. Specific lists of parameters define firewall-security classes. Firewall classification by security levels is required, in order to develop and adopt economically justified measures ensuring the required security level for all internetwork communications.

In terms of security level, there are five firewall classes. The lowest level of protection is the fifth, while the highest is the first level, which is used in order to secure all interactions of the A1 class computer systems with the external environment. Including a firewall into an automated system of a specific security level must not degrade the security level of the whole system. Depending on the importance of the confidential information, the following firewalls must be used:

- When processing information labeled as "confidential"-not lower than the 3rd class
- When processing information labeled as "secret"-not lower than the 2nd class
- When processing information labeled "top secret"-not lower than the 1st class

Let's remember that, in accordance with the Department of Defense Trusted Computer System Evaluation Criteria, known as the "Orange Book", there are seven security levels for computer systems:

- D1-minimal protection. Reserved for systems that have undergone evaluation and testing procedures, but failed to meet the requirements for higher security classes.
- C1-discretionary security protection. C1-class systems must satisfy the discretionary security requirements by providing separations of users and data, establishing access limitations on an individual basis. In other words, such systems allow users to protect private information and prevent other users from accidentally reading or destroying their data.
- C2-controlled-access protection. In comparison to the C1-level, C2-level systems enforce more finely grained discretionary-access control. Users are individually accountable for their activities through login procedures, auditing of security events and resource isolation.

- B1-labeled security protection. B1-level systems include all features required for C2 class, plus an informal statement of the security policy model, data labeling and mandatory access control over named subjects and objects.
- B2-structured protection. The B2-class system must be based on a clearly defined and documented formal security policy model, which requires the discretionary and mandatory access control enforcement found in the B1-class. Additionally, discretionary and mandatory access control is extended to all subjects and objects. Authentication mechanisms are strengthened, and trusted facility management has been provided. This is the first level, characterized as relatively resistant to penetration.
- B3-security domains. The systems of this class must satisfy the reference monitor requirements that mediate all access of subjects to objects, be tamper-proof, and support extended security-auditing mechanisms, signal security-related events and provide system-recovery procedures. Systems of this class must be highly resistant to penetration.
- A1-verified design. Systems satisfying the A1-class requirements are functionally equivalent to those in class B3, but additionally are distinguished by the use of formal security verification methods, to ensure that the mandatory and discretionary security controls implemented in the system can effectively protect information stored or processed in the system.

The requirements to security subsystems for the Automated Data Processing (ADP) system of each level are outlined in Table 2.1.

Table 2.1: Security Classes Requirements According to the Department of Defense Trusted Computer System Evaluation Criteria (the "Orange Book")

Subsystems and requirements	**Security classes**						
	D1	**C1**	**C2**	**B1**	**B2**	**B3**	**A1**
Audit	NR	NR	NEW	ENH	ENH	ENH	NAR
Configuration management	NR	NR	NR	NR	NEW	NAR	ENH
Covert channel analysis	NR	NR	NR	NR	NEW	ENH	ENH
Design documentation	NR	NEW	NAR	ENH	ENH	ENH	ENH
Design specification and verification	NR	NR	NR	NEW	ENH	ENH	ENH
Device labels	NR	NR	NR	NR	NEW	NAR	NAR
Discretionary access control	NR	NEW	ENH	NAR	NAR	ENH	NAR
Export of labeled information	NR	NR	NR	NEW	NAR	NAR	NAR
Export to multilevel devices	NR	NR	NR	NEW	NAR	NAR	NAR
Export to single-level devices	NR	NR	NR	NEW	NAR	NAR	NAR
Identification and authentication	NR	NEW	ENH	ENH	NAR	NAR	NAR
Label integrity	NR	NR	NR	NEW	NAR	NAR	NAR
Labeling human-readable output	NR	NR	NR	NEW	NAR	NAR	NAR
Labels	NR	NR	NEW	ENH	ENH	NAR	NAR
Mandatory access control	NR	NR	NR	NEW	ENH	NAR	NAR

Table 2.1: Security Classes Requirements According to the Department of Defense Trusted Computer System Evaluation Criteria (the "Orange Book")

Subsystems and requirements	**Security classes**						
	D1	**C1**	**C2**	**B1**	**B2**	**B3**	**A1**
Object reuse	NR	NR	NEW	NAR	NAR	NAR	NAR
Security features user's guide	NR	NEW	NAR	NAR	NAR	NAR	NAR
Security testing	NR	NEW	ENH	ENH	ENH	ENH	ENH
Subject sensitivity labels	NR	NR	NR	NR	NEW	NAR	NAR
System architecture	NR	NEW	ENH	ENH	ENH	ENH	NAR
System integrity	NR	NEW	NAR	NAR	NAR	NAR	NAR
Test documentation	NR	NEW	NAR	NAR	ENH	NAR	ENH
Trusted distribution	NR	NR	NR	NR	NR	NR	NEW
Trusted facility management	NR	NR	NR	NR	NEW	ENH	NAR
Trusted facility manual	NR	NEW	ENH	ENH	ENH	ENH	NAR
Trusted path	NR	NR	NR	NR	NEW	ENH	NAR
Trusted recovery	NR	NR	NR	NR	NR	NEW	NAR

Here:

- NR (No Requirement)-this requirement is not included with this class.
- NEW-this requirement has first appeared in this class and supersedes any criteria found in the lower class.
- ENH-the criteria in this requirement have either been enhanced to strengthen security or newer criteria have been added, compared to the lower classes.
- NAR (No Additional Requirements)-this requirement has not been changed from the previous class.

The requirements of a firewall according to protection class are given in Table 2.2.

Table 2.2: Firewall Parameters According to Security Classes

Protection criteria	**5**	**4**	**3**	**2**	**1**
Access control (data filtering and address translation)	+	+	+	+	=
Identification and authentication	−	−	+	=	+
Logging	−	+	+	+	=
Administration: identification & authentication	+	=	+	+	+
Administration: auditing	+	+	+	=	=
Administration: ease of use	−	−	+	=	+
Integrity	+	=	+	+	+
Recoverability	+	=	=	+	=
Testing	+	+	+	+	+

Table 2.2: Firewall Parameters According to Security Classes

Protection criteria	5	4	3	2	1
Security administrator manual	+	=	=	=	=
Test documentation	+	+	+	+	+
Project documentation	+	=	+	=	+

Here:

- "−"-no requirements for the given class
- "+"-new or additional requirements
- "="-the requirements are unchanged in comparison to the previous class

2.5.2. Requirements for Firewall Protection Classes

Requirements for Firewalls of the 5th Protection Class

Access control. A firewall must provide filtering on the network level. On account of sender and recipient network addresses, or equivalent attributes for each network packet, the decision of filtering can be made independently.

Administration: Identification and authentication. A firewall must provide identification and authentication for the administrator at a local request for access. Furthermore, it must provide an option for providing identifier and password of an alternative operation.

Administration: Logging. A firewall must provide auditing capabilities of an administrator's attempt to login or logout, as well as for system startup and shutdown. Logout event is not logged when the firewall hardware is powered off. The following data must be included as logged parameters:

- Date, time and code of the registered event
- Result of the operation - success or failure
- Firewall administrator login ID, supplied when attempting the logged action

Integrity. A firewall must include integrity-control features for its software and informational components.

Recovery. A firewall must provide recovery procedures to be activated after hardware failures and malfunctions. These procedures must ensure successful recovery of the firewall properties.

Testing. In a firewall, the feature of scheduled testing (ST) must include:

- Implementation of ST of filtering rules
- ST of the process of the administrator's identification and authentication
- ST of the process of auditing administrator's activity

- ST of the process of integrity control for software and informational components of the firewall
- ST of recovery procedure

Firewall administrator's manual. The manual must include:

- Description of manageable and controllable firewall functions
- Firewall setup and configuration manual
- Description of the firewall startup and initialization, and procedures for startup validity check
- Firewall recovery manual

Test documentation. This documentation must include test results and the types of tests that a firewall has undergone.

Project documentation: This documentation must include the following components:

- Generalized firewall scheme
- General descriptions of the following aspects:
  - Firewall's working principles
  - Filtering rules
  - Identification and authentication tools and processes
  - Auditing tools and processes
  - Integrity control tools
  - Recovery procedure

Additional Requirements for 4th-Class Firewalls

Access control. In addition, a firewall must provide:

- Network-packet filtering of protocols applied for diagnosing and controlling network-device operation
- Filtering of inbound and outbound network traffic to ensure validity of network addresses
- Contents filtering for any significant field of network packets

Logging. In addition, a firewall must provide logging capabilities to register any filtered packets. Address, time, and the result must be included as logged parameters.

Administration: Auditing. In addition, a firewall must provide auditing capabilities to audit events such as starting programs or processes for execution.

Testing. In addition, scheduled testing of the logging process.

Test documentation (TD). This must include a description of types of tests that a firewall has undergone, and the test results.

Additional Requirements for 3rd-Class Firewalls

Access control. In addition, the firewall must provide:

- Transport-layer filtering of requests for installing virtual connections. At least sender and recipient addresses must be taken into account
- Application-layer filtering of the application layer requests to application services. At least sender and recipient addresses must be taken into account
- Filtering with respect to date/time

Identification and authentication. The firewall must be able to authenticate input/output requests using methods resistant to passive and active data snooping.

Logging. In addition, the firewall must supply:

- Logging any requests for installing virtual connections
- Local warnings on any attempts of filtering rules violation

Administration: Identification and authentication. The firewall must deny access to any subject who failed to be identified or authenticated. Administrator's requests for remote authentication must be secured using methods resistant to passive and active data snooping.

Administration: Auditing. Additionally, the firewall must audit administrative attempts to change filtering rules.

Administration: Ease of use. A multicomponent firewall (MFW) must provide the capability of remotely controlling its components as long as necessary, in order to provide filter configuration, verification of mutual consistency of all filters and examination of logged information.

Integrity. The firewall must provide integrity control for its software and informational components using the checksums method.

Testing. In addition, the user must also be able to conduct scheduled testing of identification and authentication processes.

Test documentation (TD). This must include the description of test types that a firewall has undergone and test results.

Project documentation. In addition, this documentation must include the description of the process of centralized control of a firewall's components.

Additional Requirements for 2-nd Class Firewalls

Access control. In addition, the firewall must be able to:

- Hide objects and/or application functions of a protected network
- Translate network addresses

Registration. In addition, the firewall must provide:

- Remote signaling of any attempts to violate filtering rules
- Logging and auditing capabilities for requested application services
- A programmable response to events in a firewall

Administration: Identification. A firewall must offer identification and authentication according to the identifier (code) and temporary-access password. A firewall must deny access to any person who failed to be identified or authenticated. Administrative requests to remote access must be protected, using methods resistant to passive and active data snooping.

Integrity. The firewall must include integrity-control tools for its software and informational components based on checksums. The check must take place both during startup and dynamically.

Recovery. The firewall must provide a recovery procedure to restore the firewall after hardware failures and malfunctions. This feature must ensure fast and efficient recovery of the firewall's properties.

Testing. In a firewall, scheduled testing must be provided for the following aspects:

- Implementation of filtering rules
- Logging
- Identification and authentication of requests
- Identification and authentication of administrator
- Auditing of administrative activity
- Integrity control process for software and informational components of the firewall
- Recovery procedure

Test documentation. Must include the description of the tests that a firewall has undergone, and test results.

Additional Requirements for 1st-Class Firewalls

Identification and authentication. In addition, a firewall must provide identification and authentication of all application-layer subjects.

Administration: identification and authentication. A firewall must provide an administrator's identification and authentication at the time of his or her requests for access. A firewall must be capable of performing identification and authentication according to biometrics or special devices (tokens, cards, or electronic keys) and an alternative password. A firewall must deny access to anyone who fails to be identified or authenticated. Administrative requests for remote access must be protected using methods resistant to passive and active data snooping.

Administration: ease of use. A multicomponent firewall (MFW) must provide centralized control over all its components as long as it necessary in order to provide filter configuration, verification of mutual consistency for all filters and examination of logged information. A Graphic User Interface (GUI) must be provided.

Integrity. A firewall must include integrity control for its software and informational components. The procedure must be based on certified algorithm, and it must be performed both during startup and dynamically.

Testing. In addition, a firewall must include a scheduled testing process of the centralized control of its components and the GUI to control the firewall.

Test documentation. Must include the description of tests that a firewall has undergone and test results.

Design (project) documentation. In addition, this documentation must include a description of the GUI provided to control the firewall.

2.6. Overview of Contemporary Firewall Systems

2.6.1. General Overview

Practically none of the currently popular firewalls can be classified as belonging to a specific type. Contemporary firewalls are either complex, or at least combine functions of several firewall types: filtering router, session-level gateway, application gateway and expert-level gateway. Each of the modern firewalls is designed on the basis of a specific concept using a unique approach.

As a result of the current competitive environment, firewalls are easier to use, less expensive and possess extended functionality. Currently, firewall systems that don't provide VPN protection based on traffic encryption are quite rare. Furthermore, most firewalls provide strengthened user authentication, packet filtering with stateful inspection as well as high availability. Graphic interfaces, configuration wizards and preconfigured devices help to turn firewall systems into technology intended for massmarket consumption. At the moment, a multifunctional corporate firewall costs approximately $3,000. On the other hand, it is necessary to take note of the fact that the cost of the most advanced firewalls starts from $10,000.

Now let us consider some specific features of popular contemporary firewalls.

Hardware Implementation

The need for typical and unified solutions has stimulated the development of hardware firewalls. As a rule, such devices represent specialized computers with a preinstalled and preconfigured operating system, as well as with firewall software. Although most firewall implementations run some version of Unix or Linux, stripped of unnecessary programs and utilities, firewall manufacturers also provide GUI administrative tools. The main advantage of hardware firewalls is the simplicity of putting them into effect, since they require neither installation of an operating system nor firewall software. Configuration of hardware firewall is also simplified thanks to the existence of unified templates for configuring firewall rules.

Some examples of hardware firewalls are: Secure PIX Firewall 520/525, from Cisco Systems; NetScreen-100/500, from NetScreen Technologies; and WatchGuard's Firebox 1000.

Devices such as Secure PIX Firewall 520 and Secure PIX Firewall 525 run a specialized, protected operating system, PIX OS. The throughput of models 520 and 525 is 370 Mbit/sec. Secure PIX Firewall 520 can simultaneously serve up to 250,000 sessions, while Secure PIX Firewall 525 can serve up to 280,000 sessions. Cisco uses the Adaptive Security Algorithm (ASA) as its protection method. ASA is a variation of the context-check algorithm. ASA records sender and recipient addresses, TCP sequential numbers and other data required for identifying connections. Both devices provide VPN functions, such as tunneling and DES/TripleDES encryption. In 2001, Cisco announced a new version of the OS firewall-PIX OS 6.0. In contrast to its predecessors, this OS provides the GUI PIX Device Manager (PDM) as an alternative to the command-line interface. Furthermore, version 6.0 provides improved support for firewall functions for the Voice over IP (VoIP) protocol.

NetScreen Technologies supplies efficient models of hardware firewalls, such as NetScreen-100 and NetScreen-500. In NetScreen-100, the developers have provided three 10/100 Mbit/sec Ethernet ports simultaneously supporting 128,000 TCP sessions and 1,000 VPN tunnels. Two NetScreen-100 devices can be joined to form a "masterslave" configuration, which provides for reliable hot-swap backup. NetScreen-500 has a throughput of 700 Mbit/sec, and can simultaneously serve 250,000 connections. A VPN functions at 250 Mbit/sec when using Triple DES encryption, and supports up to 10,000 tunnels. Consumers can choose various combinations providing up to three 10/100 Mbit/sec Ethernet or Gigabit Ethernet ports. Administrators have the possibility to implement protection of several network segments using one NetScreen-500 module owing to the creation of the Virtual System. The Virtual System is a separate domain within the device that has its own security rules and management functions. NetScreen-500 supports up to 25 Virtual System domains. All NetScreen firewalls run the specialized ScreenOS operating system.

Firebox 1000 is a small device, only a little bit larger than a portable PC, and has three 10/100 Mbit/sec network interfaces. It runs the protected Linux kernel. It can also simultaneously serve up to 1,000 users. This is a hybrid firewall with context checking and agents for HTTP, SMTP and FTP protocols. WatchGuard specialists evaluate Firebox 1000's performance as 95 Mbit/sec for the context-checking mode and 25 Mbit/sec in the agent mode. Firebox 1000 can create a virtual private network with 150 tunnels and perform TripleDES encryption at the rate of 5.2 Mbit/sec.

The drawbacks of hardware firewalls include lower scalability in comparison to their software analogues, and difficulties of providing interaction with third-party software products (such as anti-viral software, URL-filters and programs intended for filtering the contents of electronic messages and documents).

Software Implementation

As a rule, software implementations of firewalls use standard operating systems (usually Windows or Unix) that run on normal computers. Typical examples of such firewalls are Firewall-1 from Check Point, Symantec Enterprise Firewall and CyberGuard Firewall. In addition, there are software firewalls oriented towards modified operating-system versions, from which firewall developers have eliminated all functions that could be potentially dangerous for a firewall. For example, software firewalls from some companies, such as Global Technology Associates or Secure Computing, use custom operating systems.

The Firewall-1 system deserves its leading position in the current competitive market of network protection and security tools. It provides all the components required to build a security system both for small businesses and large companies, whose offices are distributed all over the country (or even the world). Security-policy management in such systems is performed from a single point, namely, from an organization's network-security center. Firewall-1's features include the following: enables the administrator to manage internetwork access efficiently; supports multiple user-authentication algorithms; provides security management on routers; performs NAT (Network Address Translation); implements auditing; supports statistics; and provides a user-friendly graphical interface. Firewall-1 supports virtual private networks (distributed under the VPN-1 trademark). This allows the organization of protected data-transmission channels for exchanging data between company affiliations, and guarantees protected remote-user access to an organization's network resources, using the Secure Remote software.

One of the most significant drawbacks of software firewall implementations is the fact that the operating systems themselves are vulnerable. On account of this, most developers of software firewalls include specialized installation procedures that build a secure installation of an OS being used before installing the firewall software.

Support of Different Operating Platforms

Users must be able to select an operating system that is the most suitable for their network. When choosing an OS, it is necessary to consider a large number of factors, including security level, background experience with the operating system, required performance and available hardware.

Most software firewalls can run under different operating systems.

For example, Firewall-1 has versions specially intended to run under a wide range of operating systems, including Windows NT/2000, AIX, HP-UX, Solaris, SunOs and Linux. Symantec Enterprise Firewall runs under Windows NT/2000 and Solaris.

Statistical data have shown that systems running under Windows NT/2000 are significantly different from Unix-based systems. Some firewall manufacturers don't limit themselves by supplying only software. Instead, besides software, they also include hardware with their firewall implementation. This approach stems from the fact that the task of customizing the Unix operating system to run firewall software on standard Intel equipment is not easy. In contrast to this situation, installing firewall software under Windows NT/2000 is much easier. As firewalls gradually turn from specialized Unix-based "fortresses" to general-purpose products, Windows NT/2000 becomes a more appropriate platform for them. The capability to operate under Windows NT/2000 is provided by many firewalls, including Firewall-1, Symantec Enterprise Firewall, CyberGuard Firewall (only under NT 4.0) and Firewall/Plus from Network-1 (only for Windows NT 4.0).

For contemporary hardware firewalls, the OS choice isn't an aspect of primary importance as it is with software firewall implementations. This is due to several reasons, including:

- Each hardware firewall is supplied with a preinstalled and preconfigured OS. Consequently, upon initial operation of such a firewall, the administrator doesn't need to install and configure a operating system.
- The administrator doesn't need to work in the environment of the firewall's operating system, since GUI administrative tools are supplied with firewall software. For example, along with Secure PIX Firewall, Cisco Systems supplies a special Cisco Secure Policy Manager tool.
- The manufacturer of the firewall uses a protected, robust and reliable OS, from which any unnecessary programs and utilities have been eliminated. For example, the Firebox 1000 firewall runs under a secure Linux kernel.

Number of Network Interfaces

To understand the working principles of different firewalls, it is necessary to become aware of how network traffic passes through them. Any firewall allows a computer network to be divided into the following two types: an internal network, where trusted users work, and a potentially dangerous external network. This approach is justified, since, for most Internet-oriented firewalls, it is often sufficient to have only two network interfaces. The external-network interface connects to the Internet Service Provider (ISP) via the router's WAN-port, while the internal-network interface is connected to the internal LAN. However, many organizations require three network interfaces: one for the Internet, another for public servers (such as a Web server, news servers and FTP servers), while the third interface is required for the internal LAN. Finally, if firewalls are used to separate an organization's internal networks, then more than three network connections might be required in order to implement the corporate security policy. Most contemporary firewalls support at least three network interfaces. Examples of such firewalls are Firewall-1, Symantec Enterprise Firewall, CyberGuard Firewall, Secure PIX Firewall 520, and NetScreen-100.

If the chosen firewall only supports two network interfaces, for example, like Firewall/Plus from Network-1, or AltaVista Firewall from Digital (Compaq), then, for adequate protection of the internal and external subnetworks of any organization, it is desirable to install two such firewalls: external and internal. Between these firewalls, there is the external or public subnetwork (the so-called demilitarized zone, or DMZ, joining an organization's public WWW, FTP and SMTP servers, as well as the terminal server with a modem pool).

Working Technology

For efficient protection of all internetworking interactions, firewalls must use all available methods: packet filtering, proxy technology, and expert-level filtering, known as context-checking or stateful inspection. Using the above-listed technologies separately does not provide the most effective level of protection.

Packet filtering based on the analysis of the service fields of IP, ICMP, UDP and TCP packets is vulnerable on account of the possibility of replacing IP addresses.

Proxy servers are vulnerable to Denial of Service attacks. So, using them without applying expert-level filtering degrades firewall performance, since analysis of the internetworking interaction at the application level requires a separate proxy for each protected network service, and, therefore, consumes more resources

Expert-level filtering, also including packet filtering, is based on a multilevel analysis of network packets. However, without using proxies, it does not provide authentication of the interacting parties, network address translation or any other important security functions, such as cryptographic transform of traffic and antiviral protection.

This is why firewalls supporting all above-listed methods provide the highest level of security for a corporate network connected to the Internet.

The following popular products are examples of complex firewalls: Firewall-1, CyberGuard Firewall, NetScreen-100/500 and Firebox 1000. They provide expert-level filtering, including packet filtering and proxy technology.

The Symantec Enterprise Firewall, formerly known as the Raptor Firewall from AXENT, provides a higher security level based on the complex implementation of packet filtering combined with proxy technology.

Some firewalls are, in principle, improved packet filters. Sometimes, they don't even have an IP address, which allows masking of the firewall from external attacks. The system of filtering rules of such firewalls ensures the analysis and filtering of network packets taking into account temporary parameters based on MAC and IP-addresses. Service-field data at the level of protocols, such as ARP, RARP, IP, ICMP, UDP and TCP, can also be considered.

Management Efficiency

Some important characteristics of firewalls are the availability of an intuitive Graphical User Interface (GUI); convenience of configuration; and capabilities of centralized management of all firewalls installed in the organization's network. When managing all firewalls centrally, all connections between the managing module and firewalls must be authenticated, and the controlling traffic must be encrypted. Most firewalls have an intuitive, user-friendly GUI, which enables administrators to manage them easily, even if the firewalls are installed in remote offices distant from the headquarters.

However, bear in mind that the availability of a satisfactory GUI does not, in itself, guarantee that the configuration will be a simple procedure. For example, Firewall/Plus provides a graphical interface for all configuration operations, but there is still no easy method for specifying the range of ports that can be accessed from both directions.

Most popular firewalls, such as Firewall-1, Symantec Enterprise Firewall, and Firebox 1000, provide convenient configuration. Even a novice network administrator would hardly make a mistake that would result in a disruption of the protective functions of these firewalls. The manufacturers have done quite a good job in improving the ease of use of their respective products. Thanks to this development, administrators can now easily implement the security policy adopted in their organizations.

Quite an important aspect when configuring firewalls is the possibility of checking the rules database, created by the administrator for automatic consistency. The role and importance of such monitoring increases for advanced network configurations with multiple components and a complex network-interaction policy. Without such a capability, administration of the firewall will inevitably result in numerous configuration errors and vulnerabilities.

Firewall-1 provides rare features of configuring a group of firewalls as a unified system for automated consistency checking of the formulated rules. Using the GUI, one can specify the rules and install them on several firewalls and filtering routers all over the corporate network. This simplifies the procedures involved in supporting corporate domains in a consistent state, thus making Firewall-1 an ideal solution for those organizations which need internal firewalls.

The Gauntlet firewall, released by TIS, is rather hard to configure. In order to change its configuration correctly, it is often necessary to edit configuration files with the text editor. Furthermore, one must know for certain which configuration files have to be edited. TIS has provided a graphical interface for the Gauntlet firewall, however, it only allows management of a limited number of files and options. In order to perform even the simplest customization, one has to tweak additional configuration files, which use rather complex syntax rules and product-specific semantic expressions.

Supported Security Functions

The number of functions supported by firewalls gradually increases with advances in networking technologies. For example, initially, Firewall-1 only included dynamic packet-filtering functions. Currently, this firewall is a network address translator, an encrypting gateway for Virtual Private Networks (VPNs), as well as an application-level proxy server for most common network services.

The Symantec Enterprise Firewall also provides quite a wide range of functions. Due to its unique hybrid architecture, the Symantec Enterprise Firewall provides multiple-level protection of the corporate network, joining the functions of application-level proxies with thorough checking and filtering of network packets. Securing the enterprise at all levels of the protocol stack, Symantec Enterprise Firewall also includes intuitive cross-platform administrative tools, a multiprocessor and multithreading support, flexible authentication methods, built-in protection against Denial of Service attacks, and special tools for improving OS-level security.

Firewalls significantly differ from one another in regard to their SMTP-translation efficiency, support for VPN and other functions. For the moment, no firewall provides sufficient intellectual support for SMTP translation. As a result, one must use a protected mail server and redirect e-mail via a firewall using a proxy server rather than an SMTP translator.

VPN support is also considered to be one of the most important firewall functions. Using a VPN, one can organize a protected connection, either within the internal network or via any public network, such as the Internet. Practically every contemporary firewall provides built-in VPN support. Most often, VPN connections are established between two firewalls in order to organize an encrypted-data exchange. Currently, however, it is highly recommended to make sure that the chosen firewall supports another new function: personal tunneling. This enables a user working in a public network (such as the Internet) to establish a secure connection to the corporate network via the firewall. The following products support personal tunneling: Firewall-1, Firebox 1000 and Aker from Aker Security Solutions. The Symantec Enterprise Firewall can create a VPN itself, as Symantec distributes another product for this purpose: Symantec Enterprise VPN. This allows the widening of the boundaries of a corporate network by allowing remote users and business partners to connect in protected mode.

Besides the above-listed functions, firewalls also differ in their range of authentication functions. To increase the security level, most network administrators prefer to use single-use passwords to access the network. The most popular authentication tools include the S/Key system from Bellcore and SecurID from Security Dynamics Technologies. In this respect, the most important function is the integration of a firewall's authentication system and the system used for authenticating all users of the corporate network. Such firewalls as Symantec Enterprise Firewall and Firewall-1 are able to authenticate the user based on user permissions and access rights in the Windows NT/2000 domain. In the Aker firewall, users are authenticated at LAN servers running Windows NT or Unix.

Gradually, some new capabilities have begun to penetrate firewall technology. One such capability is that of HTTP filtering, based on the analysis of URL and contents of transferred data. Currently, contents filtering can be used for protection from malignant Java applets and ActiveX controls. Firewall-1, Symantec Enterprise Firewall and Firebox 1000 support this function.

Notification and Auditing

Notification and auditing functions, including event logging and report generation, are of primary importance for contemporary firewalls. In contrast to event-logging functions, which are implemented in all current firewalls, notification and alerting functions that are activated when predefined events occur, as well as report generating functions, are sometimes lacking or inefficient. Just about any firewall that is unable to send administrative alerts when detecting an attack is vulnerable. When report-generation functions are deficient or inefficiently implemented, log-file analysis aimed at detecting unauthorized traffic and/or configuration errors becomes complicated and time-consuming.

The best implementations of alerts and auditing functions are found in such firewalls as Firewall-1, Symantec Enterprise Firewall, Aker, and Firebox 1000.

Firewall-1 allows the administrator to edit the logging and notification rules for any attempt to establish a connection, regardless of whether this connection is allowed or not. Data formats for logging and notification are open and can be customized by the administrator. Standard formats contain information on the following: the source, destination, service, used protocol, date and time, source port, action attempted (connection established, passed, locked, encryption keys exchanged, address translated), log file, notification type, and the firewall module that has initiated a specific event. Any information on an attempt to establish a connection can be logged and/or used for administrative notification or any other type of response (for example, such as displaying a warning message on the console, sending an e-mail message, running any user-defined program or sending an SNMP trap). Additionally, Check Point supplies a module for generating reports: the Reporting Module, which collects data from Check Point security tools distributed all over the network (Firewall-1, VPN-1 and Flood-Gate-1), summarizes this data, and generates reports on the basis of existing rules.

Symantec Enterprise Firewall provides several notification and alerting capabilities, based on event frequency. For example, it is possible to specify the following rule: "If someone requests a Telnet session more than 100 times over a period of 5 minutes, this means that network security is at risk." When the alerting function is enabled, Symantec Enterprise Firewall produces a sound alert, sends an e-mail message or notifies the administrator using another method. The Symantec Enterprise Firewall logs contain various data, such as session duration, byte counters, full URLs, user names and authentication methods used. Using this information, it is possible to generate detailed statistical and session-trend reports. The log file in ASCII format is dynamically displayed on the management console, thus providing administrators with realtime information. Log files are overwritten once every 24 hours, and can be imported into various report generators, such as Telemate. Net for creating a graphical representation of the network workload.

The Aker firewall contains an event-response subsystem. During firewall configuration, the network administrator can describe the firewall response in case a specified event (unauthorized access, address substitution, etc.) occurs. Types of reactions include recording the event in the statistics file, sending an SNMP trap or e-mail message to the administrator, displaying a warning message (during an active administrative session), or running a program to which one can pass some parameters, such as the message number, message text, and so on.

For auditing purposes, Aker provides a specialized module for analyzing the collected statistics, which enables selection of data on the basis of the most important administrative events and system messages. When selecting the subset parameters, it is possible to specify the packet type (passed through the filter, rejected, etc.), protocol type, module, messages which are of primary importance for the administrator, source and destination addresses and date range. Messages can be sorted by IP addresses or by objects, and selected information can be saved in a file. The filters defined can be saved for future use. The module also allows manipulation of the statistics file, to delete outdated records, compress statistics database and so on.

The Firebox 1000 firewall includes advanced Logging/Notification and Historical Reports modules. The Logging/Notification subsystem integrates with the most common Intrusion Detection Systems (IDS), such as IDS Snort. The Historical Reports module provides convenient functionality for creating professional-looking reports in HTML format, FireBox log file analysis, as well as the capabilities of creating data on the basis of log files and exporting this information to other systems.

Firewall Certification

When choosing a firewall, international certification, such as that provided by ICSA (International Computer Security Association), is of primary importance. The ICSA certification program checks the product for availability of a predefined set of functions according to firewall requirements. Particularly, firewall implementations presented for certification must meet the following groups of requirements:

- A list of services that must be controlled by the firewall (the required minimum includes FTP, Telnet, HTTP, SMTP, DNS, SSL/SHTTP)
- Requirements for the security level of the firewall-management channel (access to firewall management functions must be achieved only after successfully accomplishing authentication, and control traffic must be encrypted)
- Requirements for the list of attacks detected and stopped by the firewall, for example, IP-spoofing, Denial of Service

Popular firewalls, such as Firewall-1, Symantec Enterprise Firewall, Firebox 1000, and PIX Firewall have successfully undergone ICSA certification.

The complete listing of firewalls that have successfully gained ICSA certification can be found at http://www.icsa.net.

The main characteristics of the most popular firewalls are presented in the following table.

Table 2.3: Main Characteristics of Popular Firewalls

Characteristic\Tool	**Secure PIX Firewall 520 from Cisco Systems**	**Firebox 1000 from WatchGuard**	**Firewall-1 from Check Point**	**Aker from Aker Security Solutions**	**Symantec Enterprise Firewall from Symantec**
Availability of hardware	+	+	+[1]		+[2]

Table 2.3: Main Characteristics of Popular Firewalls

Characteristic\Tool	Secure PIX Firewall 520 from Cisco Systems	Firebox 1000 from WatchGuard	Firewall-1 from Check Point	Aker from Aker Security Solutions	Symantec Enterprise Firewall from Symantec
implementation					
Availability of software implementation:				+	+
For Windows NT/2000			+		+
For Solaris			+		
For Linux			+	+	
For FreeBSD				+	
Number of supported network interfaces:					
Two			+	+	+
Three or more	+	+[3],[4]			
Technologies:					
Packet filtering	+	+	+	+	+
Proxy usage		+	+	+	+
Stateful inspection	+	+	+	+	
Management efficiency:					
GUI	+[5]	+	+	+	+
Control traffic encryption	+	+	+	+	+
Centralized management by several firewalls	+[5]		+	+	+
Consistency check of the rules database	+[5]	+	+		+
Authentication:					
Administrator authentication	+	+	+	+	+
User authentication	+	+	+	+	+
Integration with the existing user-account database		+	+	+	+
IP-address translation (NAT)	+	+	+	+	+
Traffic encryption support (VPN):					
Between firewalls	+	+	+[6]	+	+[7]

Table 2.3: Main Characteristics of Popular Firewalls

Characteristic\Tool	**Secure PIX Firewall 520 from Cisco Systems**	**Firebox 1000 from WatchGuard**	**Firewall-1 from Check Point**	**Aker from Aker Security Solutions**	**Symantec Enterprise Firewall from Symantec**
Between remote client and firewall	+	+	+[6]	+	+[7]
HTTP-filtering, including mobile code locking	+	+	+	+	
Protection against typical attacks (IP-spoofing, Denial of Service, etc.)	+	+	+	+	
Port scanning detection		+	+	+	
Auditing and notification					
Event logging	+	+	+	+	+
Alerting, including via e-mail	+	+	+	+	+
Report generation		+	+[9]	+	+
Certificates					
ICSA	+	+	+		+

[1] Hardware implementation of the Symantec Enterprise Firewall, supporting VPN, is known as Symantec VelociRaptor.

[2] Hardware implementation of Firewall-1, supporting VPN, is known as VPN-1 Appliance.

[3] The Firebox 1000 firewall only supports three network interfaces.

[4] For consistent centralized-firewall management over the Secure PIX Firewall 520 systems using GUI, the Cisco Secure Policy Manager product is required.

[5] To provide VPN support by Firewall-1, VPN-1, combining FireWall-1 and VPN building modules is required.

[6] To provide VPN support by Symantec Enterprise Firewall, Symantec Enterprise VPN is required.

[7] To generate reports based on statistics gathered by Firewall-1, Check Point supplies an additional Reporting Module.

2.6.2. Firewall-1

General Information

Currently, Firewall-1 is one of the leading and most popular firewalls present in this segment of the IS market.

The set of products known as Check Point Open Platform for Secure Enterprise Connectivity (OPSEC) is based on the concept of joining information-security technologies around a common complex security policy. Such an approach allows the implementation of a closer integration of products from other vendors on the basis of FireWall-1. This provides the features of centralized tracking of these systems' activities, along with management and configuration functionality.

FireWall-1 allows the organization to create a unified, integrated security policy, which would be applicable to a whole set of firewalls and managed from any location within an enterprise's network. This product has quite a wide range of add-on capabilities, such as management of access-control lists at the hardware routers, balancing network workload on the servers, and building highly secure systems completely integrated with the global security policy. FireWall-1's operation is seamless, and provides an extremely high level of performance for practically any IP protocol and high-speed data transmission.

Based on packet-inspection technology as related to the protocol state, which is currently the most advanced method of controlling network traffic, FireWall-1 provides the highest possible security level. This method allows information to be collected from data packets both at the communication and application levels. This is achieved by information saving and accumulation in specialized context tables that are dynamically updated. This approach provides full control over applications' activities, without needing to introduce a special proxy for each protected network service. Thus, the user will benefit from performance improvement; be able quickly to scale the system; speedily and reliably secure new applications and protocols; and not need to develop proxies.

FireWall-1 is supplied with built-in support for hundreds of standard network services, protocols and applications. In addition to the standard protocols and services, FireWall-1 allows the creation of custom protocol handlers using the built-in, high-level INSPECT programming language. The INSPECT virtual machine creates the basis of FireWall-1's technology.

FireWall-1 uses the distributed client-server architecture, which provides unique system-scaling capabilities, as well as centralized management functionality. Support of various operating systems, including: Windows 9*x*, Windows NT/2000, Unix routers, switches, and remote-access devices (via OPSEC partners of the Check Point company) and cross-platform interaction provides high flexibility and convenience of system deployment.

Check Point's world leadership in the field of development of integrated network-security products allows customers to build complex solutions for the problems of improving system security, based on Check Point products combined with third party products supporting open standards of the OPSEC platform. Currently, Check Point puts out several products lines, providing solutions for several aspects of network protection and network management:

- FireWall-1-a complex of modules, composing the core of any security system based on Check Point products. Besides firewall functions based on the registered Stateful Inspection technology, it supports user authentication, network-address translation, content-based access control and auditing. Includes a centralized policy-based management system, which is able to control the operation of FireWall-1 modules as well that of other products, such as VPN-1 and FloodGate-1.
- VPN-1-set of products intended for organizing virtual private networks, both between LANs and when connecting remote users to a protected LAN. This product includes FireWall-1.
- Account Management Module-a module integrated into the LDAP (Lightweight Directory Access Protocol) infrastructure. It allows control and management of users' security profiles.
- MetaIP-a system for managing an organization's IP-address infrastructure. Integrates DHCP, DNS and authentication services, supplementing them with its custom UAM service for mapping user names to their IP-addresses. The UAM service can be used by FireWall-1, thus providing user-level access control in a dynamic DHCP environment.
- FloodGate-Quality of Service (QoS) management tools. These tools provide flexible distribution of the bandwidth for different classes of traffic as well as for specific connections. Supports centralized management based on the rules integrated with FireWall-1/VPN-1 rules.
- ConnectControl-a system for managing network workload on servers.
- High Availability Module-a subsystem for building clusters from two or more firewalls, "hot-swapping" when one of them fails.
- Reporting Module-a powerful report-generating system for processing statistical data for VPN-1/FireWall-1.

Basic Packet-Filtering Technology

The working technology of any application gateway as a part of a firewall system is based on the usage of proxies that check authenticity of clients trying to access those gateways, establishing the required connections and performing other operations for securing internetwork connections. Proxies act as an intermediate tier for transmitting data between clients and servers. First, it is necessary to establish a connection to the proxy, then the proxy makes the decision as to whether a connection to the recipient should be established or not. Consequently, the application gateway duplicates any allowed connection as long as it is in operation. Consequently, its operation isn't seamless for the user, and involves additional overheads for connection processing.

To eliminate this drawback, Check Point has developed a new technology of packet filtering, also known as expert-level filtering or stateful inspection. When using this technology, filtering is performed on the basis of special methods known as Stateful Multi-Layer Technique (SMLT). This hybrid technology traces the network-connection status by intercepting packets at the network level and retrieving application-level information that will later be used to control the connection. A quick comparison of the incoming packets to the known state of "friendly" packets significantly reduces the processing time as compared to application-level firewalls.

The licensed implementation of Stateful Inspection technology used in FireWall-1 provides the maximum possible level of control and security. FireWall-1 controls connections from levels 3 to 7 of the OSI network model, while proxy servers can only control levels 5 to 7. Thus, FireWall-1 has unique information on the contents of network packets, connections and applications. This accumulated data on the state of the connection, application context, network topology and policy rules are used to ensure that the security policy of an organization is all encompassing. Additional protection is also provided for computers running FireWall-1, since this software intercepts and analyzes all network connections, then takes all required actions, and only then informs the operating system. It then passes the packets to the gateway OS, thus preventing unauthorized access to the operating system.

In the implementation of stateful-inspection technology, Check Point uses dynamical tables for storing information on the connection context, for active connections as well as for closed ones. The contents of these tables are checked when processing the connection attempt. This approach guarantees the best performance and makes certain that the connection will be processed with the account of the latest information on the state of communications. State tables are located within the OS kernel, and can not be corrupted or overwritten, for example, as files stored on the hard disk. In cases of rebooting, the FireWall-1 system starts to create new tables, which prevent operations over corrupted data. Clearing the table is equivalent to locking all connections completely, thus guranteeing network security in such cases.

The stateful-inspection architecture of FireWall-1 uses a unique INSPECT mechanism, which ensures the security policy at the gateway where it runs. INSPECT views the packets and extracts all required information at all levels of the network model. This approach makes high quality performance and efficiency possible, allows for the implemetation of support for a wide variety of protocols and applications, and easily supports newer applications and services.

The INSPECT virtual machine is programmable using a rather flexible internal language. This ensure the rather important characteristic of extensibility, allowing software developers to create built-in applications, services and protocols without installing new software. Support of the application network model, including support for custom user-defined applications, can be built into INSPECT simply by means of modifying one of the standard FireWall-1 templates via the GUI.

To guarantee network security efficiently, FireWall-1 manages the entire flow of data passing through it, and tracks its own state. In order to help in decesion-making, it collects, records retrieves and processes information from all communication levels, from other applications, and from the logged connection history.

Besides packet filtering in stateful-inspection mode, Firewall-1 version 4.0 also provides the following functional capabilities:

- *User and session authentication.* Users do not need to open a separate session to connect to the firewall in order to get authenticated. Instead of this, FireWall-1 intercepts FTP, HTTP, Telnet and Rlogin sessions passing through it, and redirects them to respective proxies. Session authentication can be used with precision for any service up to a specific session.
- *Contents checking.* This capability provides the network administrator with excellent tools for managing Web, Mail, and FTP connections, including content scanning for viruses, file checking and access management for specific resources (such as URL's files and so on) and SMTP commands.
- *Network-address translation.* Windows and X/Motif GUI users can now assign network-address translation rules using common rules database view. Translation rules can be automatically generated for any platform.
- *Cryptographic protection of the message flow.* FireWall-1/VPN-1 provides secure, bidirectional connections via the Internet, based on cryptographic protection and digitally signing packets. Data-exchange security is maintained both for LANs and when remotely accessing a protected LAN.
- *Active network management.* Fully integrated with network security, this manages network configuration in real-time mode, including auditing and monitoring current connections, workload balancing and export of log files to databases.
- *Synchronization and workload balancing.* Different firewall modules running on different computers can be share information about the state of network connections and make updates by synchronizing each other's data. Additionally, synchronized FireWall-1 modules can replace one another in case of failure. Workload balancing allows any number of servers to distribute the workload.

Firewall-1 Architecture

Firewall-1 implements a three-level architecture and comprises the following components (Fig. 2.22):

- Firewall Module-which implements all functions related to access control, event logging, alerting, etc.
- Management Server-a module which manages all other components connected to it. The Management Server can control both the Firewall Module and all other components comprising the Check Point solution (for example, VPN Module, FloodGate Module and so on).
- Management Console (GUI)-which implements the GUI, simplifying management of all modules connected to it.

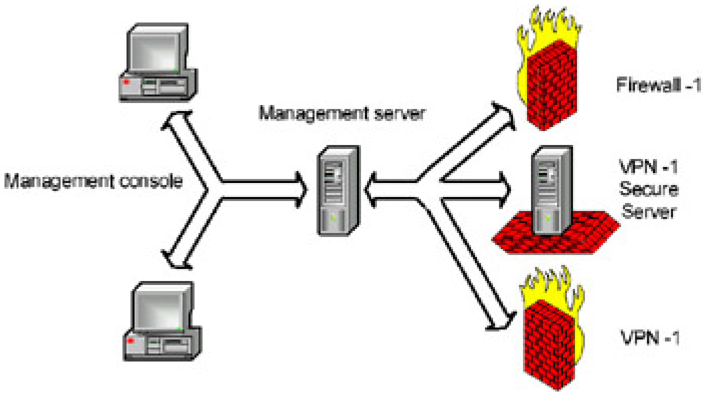
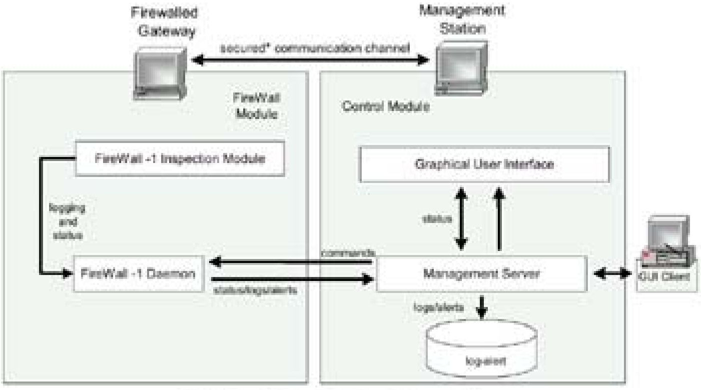


Fig. 2.22: Firewall-1 architecture

All above-listed components can be installed on the same computer.

FireWall-1 includes an inspection module, proxies for application services, and special tools for interaction with the management module. FireWall-1 directly implements a protection strategy, event logging, and communicates with the management module using communication daemons (Fig. 2.23).



* This channel is authenticated, and also encrypted if the FireWall-1 Encryption feature is installed.

Fig. 2.23: Scheme of interaction between FireWall-1 components

FireWall-1's security strategy is defined in terms of network objects, services, users and rules specifying interactions between these objects. As long as these objects and rules are defined, the management module generates an inspection code, which is automatically installed on the firewall implementing the security strategy.

The management module can also be installed in the client-server configuration.

The graphical interface on the client can run under Windows 9*x*, Windows NT/2000 or X/Motif GUI for Unix and manage a server running on one of the supported platforms (Windows NT/2000, SunOS, Solaris, HP-UX or Linux). The client component interacts with the user by means of a graphical user interface. All data (including the rules database and configuration files) are stored and processed on the server, which takes an intermediate position between the client part and management module. For example, when a user requests installation of the Security Policy, the server addresses the management module, which translates the request further and returns the results to the requesting server. The server, in turn, passes this data to the client. Additionally, the user interface based on the OpenLook GUI is available. It comprises both the client and server, and can be used on any platform supporting OpenLook (SunOS, Solaris, HP-UX, and Linux).

Depending on the information-processing technology adopted in an organization, one can implement one of two possible schemes for using Firewall-1/ VPN-1: local or distributed.

The first scheme is intended for instances in which an organization (typically, a small one) has only one point of connection to public networks, and that point is protected by use of a firewall. In this case, it makes sense to use Firewall Internet Gateway (or VPN-1 Internet Gateway), which includes the Control Module and Management Server (or VPN), and is installed on the computer designated for this purpose. The management console (GUI) is installed in a location convenient for the system administrator. In this case, the management server only controls one Firewall (or VPN) Module.

The second scheme is implemented in far-reaching organizations, with wide geographical ties. It assumes distributed installation of the Firewall (or VPN) Module and Management Server. Modules responsible for controlling access to corporate resources and VPN-building are installed at predetermined locations and managed centrally from a single management server. As in the previous case, the management console (or multiple consoles) can be installed anywhere. This variant of installation can be described by the following scheme: *n* × Firewall (VPN) Module + Enterprise Console (where *n* is the number of locations where the Check Point components are installed).

The FireWall-1 Inspection Module

The FireWall-1 inspection module is dynamically loaded into the operating system kernel between the channel and network layers of the OSI model (Fig. 2.24). It intercepts all packet traffic in order to check if the packets satisfy the predefined conditions. When the first packet of a new connection arrives, the FireWall-1 module checks the rules database to determine if this connection should be allowed. As soon as a connection is established, FireWall-1 adds it to the internal table of connections. For the sake of efficiency, subsequent packets of the established connection are checked against the connection table rather than against the rules database. Transmission of the packet is allowed via the network only when the connection is present in the connection table.

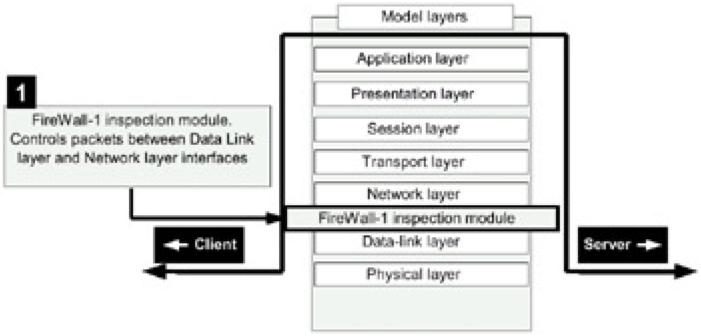


Fig. 2.24: The scheme of inspection-module operation

When filtering data, the FireWall-1 inspection module retrieves and analyzes data obtained from all communication layers. This information on the "state" and "context" is saved and dynamically updated, thus providing virtual-session information for tracing connectionless protocols (such as RPC, and UDP-based applications). Data collected from connection and application-status information, network configuration and security rules, is then used in the decision-making process. Any traffic that is not explicitly allowed by the security rules is by default locked. At the same time, the system generates alerts in real-time mode, thus providing the network administrator with complete information on the network state.

The Check Point approach is rather different from those adopted by most other firewalls. Since the Firewall-1 inspection module resides directly in the OS kernel and intercepts packets one level below the level where all software runs, there is no need to lock IP-routing, and it isn't necessary to suppress processes and daemons running at the firewall host.

The FireWall-1 security policy represents the rules and properties database. The rules database is an ordered set of rules that lets one check any connection. If the connection source, destination and service type satisfy the rule, the system will process that connection according to the prescribed rule. If the connection doesn't correspond to any rule, it will be locked according to the following principle: "Anything that isn't explicitly allowed is prohibited."

As soon as the network administrator defines the security strategy, the rules database and object parameters (networks, services, host addresses and user addresses), used in the rules, the strategy will be transformed into an inspection scenario. An inspection code generated from the inspection scenario is then passed via a secure channel from FireWall-1's management station supporting the database to the computer running FireWall-1 daemons. This means that the data is transferred to the FireWall-1 modules that ensure the network security strategy. The FireWall-1 daemon will load the inspection code into its inspection module. The network object where the FireWall-1 inspection module is installed is known as the "FireWalled system" (or protected gateway).

A system with FireWall-1 will run its related components of the inspection code, but all registration information and alerts will be sent to a network object, designated as the master. The master stores the whole inspection code for all FireWalled systems managed by it. If a FireWalled system loses the inspection code for any reason, it can restore its status by retrieving the required information from the master. In practice, the master and control station always run on the same computer. Moreover, one can always assign a backup master, which takes control if the primary master fails. Communications between inspection modules and the control station are performed in secure mode. Inspection modules communicate with the control station using an SNMP agent.

The architecture of the inspection module is shown in Fig. 2.25.

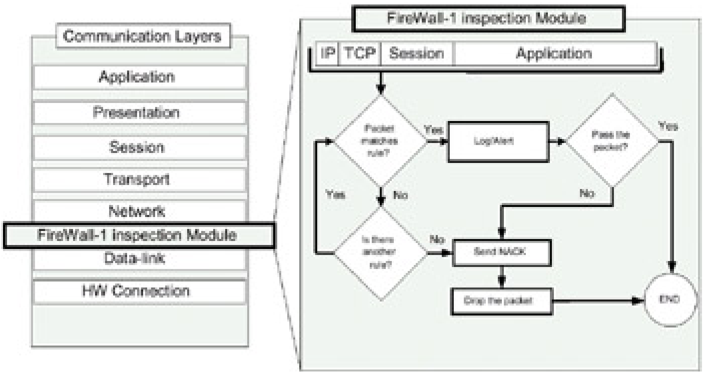


Fig. 2.25: Architecture of the inspection module

As was already mentioned, the inspection module dynamically loads into the OS kernel between the data-link and network layers (Layers 2 and 3 of the OSI model). Since the data-link layer is actually implemented by the network interface card (NIC), while the network layer, by the first layer of the protocol stack (for example, of TCP/IP), the inspection module resides at the lowest program level.

Since the inspection module has access to "raw messages", it can view all inforamtion contained in messages, including data related to higher protocol levels, as well as that, which is contained within the packet. The inspeciton module investigates the IP header, port numbers and other required infomation, in order to determine if the packets must be transmitted according to the inspection code from the rules database.

FireWall-1 can interpret internal structures of the IP protocol family and applications based on its higher levels. For stateless protocols, such as UDP or RPC, FireWall-1 retrieves and saves UDP (User Datagram Protocol) and RPC (Remote Procedure Call) data. It is able to retrive data from packets and store this information if the application doesn't provide this capability. Furthermore, FireWall-1 can dynamically enable or disable connections as necessary. The above-described dynamic features have been developed in order to provide the highest level of security for complex protocols. However, they can be disabled if necessary.

FireWall-1's ability to work with data that is encapsulated within a packet allows one to enable or disable specific commands within an application. For example, FireWall-1 will allow the ping ICMP command while disabling redirection, or enable the get SNMP function while disabling the put function, and so on. FireWall-1 can save and retrieve table data (in a dynamic context) and perform arithmetical and logical operations on the data contained in any part of the packet. Additionally, the user can create custom expressions and tables.

Packets that are not explicitly allowed by the security strategy are discarded according to the principle that "everything that isn't explicitly allowed is prohibited". While defining the rules, one can specify the host, interface and traffic direction to which a specific rule is applicable. Packets can be checked in any of the following directions:

- Eitherbound-Packets are inspected at the interface, both incoming and outgoing.

- Inbound-Packets are inspected when they enter the firewall system.
- Outbound-Packets are inspected when they leave the firewall system.

Usually, incoming packets are inspected at most gateways, although it is possible to check outgoing packets as well.

Controlling Internetwork Access

Access control protects the organization's network by means of specifying and controlling information flow and communications passing through the perimeter gateway of the network. The key feature of access-control devices is the comprehensive nature of the information on all communications, network services and applications. The first implementations of the packets based on packet filtering (usually implemented on routers) have no data on the application state, and can process neither UDP traffic nor dynamic protocol traffic. Network-security tools of the second generation, based on proxy technology, often require quite a powerful computer and tend to adapt to new Internet services (which appear regularly) rather slowly. In contrast to this, the stateful-inspection technology implemented in FireWall-1 provides the gateway with complete information on communications. This, combined with the object-oriented approach to describing network resources and services, allows the system to adapt to a newer Internet service quickly and easily. FireWall-1 provides broad capabilities for controlling access to more than 160 predefined Internet services, and has special tools for convenient certification of new services. In addition to the above-listed functionality, with FireWall-1, access to network resources can be controlled through a strict time accounting. Detailed information on access to network resources is provided by means of rules, created to allow access network resources at specified time periods (minutes, hours, days of week, month, year can be stipulated). For example, an organization can decide to disable Internet access during business hours and enable it for all other time intervals. Another example is to deny access to a business' critical servers during full backup procedures.

Specifying access control rules in FireWall-1 is an understandable and straightforward process. All aspects of the information security policy can be specified using the FireWall-1 GUI, which has received many awards. An object-oriented approach is actively used when defining network elements. Once created, the object is then used to define the security policy using the policy editor. Each rule can operate over any combination of network objects and services, and contains within it a definition of the actions to be taken and notification methods to be used when the rule is activated. Additionally, one can specify the hosts for which this rule is applicable. By default, the rules work on all gateways with FireWall-1 installed. Various platforms are supported, including Unix and Windows NT/2000, as well as various internetwork communication equipment of the Check Point OPSEC partners. One unique advantage of FireWall-1 is its ability to create a unified security policy for an entire business. After the security policy has been created, FireWall-1 checks it for consistency, and compiles and distributes it to all hosts controlling network traffic.

Administrative functions of FireWall-1 are also oriented towards multi-user access, and can provide an organization with the capability of delimiting the functions of security administrators. After authorization, the FireWall-1 administrator inherits the rights and permissions that were established for this firewall by the security administrator. The policy editor can specify these rules. This means several FireWall-1 systems can be simultaneusly administered from a single location. FireWall-1 supports different levels of administrative access:

- *Read/Write*: full access to full functionality of administrative tools.
- *User Edit*: is able to modify user accounts. All other functions are limited by read-only access.
- *Read Only*: access to all functions is allowed in the read-only mode.
- *Monitor Only*: read-only access to statistical data.

The distributed architecture of the FireWall-1/VPN-1's security system and its centralized management make it the best choice for organizing access control and building secure VPN channels in several locations of a corporate network.

In access-control locations in a central enterprise network, and in remote networks of its affiliations, it is possible to install FireWall-1/VPN-1 gateway modules, protecting all hosts within an organization's subnetworks. Gateways can be installed in the form of software modules on the standard platforms (Windows NT/2000, Solaris, HP-UX, IBM AIX and Linux), or in the form of specialized software and a hardware complex.

VPN-1 SecureClient products are intended for remote computers' protection. These products, as well as being able to establish a sucure channel to the respective gateway, perform the same fully functional access control as FireWall-1. This functionality is especially useful if one constantly makes remote connections to the Internet (for example, when using xDSL or cable moderms), during which a client computer might be exposed to hacker attacks over a long period of time. Using VPN-1 Secure-Server, one can provide similar protection for a standalone application server running within an organization's network. This is especially useful for improving a security policy's flexibility, or the security of a standalone server.

The number of locations in which traffic control is performed can also be increased by means of installing an inspection module on network switches and routers in order to perform stateful inspection. Currently, the list of models supporting the inspection module includes Nortel Networks routers, Xylan and Nortel Countivity switches. In the future, this list will be enhanced in order to support other models.

All FireWall-1/VPN-1 modules are managed centrally, and installed in the central network of an enterprose in remote affiliations and on remote computers. The management server stores the enterprise-wide set of access and security rules, which is loaded onto all FireWall-1VPN-1 modules, located in any segment of a corporate network. Policy rules can be created and edited remotely by several security administrators with discretionary access and perminssion rights.

When necessary, all of an enterprise's FireWall-1/VPN-1 modules can synchronize their work, which allows session support under conditions of dynamic routing.

Thus, a security system based on FireWall-1/VPN-1 can coordinate and synchronize the operation of a whole set of firewalls, VPN gateways, personal firewalls and VPN-clients required for reliable and flexible protection of an organization's information resources.

User Authentication

FireWall-1/VPN-1's security tools support various user-authentication schemes that can be activated when specifying respective access rules. At the moment, these schemes include authentication on the basis of passwords stored in FireWall-1, X.509 digital certificates, OS authentication, RADIUS, TACACS, SecurID and more. Using the account-management module included by FireWall-1/VPN-1, the access rules can incorporate information on users and groups stored in the external user-account databases that support the LDAP protocol (such as NDS or Active Directory). This relieves the security administrator of having to duplicate user data in the FireWall-1/VPN-1 system.

Support for digital certificates and Public Key Infrastructure (PKI) in Check Point products provides a scalable solution of the problem of user-authentication. FireWall-1/VPN-1 products currently support PKI systems of the following companies: Entrust, VeriSign, Netscape, Microsoft, Baltimore Technologies and Data Key. This support makes it possible to authenticate users in a heterogeneous environment when certificates are issued and signed by different organizations. Using the VPN-1 Certificate Manager system, which includes the Entrust Technologies certificate server and LDAP-compatible directory service from Netscape Communications, an enterprise can support its own public key infrastructure, and administer it using the standard account management GUI tool.

FireWall-1 lets administrators determine the security policy for each user, in order to authenticate each user in addition to checking each source, destination and service. Furthermore, connections can be allowed or denied on the basis of transmitted contents. For example, e-mail related to a specific source or destination address may be allowed, denied or forwarded. What's more, one can deny access to specific URLs and enable scanning of transmitted files for viruses. User and destination host authentication, as well as Firewall-1 proxies for application services ensure checking of packets' contents. These proxies run at the application level of the OSI model.

FireWall-1 provides the following three types of authentication:

- *User authentication*-allows the administrator to assign each user specific combinations of access permissions. User authentication is implemented for HTTP, FTP, Telnet and Rlogin protocols.
- *Client authentication*-provides functions for authenticating any application, both standard or custom.
- *Transparent session authentication*-enables transparent authentication of any session. This provides integration capabilities for any application.

With user authentication, administrators can specify any specific combination of access permissions for a particular user in a multi-user environment, for FTP, Telnet, HTTP and Rlogin protocols, independent of the client computer's IP-address. For example, if the user must work with the organization's servers remotely, the security administrator can allow access to an internal LAN without propagating the user's privileges to other users of the computer. FireWall-1 authenticates the user via a special security server that runs on the gateway. This firewall also intercepts all attempts to authorize a user on the server, and redirects them to the appropriate security server. After user authenticity has been confirmed, the FireWall-1 security server opens a second connection to the required application server. All subsequent packets of the session are also intercepted and inspected by FireWall-1 on the gateway.

With client authentication, the administrator can grant access privileges to specific IP addresses, whose users must undergo appropriate authentication procedures. In contrast to user authentication, client authentication is not only limited to specific services, and can provide authentication for any application, both standard and custom. Client authentication of the FireWall-1 system is not transparent for users, but, on the other hand, it doesn't require installation of any add-on software or modification of the existing one. For this type of authentication, the administrator can specify how each user must be authenticated, along with the following: which server will be available; when it will be available; for how long; at what days and hours, and how many sessions can be opened.

The Transparent Session Authentication mechanism can be used for any services. Authentication will take place for each session. After the user initiates a direct connection with the server, the gateway with FireWall-1 software installed on it detects that the client needs authentication, and initiates a connection with the Session Authentication Agent. The agent performs the required authentication, after which FireWall-1 allows connection upon successful authentication.

For each user, FireWall-1 supports the following authentication schemes:

- S/Key-the user must enter a single-usage password for current iteration by the S/Key protocol.
- SecurID-the user must enter the number shown in the SecurID field of the Security Dynamics card.
- By password-the user must enter a password to the operating system.
- Internal-the user must enter the internal password of the FireWall-1 system.
- RADIUS-the user must enter an answer to a request from the RADIUS server.
- Assurenet Pathways-the user must enter an answer to a request from the AssureNet Pathway server.

Additionally, the LDAP Account Management Module subsystem allows various information about users to be saved on the LDAP server, significantly simplifying centralized management of an enterprise's information-security system. The LDAP server is the single source of consistent information on user security parameters (Fig. 2.26). These parameters might include:

- Identification characteristics, such as fully qualified user name, an identifier, e-mail address, and so on
- Authentication characteristics, such as password, authentication scheme and server, user group
- Access control parameters, such as authorized source and destination addresses

- Temporary parameters, such as days and time of day when user access is allowed
- Cryptographic parameters, such as key distribution scheme, cryptographic algorithms used for encryption and decryption and integrity control

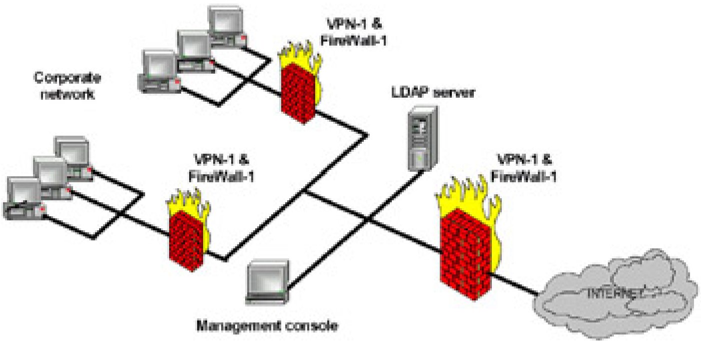


Fig. 2.26: The scheme of LDAP Account Management Module usage

Firewall-1/VPN-1's components use specified information for delimiting access of external and internal users to protected corporate resources. When requesting access to a protected resource, Firewall-1 (or VPN-1) requests the access rights of a specified user from the LDAP server, and provides access on the basis of these rights. All interaction between the LDAP server and the requesting component is performed via the SSL protocol.

Protected Management of the IP-Address Infrastructure

FireWall-1 systems support address translation using NAT algorithms adopted on the Internet by encapsulating IP addresses within packets sent via the Internet. The NAT service provided by FireWall-1 modules supports two translation modes: static and dynamic. When using static translation, packet addresses are replaced according to rules defined by the graphical-policy editor of FireWall-1. An administrator can create an address-translation table based on source and destination addresses, and the service (specified, for example, by the number of TCP/UDP port or by any other method accepted for creating FireWall-1 objects). During dynamic address translation, internal addresses are automatically replaced by a single address globally valid for the Internet. NAT functions allow internal network addresses to be hidden and/or use private addresses as internal ones which are not supported by Internet routing. In most cases, this reliably protects the corporate network from external attacks.

The FireWall-1/VPN-1 products are able to hide internal network addresses by encapsulating original packets from internal network hosts into a new packet, whose address corresponds to the address of the external interface of the FireWall-1/VPN-1 gateway. Since addresses of internal hosts are reliably protected thanks to VPN technology, which also guarantees authenticity and integrity of each transmitted packet, this method of packet exchange significantly reduces the likelihood that internal network hosts will be attacked by their IP-addresses.

Meta IP is a complex system for managing IP addresses, and is closely integrated with FireWall-1. Besides the above-described UAM service, mapping user names to their IP addresses, and serving as a basis for efficient operation of FireWall-1 at the user level, the Meta IP system also performs some rather useful functions in large corporate networks.

Meta IP can create a fault-tolerant system for managing IP addresses and DNS names. Such a system will continue to run efficiently in case some DHCP or DNS servers fail. Furthermore, such systems support dynamic record updates in DNS zones when using DHCP for distributing IP addresses.

The Meta IP system includes:

- DNS server implementation for the Windows NT/2000 environment based on the latest version of BIND 8.1.2 and complemented by an interface to the Microsoft WINS service. This implementation supports the most complete version of the Dynamic DNS standard for the Windows NT environment.
- Commercial implementation of a DHCP server by the Internet Software Consortium for the Windows NT environment, complemented by the registration messages service for Dynamic DNS, as well as by the capability of automatic recovery in case one of the DHCP servers fails. Meta IP doesn't contain the flaws of the DHCP standard, which lacks measures of protection when DHCP servers fail. Instead of a traditional, inefficient solution of this problem at the expense of splitting the pool of leased IP addresses, DHCP servers are classified into primary and backup servers in the Meta IP system. The backup DHCP server constantly tracks the process of leasing IP addresses, performed by the primary DHCP server. If the primary server fails, the backup servers take its functions without disrupting the logic of DHCP service operation.
- The UAM service, tightly integrated with FireWall-1, and UAT agents, running in authentication services of the operating systems.
- A centralized management system providing protected configuration and management for any DNS, DHCP or UAM service within the corporate networks. In each geographically distinct segment of a corporate network, a management system uses a separate management service. This service searches an LDAP-compatible database for all required information on the DNS, DHCP and UAM servers managed within this part of the corporate network, and on the Meta IP administrators and their access rights to the managed servers. The management service ensures data replication between all LDAP databases of the corporate network, so that all management is performed on the basis of centralized information, and the single failure of a specific LDAP server does not result in data loss. Meta IP includes implementation of an LDAP-compatible database, but can also work with any standard directory service supporting LDAP, for example, Microsoft Active Directory, Novell NDS and Netscape Directory Server.

All communications between a management service and managed services are performed via protected channels. Administrators control the IP system using a graphical console, which has two implementations: one for the Win32 environment and the other in the form of a Java applet, capable of working in any system that supports the Java browser. Using Meta IP, it is also possible to manage standard DNS servers.

Fault tolerance, integration of DNS and DHCP services in the Windows NT environment, as well as centralized and coordinated management of multiple DNS and DHCP running on any platform, make the Meta IP system a required component of the infrastructure of any corporate-level network.

Checking of Information Flow

The wide capabilities of checking information flow provided by FireWall-1 enhance high-level inspection functions in their work toward the highest level of information security. This allows users to be protected from various risks, including computer viruses and malignant Java and ActiveX applets, which can be achieved by a precise setting of an Internet access-control mechanism. A mechanism of checking information flow is fully integrated with FireWall-1's capabilities, which is achieved on account of centralized management via a centralized common graphical interface. Besides built-in functions of information checking, FireWall-1 provides an open API for connecting OPSEC-compatible third-party applications checking the information flow. Thanks to Check Point's initiative known as OPSEC, organizations are free to choose applications that best satisfy their requirements. Check Point certifies such applications, which guarantees their full compatibility with FireWall-1.

Virus Detection

Timely detection of computer viruses is vitally important for the information security of any enterprise. The fight against viruses can't be won merely by scanning users' personal data on servers and file systems of their computers. The most reliable protection against computer viruses can be ensured only, when incoming data flow is scanned for viruses at corporate networks' entry points. Mechanisms for checking the information flow's contents, as implemented in FireWall-1, represent an integrated anti-viral solution, and provide specialized anti-viral tools developed by Check Point's OPSEC partners. These applications can be deployed both directly on the computer with the installed FireWall-1 product, and on a stand-alone computer, specially devoted to anti-virus software. Due to this approach, a security administrator can easily implement an optimal scheme of anti-virus protection, quickly deploy it and administer the whole complex from a common center. Instead of customizing two independent software products, a security administrator defines the rules for access control and rules for checking information flow by means of a security-policy editor (FireWall-1 Security Policy Editor).

For example, if an organization needs to ensure scanning of the whole flow of email for viruses and malignant content, this can easily be achieved by checking incoming mail at the FireWall-1 gateway. In this case, FireWall-1 will intercept all data flow corresponding to the specified security policy rules, and, using the Content Vectoring Protocol (CVP), redirect them to the antiviral scan server. Before returning the incoming data, the server will perform all required actions to scan the mail and mail attachments for viruses and clean the infected data. Having received the returned data, FireWall-1 will redirect this data to recipients. Thus, no connection will be established without an appropriate check.

URL Scanning

URL scanning enables a company to regulate channel throughout and to avoid wasting time during the workday by limiting employees' visits to undesirable Web pages. This allows the security administrator to create a flexible policy for using Internet resources by only allowing users access to specific Web pages during designated time intervals. Additionally, this mechanism can be used to accumulate statistics of accessing specific resources, which can be useful when preparing various analytical reports.

FireWall-1 provides several methods of defining mechanisms of comparing requested URLs:

- Template-based descriptions
- Templates contained in the external file
- External databases and comparison tools

Each of the above-listed methods is designed to provide the administrator with a flexible customization mechanism. The most advanced management capabilities are achieved when using third-party tools using external databases for storing templates. Usually, such manufacturers provide users with the possibility of subscribing to the updated versions of their databases. Comparison tools that best satisfy an organization's requirements can be found on a list of certified OPSEC products.

Java and ActiveX Locking

FireWall-1 capabilities for scanning and analyzing data flow allow efficient system protection against various attacks related to the usage of malignant Java and ActiveX applets. The security administrator can control the transmission of Java and ActiveX code according to predefined conditions, such as the network addresses of the client and server computers, and requested URL or registered user name.

FireWall-1 can perform the following actions over the detected Java and ActiveX code:

- Delete Java applets encountered within the text of an HTML page
- Delete Java applets from all data flow between the server and client, even if the information is archived or compressed
- Lock Java attacks by means of denying suspicious outbound connections
- Delete ActiveX applets encountered within the text of an HTML page
- Delete JavaScript code encountered within the text of an HTML page

Support for the SMTP Protocol

The SMTP protocol was initially designed to provide the most flexible capabilities for user interaction. At that time, it was supposed that user accessed the Internet from different geographical regions. Later this protocol was enhanced with its ability to transmit different information as e-mail attachments. As a result, it became rather difficult to provide transparency of the mail connections and, at the same time, protect the internal network from intruders.

FireWall-1 successfully protects corporate networks by means of providing detailed control over SMTP connections. The following functional features of FireWall-1 deserve special mention:

- Encapsulation of the From: address in outgoing mail by means of replacing it with some general address. This allows the internal network structure and real internal users to be completely hidden.
- Redirection of the mail sent to a specific user (for example, root).
- Discarding the mail sent to specific recipients.
- The ability to delete attachments of a specific type (for example, executable files).
- Deletion of the Received fields in outgoing mail, which prevents distribution of information on mail routing within the organization.
- Discarding mail messages that exceed a predefined size.
- Scanning mail for viruses.

Additionally, FireWall-1 supports only the required subset of the SMTP, which implicitly helps to strengthen security, since nontrivial commands, which can be used for malign purposes, are not supported.

HTTP Filtering

Resources addressed via URI determine: the access method, such as GET, POST and so on; the server where the resource is located; the path to that resource on the server; and, possibly, the specific request to that resource. All above-mentioned methods of processing information flow can be applied to such resources whose descriptions were created using wildcard characters. These descriptions can also be stored in an external file.

Processing of the FTP Protocol

The FTP security server is able to authenticate users and monitor the information being exchanged via this protocol. Management and control is performed both at the level of FTP commands (PUT/GET), by means of including limitations for possible file names, and by redirecting data flow to external servers for virus scanning.

Managing Access Control Lists on Routers, Accumulating Statistics and Generating Alerts

Routers are an integral part of an enterprise's network and usually serve in network segmentation. Quite often, the router must be located at the network perimeter. In such instances, routers can be used as "preliminary filters" to stop undesirable traffic from penetrating the network. Actually, in this configuration, routers are the first line of defense, since they perform the elementary function of packet filtering.

Router capabilities are constantly improving. However, their control interfaces are usually based on Telnet-character terminal interface. Thus, the procedures of creating router filters are complicated, labor consuming and error-prone. Since router commands usually represent combinations of keywords and IP addresses, specialists creating configurations and filters must be highly qualified. Furthermore, each router must be controlled separately, which means that centralized representation on the network's fitering features is not provided, and configuration modifications are complicated.

FireWall-1 provides a rather good solution for managing security on routers. Using the FireWall-1 GUI, administrators can create filters for 3Com, Bay Networks and Cisco routers. FireWall-1 uses an object-oriented approach for managing devices, thus allowing an administrator to define routers as network objects and use them when defining security-policy rules. Once defined, routing objects can be reused when defining and managing Access Control Lists (ACL) on routers. There is no need to remember each IP address for network interfaces. FireWall-1 allows administrators to manage router security using logical names or associations. Since all operations are performed with the use of drag-and-drop technology, the router-configuration process is simple and easy, and doesn't require knowledge of specific commands. Modifying configuration and customizing settings is also very easy, since all modifications are applied to objects. If configuration relates to multiple objects, the system will introduce modifications to all related routers, so there is no need to perform this operation manually. Once defined, each object can be modified. By simply clicking the "load policy" button on the central management console, one can easily reconfigure all routers.

The Check Point router security management center can be purchased as a separate product, exclusively for managing ACLs on routers, or supplied as a part of a complete security solution for an enterprise (FireWall-1 enterprise security solution). In each case, the ease of configuration has different levels of justification. The most important is the Check Point FireWall-1 capability of configuring and controlling ACLs on multiple routers from a single console. If configuration modifications are required, they are performed once from the central management console, and then new ACLs are automatically generated by the system and automatically distributed to appropriate routers within the enterprise's network.

Firewall-1 also provides broad capabilities for accumulating statistics and generating alerts:

Connection Accounting-FireWall-1, besides normal registration of each established connection, allows integral data to be gathered on the duration, amount of transmitted data (in bytes) and number of transmitted packets for each session. This information is registered in a log file when the session being traced is closed. Additionally, parameters of active connections can be traced.

Active Connections-in FireWall-1, the security administrator can use the same tool-Log Viewer-to trace an active connection through firewall modules. These statistics are processed in real-time mode and provided to the operator in the same way as normal records. The records are stored in a special log file, where they reside until the connection is closed. This allows one to use the same mechanisms of event selection as those used when working with a normal statistics file. Notice that, when the option of gathering additional information on connections is used, the integral statistics data are constantly updated, so the security administrator can track not only the connections themselves, but their real-time traffic as well.

Various alerting capabilities-FireWall-1 includes a variety of alerting options: from alerts sent via e-mail to the capability of sending SNMP traps for integration with various network management platforms, such as HP OpenView, SunNet Manager, IBM's NetView 6000. In addition to the basic alerting mechanisms, it is also possible to create custom variants of handling situations that require sending an alert. This provides the integration of the security system with paging services or fast reaction systems.

Integration of Access Control Tools and VPN Tools

The FireWall-1 cryptographic protection system is equipped to create a VPN by means of encrypting and digitally signing packets of the messages transmitted via open communication channels. FireWall-1 provides encryption speeds of more than 10 Mbit/sec on a desktop workstation and supports the following cryptographic standards:

- FWZ-native encryption scheme of FireWall-1
- IPSec-a set of commonly accepted open standards for creating VPN at the network level, developed and controlled by IETF
- SKIP (Simple Key-Management for Internet Protocols)-cryptographic key distribution scheme developed by Sun Microsystems

The relationship between supported cryptographic standards implemented in FireWall-1 is outlined in Table 2.4.

Table 2.4: Cryptographic Standards Implemented in FireWall-1

Encryption scheme	FWZ	IPSec	SKIP
Key-management protocol	FWZ	Manually	SKIP-key management protocol for IPSec
Authentication algorithm	MD5	MD5, SHA-1, CBC DES\3DES, MAC	MD5, SHA-1, CBC DES\3DES, MAC
Encryption algorithm	DES\3DES, FWZ1	DES\3DES, RC4, RC5, Blowfish, CAST, FWZ1	DES\3DES, RC4, RC5, Blowfish, CAST, FWZ1
Encryption is performed	In place	Built-in scheme	Built-in scheme

FireWall-1 provides completely transparent encryption for a wide range of services. Encryption and key-management functions are integrated with other FireWall-1 functional capabilities. The following two cryptographic protection modes are supported:

- Protection of the traffic between FireWall-1 modules installed within two or more local-area networks connected via the Internet

- Protection of traffic when establishing remote connection to the LAN protected by FireWall-1

In the first case, all protective functions are transparently implemented by the Firewall-1 systems, between which a connection is established.

In the second case, cryptographic protection functions are performed by FireWall-1 of the LAN which it is necessary to access remotely, and by a special FireWall-1 SecuRemote component that must be installed on a remote computer.

The SecuRemote component is an integral part of the cryptographic subsystem of FireWall-1. SecuRemote enables the users of mobile and remote Windows systems to connect to the FireWall-1 server of the corporate network directly via telephone lines or an Internet Service Provider, and work as securely as if they were located within a LAN protected by FireWall-1. SecuRemote extends a protected LAN directly to a remote computer.

The range of stand-alone VPN products supplied by Check Point offers protection of the transmitted data on the basis of open standards, for all types of communications using public networks, and for the most critical connections within a corporate network.

For protecting "net-to-net" connections within intranets (i.e., connections between enterprises' networks), as well as within extranet (connections between enterprise networks to their business partners' networks) it is possible to use the VPN-1 Gateway or VPN-1 Appliance products.

VPN-1 Gateway represents a software solution joining the firewall functions of FireWall-1 with VPN functions on the basis of Windows NT or Unix platforms. The VPN-1 Appliance product line represents several models of specialized software and hardware devices developed by Check Point in co-operation with Nokia.

Both VPN-1 Gateway and VPN-1 Appliance devices support the IPSec, IKE, X.509 digital certificates and Public Key Infrastructure (PKI). Implementation of the IPSec protocols included with VPN products by Check Point was certified by ICSA, which guarantees the possibility of establishing protected extranet channels to a partner's networks, where standard IPSec products are used. Besides IPSec, these products support the protection of the transmitted data based on the SKIP protocol commonly accepted on the Internet, as well as on the basis of the FWZ proprietary protocol. All of the above-described schemes can be employed for user authentication.

The VPN-1 Appliance line of devices joins of the funcitonal capabilities of VPN-1 Gateway with the advanced IP routing methods developed by Nokia. VPN-1 Appliance devices are intended for the organization of protected channels in large and medium organizations, requiring a high level of performance and reliability provided by a specially designed hardware platform. The VPN-1 Appliance devices are very convenient to use in remote affiliations, which are often lacking in qualified professionals. The Web-based application Network Voyager, developed by Nokia, allows the administrator to configure and manage the VPN-1 Appliance gateway using a standard Web browser from any location within the network. As a result of a test of VPN gateways from the six leading manufacturers conducted by Network World magazine in April 1999, the VPN-1 Appliance won the "Blue Ribbon" award, which serves as evidence that this new line of Check Point products has definite potential and good prospects for the future.

Remote and mobile users can support protected connections to their company's networks using such client software as VPN-1 SecuRemote and VPN-1 SecureClient. VPN-1 SecuRemote supports the IPSec and IKE standards and digital certificates, which allows VPN connections both to the gateways of the corporate network and those of business partners. The VPN-1 SecureClient product complements VPN capabilities of SecuRemote with access control functions based on the same stateful inspection technology that is used in FireWall-1/VPN-1 products. Using VPN-1 SecureClient streghtens enterprise security, since a remote computer can only establish a VPN connection when its security configuration satisfies the security rules specified by the security administrator.

The VPN-1 SecureServer product provides a standalone business-critical server within an enterprise's network with full FireWall-1/VPN-1 security functionality. This can be useful for unprotected remote offices, and for providing additional security measures for specific servers within enterprise networks.

All Check Point VPN products support various systems and tools for supporting public key infrastructures from leading manufacturers, which allows an enterprise to organize protection of data sent to partners without any significant problems.

Check Point access-control and VPN tools are fully integrated:

- The administrator creates VPN objects and rules defining the protected traffic using a unified GUI, also used for creating objects and access control rules for FireWall-1 modules. Integration of the access-control policy and VPN protection significantly enhances corporate network security thanks to coordination and consistency of the set of rules.
- Objects and rules of both types are stored on the Management Server, which distributes them to an enterprise's firewalls and gateways.
- FireWall-1 and VPN-1 modules work in coordination, correctly process protected traffic and apply the same access rules to it which apply to unprotected traffic. An integrated FireWall-1/VPN-1 device does not require two Internet connection channels. Instead, it accepts both protected and unprotected traffic via the same channel.
- Access-control and VPN modules use common user-authentication tools that significantly simplify the process of configuring the security system and minimizes the number of possible administrative errors.
- Auditing results are logged into a common log file using a common style, which improves the efficiency of event analysis for security-related events.

The FireWall-1 and VPN-1 modules as well as other products released by Check Point and its partners possess a high level of integration. Product integration simplifies the creation of a complex enterprise-wide information-security system, secures protection from all possible attacks and significantly reduces expenses for configuring and managing the whole system.

Ensuring Performance and Quality of Service

The requirements of the applications for communication performance are constantly growing. Check Point products satisfy these requirements using the following approaches:

- High speed of access-control operations and VPN traffic protection. This speed is ensured by a high efficiency of Stateful Inspection algorithms, capabilities of installing Check Point products on high-end servers (including multiprocessor ones), as well as including VPN-1 Accelerator Card if necessary. The VPN-1 Accelerator Card can be installed onto any standard platform to implement the most calculation-intensive functions at the hardware level.
- The ability to distribute security functions between several network-security tools working in parallel. Within a single network, it is possible to install several firewalls and VPN gateways, coordinating and synchronizing their work, so that multiple sessions will be distributed between them. It is also possible to organize access-control server pools to improve performance of such operations as scanning files for viruses, searching for specific keywords within the contents, protection from malignant Java and ActiveX applets and so on. A special ConnectControl module, included with FireWall-1, lets users distribute the workload between a pool of similar servers (for example, Web of FTP servers), which is very convenient for organizing a demilitarized zone.
- Bandwidth distribution is managed using a special standalone product, FloodGate-1, designed by Check Point.

FloodGate-1 is closely integrated with the FireWall-1/VPN-1 tool. However, it can run as a standalone QoS product. FloodGate-1 is mainly intended for the distribution of bandwidth between the traffic of different applications at the perimeter of the corporate network, namely, in channels connected to the interfaces of FireWall-1/VPN-1. All the external traffic of a protected enterprise's network passes via these interfaces, including critical business applications as well as less important ones. The differentiated distribution of bandwidth can significantly improve the performance of business-critical applications by providing the required bandwidth and protecting them from intensive but less important traffic, such as that of users surfing the Internet or downloading files from FTP archives.

Operating systems, on which FireWall-1/VPN-1 runs on standard platforms, do not currently support QoS functions. Installation of FloodGate-1 on hosts playing the role of the firewall of a VPN gateway eliminates this problem and enhances the QoS management system running on all modern routers and switches with the important QoS element.

FloodGate-1 manages bandwidth on the basis of the improved WFQ algorithm, often implemented in IP routers and switches. Check Point has complemented the basic WFQ mechanisms with its custom intelligent algorithms, and on account of them, FloodGate-1 supports a practically unlimited number of virtual queues and guarantees a specific part of the bandwidth both for an aggregated flow (for example, all Web traffic) and an information flow of separate connections (such as a standalone RealAudio connection). For each type of traffic, besides a guaranteed part of the whole bandwidth, one can also specify the upper and lower limits of the bandwidth. This allows for a flexible account of the applications' specific requirements and provides a privileged service for business-critical applications. At the same time, it also offers the possibility of a guaranteed minimum of the bandwidth for other applications under any circumstances. The bandwidth is distributed dynamically, quickly reacting to the changes in the existing data flow according to the predefined rules.

Flow classification is performed using the Stateful Inspection Technology used in FireWall-1. Check Point's extensive experience in this area has allowed them to implement one of the most powerful traffic classification mechanisms for the purposes of QoS management in FloodGate-1. With this mechanism, one can consider (and devise ways of dealing with) practically any and all possible situations that might arise in the corporate network and group traffic by applications and users in the most rational way.

When using FloodGate-1, the efficiency of bandwidth usage increases due to execution of the RDED (Retransmission Detect Early Drop) algorithm, which solves the common and well-known problem of decreasing useful bandwidths of TCP connections due to multiple retransmissions under conditions of a network-intensive workload. The RDED mechanism prevents transmissions of several copies of the same packet, thus increasing useful bandwidth up to 95% of the complete channel bandwidth instead of the traditional 50%-60%.

The FloodGate-1 traffic-control system is fully integrated with FireWall-1/VPN-1 and has the same distributed architecture. The rules of traffic control are specified in the same style as those of access control and VPN protection rules, and all rules can use common objects. The GUI editor is used to specify traffic control rules. Notice that, if FireWall-1 is deployed in an enterprise, this editor is represented by a separate tab on the Check Point Policy Editor. Traffic-management rules are stored centrally in the Management Server database and are distributed to all FloodGate-1 modules within the enterprise.

The FloodGate-1 system allows the administrator to monitor bandwidth distribution in real-time mode. As a result, the specified rules can be evaluated efficiently and the weight coefficients and bandwidth limits can be re-distributed in such a way as to take into account applications' and users' real requirements.

Integration of FloodGate-1 with FireWall-1/VPN-1 ensures that traffic management tools will operate accurately when combined with the VPN protection tools.

Report Generation

All the network traffic transmitted via different components included in the Firewall-1 and VPN-1 family is registered as appropriate log files. Manual analysis of this information is impossible, because of the vast amount of data collected. Furthermore, the task of manual log-file analysis becomes tedious with distributed installation of firewall components. Finally, when performing manual analysis, it is rather difficult (and often simply impossible) to combine several log files to perform a more detailed analysis of the network traffic (for example, in order to detect DoS attacks). The add-on Reporting Module subsystem solves this task quite easily (Fig. 2.27).

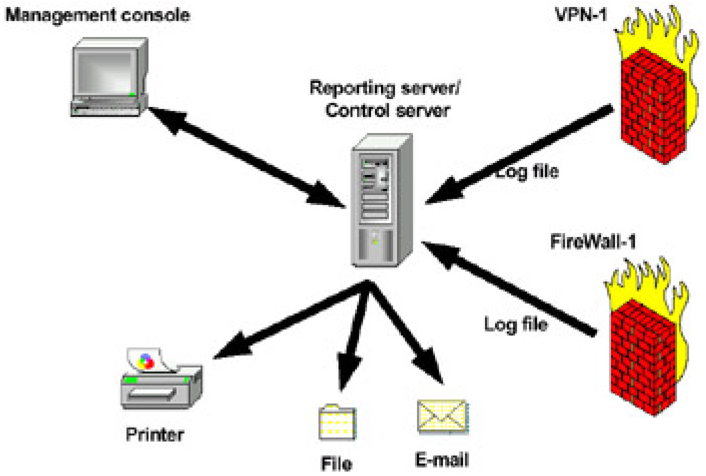


Fig. 2.27: The scheme of Reporting Module usage

The Reporting Module comprises two components:

- Reporing Server-intended to collect data from VPN-1, Firewall-1 and FloodGate-1, combine this data and generate reports on the basis of existing rules.
- Reporting Client-intended for specifying the rules for the Reporting Server and displaying the reports generated in the Reporting Module proprietary format.

The Reporting Module subsystem provides 20 predefined report templates that encompass the most common aspects of firewall usage. Besides these templates, the security administrator can create an unlimited number of custom templates, taking into account specific features involved in using Firewall-1 and VPN-1 in a specific corporate network.

All generated reports can be displayed on screen, printed, published on a Web server, sent via e-mail exported to a file and passed to the specified software application. Reports are generated and delivered automatically, according to a predefined scenario.

The Reporting Module supports various formats (including HTML and ASCII) to which the created reports can be exported.

The reports can only contain information from the required log-file fields. This feature allows the report size to be minimized without forcing the administrator to analyze execssive data. Any field of a log file can sort the information within a report.

Workload Balancing and Ensuring High Availability

Because of the explosive growth of the Internet, most companies have to install powerful servers in order to provide constant availability of information and satisfy the growing requirements for service quality. If the workload on the Web server becomes too intense, the response time to client requests might become intolerably long. In the worst-case scenario, a client might work indefinitely and still not get a reply to their request.

The ConnectControl product, representing a separate Firewall-1 module, adds flexible features of managing traffic, thus providing a balance to incoming connections and distributing them to several servers. The network administrator can replace each Web server or any other application server with a logical group of servers and provide transparent access to the whole group with a single IP address. Incoming connections are redirected to any server within the group using one of the five available mechanisms of workload balancing (Fig. 2.28).

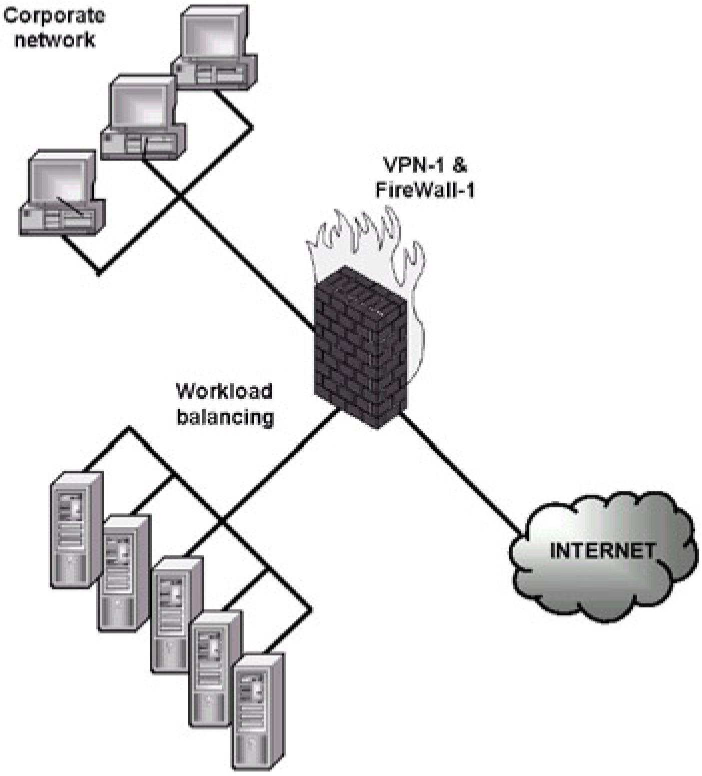


Fig. 2.28: Scheme of workload balancing

As a result of using parallel servers, users will benefit from a significant improvement in performance. The presence of the ConnectControl module is concealed from users, since it works transparently for them. Furthermore, planned server replacement can be postponed. Using this product, fully integrated with the whole security policy of the company, can significantly improve server performance, without noticeable expenses.

Another problem is related to the fact that most companies run applications whose most significant parameter is high availability. These applications must be available to users 24 hours a day, and, if a company fails to satisfy these requirements, it might suffer significant losses (both financial and in terms of customer confidence). Constant availability might suffer as a result of Denial of Service attacks on the firewall or other perimeter protection tool. Thus, ensuring high availability is a key feature of firewalls.

The Check Point Software firewall has mechanisms implemented in the High Availability Module subsystem that make it possible for the administrator to create firewall clusters comprising two or more firewalls that can replace each other in case of failure. This replacement is transparent to the end-user accessing the resources protected by the firewall supporting the high availability module. When such a transition occurs, active connections are not closed, and users do not need to be authenticated repeatedly.

All setting of the High Availability Module are performed from the central console of the Check Point Management Client (Fig. 2.29). If one of the firewalls fails, this event is registered in the log file, and the security administrator is notified by one of the available notification mechanisms (e-mail, SNMP, or pager). Furthermore, the status of all clustered firewalls is displayed on the management console in real-time mode. Adding and deleting of firewalls to/from the cluster does not require the administrator to reconfigure or restart it.

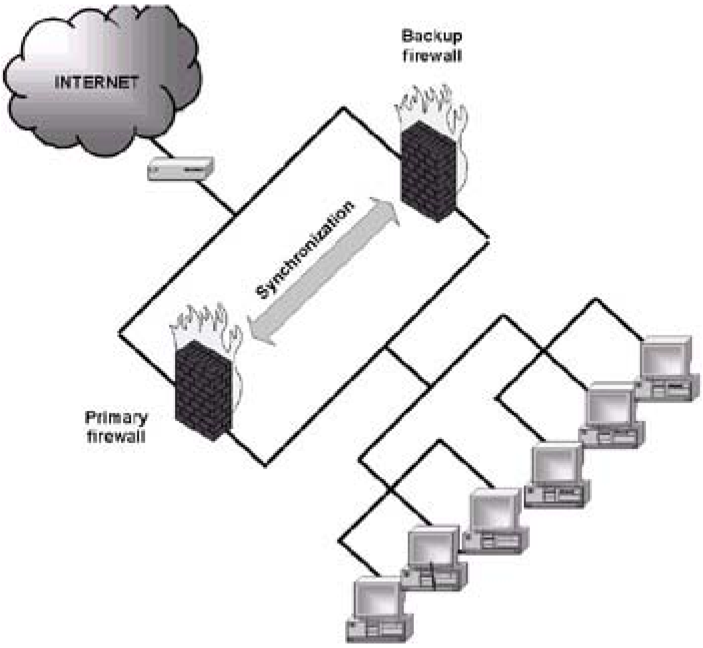


Fig. 2.29: Scheme of a cluster comprising two firewalls

Chapter 3: Construction of Protected Virtual Networks

3.1. Introduction to Virtual Private Networks

3.1.1. General Information

The safety of information interaction of local-area networks and individual computers through public networks, such as the Internet, requires the efficient solution of two basic tasks (Fig. 3.1.):

- The protection of local-area networks and individual computers connected to public communication channels from unauthorized operations performed from external environment
- The protection of information being transmitted via public communication channels

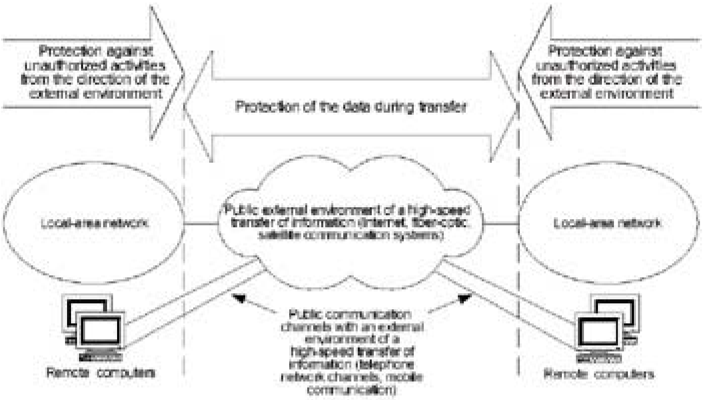

Protection against unauthorized activities from the direction of the external environment
Protection of the data during transfer
Protection against unauthorized activities from the direction of the external environment
Local-area network
Public external environment of a high-speed transfer of information (Internet, fiber-optic, satellite communication systems)
Local-area network
Remote computers
Public communication channels with an external environment of a high-speed transfer of information (telephone network channels, mobile communication)
Remote computers

Fig. 3.1: Tasks of providing security of information interaction

Using firewalls, which were covered in the previous chapter, solves the first problem. Firewalls used for this purpose must support information security by filtering a two-way message flow, and also by implementing the functions of a proxy during the information exchange. For protection of local-area networks, the firewall is located at the junction between a local and public network. For protection of a remote computer connected to a public network, the firewall software is installed on the same computer, and, in this case, the firewall itself is identified as personal.

Information protection during a transfer via public communication channels can be achieved through the following:

- Authentication of communicating parties
- Cryptographic protection of transmitted data
- Confirmation of the authenticity and integrity of the delivered information
- Protection from duplication, delay, and removal of messages
- Guaranteed recognition of message sending and reception

In many respects, the above-listed functions are interrelated, and their implementation is based on cryptographic protection of the transmitted data. The high efficiency of such protection is provided by means of combining symmetric and asymmetric cryptosystems.

The combination of local-area networks and individual computers via the public external environment is identified as a Virtual Private Network (VPN). This network is formed through communication channels of a public network. The term "virtual" itself emphasizes that the communication channels of a virtual network are simulated using real communication channels. A public network can form the basis for simultaneous coexistence of a set of virtual networks, the quantity depending on the throughput of public communication channels.

A public environment can be divided into a high-speed data-transfer environment (such as the Internet, for example), and slower public communication channels (most often, telephone network channels are used for this purpose). The Internet provides the most efficient way of building VPNs by combining LANs and remote computers (Fig. 3.2). If a direct Internet connection isn't available, one can access the Internet via a telephone line. Organizing virtual networks on the basis of the Internet provides several important advantages:

- Guarantees a high quality of information exchange, since Internet backbone channels and routers are distinguished by a high throughput and reliability.
- Provides scaled support for remote access to local-area network resources. Mobile users can connect to local Internet Service Providers (ISPs) in order to login to their corporate networks.
- Eliminates the need in modem pools for organizing remote access to a local-area network. RAS traffic can be managed in just the same way as any other Internet traffic.
- Reduces expenses required for information exchange via public communication lines:
    - Using the Internet to merge local-area networks is much cheaper than renting a telephone and other global network communication channels, such as frame relay networks, not to mention the cost of independently constructing communications.
    - With remote access, instead of installing expensive direct connections with a local-area network with inter-urban or international telephone communication, remote users can connect to the Internet and, furthermore, they can contact their organization's network through this global network.

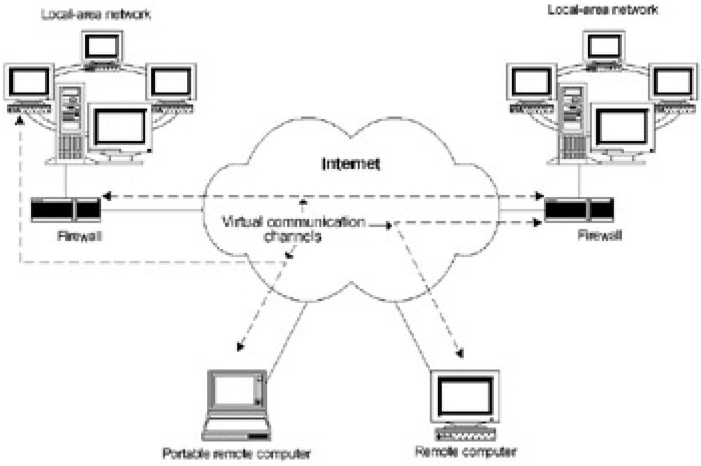


Fig. 3.2: Construction of a virtual network on the basis of the Internet.

A virtual network that uses the Internet as an external information-transfer environment is also known as *extranet*.

In many ways, VPN efficiency depends on the level of security of the information circulating on open communication channels. It is necessary to ensure the safety of an information exchange both in the case of merging local-area networks and when organizing remote user access to the LAN (Fig. 3.2).

The protection of the information during its transfer via open channels is based on the construction of protected virtual communication channels, called cryptographic channels, or VPN channels. Every such tunnel represents a connection established via a public network and used to transmit encrypted packets.

For protection against duplication, removal and delays of message packets transmitted on VPN tunnels, the built-in functional capabilities of the TCP/IP protocol stack are used. For protection against duplication, removal, and delays at the level of single messages, the application-level security subsystem must supplement each message with additional information. As such, ordinal numbers, random numbers and timestamps can be used.

In order to avoid discarding messages and denying reception, the application-level security subsystem should provide the capability of sending the receipt notification to the sender. Such notification must be digitally signed by the message recipient. To prevent users from refusing to send the notification message, that is, to provide the protection against failures to recognize a digital signature, it is necessary to develop and implement organizational meausres that would legalize a digital signature. To prevent rejection of open keys and accordingly, the rejections of a digital signature, an exchange of open keys should be accompanied by a legal procedure.

3.1.2. Methods of Creating VPN Channels

Any two hosts of a virtual network connected by a protected tunnel can belong to the endpoint or to an intermediate point of a protected data flow. Consequently, various ways of creating the protected virtual channel are possible (Fig. 3.3).

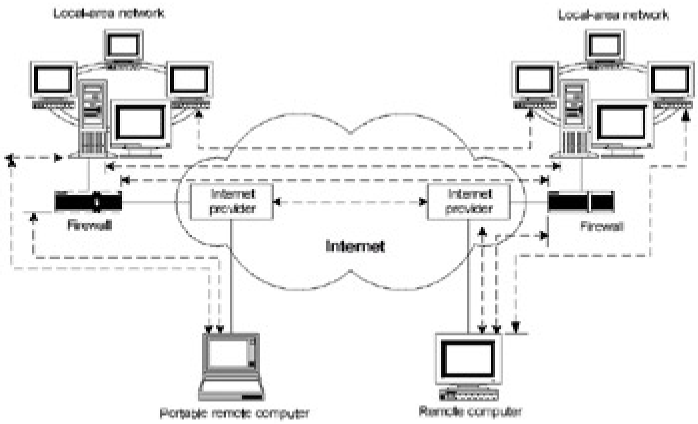


Fig. 3.3: Versions of protected virtual channels

In terms of safety, the best version is the one in which the endpoints of the protected tunnel coincide with the endpoints of the data flow that needs protection. In this case, complete security is provided for message packets along the whole message route. However, such a version leads to management decentralization and excessive resource consumption. In particular, implementation of this scheme requires installation of VPN tools on each client workstation participating in the LAN. This complicates centralized access control to computer resources. Furthermore, quite often, such an approach is not economically justified. In a large network, the independent administration of each client computer in order to configure security tools is quite a tedious procedure.

Therefore, if there is no need to protect the traffic within a local-area network included in a virtual network, it would be expedient to designate a firewall or a boundary router of this LAN as the end-point of the protected tunnel. On the other hand, when the traffic within the LAN also needs protection, the computer representing one of the parties participating in the protected communication must play the role of the endpoint of the protected tunnel. When a remote user wants to access a local-area network, his computer should also act as an endpoint of the VPN channel. A version characterized by lower safety but higher application convenience is also widely used. According to the given variant, neither workstations, local-area-network servers, nor remote computers participate in the creation of the protected tunnel, which is only laid inside a public network with packet switching, for example, inside the Internet. Internet providers and/or boundary routers (firewalls) of a local-area network more often act as endpoints of such a tunnel. Upon remote access to a local-area network, a tunnel is created between the Internet provider's remote access server and, also, the boundary Internet provider or the local-area-network router (a firewall). At integration of local-area networks, a tunnel is only formed between boundary Internet providers and/or local-area-network routers (firewalls).

One argument in favor of this version of creating virtual networks is that, for intruders, networks with packet switching, such as the Internet, are significantly more vulnerable than telephone channels or dedicated connection lines. Virtual networks constructed on the given version have good scalability and controllability. For client computers and local-area-network servers included in a virtual network (meaning a local-area network is included), protected tunnels are completely transparent and the software of these hosts remains unchanged. However, since part of the protected traffic passes in an unprotected mode via public communication channels, this approach significantly reduces the security level. Furthermore, since most part of the work is delegated to ISPs, they must be paid and trusted.

A protected network is created by virtual-network components on hosts between which a tunnel is formed. These components are called the tunnel's initiator and terminator. The initiator of the tunnel encapsulates packets into a new packet that contains the initial data and a new heading, with information about the sender and recipient. Though all packets transmitted on the tunnel are IP packets, any type of protocol, including the ones without routing support, such as NetBEUI, can possess encapsulated packets. A standard routed IP network, which can be different from the Internet, defines the route between the tunnel initiator and terminator. The terminator of the tunnel carries out the process opposite to encapsulation-it deletes new headers and forwards each original packet to a local protocol stack or to the recipient in a local-area network.

The encapsulation itself does not affect the security of the packets transmitted on a VPN tunnel. But full cryptographic protection of encapsulating packets is possible, on account of encapsulation. Information on encapsulating packets is confidential as a result of encryption. Data integrity and authenticity are ensured by a digital signature. As there are various methods of cryptographic protection available, it is very important that the tunnel's initiator and terminator use the same methods and coordinate this information with each other. Besides, in order to allow the data checking of the digital signature to be deciphered upon reception, the tunnel's initiator and terminator should support a safe key exchange. In order to create VPN tunnels between authorized users only, communicating parties need to be authenticated.

3.1.3. Protocols Overview

Technically, organization of virtual private networks became possible quite a long time ago. Encapsulation was used earlier and is now applied to transfer non-routed traffic through routed networks, and for the purpose of limiting multi-protocol traffic by a single protocol. The technology of encryption also appeared long before global network technologies became widely available. However, the standard protocols for creating virtual private networks have been developed only recently, and, now, they are still undergoing further improvement. They are open, that is, free for distribution and implementation.

For independence from application-level protocols and software applications, virtual private networks are formed on one of the lower levels of the OSI model-on the data-link, network or session layers. Such protocols of VPN implementation as PPTP (point-to-point tunneling protocol), L2F (Layer-2 Forwarding) and L2TP (Layer 2 Tunneling Protocol) correspond to the data-link layer (the second layer of the OSI model); IPSec and SKIP-to the network layer (the third layer); while protocols such as SSL/TSL and SOCKS correspond to the session layer (the fifth layer of the OSI model). The lower the reference model level on which protection is implemented, the more transparent for applications, and more imperceptible for users, it will be. However, the lower this level is, the smaller is the set of security services possible to implement, and the more complicated becomes the management. The higher the OSI model level at which security services are implemented, the wider becomes the set of security services. Besides this, access control becomes more reliable, and system setup and configuration become easier. However, in this case, the implementation becomes more dependent on communication protocols used, as well as on the network applications.

In a virtual network, cryptographic protection can be simultaneously carried out at several levels of the OSI reference model. With that, security increases, but, owing to a reduced general rate of cryptographic transformations, the throughput of a virtual network decreases. Therefore, in practice, virtual private networks are formed at one of the layers of the OSI model (the data-link, network, transport or session layers).

The Data-Link Layer of the OSI Model

For standard formation of cryptographic tunnels at the data-link layer of the OSI model, Microsoft, with the support of Ascend Communications, 3Com/Primary Access, ECI-Telematics and US Robotics, developed PPTP tunneling protocol, representing an expansion of the PPP protocol. Specific methods of authentication and encryption are not specified in the PPTP protocol. Remote clients in Windows NT 4.0 and Windows 98 with Dial Up Networking are delivered with the DES version of encryption put out by RSA Data Security, which has received the name Microsoft Point-to-Point Communication Encryption-MPPE (Microsoft Point-to-Point Encryption).

The L2F tunneling protocol, developed by Cisco Systems with the support of Shiva Corporation and Northern Telecom, also corresponds to the OSI model data-link layer. In the given protocol, specific methods of authentication and encryption are also not specified. In contrast to the PPTP protocol, the L2F protocol for remote access to an Internet provider allows one to use not only the PPP protocol, but also other protocols, for example SLIP. Upon forming protected channels on wide-area networks, Internet providers do not need to carry out address configuration and authentication. Furthermore, to transfer the data through a protected tunnel, various network-layer protocols, and not just IP, such as PPTP protocol, can be used. The L2F protocol became a component of the IOS (Internetwork Operating System) operating system developed by Cisco and is supported in all internetworking and remote access devices that have been released by this company.

PPTP and L2F protocols were presented to the Internet Engineering Task Force (IETF) and, in 1996, the appropriate committees decided to combine them. The resulting protocol was named Layer-2 Tunneling Protocol (L2TP). This protocol is supproted by Cisco, Microsoft, 3Com, Ascend and many other manufacutrers. As well as preceding data-link layer protocols, L2TP specification does not include authentication and encryption methods. On the data-link layer, the L2TP protocol is an extension of the PPP protocol and may support any high-level protocols.

Protected-tunnel formation protocols on the data-link layer are independent of network-layer protocols of the OSI model, in which local-area networks are included in virtual networks' functions. They allow the creation of protected channels for data exchange between remote computers and local-area networks functioning under various network-layer protocols-IP, IPX or NetBEUI. Packets of these protocols are cryptographically protected and encapsulated in IP packets that are transferred to a destination, forming VPN channels. Its status as a multi-protocol is the basic advantage of the data-link layer encapsulating protocols.

At the same time, the formation of cryptographically protected tunnels between united local-area networks via data-link-layer protocols results in complex configuration and support of virtual communication links. PPP-based tunnels require the endpoints to support the information on the status of every channel (for example, time-outs) and, therefore, do not provide a good scalability if it is necessary to have several tunnels with common endpoints. Furthermore, protocols of formation of the protected tunnels do not specify particular methods of encryption, authentication, integrity checks of every transmitted packet, nor means of key-control on the data-link layer.

Proceeding from the aforesaid points, it is possible to draw the conclusion that protocols for creating VPN channels on the data-link layer deal with protecting information interaction upon remote access to a local-area network in the best way. Taking into account that the structure of Windows 98/NT OS includes PPTP protocol implementation, this protocol has received the widest distribution for organizing protected remote access. A reorientation of remote access towards a more perfect protocol is expected in the near future.

The Network Layer of the OSI Model

The Internet Protocol Security (IPSec) protocol, which is appropriate to the network layer of the OSI model, is a specification in which standard methods for all components and functions of VPNs are described. It is included in the new version of the IP-IPv6 protocol-and is sometimes referred to as the Layer-3 Tunneling Protocol (L3TP). IPSec provides standard user or computer authentication methods during a tunnel's initiation, and standard methods of encryption by the tunnel's endpoints and of the exchange and management of cryptographic keys between endpoints. It is flexible, and offers several ways to implement each task. The chosen methods of one task do not usually depend on the methods of implementation of other tasks. For authentication functions, IPSec supports digital certificates corresponding to the popular X.509 standard.

An IPSec tunnel between two local-area networks may support a set of individual data links. Therefore, applications of the given type receive advantages in terms of scaling in comparison with the technology of the second level. The IPSec protocol may be used in combination with the L2TP protocol. These two protocols jointly provide the highest level of flexibility upon protection of virtual channels. The point is that IPSec specification is based on the IP protocol and thus, is useless for the traffic of any other network-layer protocols. The L2TP protocol is notable for its independence from the transport layer, which may be useful in heterogeneous computer networks consisting of IP, IPX, and Apple Talk segments. IPSec has swiftly gained popularity and is probably going to be the dominant standard for the creation and support of virtual private networks. For management of cryptographic keys on the network layer of the OSI model, such protocols as SKIP (Simple Key Management for Internet Protocols) and ISAKMP (Internet Security Association and Key Management Protocol) have garnered the widest circulation. SKIP is easier in implementation, but it does not support negotiations concerning encryption algorithms. If the recipient is not able to decipher the packet while using SKIP, he will no longer be able to coordinate encryption methods with the opposite party. The ISAKMP Protocol supports such negotiations and is chosen as the prerequisite protocol for key management in IPSec for IPv6. In other words, the ISAKMP protocol is a component of the IPSec protocol. In the current version (in the IPv4 protocol), both the ISAKMP and SKIP protocols may be applied.

The Session Layer of the OSI Model

VPN channels can be organized at the OSI model's session layer. So-called circuit proxies are applied for this purpose. The proxy operates above the transport layer. It ciphers and retransmits traffic from a protected network to the Internet public network for each socket separately. Upon reception, the return procedure is carried out. An IP socket is identified by a combination of TCP connections and a specific port or a given UDP port. For encryption of the information on the session layer, Netscape's SSL/TSL (Secure Sockets Layer/Transport Layer Security) protocol has gained the greatest popularity. This protocol creates a protected tunnel between endpoints of a virtual network, providing mutual subscriber authentication, confidentiality, authenticity and integrity of the data circulating on the tunnel. The kernel of the SSL/TSL protocol is the technology of multiple use of asymmetric and symmetric cryptosystems distributed by RSA Data Security. Digital certificates of users' public keys (i.e. client's and server's), certified by digital signatures of special certification centers are used for the authentication of the cooperating parties and cryptographic protection of the symmetric encryption key. Digital certificates appropriate for the generally accepted X.509 standard are supported.

In order to standardize the procedure of interaction of client/server applications via TCP/IP through a proxy server (a firewall), the IETF committee has approved a protocol under the name SOCKS, and now the fifth version of the given protocol (SOCKS 5) is applied to the standardized implementation of proxy channels. SOCKS supports applications demanding control over data streams' directions and adjustment of access conditions depending upon user and/or information attributes.

According to SOCKS, the client computer establishes an authentication session with the server implementing the role of the proxy. The use of this proxy is the way to gain connection through a firewall. The proxy, in turn, carries out any operations required by the client. As the proxy knows about the traffic at the session level, it may carefully exercise control, for example, by blocking specific users' applications if they do not have sufficient rights.

In contrast to VPNs on the session layer of the OSI model, those on the data-link or network layer usually simply open or close the channel for all traffic on the authenticated tunnel. This may represent a problem if the local-area network on the other end of the tunnel is unreliable. Besides, the created tunnels of the data-link and network layers function equally in both directions, and virtual networks on the session layer assume an independent control of transfer in each direction.

Virtual networks with a channel proxy such as IPSec are based on the IP protocol. If IPSec, in essence, differentiates VPN channels between different pairs of cooperating sides, then the SOCKS 5 protocol will create protected tunnels for each application and session separately. Similar to the IPSec and data-link layer tunneling protocols, virtual networks of the session layer can be used with VPN types, since the given technologies are not mutually exclusive.

It is necessary to note that there are protocols for protected interaction at the OSI model's application layer. These protocols, as a rule, are additions to various protocols of an application level. For example, the Secure HTTP (SHTTP) protocol is an addition to the protection functions of the hypertext transfer protocol, HTTP, and Secure MIME (S/MIME)-an addition to the protective functions of the MIME e-mail protocol. Applied protocols of the protection of information interaction do not carry over to protocols of construction of VPNs as they completely depend on used services and applications. Protocols that create VPN channels are transparent for application protocols of protection. Accordingly, using applications that implement, for example, SHTTP or S/MIME, alongside cryptographic protection at a lower level, does not weaken, but only strengthens security.

3.2. Tunneling on the Data-Link Layer

3.2.1. The PPTP Protocol

The PPTP (Point-to-Point Tunneling Protocol) protocol, developed by Microsoft with the support of other companies, represents the extension of the PPP (Point-to-Point Protocol) protocol for creating VPN channels upon remote user's access to local-area networks through the Internet. It creates a cryptographic tunnel on the OSI model data-link layer, as in the case of a direct connection of a remote computer with a public network, and in instances where it is connected to a public network using a telephone line through a provider.

The given protocol was submitted to IETF as an applicant for a standard protocol creating protection of a virtual channel upon remote users' access to local-area networks through public networks (first of all, through the Internet). PPTP has received the status of a project at the Internet standard, however, in spite of its wide distribution, the standard was never authorized. Now IETF is considering accepting the L2TP protocol as the standard, which integrates the best aspects of the PPTP protocol and similar L2F protocols, and is offered by Cisco Systems.

For a remote user, connected through a public IP network to the Remote Access Service (RAS) of a local-area network, PPTP simulates the presence of this user's computer in an internal network by tunneling message packets. The data through the tunnel is transferred with the help of the PPP standard remote-access protocol, which in PPTP protocol is used not only for connection of the remote computer with the Internet provider, but also for interaction with the LAN's RAS through the tunnel. The IP packet containing encapsulated PPP packets is applied for data transfer. The encapsulated PPP packets, in turn, include ciphered encapsulated initial packets (IP, IPX, and NetBEUI), intended for interaction between a remote computer and a local-area network. Packets circulating within the limits of a PPTP session have the following structure:

- The data-link-layer header used inside the Internet, for example, inside an Ethernet frame
- The IP header
- The GRE (Generic Routing Encapsulation) header
- The PPP initial packet, including the IP, IPX and NetBEUI packets

The standard for the GRE v.2 protocol's Internet header, employed during the encapsulation of various types, is used for an indication that there is a PPP packet encapsulated within the IP packet.

This method of encapsulation provides independence from OSI model network-layer protocols and allows one to carry out protected remote access through public IP networks to any local-area networks (IP, IPX, and NetBEUI).

The technology of creating a VPN channel under the PPTP protocol provides both remote user authentication, and a ciphered data transfer.

Various protocols can be used for authentication. In PPTP implementation included in Windows 98/NT, PAP (Password Authentication Protocol-the protocol of password recognition) authentication and CHAP (Challenge Handshaking Authentication Protocol-the protocol of recognition at handshaking) protocols are supported. When using the PAP protocol, identifiers and passwords are transferred via communication line as plain text. While using the CHAP protocol, each password for transfer on the communication line is ciphered on the basis of a random number received from the server. Such technology also provides protection against reusing intercepted packets with the ciphered password.

The remote-access software implementing PPTP, may use any standard of cryptographic protection of the transmitted data. For example, the RAS of Windows NT uses an RC4 standard and, depending upon the version, 40 or 128-digit session keys, which are generated on the basis of the user password.

In the PPTP protocol, three schemes of its application are determined: one scheme-for cases of direct connection of a remote computer with the Internet, and two-when a remote computer is connected to the Internet via a telephone line through an ISP.

Upon direct connection of a remote computer to the Internet (Fig. 3.4), for example, at access from a local-area network directly connected to the Internet, the user initiates the remote connection with the help of the RAS client part. It makes contact with the LAN's RAS, specifying its IP-address, and initiates a connection under the PPTP protocol. A LAN perimeter router may carry out RAS functions. PPTP determines the number of service messages that the cooperating parties may exchange. Service messages are transferred using the TCP protocol. After successful authentication, the process of protected data exchange begins.

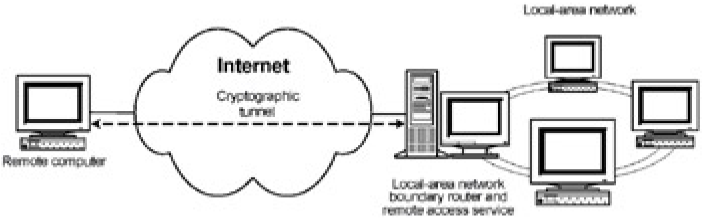


Fig. 3.4: The tunneling scheme at direct connection of a remote computer to the Internet

In this version, the RAS client and PPTP driver, which are included in Windows 98/NT structure, must be installed on a remote computer, and on the LAN's RAS. The RAS server and PPTP driver are included with the Windows NT Server operating system. LAN internal servers aren't required to support the PPTP protocol as the perimeter router takes PPP frames from IP packets and sends them on a network in a necessary format-IP, IPX, or NetBIOS.

For connection of a remote computer to the Internet on a telephone line, two schemes are stipulated (Fig 3.5).

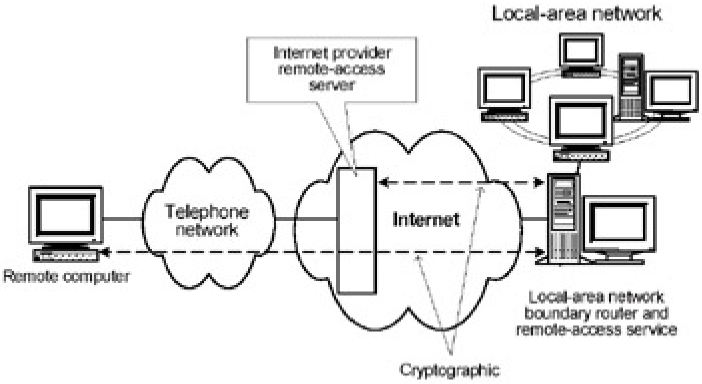

Local-area network
Internet provider remote-access server
Telephone network
Internet
Remote computer
Local-area network boundary router and remote-access service
Cryptographic

Fig. 3.5: Tunneling scheme upon connection of a remote computer to the Internet through a provider using a telephone line

The first scheme is designed for the maintenance of a protected channel between the RAS of the Internet provider and a LAN boundary router. In this version, the remote computer should not support the PPTP protocol. It contacts the RAS of the provider with the help of the standard PPP protocol and passes the authentication at the provider. The provider's RAS should support the PPTP protocol. According to the user name, the provider's RAS finds in the user database IP address of the router, being also the boundary router, and the LAN RAS of the given user. With this router, the ISP of the RAS initiates a session under the PPTP protocol.

The Local-Area Network Perimeter Router and Remote-Access Service

The provider's RAS transfers the user ID and other data to the LAN RAS, on the basis of which the server authenticates the user for a second time under the CHAP protocol. If the user is successfully authenticated, which is transparent, the provider's RAS sends a message using the PPP protocol, and then a VPN channel between the provider's RAS and LAN is formed. The remote computer sends LAN interaction packets (IP, IPX or NetBIOS) to the provider's RAS, encapsulating them in PPP frames. The RAS of the provider encapsulates PPP frames in IP packets, specifying a LAN boundary router address as the destination address, and its own IP address as the source address. PPP packets for transit between the provider's RAS and that of a LAN are ciphered with the help of a private key. The key's value is used as a hash-function from the user password, which is stored in the RAS' registration database of the provider for authentication under the CHAP protocol.

The scheme described above does not find wide application, as the PPTP protocol is basically implemented in Microsoft products-in Windows NT 4.0 client and server parts of the RAS, and in the client part of Windows 98. More often, providers use more powerful means than the RAS of Windows NT. In addition, the given scheme does not provide a high degree of security as between a remote computer and an Internet provider, the data are transferred as plain text.

When a remote computer is connected to the Internet through a telephone line via a provider, the second scheme (see Fig. 3.5) has received a wider application when the cryptographic tunnel is formed between endpoints of the interaction. Users initiate the remote connection twice with the help of the client part of the RAS, for example, with the Dial-Up Networking utility from Windows 98/NT. At first, they call on the provider's RAS modem and initiates a connection under the PPP protocol, passing the authentication using one of the methods supported by the provider-under the PAP or CHAP protocols or with the help of a terminal dialogue.

After successful authentication with the provider, the user initiates a connection with the LAN's RAS, indicating its IP address. Then, a session under the PPTP protocol is initiated between the remote computer and the LAN's RAS. The LAN's RAS verifies user authenticity with the help of the registration database. Upon successful authentication, the process of protected data exchange begins. To reduce manual operations, Microsoft offers script language resources in the Windows 98/NT RAS.

For interaction between boundary devices of a cryptographic tunnel in the PPTP protocol, the control messages intended for establishing, maintaining and closing the tunnel are stipulated. The control messages exchange is carried out on the TCP-connection initiated between the client and PPTP server. The packets transmitted through this connection besides the data-link-layer header contain the IP header, the TCP header and a PPTP control message in the data area of the packet. If the remote computer does not have PPTP support, which is typical for the scheme of creating a protected channel between the RAS of the Internet provider and a LAN boundary router, then this computer is not involved in the process of controlling the exchange of messages. In this case, the functions of the PPTP client are assigned to the RAS of the Internet provider. At PPP message-exchange control, the logical port of the TCP protocol with the number 1723 is used, and, when transmitting bagged PPP packets to the IP header, the protocol identifier equal to 47 is installed. These agreements allow one to use PPTP technology together with firewalls. To increase security tone should block any traffic different from PPTP traffic with the help of a LAN firewall.

3.2.2. The L2F Protocol

The L2F protocol was developed by Cisco Systems with the support of Shiva and Northern Telecom as an alternative to the PPTP protocol in creating virtual private networks on the data-link layer of the OSI model. Unlike PPTP, the given protocol is more convenient to use for Internet providers, and also for the support of different network protocols. According to the L2F protocol, various protocols of remote access can be applied for connection of a remote computer to a provider's server-PPP, SLIP, and others. A public network used for data transfer through the tunnel may function not only under the IP protocol, but also on the basis of other protocols, for example, the X.25 protocol. The L2F protocol is initially implemented for TCP/IP networks.

Besides the orientation to various data exchange protocols, the development of the L2F protocol was, as well, aimed towards the following:

- Flexibility of authentication, which does not contain rigid binding to specific authentication protocols.
- Transparency for communicating systems-both remote system, and the LAN workstations should have special software to use a protective service.
- Transparency for proxies:
    - Authorization should be carried out in the same way as in the case of direct connection of users to the RAS of a LAN.

    - The LAN server, rather than the provider's server, must assign the address of the RAS.
- The completeness of the audit providing the event logging about access to the LAN server not only by the RAS of this network, but also by the provider's server.

It is possible to allocate three types of protocols participating in the creation of a protected tunnel:

- The initial encapsulated protocol is the protocol on which the LAN functions and which is used for direct interaction with this LAN, for example, with the IP, IPX or NetBEUI protocols.
- The passenger protocol (the protocol that is transmitted with the help of the other protocol) in which the initial protocol is encapsulated and which, in turn, is required to be encapsulated through a public network upon remote access (the PPP protocol is recommended as the given one).
- The encapsulating protocol is a controlling protocol that is used for creating, maintaining and breaking the tunnel (in this case, the L2F protocol acts as such a protocol).
- The provider protocol, which is used to carry encapsulated protocols (the initial protocol and the passenger-protocol). The most flexible and popular provider protocol is the IP protocol.

The formation of a virtual channel under the L2F protocol is carried out according to the following stages (Fig. 3.6):

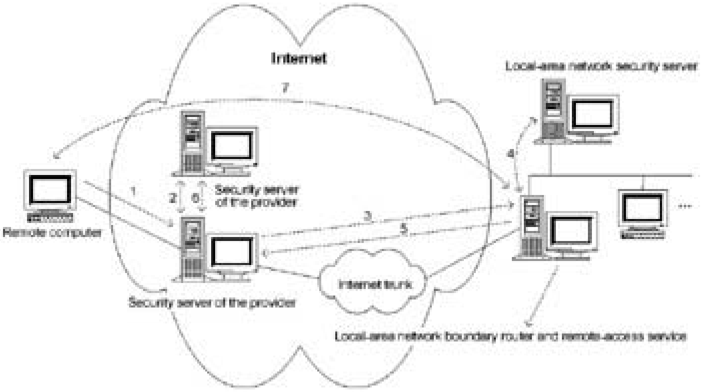

Internet
7
Local-area network security server
4
Remote computer
1
2
6
Security server of the provider
3
5
Security server of the provider
Internet trunk
...
Local-area network boundary router and remote-access service

Fig. 3.6: Interaction scheme under the L2F protocol

Stage 1. The remote user initiates connection with the provider's RAS server under the PPP protocol. Then, the provider's RAS starts the process of authentication on the basis of the protocol being used (CHAP, PAP or some other one). After initiating a connection according to the CHAP protocol, the provider's RAS server, using a "Challenge" packet, including a pseudorandom number, asks the user to enter a password. The party undergoing the test responds using a "Response" packet, containing the user ID, and also a number generated with the help of a one-way hash-function on the basis of a pseudorandom number and the user password.

Stage 2. During authentication, the provider's RAS only interprets the user ID in order to determine whether it is necessary to create a virtual channel, and, if it is, to find the address of the required LAN boundary router. The user ID and other data received at this stage of authentication are kept for transfer to the RAS of the LAN, which will carry out final user authentication. As for the CHAP protocol, it uses for authentication the value of the hash-function from the password and the pseudorandom number along with that number itself.

Whether it's necessary to create a virtual channel is determined by a database that is operated by the provider on their security server.

For a quick definition of the required LAN boundary router address, it is supposed that user IDs will be compound, for example, vlad@spectrum.widgets.org. The user name registered in the LAN is placed to the left of the ‘@' symbol, and, to the right, the domain address of the LAN boundary router, which is also the RAS, is indicated. If ISP doesn't use the procedure of defining addresses by compound identifiers, then it is necessary to maintain the database, in which user IDs are mapped to IP-addresses of the LAN perimeter routers.

Stage 3. If the virtual channel service is necessary after defining the LAN boundary router's address, a check is made to see if a tunnel to this boundary router exists. If it does not, then one is created. L2F specifications requires that the tunnel provide a "point-to-point" connection. In the first L2F protocol implementation, the UDP protocol is chosen to make such a connection.

After creating the tunnel, the provider's RAS server appoints the unused multiplexes' channel ID (Multiplex ID - MID) to the new virtual connection, generates a pseudorandom key for it and sends the message to the RAS of the LAN during a new dial-up session, including the information for authentication. The generated pseudorandom key is used to protect data packets against forgery.

Stage 4. On the basis of the information received from the provider's RAS server, the RAS of the LAN performs final user authentication and, according to the received results, accepts or rejects a new session. The authentication is performed on the basis of interaction with the LAN security server.

So, if the CHAP protocol is applied for user authentication, then, in the message about the new dial-up session, besides the service data, the user ID, the value of a hash-function from the password and a pseudorandom number, and also this pseudorandom number are transferred to the RAS of the LAN. According to the received user ID, the LAN's RAS server takes the ciphered reference value of the password from its security server database and, on the basis of this value, as well as that of the received pseudorandom number, determines the reference value of the hash-function. In cases where the reference value of the hash-function coincides with the received value, then authentication is considered to be successful. Otherwise, the user is not admitted to the LAN and the new session is rejected.

Stages 5 and 6. If the LAN's RAS server accepts a new session, it creates a virtual channel under the PPP protocol (stage 5) just as it occurs at a direct phone call to the RAS. The provider's RAS may carry out the event logging when the virtual channel is created or closed (stage 6).

Stage 7. After the virtual channel is created, the PPP frames are transferred in both directions. On the provider's LAN and RAS servers, the PPP frames for transfer on the Internet are cleared of start and stop flags, encapsulated under the L2F protocol in IP packets, and sent. Upon receiving IP packets with enclosed PPP frames on Internet boundary points, the provider's LAN and RAS servers carry out the inverse procedure. So, an emulated PPP "point-to-point" connection functions where final points are represented by a remote user network program and a program supporting PPP and the LAN's RAS server functions.

For encrypting the traffic between the endpoints of the created virtual channel, the IPSec protocol is used.

The stages involved in creating a virtual channel under the L2F protocol described above, and also the methods of using the PPTP protocol allow one to pick out the important differences between the given technology and that of a traditional remote access service through the Internet.

With a traditional remote-access service, the provider's server carries out user authentication. When using L2F technology, the provider's RAS only uses authentication to determine whether it is necessary to create a virtual channel and to find the address of a required RAS of a LAN. In both the L2F and PPTP protocols, final authentication with the LAN RAS server is performed after the provider's server initiates a connection with it.

The LAN's RAS executes the procedure of authentication with the help of standard remote-access authentication protocols through the information received from the provider's server. As the LAN server performs authentication, this network's security service, not the Internet provider, dictates the access policy. Such an approach provides a higher security level as, in this case, protection functions are assigned to the person interested in it. The Internet provider may be unreliable or negligent with its duties. Besides, L2F technology is much more flexible, since, when completely implemented, it doesn't require binding account information on remote users to Internet providers.

In terms of disadvantages of the L2F protocol, it is pertinent to note that the current version of the IP protocol (IPv4) lacks functionality, allowing one to create a cryptographic tunnel between endpoints of information interaction. A VPN channel may only be created between the provider's RAS and a LAN boundary router, and the area between a remote computer and the provider's server remains open.

3.2.3. Features of the L2TP Protocol

The L2TP (Layer-2 Tunneling Protocol) protocol was developed by the Internet Engineering Task Force (IETF), with the support of Microsoft and Cisco Systems, as the protocol of protected tunneling of PPP-traffic through general-purpose networks within an arbitrary environment. Work on this protocol was conducted on the basis of PPTP and L2F protocols, and so it combines the best features of the both projects. L2TP, unlike PPTP, is not bound to the IP protocol, and, therefore, may be used in networks with packet switching, for example, in ATM networks. Besides, L2TP provides the control of data flow-a feature lacking in L2F. The main thing, in the developers' opinion, is for the new protocol to become the generally accepted standard recognized by all manufactures.

To understand the essence of the concept of L2TP, it is necessary to represent the goals that Microsoft and Cisco laid out when they developed PPTP and L2F.

According to the aims outlined in the development of PPTP and L2F, various organizations should have the opportunity to delegate the functions of secure remote access to Internet providers. This, in turn, would allow a reduction in expenses for administration and hardware as these organizations could do without scores of modems and additional telephone channels. In both protocols, the desired aim was achieved. Both L2TP, and PPTP allow Internet providers to carry out remote sessions under the PPP protocol, using queries to LAN security servers for authentication.

Differences between L2F and PPTP are explicated by the specialization of their developers. Cisco makes hardware routers for network infrastructures, whereas Microsoft puts out operating systems. To operate with L2F for providers it is necessary that their routers and remote-access servers support this protocol. As for PPTP, it is not necessary for providers to have the means to organize tunnels as special software may form tunnels at the final points of interaction-remote computers and the LAN's RAS. Nevertheless, in comparison with PPTP, L2F has several advantages. Thus, PPTP requires application through IP-routing whereas L2F is not bound to any specific network layer protocols used for the transportation of PPP-frames.

In the L2TP hybrid protocol, not only are the best features of the PPTP and L2TP protocols combined, but new functions are also added.

As well as PPTP, the new specification does not require built-in hardware support, though equipment specially intended for it will strengthen system performance and security. In L2TP, there are a number of protection functions lacking in PPTP specification, including the capability to work with the IPsec protocol.

L2F hereditary traits are shown in the following: the L2F protocol is not bound to the IP environment and, with the same success, is capable of working in any network with packet switching, for example, in ATM networks or in frame relay networks.

The important data-flow-control function that does not admit more information than it is capable to process to the system is added to the L2TP protocol. Besides, in contrast to its predecessors, L2TP allows for several tunnels to be opened at once between the final subscribers, any of which the administrator can allocate to one application or another. These features provide safety and flexibility in tunneling, and also, essentially raise the quality of service of virtual communication channels.

In essence, the L2TP protocol represents an extension of the PPP protocol with the following functions: remote user authentication, installation of a protected virtual connection and data flow control. According to the L2TP protocol (Fig 3.7), the access concentrator must play the role of the provider's RAS server. Access concentrator implements the client part of the L2TP protocol and provides, the user access to the LAN via the Internet. The L2TP network server functioning on any platforms compatible with the PPP protocol should perform the role of the LAN's RAS.

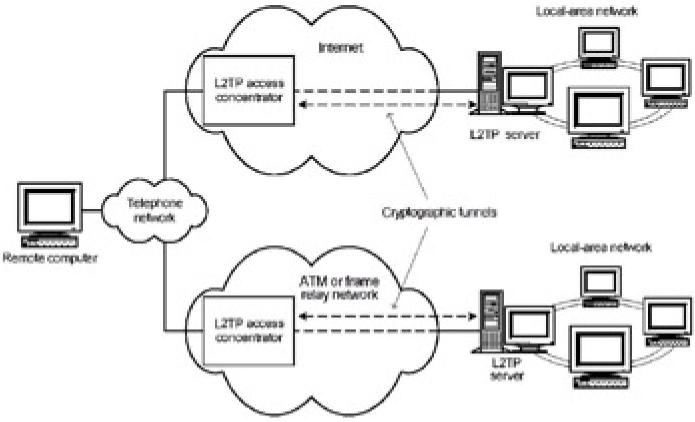


Fig. 3.7: Interaction scheme under the L2TP protocol

Similar to the PPTP and L2F protocols, the L2TP protocol provides three stages of formation of a protected virtual channel:

- Initiation of connection with the LAN's RAS
- User authentication
- Configuration of a cryptographic tunnel

To initiate a connection with the LAN's RAS (L2TP network server performs this role) the remote user contacts the access concentrator of the L2TP functioning on the Internet provider's server under the PPTP protocol. At the given stage, the L2TP access concentrator may perform user authentication on behalf of the provider. Furthermore, according to the user name, the access concentrator of the provider determines the IP address of the L2TP network server which belongs to the required LAN. Between the access concentrator of the provider and the LAN's L2TP server, the session under the L2TP protocol, or an L2TP session, is initiated.

After initiation of an L2TP session, the process of user authentication on the LAN's L2TP server takes place. For this purpose, any of the standard authentication algorithms can be used, for example, CHAP. As well as in PPTP and L2F protocols, L2TP specification does not include descriptions of authentication methods.

In instances of successful user authentication, a cryptographically protected tunnel between the provider's access concentrator and the LAN's L2TP server is created. With the help of control messages, a tunnel's various settings may be adjusted. In one tunnel, several L2TP sessions can be multiplexed. The L2TP protocol does not specify any particular methods of cryptographic protection and provides the opportunity to use various encryption standards. However, if the tunnel is formed in IP networks, the cryptographic protection must be executed according to the IPSec protocol. In this case, L2TP protocols are encapsulated in UDP-packets, which are transferred between the provider's access concentrator and the LAN's L2TP server through the IPSec tunnel. The UDP-port 1701 is enabled expressly for this purpose.

Despite its advantages, the L2TP protocol does not eliminate all problems associated with tunnel data transfer on the data-link layer. In particular, L2TP providers support is necessary. The L2TP protocol limits all traffic to the frameworks of the chosen tunnel and deprives users of the access to other parts of the Internet. The current version of the IP protocol (IPv4) does not include a cryptographically protected tunnel between the endpoints of information interaction. As well, the offered specification only provides standard encryption in IP networks during use of the IPSec protocol, which has not yet received wide enough circulation. And this may result in problems of compatibility, as each manufacturer will use proprietary technologies of cryptographic protection in L2TP products.

3.3. Protecting Virtual Tunnels on the Network Layer

3.3.1. The IPSec Architecture

Creating protected tunnels on the data-link layer of the OSI model provides independence from network-layer protocols, but, at the same time, it results in complications with configuring and supporting virtual communication channels. Besides this, when organizing the protection of informational exchange on the data-link layer, the set of implemented security functions decreases, and it becomes more complicated to manage cryptographic keys. One can achieve an optimal balance between protection transparency and the quality of its operation by creating protected virtual channels on the OSI model's network layer. This approach makes virtual channels transparent for applications since in this case, the network protocol will always be implemented between the network layer and application. Thus, there will be no need to redesign the application. On the other hand, one will be able to implement traffic protection and key management functions more completely on the network layer, as that is where packet routing takes place.

Standard methods of information protection on the OSI model's network layer for IP networks (which is the most common type of public networks) are determined by the IPSec (Internet Protocol Security) protocol. This protocol makes up the key part of the newest version of the IP protocol (IPv6). It is also applicable to the current version (IPv4). For IPv4, IPsec support is desirable, while it is mandatory for IPv6. IPsec ensures data-source authentication, cryptographic protection of the packets being transmitted-checking them for integrity and authenticity after reception-and it provides limited protection against traffic analysis. Standard IPSec protection functions can and must be employed by higher-level protocols, particularly by management, control, configuration and routing protocols.

According to the IPSec protocol, the security tools' architecture has three layers (Fig. 3.8).

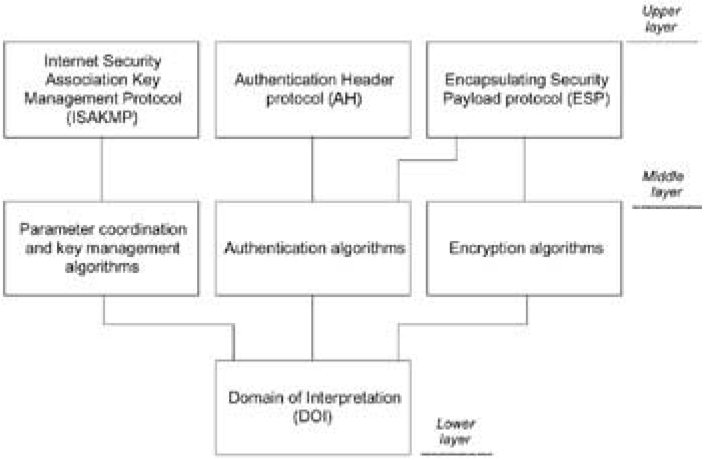

Upper layer
Internet Security Association Key Management Protocol (ISAKMP)
Authentication Header protocol (AH)
Encapsulating Security Payload protocol (ESP)
Middle layer
Parameter coordination and key management algorithms
Authentication algorithms
Encryption algorithms
Domain of Interpretation (DOI)
Lower layer

Fig. 3.8: The IPSec architecture

The following protocols exist on the highest layer of the IPSec architecture:

- A protocol for coordinating virtual channel parameters and key management-Internet Security Association Key Management Protocol (ISAKMP)-ensuring general management of a protected virtual connection, including coordination of the cryptographic algorithms used and key information distribution
- The Authentication Header (AH) protocol, which ensures data-source authentication, checking data for integrity and authenticity after reception and protection against enforced message duplication
- The Encapsulating Security Payload (ESP) protocol, which encrypts the transmitted message packets and fulfills all AH protocol functions.

IPSec uses two different protocols for virtual tunnel protection (AH and ESP), on account of the common practice of export/import limitations for cryptographic tools adopted by most countries. Each of these protocols can be used both independently and in combination with another one. The ESP and AH protocols have been registered by the Internet Address Naming Authority (IANA) and included in the protocol registry under the registration numbers 50 and 51, respectively.

Authentication and encryption algorithms used in the AH and ESP protocols make up the intermediate layer of the IPSec architecture. This layer also includes protocol coordination and key management algorithms used in the ISAKMP protocol. Higherlevel virtual tunnel protection protocols (AH and ESP) do not depend on specific cryptographic algorithms. This means that it is possible to use any authentication method, any key type (symmetric and asymmetric) and encryption and key distribution algorithms. For example, any country might use proprietary encryption algorithms, corresponding to national standards.

Algorithmic independence of the AH and ESP protocols requires the interacting parties to coordinate a set of algorithms and their parameters before communicating. This function is delegated to the ISAKMP protocol, requiring the communicating parties to coordinate a common security context (Security Association, SA) before forming a virtual protected channel. After accomplishing this, the parties can use elements of that context, such as algorithms and keys.

The IPSec architecture is based upon the so-called Domain of Interpretation (DOI), which represents a database that stores information on all protocols and algorithms used in IPSec, along with their parameters, protocol identifiers, etc. The IPSec architecture is fully open, flexible and versatile. Any protocols and algorithms can be employed in IPSec, even those that were not originally designed for this architecture. This is why the DOI is necessary, since it ensures the coordinated operation of all protocols and algorithms included in the IPSec architecture. In order to be able to use proprietary authentication and encryption algorithms in the AH and ESP protocols, these algorithms need to be registered in the DOI.

Currently, there are two registered authentication algorithms for the AH and ESP protocols-HMAC-MD5 (Hashed Message Authentication Code-Message Digest-version 5) and HMAC-SHAI (Hashed Message Authentication Code-Secure Hash Algorithm-version 1). These are authentication algorithms, each with a secret key. If only the transmitting and receiving parties have knowledge of the secret key, data-source authentication and integrity of the packets exchanged by the communicating parties will be guaranteed. To ensure hardware and software compatibility upon the initial stage of IPSec protocol implementation, one of the registered authentication algorithms must be used by default. The HMAC-MD5 algorithm is designated for this role.

There are seven registered encryption algorithms for the ESP protocol. The DES (Data Encryption Standard) algorithm and HMAC-MD5, are used by default and are needed for IPSec compatibility. As an alternative to DES, one can use the Triple DES, CAST-128, RC5, IDEA, Blowfish and ARCFour algorithms.

AH and ESP support the following two IPSec modes:

- The tunnel mode, in which IP packets are fully encrypted, including their headers
- The transport mode, in which only an IP packet content is encrypted

The tunnel mode is the main mode. When operating in this mode, each normal IP packet is encrypted and enclosed in a protected IPSec envelope, which is then encapsulated into another IP packet. The tunnel mode is usually implemented on the dedicated security gateways, which can be routers or firewalls. IPSec protected tunnels are formed between such gateways. Before transmitting data via such a channel, original IP packets of the transmitting LAN are encapsulated into the IPSec-protected IP packets. When protected packets are delivered to the destination point, they are "unpacked", and the resulting IP packets are transmitted to computers located within the receiving LAN, according to the standard rules of that particular network. IP-packet tunneling is fully transparent for the hosts in local-area networks connected by tunnels. The tunnel mode can be used for supporting remote and mobile users in termi-nal systems. In this case, special software implementing the IPSec tunnel mode must be installed on their computers.

In the transport mode, only the contents of the original IP packets are encrypted and enclosed in a protected IPSec envelope. The original IP header is then added to the resulting envelope. Consequently, the IPSec header is transmitted between network (IP) and transport (TCP or UDP) headers of the normal IP packet in transport mode.

Transport mode is faster than tunnel mode. It has been specially designed for use with terminal systems. This mode can be used for supporting remote and mobile users, as well as for protecting information flows within local-area networks. Furthermore, transport mode can also be used on gateways for protecting internal connections between gateways. This ensures efficient and adequate protection for the process of remotely controlling routers, ATM switches, firewalls and other key components of the network infrastructure. Operation in transport mode influences all systems included in the protected group and, in most cases, network applications must be redesigned.

3.3.2. Authentication Header Protocol

The Authentication Header (AH) protocol ensures IP-packet integrity and data-source authentication, as well as protection against packet duplication. This protocol provides full protection against forgery or accidental corruption of the transmitted IP packets, including the data of higher-level protocols. The degree of protection provided depends on the operational mode-tunnel or transport.

All fields of the IP header are protected in tunnel mode (Fig. 3.9). Each normal IP packet is completely enclosed in an IPSec envelope, which then is encapsulated into another IP packet. In a protected IP packet, a nested (original) IP header contains the packet's target address, while the external IP header contains the tunnel endpoint's address.

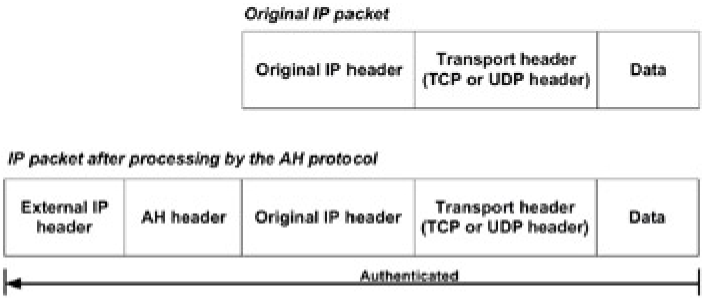

Original IP packet
Original IP header
Transport header (TCP or UDP header)
Data
IP packet after processing by the AH protocol
External IP header
AH header
Original IP header
Transport header (TCP or UDP header)
Data
Authenticated

Fig. 3.9: The IP packet before and after processing by an AH protocol in tunnel mode

When using the AH protocol in transport mode, only those fields of IP headers whose contents change unpredictably along the delivery route remain unprotected. For IPv4, one such field is the Time To Live (TTL) field, specifying the life-time of the IP packet. Another example of such a field is the Type of Service field, which defines the packet servicing type. For IPv6, protection is not applied to the Prio. and Flow Label fields, similar to the Type of Service field in IPv4, nor on the Hop Limit field specifying the maximum number of intermediate systems along the packet route (as its name implies, each router decreases this number by 1 when it transmits the packet). In transport mode, only the data contents of the protected IP packet are enclosed in the IPSec envelope, and the original IP header is then added to the resulting envelope (Fig. 3.10).

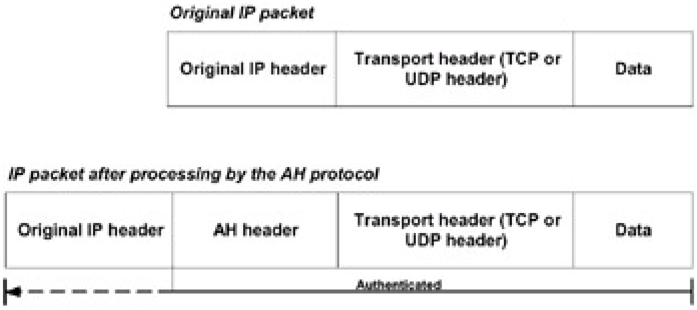
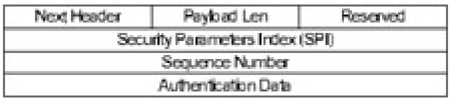

Original IP packet
Original IP header
Transport header (TCP or UDP header)
Data
IP packet after processing by the AH protocol
Original IP header
AH header
Transport header (TCP or UDP header)
Data
Authenticated

Fig. 3.10: The IP packet before and after processing by the AH protocol in transport mode

Fig. 3.11 shows the format of the AH header. Let us describe the meaning of the fields contained in this header.



Next Header	Payload Len	Reserved
Security Parameters Index (SPI)		
Sequence Number		
Authentication Data		

Fig. 3.11: The AH header format

- Next Header-a single-byte field containing the code of the next header type enclosed in the IPSec packet. For example, if the IPSec packet contains the TCP packet, this field will contain the number 6-the TCP-protocol code.
- Payload Length-the length of the AH header, in 32-bit words minus 2.
- SPI-a 32-bit index of the security parameters, determining the SA (Security Association) structure, which contains all parameters of the IPSec tunnel, including the cryptographic algorithm and encryption key types.
- Sequence Number-an unsigned 32-bit integer, incremented by 1 after transmitting each IP packet protected according to the AH protocol. This field ensures protection against reproduction of IP packets that were transmitted earlier. The sender must maintain this counter. When creating each protected session of information exchange via the IPSec tunnel, both parties reset their counters to zero, and then they begin to increase their counters in a coordinated way.
- Authentication Data-a variable-length field containing information used for packet authentication. This information is known as the MAC-code (Message Authentication Code). This term is synonymous with "digital signature", "hash value" and "cryptographic checksum" (Integrity Check Value, ICV). The method used to calculate the value of this field depends on the authentication algorithm.

Various algorithms can be used to calculate the contents of the Authentication Data field. Currently, one must provide support for HMAC-MD5 and HMAC-SHA1 based on the usage of one-way hash functions with secret keys. Secret keys are generated in accordance with the ISAKMP protocol.

Thus, regardless of the operation mode, the AH protocol provides protective measures against attacks aimed at data corruption or counterfeiting. Using this protocol, each packet is authenticated which renders the network-sniffing software inefficient. Despite the fact that the IP header is located outside the protected IPSec envelope, the AH protocol ensures authentication of both the IP packet contents and headers. However, bear in mind that authentication according to the AH protocol does not allow any manipulations with the main fields of an IP packet during packet transmission. Hence, the AH protocol cannot be used in an environment using Network Address Translation (NAT), since this capability is required for NAT operation.

3.3.3. The Encapsulating Security Payload Protocol

The Encapsulating Security Payload (ESP) protocol provides the following functions:

- IP-packet encryption
- Limited protection against traffic analysis
- Creating and checking digital signatures of IP packets, in order to protect them from corruption and authenticity violations
- Protection against IP-packet duplication

The list of functions provided above shows that the ESP protocol provides a wider range of security functions than the AH protocol. The ESP protocol ensures data confidentiality, and supports all AH functions for protecting encrypted data flow from counterfeit, duplication or accidental corruption.

In general, all security functions supported by the ESP protocol can be limited to authentication, which can also be provided by the AH protocol, and encryption of the transmitted IP packets. The IPSec specification allows the ESP protocol to be used for encryption of IP packets without using authentication functions. Furthermore, when performing AH protocol functions, selective encryption is allowed. Thus, according to the ESP protocol, authentication and encryption functions can be used either together or separately. When performing encryption without authentication, it is possible to use the NAT mechanism, since, in this case, addresses in IP-packet headers can be modified.

Independent of the ESP operation mode, its header is formed as an encapsulating envelope for encrypted contents (Fig. 3.12).

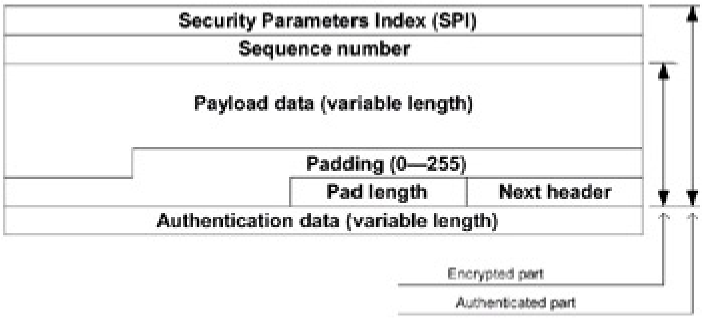

Security Parameters Index (SPI)
Sequence number
Payload data (variable length)
Padding (0—255)
Pad length
Next header
Authentication data (variable length)
Encrypted part
Authenticated part

Fig. 3.12: The ESP packet header

The SPI, Sequence Number, Next Header or Authentication Data fields are the same as in the AH protocol. The Authentication Data field is placed into the ESP header only when authentication is enabled. The variable length Payload Data field includes an encapsulated packet which is encrypted along with the Padding, Pad Length and Next Header field. The Padding field represents bytes added in order to ensure the evenness of the encapsulated packet lengths and block size of the encryption algorithm. The Pad Length field contains the length of the Padding area. When the ESP protocol operates in tunnel mode, it encapsulates the whole IP packet (Fig. 3.13), and, when it operates in transport mode, only the contents of the IP packet are encapsulated (i.e., the original TCP or UDP packet). The algorithm, according to which the ESP protocol is applied to the outbound IP packets, comprises the following steps:

- The packet to be encapsulated is copied to the clipboard.
- The copied packet is complemented by the padding bytes (the Padding field), their number (the Pad Length field) and the first header type of the packet to be encapsulated (the Next Header field). The Padding field must ensure that the Next Header field is aligned to the end of the 32-bit word, and that the size of the packet satisfies the encryption-algorithm requirements.
- The current contents of the clipboard are then encrypted.
- The SPI and Sequence Number fields are added with the appropriate contents to the starting position of the clipboard contents.
- The complemented contents of the clipboard are processed according to the authentication algorithm used. Upon completion, the Authentication Data field is added to the ending position of the data.
- The resulting IP packet is then created by means of adding an appropriate IP header to the starting position of the clipboard data.
- Thus, according to the ESP protocol, if both encryption and authentication are enabled, the encrypted packet will be authenticated. For incoming packets, all actions are performed in reverse order, i.e., authentication takes place first. This allows one to avoid extra resource consumption for decrypting fraudulent packets, which partially provides protection against DoS attacks.

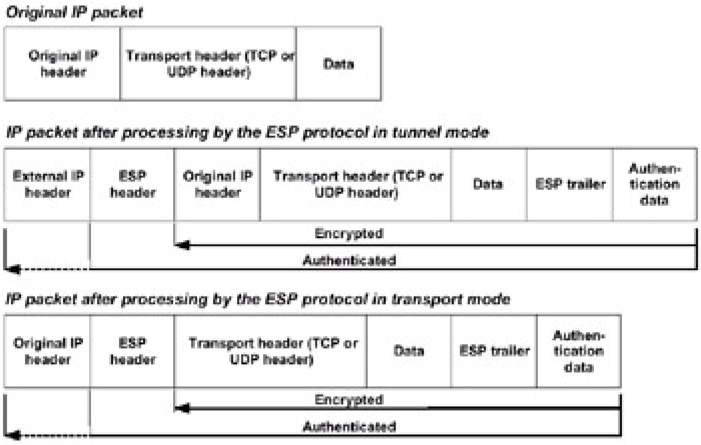

Original IP packet
Original IP header
Transport header (TCP or UDP header)
Data
IP packet after processing by the ESP protocol in tunnel mode
External IP header
ESP header
Original IP header
Transport header (TCP or UDP header)
Data
ESP trailer
Authen-tication data
Encrypted
Authenticated
IP packet after processing by the ESP protocol in transport mode
Original IP header
ESP header
Transport header (TCP or UDP header)
Data
ESP trailer
Authen-tication data
Encrypted
Authenticated

Fig. 3.13: The IP packet before and after applying the ESP protocol

When the ESP protocol is used in tunnel mode, each original IP packet is encrypted and then enclosed in an IPSec envelope, which is then encapsulated into another IP packet. In a protected IP packet, the nested (original) IP header is located within the encrypted part and contains the packet's target address. The external IP header, on the other hand, contains the tunnel's endpoint's address. When ESP is used in transport mode, only the contents of the original IP packet are enclosed in the IPSec packet, and the original IP header is added to the resulting envelope.

Currently, the IPSec specification for IP packet encryption according to the ESP protocol requires the DES-CBC (Data Encryption Standard in Cipher Block Chaining mode) algorithm support. This encryption algorithm is used in ESP by default, and IPSec compatibility must be provided. Other algorithms, such as Triple DES, CAST-128, RC5, IDEA, Blowfish and ARCFour can be used as alternatives to DES-CBC.

The CAST algorithm (RFC 2144) is considered to be as high-grade as the Triple DES algorithm with a 128-bit key. Besides this, CAST is faster than DES. The RC5 algorithm (RFC 2040) is a data-stream encryption algorithm which uses a variable-length key. It is commonly believed that the resistance of the RC5 algorithm depends on its key length (keys with a length of up to 256 bits can be used). The IDEA (International Data Encryption Algorithm) algorithm is thought to be the "fast" equivalent of the Triple DES algorithm. Another example of a variable-length key algorithm is the Blowfish algorithm, which is also sufficiently resistant. The ARCFour algorithm represents the public version of the RC4 algorithm.

Selection of the encryption algorithm, with the exception of DES (which is required) depends entirely on the developer. Being able to select the encryption algorithm provides an important advantage, since intruders will have to first determine which cipher they have to break before actually doing so. Along with the necessity to identify the keys, this significantly reduces the risk of compromising the cryptographic system.

AH and ESP protocols can be used in any combination. If the transport mode is used, then, similar to the processing by ESP, the AH protocol must be applied after ESP when authentication follows encryption. In tunnel mode, AH and ESP protocols are applied to different encapsulated packets, and, furthermore, this mode allows multiple-level nesting of tunnels with different starting and ending points. On account of this, tunnel mode provides a larger number of combinations when using AH and ESP protocols.

3.3.4. Protected Tunnel Management

It is impossible to create and maintain protected virtual channels without implementing control and management functions. In the IPSec specification, these functions fall into two groups:

- General management functions based on the use of the Security Policy Database (SPD)
- Control functions oriented towards coordinating tunnel parameters and creating the Security Association (SA), which describes the common security parameters of a VPN tunnel

In accordance with the general management functions, all incoming and outgoing IP packets must be compared to the ordered set of security-policy rules, which must be defined for the following objects:

- For each network interface with IPSec tools enabled
- For each inbound and outbound data flow

According to IPsec specifications, the security policy must involve independent processing of IP packets at the OSI model's network level, according to contemporary filtering technology. Accordingly, special tools must be available to administer a security-policy database. These tools must be similar to the tools for firewall rules database administration. The Security Policy Database (SPD) is an ordered set of rules, each including a set of selectors and valid security contexts. Selectors serve for packet selection, while contexts specify required processing.

When comparing data against an ordered set of rules, selectors that specify the set of the analyzed fields of the network and higher protocol layers are processed first. All IPSec implementations must support IP packet filtering based on the analysis of the following data items:

- Source and target IP addresses. These addresses might be both individual and group addresses (when specifying address ranges, wildcard characters must be allowed).
- A user or host name (in DNS or X.500 formats).
- The numbers of the transport protocols used.
- The source and target port numbers (ranges and wildcard characters are allowed).

The first applicable rule from the security-policy database (SPD) defines any further actions in respect to the packet:

- The packet is discarded.
- The packet is processed without applying IPSec tools.
- The packet is processed using IPsec tools, taking into account the security context associated with that rule.

When the user decides to process the packet using IPSec tools, the security associations of the chosen rule need to be analyzed. Each Security Association (SA) describes parameters of the valid IPSec connection, including types of cryptographic algorithms, encryption keys and other service information. If the rule refers to a non-existent SA, this SA must be created in order to form a protected IPSec tunnel. In this case, automatic SA and key management must be supported.

When creating an SA, communicating parties must authenticate each other and coordinate tunnel parameters, including the types of cryptographic algorithms, encryption keys and other service information. To solve this problem, IPSec uses a special protocol, ISAKMP, which provides general management of the VNP connection. ISAKMP describes basic authentication technology, procedures of exchanging keys and coordinating other parameters of the IPSec tunnel. However, ISAKMP does not contain specific algorithms for a cryptographic key exchange. As a result, other protocols can be used to exchange the keys. Currently, the Oakley protocol, based on the Diffie-Hellman algorithm, has been adopted for this purpose. The combination of ISAKMP and Oakley protocols is known as ISAKMP/Oakley.

According to the ISAKMP protocol, coordination of secure communication parameters is required both for the creation of an IPSec tunnel, and for creating protected one-way connections within that tunnel. The tunnel's global parameters make up the management context, and make up the above-mentioned Security Association, also called the protocol context. The so-called Security Parameters Index (SPI) is used to identify each SA. This index is included in protected IPSec packet headers, so that the recipient will be able to decrypt and/or authenticate them correctly, using the specified Security Association.

Cryptographic keys for each protected one-way connection (included in the Security Association of this connection) are generated based on the keys developed within the management context. This operation is performed taking into account authentication and encryption algorithms used in AH and ESP.

According to the IPSec specification, processing of inbound and outbound traffic is not symmetrical. For outgoing packets, IPSec browses the SPD database, finds an appropriate rule, extracts SAs associated with this rule and applies the required security functions. The SPI value is present in all incoming packets for each protected protocol, which explicitly identifies the SA. In this case, there is no need to search the security-policy database, since the security policy was already considered when an appropriate Security Association was formulated.

How flexible a security policy is when using the IPSec protocol depends on the selectors and SAs used in the rules. For example, when only IP addresses are used in selectors, the pair of interacting computers can use one set of SAs. If TCP and UDP port numbers are analyzed, then the set of security associations can be specific for each application. Accordingly, two security gateways can organize a common tunnel for all computers they serve, but, on the other hand, they can delimit the tunnel by means of organizing different contexts by pairs of computers or even applications.

3.4. Creating a Virtual Private Network on the Session Level

3.4.1. Specific Tunneling Features

The session layer is the highest layer of the OSI model at which it is possible to create VPN channels. When creating virtual private networks on the session layer, it is possible to achieve high-quality performance and functionality parameters for information exchange, access-control reliability and ease of administration. Protocols for creating VPN channels at the session layer are transparent for application layer security protocols and for higher-level protocols providing various services (such as HTTP, FTP, POP3, SMTP, NNTP, and so on). However, implementation at the session layer is directly dependent on the applications that implement higher-level protocols. Because of this, in most cases, implementation of security protocols at this layer requires a modification of the code of higher-level network applications.

Since the session layer of the OSI model is responsible for establishing and managing logical connections, security-protocol implementation at this level enables the user to employ proxy software. Proxy software checks if requested connections are allowed, and provides other functions of securing internetwork communications. In general cases, proxy software, which is usually employed in firewalls, can perform the following functions:

- User identification and authentication
- Cryptographic protection of transmitted data
- Access control for resources located within the internal LAN
- Access control for resources located in the external network
- Filtering and converting message flow, for example, dynamic scanning for viruses and transparent information encryption
- Translation of internal network addresses for outgoing packets
- Event logging and reacting to predefined events
- Caching data requested from the external network

Thus, when creating virtual networks on the session layer, in addition to cryptographic protection (including authentication), it is also possible to implement proxy technologies.

For cryptographic protection of information exchange on the session layer, the Netscape Communications' SSL/TLS (Secure Sockets Layer/Transport Layer Security) protocol, is currently the most popular. From this point on, we'll refer to this protocol as simply SSL. The SOCKS protocol has been adopted by IETF (Internet Engineering Task Force) as the standard for performing proxy functions on the session layer.

3.4.2. The SSL Protocol

The Secure Sockets Layer (SSL) protocol, originally oriented towards securing information exchange between clients and servers within computer networks, is the industry session-layer protocol, and it uses cryptographic methods for information protection. The transmitted data is kept confidential by encryption, while creating and checking digital signatures guarantees authentication and information integrity.

The core of the SSL protocol can be found in its combination of symmetric and asymmetric cryptographic systems. The range of asymmetric encryption algorithms includes RSA (developed by RSA Data Security, Inc.) and the Diffie-Hellman algorithm. For calculating hash functions, one can use such standards as MD5 and SHA-1. Supported symmetric encryption algorithms are RC2, RC4, DES, and TripleDES. It is also the case that the set of cryptographic algorithms can be extended in SSL_V3. Open-key digital certificates are used for authentication and cryptographic protection of the symmetric encryption key. Special certification centers verify users' digital certificates (i.e. the client and server) by way of a digital signature. SSL supports digital certificates that satisfy the generally adopted X.509 standard.

The SSL protocol was originally developed by Netscape, and then supported by most leading software manufacturers. Thanks to its advantages, SSL has practically pushed all competing higher-level security protocols such as SHTTP (Secure HTTP) out of the market, and it has become the generally accepted security standard for protecting information on the Internet and intranets. SSL specifications were suggested for adoption as the official Internet security standard, but were not granted this status, due to some formalities. Furthermore, it is quite possible that in the long run, SSL will be officially accepted by IETF as the official standard, since it has already become the de facto industry standard, adopted and developed by most leading manufacturers, despite the IETF's reticence. The latest version of SSL is version 3.0, which will be convered in detail later on in this chapter.

The client part of the SSL is implemented in most popular Web browsers, including Netscape Navigator and Internet Explorer. The server part of the SSL is implemented in most WWW servers, both commercial and shareware, including server applications from IBM, Netscape, Microsoft, Spyglass and Open Market.

According to the SSL protocol, cryptographic tunnels are created between the endpoints of the virtual network (Fig. 3.14). Client and servers running on those computers that represent endpoints of the tunnel are initiators of each protected tunnel. SSL makes provisions for two stages of client/server interaction when creating and maintaining the protected connection:

- Establishing an SSL session
- Protected communication

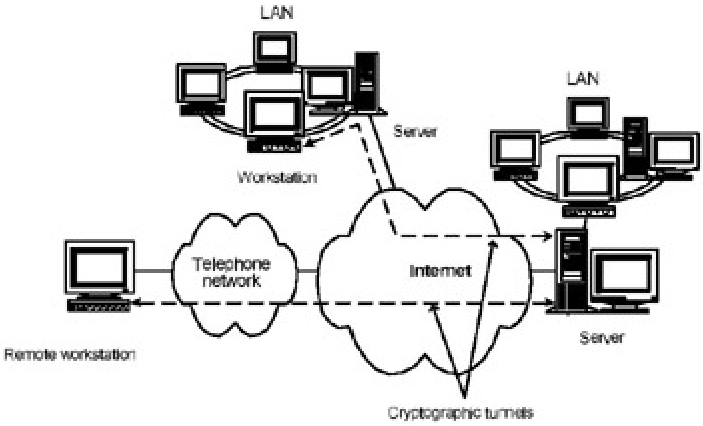


Fig. 3.14: Cryptographic tunnels based on the SSL protocol

The procedure of establishing an SSL session, also known as the "handshake" procedure, is performed before direct protection of an information exchange, in accordance with the Handshake Protocol implemented within SSL. During this procedure, the following tasks are accomplished:

- Client and server authentication
- Coordination of cryptographic and compression algorithms that will be used
- Creation of a common secret master key
- Generation of secret session keys on the basis of a master key

SSL implementations most often use the RSA algorithm developed by RSA Data Security, Inc., in order to perform authentication.

Unambiguous and certain correspondence between public keys and their owners is established using digital certificates, provided by special certificate centers. A certificate is the data block containing the following information:

- The name of the certificate center
- The name of the certificate owner
- The certificate owner's public key

- The period of the certificate's validity
- The identifier and parameters of the cryptographic algorithm that will be used when processing the certificate
- The certificate center's digital signature

The certificate center's digital signature ensure reliable and explicit correspondence between the public key and its owner. The certificate center performs "notarization" functions when confirming the public keys' authenticity, enabling their owners to use protected interaction services without actually meeting in person. Since the certificate center must be a trusted entity on the part of all paritcipants of the protected communications, it must satisfy quite stringent requirements in verifying the public keys' authenticity. One such center is VeriSign, founded by RSA Data Security, Inc., in cooperation with other companies, including Visa, IBM, Netscape, Microsoft and Oracle.

The third version of the SSL protocol supports three authentication modes:

- Mutual authentication of the interacting parties
- One-way authentication of a server without client authentication
- Anonymous mode

When using the latter mode, neither of the interacting parties is protected from attacks oriented towards substitution of one of the communication participants. In this mode, only information exchange is protected without any guarantees in relation to the interacting parties' authenticity.

In the mode of one-way server authentication without client authentication, establishing an SSL session between a client and server includes the following steps:

1. The client sends a request to the server to establish a protected connection. This request contains several requisite connection parameters, including:
    - The current date and time
    - The random sequence (RAND_CL)
    - A set of symmetric encryption and hash-function calculation algorithms supported by the client
    - A set of supported compression algorithms, etc.
2. The server processes the client's request and passes the coordinated set of parameters to the client:
    - The identifier of the SSL session
    - Specific cryptographic algorithms selected from a set of algorithms suggested by the client (if, for some reason, the suggested algorithms or their parameters don't meet the server's requirements, the session will be discontinued)
    - The server's certificate, verified by the certificate center's digital signature
    - Random sequence (RAND_SERV)
3. The client checks the certificate received from the server using the public key supplied by the certificate center. If the result is negative, the session is closed. Otherwise, the client performs the following actions:

    - Generates a random 48-byte sequence (Pre_MasterSecret using the server's public key received within the server's certificate, and sends it to the server.
    - Using coordinated hash algorithms, forms a common secret master key (MasterSecret). Parameters used in the course of this operation include the Pre_MasterSecret sequence, a RAND_CL random sequence sent to the server and the RAND_SERV random sequence received from the server.
    - Using MasterSecret, the client calculates cryptographic parameters of the SSL session, forms common session secret keys for symmetric encryption and hash-function calculation.
    - Switches to a protected communication mode.

4. The server decrypts the Pre_MasterSecret sequence obtained from the client with its own secret key and performs the same operations as the client, based on the value obtained:
    - Using coordinated hash algorithms, forms a common secret master key (MasterSecret) based on the following parameters: the Pre_MasterSecret, RAND_SERV sequence sent to the client and RAND_CL random sequence obtained from the client
    - Uses MasterSecret to calculate cryptographic parameters of the SSL session, and forms common session secret keys for symmetric encryption and hash-function calculation
    - Switches to a protected communication mode

Since both the client and server used the same source data (coordinated algorithms, the common secret random sequence Pre_MasterSecret, and common RAND_CL/RAND_SERV random sequences), it is obvious that they will produce the same secret session keys as a result of this operation. To check the identity of the SSL session parameters, the client and server exchange test messages, of whose contents they have knowledge:

- The client forms the message from the messages sent to the server during step 1 and the messages received from the server during step 2, introducing some randomizing factor in the form of the MasterSecret sequence unique for the current session. Then the client forms the code for checking the message authenticity (MAC), encrypts the message according to the common session secret key, and sends it to the server.
- The server then forms a message from the messages sent to the client during step 2, those received from the client during step 1, and the MasterSecret sequence; forms the code for checking the message authenitcity (MAC); encrypts the message using the common session key; and sends it to the client.
- If both parties succesfully decrypt and verify test messages, then this means the SSL session has been established, and both parties switch to a standard secure-connection mode.

In the course of a secure connection with established cryptographic parameters of the SSL session, the following actions take place:

- When transmitting each message, both parties form the MAC code for subsequent message integrity verification, then encrypt both the original message and MAC code using the secret session key.
- When receiving a message, each party decrypts it and performs an integrity check (i.e. calculates the current MAC code and checks it against the MAC code received with the message). If the check produces a negative result, the SSL session will be terminated.

Despite the fact, that SSL protocol is supported by server and client software from most leading manufacturers, practically all existing products supporting SSL are implemented in the United States. Therefore, due to export limitations, outside the U.S., only limited versions of these products are available (with session-key length up to 40 bits for symmetric encryption algorithms, and 512 bits for RSA algorithms). These limitations are also applicable to cryptographic modules of the most popular Web browsers, including Netscape Navigator and Internet Explorer. Notice that the latest export releases of these products support some algorithms with sufficient key length, but, again, with specific limitations. Also, Certificate Centers, such as VeriSign, provide strong cryptography services only to the licensed banking servers.

To solve the above-described problem, it makes sense to use software products complementing client and server software from leading software manufacturers with SSL support free from export limitations for key length, as well as for the capability of using national cryptographic standards and certificate servers. The example of such system is the Fortify freeware program, developed by Farrel McKay (http://www.fortify.net).

3.4.3 The SOCKS protocol

The SOCKS protocol was developed in 1990 by David Koblas to organize mediation during interaction between client/server applications on the OSI model's session layer. Initially, this protocol was only developed to redirect queries to servers on the part of client applications, and also for the return of the received answers to these applications. However, even merely the redirection of inquiries and answers between client/server applications allows one to implement the network IP addresses' translation function (Network Address Translation, or NAT). When replacing IP addresses of outgoing packets with one gateway IP address, the topology of the internal network is completely hidden from external users, thus complicating attempts of unauthorized access. Besides enhancing safety, network-address translation allows the expansion of network-internal address space at the expense of support of an addressing system that is not coordinated with external network addressing.

On the basis of the SOCKS protocol, any other proxy functions can be implemented in order to protect internetwork communications. For example, SOCKS can be used to control data-flow directions and for discretionary access control, depending on user and/or information attributes. Using this protocol to implement protection functions is efficient on account of its orientation to the OSI model's session layer. In comparison with application layer proxies, on the session layer, higher speed and independence of higher-level protocols (HTTP, POP3, SMTP, NNTP and others) can be reached. Besides, the SOCKS protocol is not tied to the IP protocol, and also does not depend on operating systems. For example, for interaction between a client application and a proxy, it is possible to use the IPX protocol.

Due to the SOCKS protocol, virtual private networks and firewalls can organize secure interaction between different networks. Furthermore, SOCKS lets these systems operate under safe control on the basis of a unified strategy. One should note that if VPN channels are formed between different pairs of interacting parties through the SOCKS protocol, protected tunnels for each application and session can be formed separately on the basis of channel and network-layer protocols.

According to the SOCKS protocol, the SOCKS server, which is expedient to install on a network gateway, and the SOCKS client that is installed on every user computer, are recoginized. The SOCKS server provides interaction with any application server on behalf of client application appropriate to this server. The SOCKS client is intended to intercept all inquireis to the application server on the part of the client, and to transfer them to the SOCKS server. As the SOCKS server knows about the traffic at the level of a session, it can execute careful control, for example, to block the work of specific user applications, if they lack the necessary data-exchange authorities.

Form the moment of development, the 4th and 5th versions (v.4 and v.5) of this protocol have been widely used. Nowadays, SOCKS v.5 is approved by IETF as the Internet standard and is included in Request For Comments (RFC 1928)-a series of documents in which all members of the Internet community take part.

The generalized scheme under the SOCKS v.4 protocol takes the following form:

- Request from a client application to establish a connection with any network application server. The SOCKS client installed on the same computer intercepts this request.
- The SOCKS client connects to the SOCKS server and transfers the application server request to it.
- The SOCKS server connects to the requested application server.
- The application client and application server cooperate using a connections train, in which the SOCKS server simply relays the data.

In SOCKS server v.4 implementation, except for IP-address translation, other proxy functions of network-interaction protection can be provided. However, in the SOCKS v.4 protocol, such functions are not specified.

The SOCKS v.5 protocol shows significant development of the fourth version, and offers the following additional features:

1. The SOCKS client can transfer to the SOCKS server not only the IP address of the computer with which it is necessary to establish a connection, but also its DNS name. The SOCKS server will receive the IP address by the DNS name itself. Thus, in local-area networks using SOCKS v.5, it is possible to do without a local DNS server, whose availability the SOCKS v.4 protocol required.

2. In SOCKS v.5, in addition to TCP protocols, UDP protocols are also supported. Together, these protocols cover practically every set of application protocols. Of widely used programs, only the PING and TRACERT diagnostic utilities use the ICMP protocol, and cannot work through TCP and UDP.
3. User authentication, which the SOCKS client's address is stipulated on behalf of. The SOCKS server can coordinate the authentication mode with the SOCKS client. Authentication makes access differentiation to computer resources possible. Two-way authentication is also possible, i.e., the user, in turn, can make sure that they have connected to the necessary SOCKS server.
4. SOCKS v.5 permits the use of current addressing schemes according to the IPv.4 protocol, and also the prospective ones stipulated by IPv.6.

The generalized communication chart under the SOCKS v.5 protocol looks as follows:

1. The application-client request, wishing to establish a connection with any application server in a network, is intercepted by the SOCKS client installed on the same computer.
2. The SOCKS client connects to the SOCKS server and gives all authentication methods ids it supports.
3. The SOCKS server decides which authentication method to take advantage of, and, if the SOCKS server does not support any of the authentication methods offered by the user, the connection is terminated.
4. With the support of any offered authentication methods, the SOCKS server, according to the chosen method, authenticates the user on whose behalf the SOCKS client acts. If the authentication is unsuccessful, the SOCKS server terminates the connection.
5. After successful authentication, the SOCKS client transfers the DNS name or IP address of the required application server in the network to the SOCKS server, and, then, the SOCKS server makes a decision about establishing a connection with this application server on the basis of available access-differentiation rules.
6. In cases where a connection is established, the application client and the application server interact on a series of connections in which the SOCKS server relays the data, and also may carry out network-protection proxy functions. For example, if, during authentication, the SOCKS client and the SOCKS server have exchanged session keys, all traffic between them can be ciphered.

User authentication carried out by the SOCKS server may be based on X.509 digital format certificates and passwords. For encryption of traffic between the SOCKS client and SOCKS server, any protocols oriented toward session and lower OSI-model layers can be used. The SSL protocol will provide the maximal functionality of cryptographic protection. Besides user authentication, IP-address translation and cryptographic protection of traffic, the SOCKS server can carry out such functions as:

- Discretionary access control to internal network resources
- Discretionary access control to external network resources
- Data-flow filtering, for example, dynamic scanning for viruses

- Event logging and reacting to predefined events
- Caching of the data requested from the external network

SOCKS clients that intercept client application inquiries and interact with the SOCKS server can be built into multi-purpose client programs. For example, popular Web navigators, such as Netscape Navigator and Microsoft Internet Explorer, provide built-in SOCKS support. There are also specific programs, named "SOCKSifiers", supplementing client applications' support of the SOCKS protocol. Examples of such programs are NEC SocksCap and HummingBird SOCKS Client. At installation, the "SOCKSifier" is implanted between user applications and the communication-protocol stack. Furthermore, while working, it intercepts the communication calls formed by applications, and redirect them to the SOCKS server when needed. If the determined security rules are not violated, then the work of the SOCKS client is completely transparent for client applications and users.

Thus, to form virtual private networks under the SOCKS protocol at the junction point of each LAN with the Internet, one must install the SOCKS server on a gateway computer, and on workstations in local-area networks. It is also necessary to install SOCKS clients on remote computers (Fig. 3.15). The SOCKS server is simply a firewall supporting the SOCKS protocol.

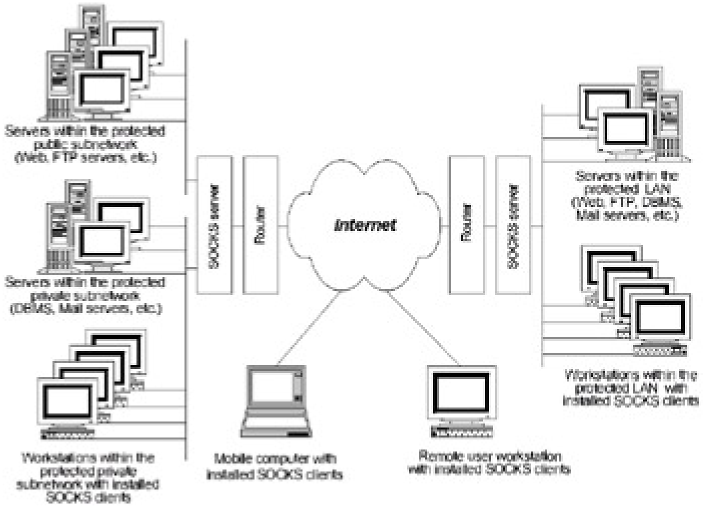

Servers within the protected public subnetwork (Web, FTP servers, etc.)
Servers within the protected private subnetwork (DBMS, Mail servers, etc.)
Workstations within the protected private subnetwork with installed SOCKS clients
SOCKS server
Router
Internet
Router
SOCKS server
Servers within the protected LAN (Web, FTP, DBMS, Mail servers, etc.)
Workstations within the protected LAN with installed SOCKS clients
Mobile computer with installed SOCKS clients
Remote user workstation with installed SOCKS clients

Fig. 3.15: The scheme of network interaction under the SOCKS protocol

Remote users can connect to the Internet using any method-via dial-up or dedicated line. When a virtual-private-network user attempts to establish a connection with any application server, the SOCKS client begins to interact with the SOCKS server. After the first stage of interaction, the user will be authenticated, and verification of access rules will show whether or not they have the right to connect to the specific server application running on the computer with the specified address. Further interaction may be performed via cryptographic tunnel.

Besides protecting a LAN from unauthorized access, the SOCKS server might be delegated the functions of controlling user access to the public Internet resources (Telnet, WWW, SMTP, POP and others). Access is completely authorized, since SOCKS server identifies and authenticates specific users rather than computers from which those users have logged on to the network. Access rules can allow or deny connections to specific Internet resources on the basis of rights and privileges assigned to specific user. Access rules can also produce different effects, depending on other parameters, such as authentication method or time of the day. In addition to discretionary access-control functions, the SOCKS server can perform event logging, and react in a predefined manner to specific events.

To achieve the highest level of security, LAN servers accessible from the Internet should be grouped into a separate network segment connected to the SOCKS server. This segment forms the so-called protected public subnetwork. Due to network-address translation, LAN servers and workstations connected to the Internet through the SOCKS server might lack registered network addresses and domain names. In this case, the SOCKS server makes all decisions concerning name and address resolution. With such configuration, users of the virtual private network can access all LAN servers within the VPN, as if they all were located within the same LAN.

3.5. Distribution of Cryptographic Keys and Coordination of Protected Tunnel Parameters

3.5.1. General Information

When virtual private networks are constructed, functions of distribution of cryptographic keys and coordination of protected tunnel parameters are of particular importance. The given functions are carried out upon creation of each cryptographic channel.

In virtual private networks, the following types of cryptographic keys are recognized according to the duration of their use:

- Long-term keys, which are applied over a relatively long time period, for example, a week, a month, or several months
- Temporary keys, which are generated for cryptographic protection of the information within the framework of one protected channel.

Long-term keys are assigned to users and servers of a computer network and provide authentication of the communicating parties, and also cryptographic protection of the distributed temporary keys. After both communication parties are authenticated, the temporary keys are safely distributed, and the parameters of the protected tunnel are coordinated. The cryptographic protection of the traffic within this tunnel is performed on the basis of the distributed temporary keys.

Long-term encryption keys (master keys), which are used over a long period of time, should be distributed before creation of VPN connections. Thus, the way these keys are distributed depends on the type of cryptosystem to which they correspond.

Symmetric cryptosystems are expensive, complex and unreliable in terms of their distribution of master encryption keys. From the end-user point of view, master encryption keys are similar to passwords, which, as well as these keys, must be kept secret. Consequently, during the exchange of symmetric encryption keys, their authenticity, integrity and confidentiality should be maintained. This requires that master symmetric encryption keys are initially distributed "hand-to-hand", or via protected communication links, which, for a Wide Area Network, will mean substantial costs. With decentralized distribution of master symmetric encryption keys, their quantity and, accordingly, complexity of distribution, sharply rises. The centralized distribution of these keys in the distribution center decreases the security level, since, in this case, the personnel of the center knows, what keys and for whom they have issued. Thus, such a scheme does not guarantee confidentiality. Also, when using symmetric encryption keys as master keys, it is impossible to implement protected-communication protocols to organize the interaction between parties that do not trust each other, since each of the parties knowing the common master key can reveal it.

Master cryptographic keys can be most efficiently distributed by using asymmetric cryptosystems when only public keys are subject to distribution. In this case, whoever has generated the private keys should be the one in possession of them. With proper support of the private key's confidentiality, nobody, except for its owner, can generate a digital signature with the help of this key nor decipher information enciphered by the appropriate public key. Therefore, when using asymmetric encryption keys as master keys, protocols of the protected interaction of parties that do not trust each other are implemented.

Sharing of public keys, in contrast to private symmetric encryption keys, does not require support of their confidentiality. One only needs to provide authenticity and integrity of the distributed public keys, which can be successfully accomplished by using digital certificates. Such a certificate includes the name of the certification center, the description of the certificate's owner, their public key, the period the certificate is valid for, and some additional parameters. The digital signature of the certification center, generated under the contents of each certificate, provides authenticity and integrity of the information specified in it, including a description of the owner, and their public key. The most popular and generally accepted standard specifying digital certificates' contents and format is the X.509 standard.

Temporary or session keys working within the framework of one tunnel are distributed on a network with the help of master keys. Since cryptographic protection of transmitted data is usually organized using symmetric encryption systems, temporary keys, as a rule, are symmetric encryption keys. To eliminate the capability of the intruder to predict the value of the temporary key, it should be generated on the basis of strictly random parameters.

Each temporary key can be safely distributed in the following ways:

- By encrypting it using an appropriate master key and then sending it via the network to the recipient
- By both communicating parties independently generating sessions keys on the basis of previously distributed master keys for symmetric encryption
- By both communicating parties independently generating session keys on the basis of open keys (distributed beforehand or exchanged by the communicating parties)

If an asymmetric cryptosystem is used for cryptographic protection of the distributed temporary keys, then temporary keys exchanged by the parties must be both encrypted and digitally signed. The signature on the public key allows the recipient to be sure that the temporary key has been encrypted by the sender specified within the message containing encrypted key.

When the communicating parties independently generate a session key on the basis of previously distributed master keys for symmetric encryption or pre-assigned passwords, each of the parties generates a temporary key based on the same parameters. To strengthen security, the parameters, besides the master key or password, must contain a coordinated random number. For this purpose, it makes sense to use the current time-stamp, including the year, date and time or the day.

When the communicating parties independently generate a session key on the basis of open keys exchanged by the parties rather than on the basis of previously distributed keys, the exchanged open keys must be verified by certificate center. If this is not the case, the communicating parties must authenticate each other before starting the procedure of temporary key generation. In order to strengthen the security of a protected tunnel, it is prudent to generate a temporary key on the basis of public keys, plus a coordinated random number, rather than on the basis of public keys only. The Diffie-Hellman algorithm is one of the most popular session key-generation algorithms used for public keys distributed or transmitted to each other.

In VPN protocols for creating protected tunnels at the data-link layer of the OSI model (PPTP, L2F, L2TP), temporary keys are mostly generated based on user passwords. Each mutual authentication, each participant in the information exchange generates the session key independently. Matching of the two keys generated by communicating parties is ensured by the fact, that these keys are calculated on the basis of the same parameters, including the coordinated random number of timestamp, as well as the hash function from the password. Taking into account that passwords are analogues of common symmetric encryption keys, a more effective way to distribute temporary keys on the data-link layer is to do so in a centralized way, for example, on the basis of the Kerberos protocol.

Mainly, VPN protocols for distributing temporary keys at the network and session layers of the OSI model (SKIP, ISAKMP and Handshake Protocol) use asymmetric cryptographic systems. When using these cryptosystems, temporary keys are distributed with the help of primary public keys. Distribution of temporary keys on the network layer is mostly carried out according to the Diffie-Hellman algorithm. On the session layer, temporary keys are, as a rule, distributed with the help of such asymmetric systems as RSA and A1 Gamal.

Depending on how simple implementation is and what degree of security can be achieved, a protected channel between two hosts of a computer network can be constructed in the following ways:

- Creation of a protected channel for each connection established on behalf of any software application
- Formation of a shared protected channel between network hosts, and creation, within this channel, of separate protected connections established on behalf of software applications

Creating a VPN channel for each connection proposes:

1. Distributing a request of one of the parties and achieving an agreement to create a protected tunnel
2. Authenticating the parties, which is carried out with the help of previously distributed master encryption keys or assigned passwords
3. Distributing temporary keys, and coordinating the protected tunnel's parameters

The second and third stages mostly intersect with one another, and authentication is carried out together with the distribution of temporary keys. The exception to this is in cases where authentication of the parties is performed through password methods.

For every connection, upon formation of a protected tunnel, temporary keys are always distributed with the help of master keys (pre-assigned passwords can also be used for this purpose). In view of this, for frequently organizing protected connections between software applications of communicating parties, private master keys are frequently used-a fact that increases the likelihood of their theft.

When a shared protected channel is created between two network hosts, within which separate protected connections are created, the stages listed above are not only carried out when the protected tunnel is established. They are implemented as each protected connection is established on behalf of the communicating parties' software applications.

At the beginning of the creation of a shared protected channel, a master session symmetric encryption key is distributed. This distribution is performed with the assistance of the communicating parties' master keys. Temporary keys for each protected connection are distributed by means of a master session key. Master keys are only used once-at distribution of a master session key, irrespective of the number of protected connections created within the shared protected tunnel. Therefore, the security level of the information exchange increases, since the probability of theft of the private master keys is minimized.

Thus, creating a protected channel for each connection established for any software application is a simple enough procedure. But interaction becomes less secure when private keys are frequently used. Besides, the frequent distribution and generation of temporary keys via master keys results in resource costs.

Creating a shared protected channel between two network hosts and creating separate protected connections on its basis is more complex. However, shared secret keys that distribute a session key in this case become less vulnerable, and computer resources can be used more effectively with regard to generating temporary keys. The resource costs of generating temporary keys can be reduced due to the fact that the basic calculations are carried over to the distribution stage and generation of a master session key.

Two striking representatives involved in constructing protected channels on the network layer are the popular SKIP and ISAKMP protocols.

The SKIP protocol deals with forming a protected tunnel for each connection. This protocol is easier to implement, but, with independent use, it does not support the coordination of parameters concerning encryption algorithms. Additional specifications that can be sustained upon implementation of the SKIP protocol provide coordination of all the parameters necessary for the development of a multi-purpose protected tunnel.

The ISAKMP protocol is aimed at creating a shared protected channel and individual protected connections. This protocol is more advanced and versatile, and stands as the obligatory standard for managing keys in the implementation of the upcoming sixth version of the IPSec protocol. In the current, fourth version of the IP protocol, the SKIP protocol is used more often.

3.5.2. The SKIP Protocol

The SKIP (Simple Key Management for Internet Protocols) protocol was developed by Sun Microsystems in 1994 and suggested as the Internet standard. Though this protocol is not yet accepted as the official standard, it has already become the open industrial protocol, supported by a number of companies.

The SKIP protocol, being an IP-compatible protocol, provides encryption-key management as well as application-transparent cryptographic protection of IP packets on the network layer of the OSI model. SKIP provides the independent generation of temporary keys by the communicating parties on the basis of public keys distributed in advance or transmitted to each other. The technology of key distribution is based on the Diffie-Hellman asymmetric cryptosystem. According to this cryptosystem, the following values are chosen for the whole virtual private network: a large prime number p and also a number a, satisfying the condition $a < p$. To provide cryptographic robustness, the following condition is imposed on the number p: factorization of the number $p-1$ must contain at least one large prime factor, and p should not be less than 512 bits.

Each VPN user generates the private key x, being a large random number, and develops the public key y, appropriate to the private key, according to the formula:

$$y = a^{x} \, mod \, p$$

All users place their public keys in a shared directory, which should be certified to exclude possible public key substitution attacks.

If two users, A and B, want to establish a protected tunnel, then the user A takes the public key of the user B from the directory and, using his private key, calculates the shared secret key:

$$K_{AB} = (y_B)^{x_A} \, mod \, p = (a^{x_B})^{x_A} \, mod \, p = a^{x_B x_A} \, mod \, p$$

where y_B is the public key of the user B, and x_B and x_A are private keys of the users B and A. There is no need to transfer the shared secret key K_{AB} on communication links, since user B can proceed the same way and calculate the required value using the public key of the user A taken from the directory.

$$K_{AB} = (y_A)^{x_B} \mathit{mod}\, p = (a^{x_A})^{x_B} \mathit{mod}\, p = a^{x_A x_B} \mathit{mod}\, p$$

where y_A is user', A public key, and x_A and X_B are private keys of the users A and B. To speed up the exchange, shared secret keys can be calculated beforehand and kept together with the asymmetric private keys in ciphered mode.

The secret master key K_{AB} is not used directly for encryption of the traffic between sites A and B. Instead, for encryption of a specific packet or a small group of packets, the transmitting party generates a random temporary key K_P which, in SKIP implementation, is known as a batch key. The following operations are carried out further (Fig. 3.16):

1. The original IP packet is encrypted using the K_P batch key and encapsulated into a SKIP packet.
2. The K_P batch key is encrypted using the secret master key K_{AB} and then placed into the SKIP header. Notice that place of authentication data (DAC) must be reserved in the SKIP header.
3. The resulting SKIP packet is encapsulated into another IP packet.
4. For the resulting IP packet, the authentication information (DAC encrypted using the K_P batch key) is calculated. The resulting value is then placed into the reserved field of the SKIP header.

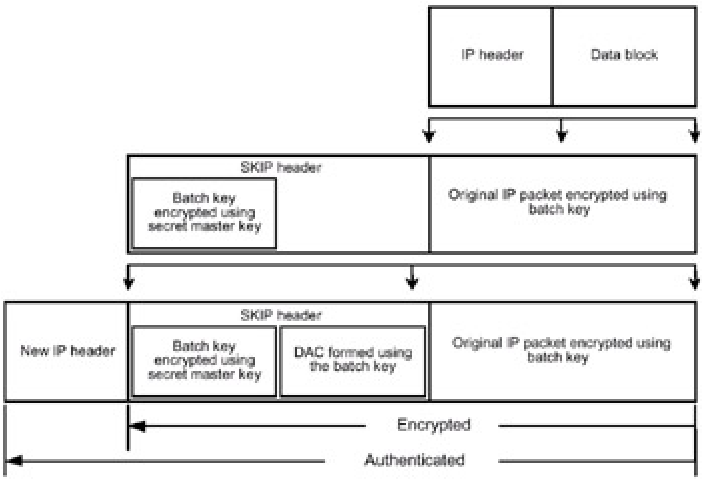

IP header
Data block
SKIP header
Batch key encrypted using secret master key
Original IP packet encrypted using batch key
New IP header
SKIP header
Batch key encrypted using secret master key
DAC formed using the batch key
Original IP packet encrypted using batch key
Encrypted
Authenticated

Fig. 3.16: IP packet before and after use of the SKIP protocol

Thus, besides effective distribution of keys, the SKIP protocol provides authentication and cryptographic protection of IP packets. As the batch key K_P is encrypted using the secret master key K_{AB} then only parties A and B can decipher this batch key. Thus, neither DAC (data authentication code) substitution, nor deciphering of the initial IP packet is possible.

The IP-packet, which is generated as a result of the listed operations, is sent to the recipient, who processes it in the reverse order:

1. Calculates the secret master key K_{AB}.
2. Deciphers the K_P batch key.

3. Using the hash function, calculates the current K_P characteristic on the basis of the batch key for the received IP packet, and then compares it with the DAC.
4. If the calculated value matches to the received one, then the integrity and authenticity of the received IP packet are guaranteed. If this is the case, the recipient decrypts the SKIP packet and extracts the original IP packet.

Applications running on the communicating computers act as sender and recipient.

If, for cryptographic protection of the traffic, the random batch key K_P, rather than the common, private key K_{AB}, is used, the security of the protected tunnel will increase. This is because the permanent private key K_{AB} cannot be compromised, as the probable opponent will not have sufficient material for fast cryptanalysis with the purpose of disclosing this key. Exchange security is also strengthened with the constant change of encryption batch keys. But nevertheless, if the batch key is compromised, any damage will affect only a small group of packets ciphered using this temporary key.

The SKIP protocol is constantly improving. In the final implementations of this protocol, additional measures for the protection of shared secret keys are accepted. Cryptographic robustness of a protocol against attacks on a shared secret key is enhanced due to encryption of the batch key K_P, using the session key generated on the basis of the K_{AB} key plus new parameter N, rather than on the basis of that key itself. To ensure the recipient's capability of calculating a session key, the number N is included in the SKIP header, together with the ciphered key K_P. Rules for working with the N parameters are up to the developer, however. To provide compatibility of the version, it is suggested that one consider that N is time, in hours elapsed since 00:00 on 01.01.1995. The problem of time synchronization on protected systems is solved simply enough-if the N parameter differs by more than 1, making the difference in time over one hour, the packet is discarded, because it is highly probable that it has been generated by the intruder.

Also, one of the essential innovations is unification of the SKIP protocol in accordance to the IP architecture. First of all, encapsulation order is now implemented according to the RFC 1827 standard. In the SKIP packet header, a new NEXT HEADER field that specifies the registration number of the protocol encapsulated within the given SKIP. Due to this, a unified procedure for packet encapsulation was implemented. In addition, changes have been made for the ESP protocol to achieve compatibility with IPSec. When combining SKIP and ESP protocols, the SKIP protocol is only responsible for transfer of the session key and the registration numbers of cryptographic protection algorithms used in ESP.

A number of additional implementations have been developed for the SKIP protocol. With the purpose of safe distribution of public keys, while avoiding violation of their integrity and authenticity, one can use digital certificates appropriate for the X.509 standard. There are also additional implementations describing interaction of supported encryption algorithms routing. The importance of standardizing this routing lay in the potential to use various encryption algorithms on different protected virtual-network member workstations. Without the coordination of used encryption algorithms, cryptographic tunnels cannot be established. The offered coordination protocol is based on the expansion of the well-known ICMP (Internet Control Message Protocol) protocol and provides authentication of coordinated parameters. For protection against reception of false information, a DAC (Digital Authentication Code) is located in each SKIP-packet header containing a transmitted ICMP packet. In case there is a discrepancy between a received packet's current characteristic and a DAC, parameters of the coordination in an ICMP packet are ignored.

3.5.3. ISAKMP Protocol

ISAKMP (Internet Security Association Key Management Protocol) is the most widely distributed protocol for the coordination of parameters and key management upon formation of a protected channel and creation individual protected connections in its framework. This protocol was developed by IETF, and has been specified as the key management standard in IPv6 protocol IPSec implementation. In comparison with the SKIP protocol, ISAKMP is more complicated in terms of implementation, but provides higher information security. When using the ISAKMP protocol, private master keys used for the distribution of temporary encryption keys are less vulnerable.

The ISAKMP protocol describes basic procedures of authentication of communicating parties, key exchange, and coordination of all other parameters of a protected IPSec tunnel. However, it does not contain specific cryptographic key-exchange algorithms. Therefore, other protocols can be used for key exchange. In IPSec implementation, the Oakley protocol, based on Diffie-Hellman algorithm, is chosen as the protocol to be used at formation of a protected tunnel. The combination of ISAKMP and Oakley protocols is designated as ISAKMP/Oakley. It is necessary to note that the ISAKMP and Oakley protocols have been developed so that they do not depend on IPSec implementation. For example, to make the session establishment process safer, the Oakley protocol can be used together with the SSL (Secure Sockets Layer) protocol.

According to ISAKMP, the coordination of parameters of protected interaction is necessary both upon formation of an IPSec channel, and at creation of each protected unidirectional connection in its framework. Global parameters of a protected channel form a managing context and at their coordination, besides the ISAKMP protocol, the key distribution of the Oakley protocol. Parameters of each protected connection are coordinated on the basis of a created managing context and form a Security Association (SA). Cryptographic keys for each protected connection, included in its security association, are generated through keys generated within the framework of a managing context. Thus, encryption and authentication algorithms, which will be used in a protected connection, are also taken into account.

Coordination of Protected Channel Global Parameters

Global parameters are coordinated at formation of a protected channel between network sites. These parameters serve for the creation of individual protected connections within the framework of a generated tunnel. In ISAKMP terminology, the coordinated global parameters of the protected tunnel are named the *managing context*.

The possible ways to coordinate global parameters of a protected channel differ from the method in which common, private keys are distributed, and also in the degree of protection final interacting sides' identifiers.

In the simplest case, a common session key for a protected channel is generated on the basis of early distributed basic symmetric encryption keys. For small networks, such an approach may be acceptable, but not scalable. The last property is provided at the use of asymmetric cryptosystems. In IPSec protocol for ISAKMP implementation, the Oakley protocol is the common session-key distribution protocol of choice. It is also named the IKE (Internet Key Exchange) protocol.

Upon coordination of global parameters of the parties communicating via the protected channel, for example, their IP addresses can be transferred openly or ciphered. If a protected channel is formed between intermediate points of interaction, for example, between LAN gateways forming a protected tunnel on behalf of endnotes, these ids can be hidden by encryption. As ISAKMP providers work in client/server mode, then the ISAKMP server functioning on the gateway can carry out cryptographic protection of ISAKMP-clients' ids functioning on workstations. Hiding the identifiers of the final cooperating sides heightens security from passive listening of an open network.

ISAKMP provides the following sequence of operations on formation of a managing context forming coordinated global parameters of a protected tunnel (Fig. 3.17).

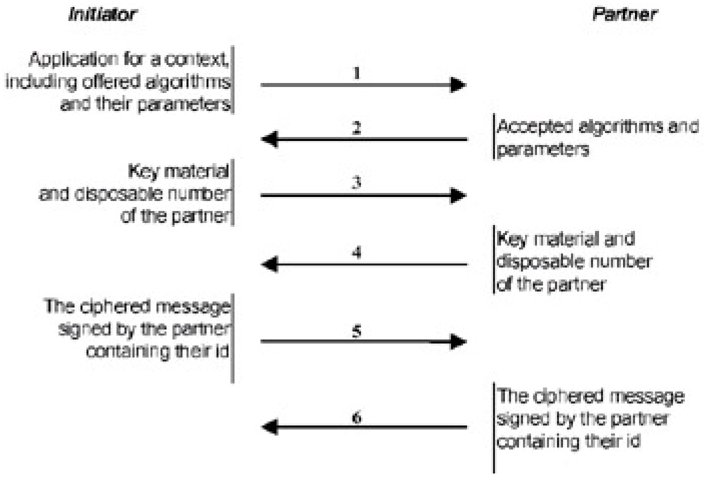

Initiator
Partner
Application for a context, including offered algorithms and their parameters
1
2
Accepted algorithms and parameters
Key material and disposable number of the partner
3
4
Key material and disposable number of the partner
The ciphered message signed by the partner containing their id
5
6
The ciphered message signed by the partner containing their id

Fig. 3.17: The scheme of coordination of global parameters of a protected channel

When protected interaction is needed, an initiator directs an inquiry about forming a protected channel, including offers of a set of protective algorithms to the opposite side (stage 1). In the initiator's inquiry, offers on the set of protective algorithms and their parameters are ordered according to their level of preference. In the response message (step 2), the partner informs the initiator about those protection algorithms and parameters that suit him or her. One algorithm and its parameters are chosen for each protective function (generation and distribution of keys, authentication and encryption).

During the third and fourth steps of interaction, the initiator and partner send their public keys to each other. These are necessary to generate a common, private key.

Accordingly, the public keys are *Y(A)* $(y_A = a^{x}{}_{A} \mathit{mod}\ p)$ and *Y(B)* $(y_B = a^{x}{}_{B} \mathit{mod}\ p)$. The large prime numbers given for a virtual private network act as the Oakley protocol for the cooperating parties *A* and *B*, and *X(A)* and *X(B)*-the private keys of subscribers A and B generated as the large random numbers.

The common, private key is calculated according to the following expression:

$$K_{AB} = (y_A)^{x_B}\ mod\, p = (y_B)^{x_A}\ mod\, p$$

Public keys are sent to each other with random one-time numbers, which protect against message reproduction. According to the X.509 standard, digital certificates technologies are used to provide authenticity of the public keys directed to each other. But ways to realize this technology together with ISAKMP in IPSec specification have been not yet determined, because of the absence of a directory service arranging all specifications.

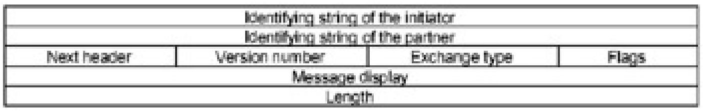
On the basis of a shared secret key, three kinds of keys are generated:

- SKEYID_d-the key material used to generate temporary keys for protected connections
- SKEYID_a-session keys used to authenticate both sides and coordinate parameters
- SKEYID_e-session keys used to encrypt coordinated parameters

The fifth and sixth steps of formation of a managing context serve for the exchange of identified information ciphered and signed on SKEYID_e and SKEYID_a keys.

Creating a managing context for a protected channel on the basis of the ISAKMP/Oakley protocol requires major resource expenses to calculate a common, private key, and to generate session keys. But, due to it, on the other hand, the expenses for generating temporary keys for protected connections are reduced. Taking into account that, for many protected connections, a common, protected channel is only formed once, such an approach, looking at the end result, is more effective than forming a protected channel for each connection. Depending on the conditions present in using the ISAKMP/Oakley protocol, a compromise is possible between cryptographic robustness and the necessary volume of calculations. At formation of a managing context through shorter public keys, the volume of calculations decreases, but owing to diminished cryptographic robustness, communications safety also weakens.

As well as providing protection against confidentiality and authenticity violation of coordinated global parameters, ISAKMP also protects against replication, delays and removal of protected messages. So-called identifying strings are applied for protection against the attacks listed above. These strings are formed by an initiator and his or her partner by using the current time and are present in all ISAKMP messages (Fig. 3.18). An exception to this is the first query about creating a protected channel in which only one of the identifying strings, i.e. the string of the initiator, is included.



<table>
<tr><td colspan="4">Identifying string of the initiator</td></tr>
<tr><td colspan="4">Identifying string of the partner</td></tr>
<tr><td>Next header</td><td>Version number</td><td>Exchange type</td><td>Flags</td></tr>
<tr><td colspan="4">Message display</td></tr>
<tr><td colspan="4">Length</td></tr>
</table>

Fig. 3.18: Format of the ISAKMP message header

Both the initiator and the partner for each sent message generate identifying strings on the basis of the current time and insert it into this message. The identifying string of the opponent side received in the message for which this answer is formed is inserted into each response message, in addition to the sender's identifying string. Upon receiving the response message, the recipient checks the presence of the identifying string sent. When setting the maximum following time of an expected message, using identifying strings allows one to get information on repeated, delayed and removed messages.

The most important types of attacks are those that can be implemented by impersonating an authorized participant of interaction. Using this method, one can get unauthorized access to confidential information and implement any other actions that run counter to the security policy. For example, it is possible to cause any protected virtual site to malfunction by imposing intensive calculations characteristic to public key cryptography. For protection against impersonating an authorized participant of interaction, global parameters must be coordinated (Fig. 3.17), together with public-key exchange cooperating sides' authentication on the third and fourth steps. Such verification is effectively implemented while using digital certificates. To prove to any participant that they are definitely the owner of a given public key, they can generate a digital signature under the sent message including their digital certificate, and also a timing mark to protect this message from being replicated. The recipient of such a message carries out the following operations:

- Extracts a public key from a digital certificate of the sender and verifies the validity of the digital signature on this key.
- Verifies the digital signature in a certificate on an available public key of the certification center.
- When verified digital signatures and the time of the received message are validated, the recipient draws the conclusion that the person sending the message is a valid partner.

A protected tunnel formed with coordinated global parameters is bi-directional, in the sense that any of the communicating parties has the opportunity to initiate an individual protected connection with the help of these parameters. Any protocol can be used to transfer ISAKMP messages, however, in IPSec realization. The UDP protocol with the port number 500 is designated for this.

The Coordination of Every Protected Connection's Parameters

After creating a common protected channel, the parameters of every protected connection are coordinated on the basis of the channel's generated global parameters and form a Security Association, also named a protocol context. Every protected connection formed within the framework of a common protected channel is a unidirectional connection, i.e. a connection from the packet's sender to its recipient.

Thus, in one protected IPSec connection, only one of two protocols of cryptographic protection message packets can be used:

- The Authenticating Header (AH) protocol, which authenticates the data source, verifies their integrity and authenticity after reception, and protects against messages being repeated
- The Encapsulating Security Payload (ESP), providing cryptographic closing of transmitted messages packets, and also implementation of all AH functions

On the basis of a common protected channel formed under ISAKMP, protected connections that are inappropriate for IPSec implementation can be also created. However, in this case, when a protected connection is created, a security association needs to be formed, and only one cryptographic protection protocol can be used in one connection.

At symmetric interaction, one must form a minimum of two protected connections, and accordingly, two security associations-one for each direction. If the AH and ESP cryptographic protection protocols are used together, four security associations need to be established.

The structure of coordinated parameters for every protected connection forming a security association includes the following elements:

- The number of cryptographic protection protocols being used (AH, ESP or a cryptographic protection protocol which is not included in IPSec implementation)
- Numbers of cryptographic protection algorithms and their parameters
- Cryptographic protection mode (transport or tunnel)
- Temporary encryption keys only valid for the current protected connection
- A Time To Live (TTL) of the protected connection
- The maximum size of packets
- Additional parameters (counter, window and flags) for protection against packet repetition, delay or removal

From the given list, it follows that, at formation of a security association, a TTL of a protected connection that depends on a required cryptographic robustness will also be coordinated. According to this term, how soon the current protected connection will need to be closed and, probably, when a new one should be opened will be defined depending on the time or volume of the transferred data. It is important to keep in mind that TTL is set for each protected channel as well as for each connection. This parameter can be set according to a maximum time interval or a maximum volume of the transferred data. For example, the TTL of a protected connection may be determined to be 15 minutes or 10 Mbytes, and the TTL of a protected channel-60 minutes or 40 Mbytes. With an increase of length of used keys and cryptographic robustness of cryptographic protection algorithms used, the allowable TTL of a protected channel and protected connection also increases. Such an approach allows a balance of cryptographic robustness of services and costs of overhead charges for ISAKMP packet transfers.

Application processes are, as a rule, users of protected connections and between two network hosts, any number of such connections may be generated within the framework of one protected channel (Fig. 3.19). One should note that a pair of network hosts can simultaneously support several protected channels, if there are applications with completely different cryptographic requirements. For example, a part of temporary keys through preliminary distributed basic symmetric encryption keys can be generated, while another part may be generated on the Diffie-Hellman algorithm.

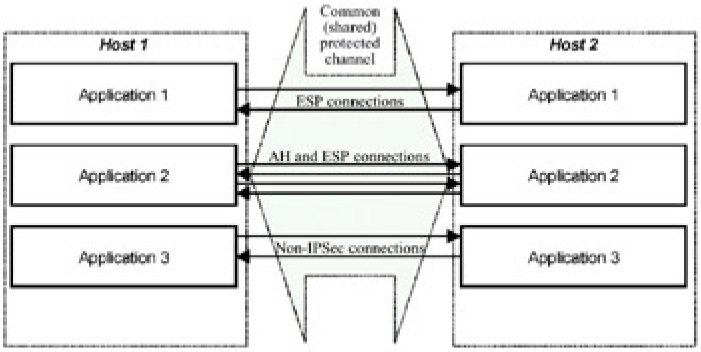


Fig. 3.19: Protected interaction under the ISAKMP protocol

The protected connection appropriate to IPSec implementation is identified by the target IP address used by the AH or ESP cryptographic protection protocol of a larger size with the Security Parameter Index (SPI). The SPI is necessary for the identification of each of the security associations as there may be several protected connections with identical IP addresses and cryptographic protection protocols. This SPI is included in protected IPSec-packet headers so that the accepting side can correctly decipher and authenticate these packets, having taken advantage of the specified security association.

It is much easier to set up a protected connection than it is to form a protected channel. The parameters for two symmetric unidirectional connections can be coordinated with the help of a generated managing context by following three steps (Fig. 3.20).

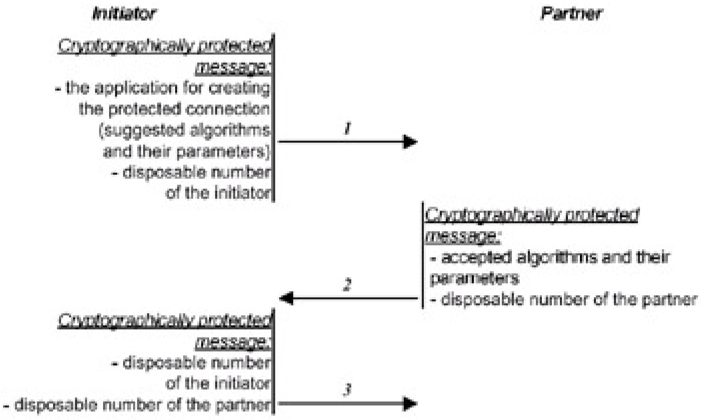

Initiator
Partner
Cryptographically protected message:
- the application for creating the protected connection (suggested algorithms and their parameters)
- disposable number of the initiator
1
Cryptographically protected message:
- accepted algorithms and their parameters
- disposable number of the partner
2
Cryptographically protected message:
- disposable number of the initiator
- disposable number of the partner
3

Fig. 3.20: The scheme of parameters of two symmetric connections

The following session keys generated upon formation of a protected channel are intended for the cryptographic protection of messages at parameters' coordination of created connections, and for distribution of temporary keys working only within the framework of these connections:

- SKEYID_a-session keys used to authenticate both sides and coordinated parameters
- SKEYID_e-session keys used to encrypt coordinated parameters
- SKEYID_d-key material used to generate temporary keys for protected connections

All three messages submitted in Fig. 3.20 are ciphered and authenticated on SKEYID_a and SKEYID_a keys. For authentication, the DAC is used and calculated with the help of a hash function, on whose arguments the authenticated message and session SKEYID_a key act. Temporary keys for a protected connection are generated by application of a hash-function to SKEYID_d key with additional parameters including the initiator and partner's disposable numbers. Messages (1) and (2) can carry additional information, for example, data for generating new keys on the Diffie-Hellman algorithm or client ids on behalf of which ISAKMP servers form protected connections. During one data exchange according to coordinated parameters, two unidirectional connections are formed, via three steps one for each direction. The recipient sets the SPI with whose help it will find an SA to process received cryptographically protected packets.

Coordinating parameters of a protected connection, including generating and distributing temporary keys, does not require a high volume of calculations, as a set of simple mathematical operations is used. Also, as the TTL of protected channels and protected connections is set, restrictions on the admissible number of protected connections within the framework of one protected channel can also be determined. Exceeding this number will cause the keys to be generated at formation of a managing context, and, thus, all temporary keys may be exposed. There is no strict rule at the moment designating the number of protected connections for one protected tunnel. So, developers of security facilities continue their work, being guided by common reasons and taking into account the conditions under which available products are used.

3.6. Securing Remote Access to a LAN

3.6.1. Organizing Secure Remote Access

Currently, remote access to LAN resources is a matter of no less importance than access in direct-connection mode. Remote access to the LAN is implemented from external physically unprotected environment via public networks. Accordingly, VPN tools must ensure security of internetwork communications, not only when joining LANs, but also when connecting remote computers to local-area networks.

Remote access to a local-area network is possible via telephone networks, global computer networks or via a combined data-transfer environment made up from telephone networks and global computer networks (Fig. 3.21). Besides this, there might be situations when LAN and remote computers are joined to make a virtual network by means of connecting remote computers via telephone lines to global computer networks. The Internet provides the most efficient way of organizing remote access to a local-area network via telephone lines. Remote access to local-area networks via the Internet provides several advantages:

- Scalable remote-access support, enabling mobile users to connect to their local ISP via telephone lines, and then logon to their LANs
- No need for modem pools within local-area networks, and the capability of controlling remote-access traffic just in the same way as any other Internet traffic
- Reduced expenses for informational exchange via the external environment. Instead of establishing an expensive direct connection to the LAN, remote users can connect to the Internet and then access their corporate networks

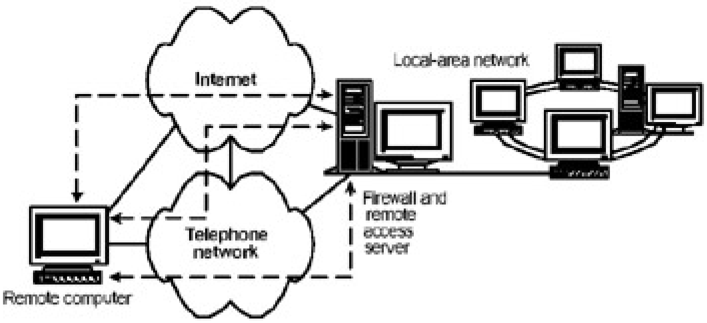


Fig. 3.21: Methods of accessing LAN resources remotely

Independently of the type of the data-transmission environment connecting the remote computer to the local-area network, within the LAN there must be a dedicated remote-access server. In the most general case, this server is intended for establishing a connection with the remote computer, authenticating the remote user, managing the remote connection and performing other proxy functions when exchanging data between the remote computer and the LAN. These functions include:

- Discretionary access control to computer resources
- Cryptographic protection of the traffic
- Event logging
- Reacting to specific predefined events.

Functions towards which the remote access server is oriented are closely related to the functions performed by the modern firewall. Taking into account firewall functionality, one can draw the conclusion that it also must perform remote-access server functions. Most firewalls support these functions. However, the remote-access server is important by itself. Therefore, in practice, it is configured separately, independently of the firewall. This relates to the fact that separate protocols are used for organizing remote access. These protocols are intended to manage and protect the remote connection.

To support a high level of security, the remote access server, similarly to the firewall, must run on the computer located at the junction point between the protected LAN and external environment, such as various public networks and telephone networks. If users connect to the LAN remotely, possibly connecting to the Internet via dial-up lines, then the remote-access server is usually installed as part of the firewall or on the same computer. If remote access to the local-area network is also performed via telephone communication lines, then it makes sense to dedicate a separate computer as a remote-access server to manage these connections. In this case, a standalone remote-access server is known as terminal server (Fig. 3.22).

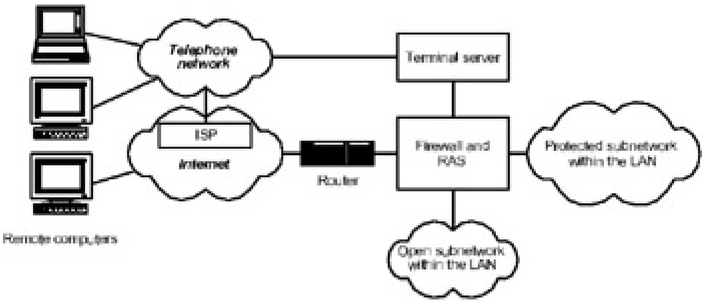


Fig. 3.22: Secure configuration of the LAN when organizing remote access

The terminal server is a system equipped with several asynchronous ports and one LAN interface. It must be connected in such a way as to ensure that data exchange between a remote computer and the LAN takes place only via the firewall. There are two possible methods of organizing such a connection:

- The terminal server is included into an open subnetwork and the firewall is configured separately to protect both the open subnetwork and the protected subnetwork.
- The terminal server is connected to a separate network interface of the firewall, as shown in Fig. 3.22.

The presence of the terminal server controlling remote access to the LAN via telephone lines does not eliminate the need for a remote-access server operating in coordination with the firewall and controlling remote access from the Internet.

The terminal server must support the important function of callback by the specified telephone number of the remote user. This function allows limiting the range of telephone numbers that can be used to access the LAN remotely. If callback mode is specified for the user, then, after successfully authenticating that user, the terminal server closes the connection and initiates the call using the telephone number of the remote user specified in the database. This approach to establishing remote connections eliminates the possibility of accessing the LAN from unknown and potentially dangerous locations.

The PPP (Point-to-Point Protocol) is the most popular among remote access protocols. It was developed with the experience gained on the basis of using similar protocols of the previous generations (such as SLIP), and has the status of the Open Internet Standard. The set of PPP functional capabilities includes establishing a remote connection, and exchanging information via that connection in the form of network-layer packets encapsulated into PPP frames. The method of forming frames employed in the PPP protocol provides the capability for several network-layer protocols to operate simultaneously via a single remote-access channel. Other important functions supported by the PPP protocol include:

- Configuring the communication line and testing its quality

- Authentication of both remote user and remote-access server
- Data compression and encryption
- Error detection and correction
- Dynamic assignment and management of IP addresses

Cryptographic protection of the traffic when accessing the protected LAN remotely can be based on any VPN protocols. However, the independence of the OSI network-layer protocols is ensured only when organizing protected tunnels at the data-link layer. This is why VPN protocols of the data-link layer, such as PPTP, L2F and L2TP are most frequently used for cryptographic protection of remote connections.

The PPTP, L2F and L2TP protocols are based on PPP, and allow the creation of protected tunnels for secure data exchange between remote computers and local-area networks using various network-layer protocols-IP, IPX or NetBEUI. For transmission via telephone channels, packets of these protocols are encapsulated into PPP frames. When it is necessary to exchange data via the Internet, protected PPP frames are encapsulated into IP packets. Cryptographic protection of the traffic is possible both within Internet channels and along the whole route connecting the remote computer and the remote-access server.

3.6.2. Remote User Authentication

General Overview

Remote access to LAN resources must be controlled according to the requirements of the security policy of the organization to which the LAN belongs. Reliable discretionary access control can be ensured only in case of reliable user authentication. For remote users, these requirements become significantly more stringent. This relates to the fact that, in contrast to local users, remote users do not need to pass security control when entering the organization's territory. When working with "invisible" users, it becomes much more difficult to ensure that only authorized users can access LAN resources.

When it is necessary to organize remote access to a LAN, the following functional capabilities of the authentication subsystem must be supported:

- Coordination of the authentication protocols used. The authentication subsystem must be flexible, and should not be strictly bound to specific authentication protocols.
- Blocking any attempts of bypassing the authentication phase after establishing a remote connection.
- Authentication of each communicating party, both the remote user and the remote-access server, which will eliminate the possibility of impersonation.
- Performing both initial authentication before accessing LAN resources as well as dynamic authentication of the interacting parties in process of the remote session. This function eliminates the risk of connection interception by the intruder and further impersonation.

- Usage of temporary session passwords or cryptographic protection of transmitted secret passwords, to exclude the possibility of reusing the intercepted information.

Having compared the existing remote-access protocols, one can draw the conclusion that the above-listed authentication capabilities can be implemented only on the basis of the PPP protocol. This protocol combines the following specifications:

- LCP (Link Control Protocol)-used to organize data transmission, choose and configure the communication channel, check the connection, maintain or close the channel, perform authentication and detect errors
- HDLC (High-level Data Link Control)-responsible for forming datagrams for further transmission via telephone channels
- NCP (Network Control Protocols)-actually the family of protocols used for determining the configuration of network-layer protocol implementations and managing IP addresses

The lifecycle of the PPP connection comprises several stages (Fig. 3.23). After a remote computer successfully calls the LAN remote-access server, both parties coordinate channel parameters by exchanging LCP packets. At this stage, it is necessary to determine if authentication is required (notice that authentication is optional, and in general might be unnecessary).

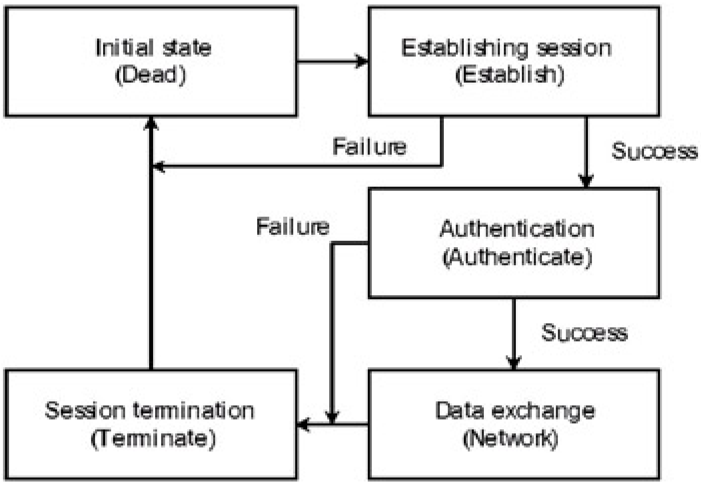


Fig. 3.23: Scheme for establishing a PPP session

For secure remote access, authentication must be enabled in PPP settings. In this case, any of the parties that establish a PPP session must explicitly enable authentication. If other party does not support the suggested authentication protocols, then the software, depending on the implementation or configuration, can either suggest another authentication protocol or terminate the established connection. If the requesting party does not support any suggested authentication protocol, the established connection will be terminated. The PPP protocol does not specify whether authentication must be one-way or mutual. In the latter case, both parties can use their own authentication protocols.

If authentication is successful, the parties pass to the information-exchange phase. At the beginning of this phase, they coordinate OSI network-layer parameters using the

NCP protocol. After that, they exchange network-layer packets, encapsulated into PPP frames according to the HDLC protocol. The method of forming the frames provides the capability of simultaneously supporting several network-layer protocols on the same remote-access channel. Consequently, from the remote user's point of view, PPP imitates the presence of the remote computer within the internal network.

The PPP protocol implies that, during the procedure of establishing a connection, each of the interacting systems can be required to use one of the two standard authentication protocols-PAP (Password Authentication Protocol) or CHAP (Challenge Handshake Authentication Protocol). However, other authentication protocols can also be used. When performing authentication according to the PAP protocol, identifiers and passwords are transmitted via the connection in plain-text mode. When using the CHAP protocol, the password transmitted via the connection is encrypted on the basis of the random number obtained from the server. When PAP protocol passwords are not encrypted, this protocol must be used only in combination with a protocol oriented towards authentication by session passwords. The S/Key protocol is the most popular one that is oriented towards using temporary passwords.

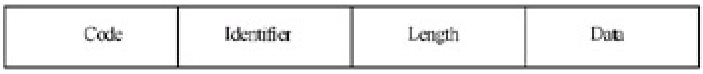
Most software products providing remote access according to the PPP protocol usually support PAP and CHAP. Sometimes, developers implement custom remote-access authentication protocols, supporting PPP. These implementations are usually modifications of the PAP and CHAP protocols that extend their functional capabilities and provide several advanced features. For example, the Remote Access Service (RAS) implemented in Windows NT uses a custom version of the CHAP authentication with the MD4 hash function, and is known as MS-CHAP. The MD5 hash function standard for the CHAP protocol is not supported. Client/server software from Shiva, in addition to the PAP and CHAP protocols, also supports of the proprietary SPAP (Shiva Password Authentication Protocol) protocol. These specific features must be taken into account when choosing remote-access clients and servers. Notice that, even provided that the same remote-access protocol is used, there still exists the risk of authentication failure due to authentication-protocol incompatibility. If this is the case, the remote computer and the remote-access server will not be able to establish the remote session.

The PAP Protocol

The PAP protocol, according to which passwords are transmitted via a connection line in plain text, must be applied only in combination with a protocol oriented toward authentication by session passwords, such as S/Key. Otherwise, the password transmitted via communication line might be intercepted by the intruder and reused, in order to get unauthorized access on behalf of the remote user.

Two parties participate in the authentication process: the authenticating party and the authenticated one. Usually, the remote user is the party being authenticated, while the remote-access server is usually the authenticating party. To initiate the authentication process according to the PAP protocol, the remote-access server, after establishing the connection, must send a special LCP packet instructing use of the PAP protocol to the remote computer. The client and server exchange PAP packets. The remote computer sends the user identifier and password supplied by the user. The remote-access server uses the received user identifier to select the reference password from the security database, and compares it to the password that it has just received from the user. If the passwords match, authentication is successful, and server informs the remote user of this.

PAP-packet format is shown in Fig. 3.24.



Code	Identifier	Length	Data

Fig. 3.24: PAP-packet format

The Code field specifies the packet type. There are three types of PAP packets:

- Authenticate-Request-request for authentication
- Authenticate-Ack-acknowledging the authentication
- Authenticate-Nak-failed authentication

The Identifier field contains a unique number that enables the recipient to determine to which request the received answer corresponds. The Length field contains the packet length in bytes. The length and format of the Data field depend on the type of the PAP packet.

The Authenticate-Request packet is used to transmit a user identifier and password to the remote-access server. A packet of this type must be repeatedly transmitted via an established connection until a valid Authenticate-Ack packet is received, or until the timer (which might be present in certain implementations) expires. The remote-user identifier and password are placed into the data field of the Authenticate-Request packet.

The remote-access server, having received the Authenticate-Request packet and having performed a password check, must send the Authenticate-Ack or Authenticate-Nak packets, depending on the result of the check. If the packet is lost, the remote computer will retransmit the Authenticate-Request packet. In this case, the remote-access server will have to recreate the Authenticate-Ack or Authenticate-Nak packet.

The Authenticate-Ack packet informs the party being authenticated that the procedure has completed successfully and that its authenticity was acknowledged. If the remote-access server detects that the password specified in the Authenticate-Request field is invalid, it will send the Authenticate-Nak packet, and then initiate the connection-termination procedure. Even if the Authenticate-Nak packet is lost, LCP packets terminating the connection will inform the party being authenticated that the authentication procedure has failed. The data field in the Authenticate-Ack and Authenticate-Nak packets contains the message, the contents of which are not strictly defined by any standard, as well as its length in bytes.

The S/Key Protocol

S/Key (RFC 1760) is one of the most popular session-password authentication protocols. This protocol is implemented in most systems that require remote-user authentication, such as FreeBSD and CISCO Tacacs+. Interception of the session password transmitted via the network during authentication, does not provide the intruder the possibility of reusing this password, since, during the next authentication, another password will be used. Because of this, an authentication scheme based on session passwords (for example, the one implemented by S/Key) allows transmission of the session password via the network in plain text, and thus compensates for the main drawback of the PAP authentication protocol.

The S/Key protocol does not eliminate the necessity of specifying a secret password for each user. However, such passwords are only used for generating session passwords. To prevent the intruder from calculating the original secret password on the basis of the intercepted session password, the generation of session passwords is performed using a one-way function (sometimes named an irreversible function). Recall that *Y* = *F(X)* is irreversible if it satisfies the following requirement: for each value of *X*, *F(X)* is easily calculated, while calculation of *X* given *Y* is problematic. For this purpose, the S/Key algorithm uses the MD4 (Message Digest Algorithm 4) hashing algorithm. Some implementations of S/Key use MD5 (Message Digest Algorithm 5) cryptographic hashing algorithm. Thus, to restore the argument by the given value of the cryptographic hash function within the acceptable time period, practically unavailable calculation resources are required.

The main idea of the S/Key protocol is as follows. The party that needs to be authenticated is assigned the random key *K*, which serves as its secret long-term password. Then the authenticating party performs the initialization procedure for generating new list of session passwords *M*. In the course of this procedure, the irreversible function *F* is calculated by the *K* key, thus producing the check value Y_{M+1} for the first session password. To calculate this value, the *K* key is substituted as the argument of the *F* function, and this function is recursively calculated *M*+1 times: $Y_{M+1} = F(F(F(\ldots(F(K))\ldots))) = Y^{M+1}$ (*K*). The user identifier and the secret key *K*, along with open numbers *M* and Y_{M+1}, are stored in the database of the authenticating party. The *M* number is considered as the session-password number for the next authentication from the list of session passwords:

$$Y_1 = F(K) = F^1(K)$$

$$Y_2 = F(F(K)) = F^2(K)$$

$$Y_3 = F(F(F(K))) = F^3(K)$$

....

$$Y_{M-1} = F(F(\ldots(F(K))\ldots)) = F^{M-1}(K)$$

$$Y_M = F(F(F(\ldots(F(K))\ldots))) = F^M(K)$$

When performing the next authentication procedure after initialization, the authenticated party provides its identifier to the authenticating party, which returns the *M* number corresponding to the identifier. Next, the authenticated party calculates the session password, using its secret key *K*, by the following formula: $Y'_M = F(F(F(\ldots(F(K'))\ldots))) = F^M$ (*K'*). When done, it sends the session password to the authenticating party. Having received this value, the authenticating party uses it to calculate the irreversible function *F:* $Y'_{M+1} = F(Y'_M)$. The resulting value Y'_{M+1} is compared to the Y_{M+1} value from the databse. If these values match, i.e., $Y'_{M+1} = Y_{M+1}$, this means that $Y'_M = Y_M$, and, consequently, authentication is successful. In case of successful authentication, the authenticating party replaces the Y_{M+1} value stored in its database for the authenticated party with the Y_M number received, and decreases the M number by one: $M = M-1$. Taking into account that, in case of successful authentication, the number of session password M for the next authentication has reduced by one, the database record for the authenticated party will now store its user identifier, the secret key K and numbers M and Y_{M+1}. Here, Y_{M+1} is the session password received from the authenticated party during the last successful authentication. After the list of session passwords has been exhausted, the authenticating party must repeat the initialization procedure.

To enable the user to choose a long-term secret password on his own, the S/Key specification prescribes that session passwords be calculated on the basis of a secret password plus a random number generated by the authenticating party. Thus, according to the S/Key protocol, each user is assigned a user identifier and a long-term secret password. To enable authentication, it is necessary to perform the initialization procedure for each user. The initialization generates the next list of session passwords, and is also known as password initialization. This phase is performed at the user's request on the remote-access server. The procedure comprises the following steps:

- The remote-access server requests the user's identifier, which will be required at step 4.
- The remote-access server then generates random number N, known as the initialization code. This number will be used to calculatea session-password list for the user.
- The remote-access server requests the number of the session password from the user (the number M), which, usually, must belong to the range $300 <= M <= 1000$. This number defines how many session passwords the user needs until the next password-initialization procedure (this number can also be predefined by the administrator).
- The authenticating party retrieves the user's secret password P by the user identifier supplied by the remote user.
- The values of N and P are used as arguments of the irreversible function F, applied repeatedly $M+1$ times

$$Y_{M+1} = F(F(F(...(F(P, N))... , N), N), N) = F^{M+1}(P, N)$$

- The numbers N, M and Y_{M+1} are stored in the security-system database, along with the user identifier and password. In contrast to the password, however, these numbers are not secret.

The M number represents the ordinal number of the session password for the next authentication from the list of all session passwords generated for the user. The password-initialization phase does not require the parties to transmit the secret password via the network. Consequently, the user, both from the computer within the protected LAN and from the remote computer, can initiate it. After password initialization, the remote user is able to use M session passwords and, accordingly, to establish M remote connections to the LAN.

The authentication procedure during remote access to the LAN using the S/Key protocol comprises the following steps:

1. The remote user supplies their user identifier to the remote access server.
2. The remote-access server searches the security database and uses the supplied user identifier to retrieve the user's secret password *P* along with the numbers *N*, *M* and Y_{M+1}.
3. The server returns to the user the value of *N*, which will be constant until the next initialization, along with the number of the session password *M*. After using each password, this number is decreased by one.
4. The remote user enters the secret password *P′* and the client software calculates the current session password: $Y'_M = F(F(F(\ldots(F(P', N))\ldots, N), N), N) = FM(P', N)$.
5. The calculated session password *P′* is then sent to the remote-access server, which applies to it the irreversible function F: $Y'_{M+1} = F(Y'_M)$.
6. Next, the server compares the resulting value Y'_{M+1}, to the Y_{M+1} value from the security-system database. If these values are matching, the authentication is successful, and remote user will be provided access to the protected LAN. Otherwise, the user will be informed of the failed authentication and the connection will be terminated.
7. If authentication is successful, the server edits the user's record in the security-system database by replacing the Y_{M+1} number with the session password Y'_M supplied by the user. The number *M* will be decreased by 1: $M = M - 1$. Taking into account that the number of the session password M for the next authentication has been reduced by 1, the session password received from the user and stored in the security system database will now be designated as Y_{M+1}.

To make authentication faster, a certain number of session passwords (for example, several dozens) can be calculated beforehand and stored on the remote computer in encrypted form. It makes sense to use the long-term secret key as an encryption key for this procedure. It is also possible to record or print the list of session passwords, and then strike out the used ones. However, in this case, it is necessary to take all precautions in order to prevent someone from stealing these passwords.

The phase of password initialization must be performed in the following cases:

- When the long-term secret key has been assigned to the user for the first time or the existing secret key has changed
- When the current session password list has been exhausted, and the number of the next session password approaches zero
- Upon suspicion that someone has stolen the list of session passwords

In the latter case, the suspicion might arise when one notices that the order of the session passwords from the initialized list has changed: For example, if, next time after using password 150, the server requests password 145, rather than 149.

The CHAP Protocol

A private, static password is used in both the CHAP and PAP protocols. However, in contrast to the PAP protocol, every user password for transfer on a communication link is encrypted on the basis of a random number received from the server. Yet, unlike the PAP protocol, every user password for transfer on a communication link is ciphered on the basis of a random number received from the server. Password encryption, according to the CHAP protocol, is one-way, and carried out with the help of the cryptographic hashing algorithm. The MD5 Algorithm (Message Digest Version 5) has been designated as the standard hashing algorithm, which must be supported in all implementations of the CHAP protocol. However, CHAP implementation does not preclude the use of other hash-function computation algorithms.

To activate the process of authentication under the CHAP protocol, after establishing a communication session, the RAS must send an LCP packet, indicating the necessity of the application of the CHAP protocol to a remote computer, as well as the required hashing algorithm. If the remote computer supports the hashing algorithm that is offered, it must answer with an LCP packet about the agreement with the offered parameters. Otherwise, the exchange of LCP packets for the coordination of the hashing algorithm will be carried out. After that, the authentication begins through the exchange of CHAP protocol packets. These packets have the same format as PAP protocol packets (Fig. 3.24). The Code field specifies the packet type. The Identifier field contains a unique number that designates which query the received answer corresponds to. The Length field contains the packet length in bytes. The type of CHAP packet defines the length and format of the Data field. There are four types of CHAP packets:

- Challenge
- Response
- Success
- Failure

The procedure of authentication begins with the RAS sending a "Challenge packet" (Fig. 3.25). The Data field in a Challenge packet includes:

- Any arbitrary numerical sequence, which must be unique for each sent Challenge packet, and also the length of this sequence in bytes
- The id entifier of the verifying party

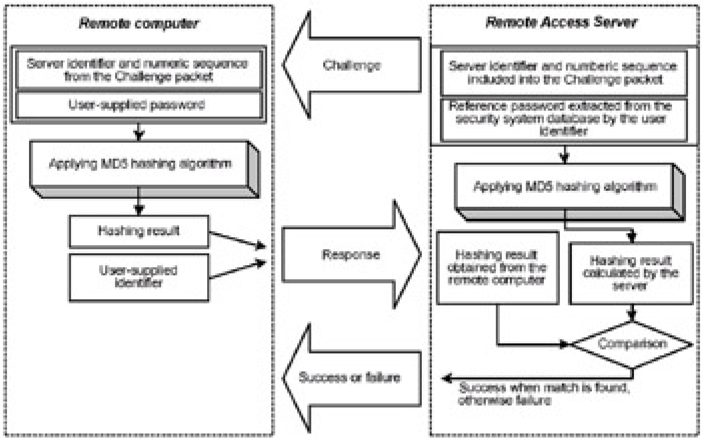

Remote computer
Server identifier and numeric sequence from the Challenge packet
User-supplied password
Applying MD5 hashing algorithm
Hashing result
User-supplied identifier
Challenge
Response
Success or failure
Remote Access Server
Server identifier and numberic sequence included into the Challenge packet
Reference password extracted from the security system database by the user identifier
Applying MD5 hashing algorithm
Hashing result obtained from the remote computer
Hashing result calculated by the server
Comparison
Success when match is found, otherwise failure

Fig. 3.25: Authentication by the CHAP protocol

The remote computer responds with a Response packet. The Data field in this packet contains the following elements:

- The result of application of the coordinated hashing algorithm on the information structure, consisting of the verifying party's id, a numerical sequence of a Call packet, and the remote user's secret password
- The identifier of the party under verification, which can be used so the verifying party can find an appropriate identifier-password pair in its database

The length of the hashing result depends on the hash-function that is used. For the MD5 algorithm, the length of the hash code and the hashing result is equal to 128 bits. As a one-way hash function is used on the Challenge and Response intercepted packets, it is practically impossible to calculate the remote user's password.

Having received a Response packet, the RAS on the identifier of the side under verification extracts the user's private reference password from the security system database. It then puts the coordinated hashing algorithm into effect on the information structure consisting of its identifier and numerical sequence, which were sent in a Challenge packet, and the private reference password. A server further on compares the contents of the result from the received Response packet to those that were calculated independently. If these results match, the authentication is considered to be successful and the server sends a Success packet to the remote computer. Otherwise, the RAS sends a Failure packet and terminates the communication session. A data field with Success and Failure packets includes an appropriate message, whose contents and length (in bytes) are not established by the CHAP protocol.

If the server receives a Response packet, it will repeatedly send a Challenge packet. A data field with Success and Failure packets includes an appropriate message, in which neither the contents nor the length are established by the CHAP protocol. Accordingly, the party being verified must send a Response packet in reply to each accepted Challenge packet.

From the authentication scheme under the CHAP protocol, it becomes clear that the numerical sequence in a Challenge packet must be unique and undecipherable. If the given sequence is not unique, an intruder can reuse a previously intercepted

Response packet, impersonating as an authorized remote user. If the numerical sequence in a Challenge packet is decipherable, an intruder, having deciphered it, can compose an appropriate Challenge packet, send it on behalf of the RAS, and keep the received Response packet. To ensure that the numerical sequence in a Call packet is unique and undecipherable in the majority of CHAP protocol implementations, it is formed as a concatenation of two elements: the current time, including seconds, date and year, and a generated random number.

Centralized Remote Access Control

In instances when a LAN is small, one RAS suffices for control of remote connections with the network. However, if a local-area network unites substantially large segments and the number of remote users significantly increases, one remote-access server will be insufficient. When using several remote-access servers in one LAN, high efficiency of network management will be achieved when communication functions vary, and access to computer resources is placed under tight control. For centralized control of remote access, a single server should be made the authentication server (Fig. 3.26) and exclusively dedicated for authentication of remote users, definition of their rights, and collection of registration information concerned with remote access.

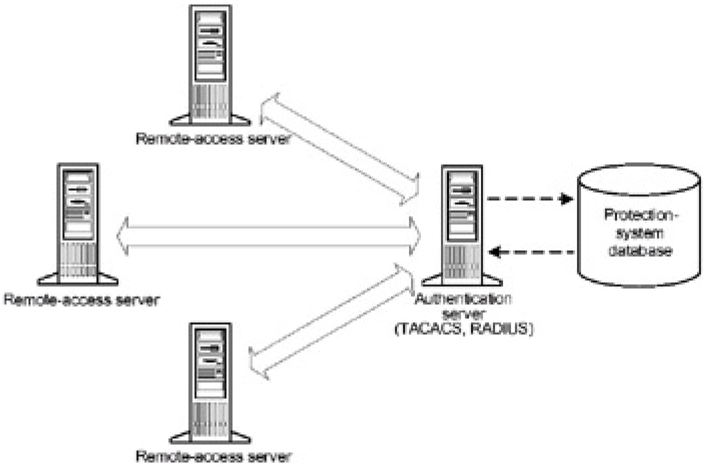

Remote-access server
Remote-access server
Authentication server (TACACS, RADIUS)
Protection-system database
Remote-access server

Fig. 3.26: Centralized remote-access control

Even if a local-area network has one remote-access server, it would be quite practical to apply a host-authentication system. Maintaining a separate database with accounts on a remote access server results in administration function redundancy and can cause inconsistency in rules of access to network resources. Effective administration and reliable protection can be increased if a remote access server requests the information necessary for authentication directly from the server, on which the general security database of a computer network is stored. For example, this could be done on the Windows NT domain controller or on the IntranetWare server.

However, in most cases, remote-access servers require a proxy (e.g. a directory service) for interaction with a security system's central database. The point is that the CHAP and PAP protocols are remote-authentication standards supported by remote-access servers, neither of which works for authentication without additional tools. The two are also unable to operate with the use of a Novell Directory Services (NDS) tree or Windows NT domain service in this context. To verify the response to a challenge faced by the server and received from the user under authentication, the implementation of the CHAP protocol uses a non-ciphered copy of the password in plain-text form. At authentication on the basis of the PAP protocol, the password is also used in plain-text mode. The majority of network operating systems and directory services keep user reference passwords with one-way hash coding. This practice does not allow remote-access servers to implement the PAP and CHAP protocols in order to extract a public reference password to verify a response.

Remote-access servers that allow interactions with the centralized database of the security system do, in fact, exist. But, as a rule, they are focused on one type of such a database and frequently require its presence directly on the remote-access server. For example, the Windows NT RAS (Remote Access Server) service will cooperate with the Windows NT domain controller registry, but only as long as the RAS server and domain controller are executed on the same server.

The role of the proxy in interactions between remote-access servers and a security system's central database can be assigned to an authentication server. Centralized control of remote access to computer resources with the help of an authentication server is carried out through specialized protocols. These protocols enable used remote-access servers and an authentication server to be united into one subsystem, implementing all control functions of remote connections on the basis of interaction with a security system's central database. The authentication server creates a unified point of supervision and verification of all remote users and supervises access to computer resources according to established rules. The TACACS (Terminal Access Controller Access Control System) and RADIUS (Remote Authentication Dial-In User Service) protocols are the most popular protocols for centralized control of remote access. In many respects, their functionalities are similar, they are open standards implemented by a large number of remote-access facilities' manufacturers.

In systems based on TACACS and RADIUS, an administrator can manage user ids and a password database, give them the access privileges, and keep account of calls on system resources. Both TACACS and RADIUS require use of a single authentication server, which besides using its own database for user authentication can also interact with modern directory services, for example, with Novell Directory Services (NDS) or Microsoft Windows NT Directory Service (NTDS). RADIUS and TACACS implementations can also serve as proxies for external authentication systems. Products based on TACACS and RADIUS provide registration of all queries for authentication and results of their execution, down to the specific port through which the user established the remote connection. Such logging functions let administrators know which users are currently connected to the RAS, and they also allow one to receive any information on previous sessions of remote connection.

Let us consider some features of centralized control of remote access by looking at the TACACS protocol.

The TACACS protocol was developed by Cisco Systems and is based on client/server technology. In a computer network including several remote access servers, one authentication server is present - which we shall simply name the

TACACS server. This server is involved with organizing a central database of remote user accounts, including their names, passwords, computers and services they can work with, and also various types of restrictions, for example, temporary restrictions. It is also possible to keep an audit database on the TACACS server that gathers registration information on each login, session time, and time of use of network resources. The clients of this server are remote-access servers accepting queries about remote users' access to network resources.

The TACACS protocol standardizes the interaction scheme of remote access servers with the TACACS server by assigning possible types of queries, responses and connections.

There are several types of queries that clients can address to the TACACS server. The server must respond to every query with an appropriate message. Several types of connections are set, each of which is defined as a sequence of pairs of queries and responses focused on the solution of a common taks. Let us consider two standard connections:

- AUTH, where only remote user authentication is carried out
- LOGIN, where authentication is executed along with the control of a logical connection of a remote computer with a LAN computer

With the help of an AUTH connection, remote-access servers redirect the traffic flow of all users' logical connections to the TACACS server. The LOGIN connection serves for redirection of queries to the TACACS server during users' logical connections to a LAN's individual computers.

At AUTH connection, the remote access server sends only one AUTH message-packet to the TACACS server, to which it responds with a REPLY to message.

An AUTH packet includes the *username, password, line* and *style* fields. The first three fields define a name and password, on which the user has established a connection. The last field can be used to specify the authentication method. With the help of available data, the TACACS server verifies the user's password and, in response, returns a REPLY packet which includes notification about the authentication's success or failure. The TACACS server can carry out authentication independently, or it can address other authetication systems, for example, the Unix authentication system, NDS NetWare directory service, Windows NT Directory Services or the Kerberos authentication system.

According to TACACS, the password is transferred between the RAS and authentication servers in open mode. Therefore, it is necessary to use TACACS together with an authentication protocol (e.g. the S/Key protocol) on one-session passwords. The LOGIN connection includes the following stages of interaction:

1. The client sends a CONNECT packet.
2. The server responds with a REPLY packet.
3. These steps are repeated 0 or more times:
   - The client sends a CONNECT packet.
   - The server responds with a REPLY packet.

- The client sends a LOGOUT packet.
- The server responds with a REPLY packet.

The LOGIN query format is the following (*username, password*, and *line*). The value of the fields is equivalent to that of an AUTH query. The server response is always (*result 1, result 2, result 3*), where all three fields are integers whose values are not stipulated in the protocol. For example, in Cisco remote-access servers, the field *result 3* corresponds to the number of the access rights' list, which needs to be applied to the given user.

A CONNECT query appears as (*username, password, line, destination IP*, and *destination port*), where the assignment of the first three fields is the same as in the previous queries, and the last two fields identify the computer's IP address and TCP port. A CONNECT query is transferred on the connection already established with the user and, therefore, usually the password is not indicated in it. A query is intended for verifying, whether or not the user is allowed to connect to a specified IP address. The server response is the same type as that for a LOGIN query.

A LOGOUT query is transferred to notify the TACACS server about termination of the user session. Answering, the server confirms that the notice has been received.

The TACACS protocol's imperfections, which are related to the public transfer of a password on a network, have been eliminated by Cisco in its release of a version named TACACS+. According to the TACACS+ protocol, passwords for transfer on a network are ciphered with the help of the MD5 algorithm. TACACS+ provides separate storage of authentication, authorization and registration information databases located on different servers. Interaction with the Kerberos system has also been improved.

It is important to note that, according to the RADIUS protocol, only the passwords for transfer between RAS authentication servers are ciphered, and, according to TACACS+, it is possible to encrypt all traffic. To improve safety, RADIUS and TACACS protocols also support CHAP.

Though TACACS and RADIUS are practically equivalent in regard to their functionalities, and both are open standards, RADIUS is more popular among manufacturers of RAS-control systems. This has do to with the fact that its server software is distributed free of charge. Besides, TACACS has fallen into disfavor among manufacturers competing with Cisco, as its newest version, TACACS+, is that company's patented technology. Also, RADIUS is less complicated in its implementation. For example, for interaction between the RAS and the authentication servers TACACS uses TCP, while RADIUS utilizes simpler, but less reliable, UDP protocol.

Protocols of centralized control of remote access can essentially differ from each other in terms of how specifically they are implemented. Therefore, when choosing specific implementation, one should pay attention to availability of required authentication services, support of tunneling protocols, opportunity of callback and definition of various restrictions, in addition to audit efficiency. Table 3.1 provides an overview of the remote-access services' generalized characteristics supporting protocols of centralized control of remote connections.

Table 3.

Product, manufacturer	RADIUS/TACACS+	Basic number of access ports	Max. number of access ports	Max. number of connections	V.34/V.34 bis/V.90	BRI/PRI	T
Total Control Multiservice Access Platform with HiPer Access System and Edge−Server Pro Module (3Com)	+/+	4 analogous or 1 PRI	360 analogous or 15 PRI	360	+/+/+	−/+	+
Max 6000 (Ascend)	+/+	96 analogous or 4 PRI	96 analogous, 4 PRI or 48 BRI	96	+/+/+	+/+	+
X1600 (Assured Access Technology)	+/−	32 analogous and 4 PRI	2016 analogous or 364 PRI	8372	+/+/+	+/+	+
CyberSwitch CSX7010 (Cabletron Systems)	+/+	24 analogous or 1 PRI, or 4 BRI	144 analogous, 6 PRI or 24 BRI	144	+/+/+	+/+	+

Table 3.

Product, manufacturer	RADIUS/TACACS+	Basic number of access ports	Max. number of access ports	Max. number of connections	V.34/V.34 bis/V.90	BRI/PRI	T
AS5300 (Cisco Systems)	+/+	48 analogous or 2 PRI	192 analogous or 8 PRI	192	+/+/+	−/+	+
DCS−5000 (Computone)	+/−	24 analogous and 1 PRI	288 analogous and 12 PRI	564	+/+/+	+/+	+
NEVADA (ECI Telecom)	+/−	12 analogous and 1 PRI	120 analogous or 5 PRI	120	+/+/+	−/+	+
LanRover Access Switch DPS (Intel)	+/+	12 analogous and 1 PRI	72 analogous and 1 PRI, 48 analogous and 4 PRI	240	+/+/+	−/+	+

Table 3.

Product, manufacturer	RADIUS/TACACS+	Basic number of access ports	Max. number of access ports	Max. number of connections	V.34/V.34 bis/V.90	BRI/PRI	T
			or 11 PRI				
PortMaster 4 (Lucent Technologies)	+/−	96 analogous or 4 PRI	864 analogous or 36 PRI	864	+/+/+	−/+	+
EdgeCommander (Mediagate)	+/+	48 analogous or 2 PRI	96 analogous or 4 PRI	96	+/+/+	−/+	+
XpressWay RLAN (Mitel)	+/−	24 analogous, 1 PRI or 8 BRI	120 analogous, 10 PRI or 64 BRI	230	+/+/+	+/+	+
CommPlete Communications (Multi−Tech Systems)	+/+	24 analogous or 1 PRI	192 analogous 4 PRI	192	+/+/+	+/+	+

Table 3.

Product, manufacturer	RADIUS/TACACS+	Basic number of access ports	Max. number of access ports	Max. number of connections	V.34/V.34 bis/V.90	BRI/PRI	T
EdgeBlaster (NBase−Xyplex)	+/−	8 analogous and 2 PRI	8 PRI, 64 might be analogous	184	+/+/+	−/+	+
5399 Remote Access Concentrator (Nortel Networks)	+/+	48 analogous or 2 PRI	576 analogous or 24 PRI	576	+/+/+	−/+	+
IQX−200 (Osicom Technologies)	+/−	8 analogous, 1 PRI or 4 BRI	96 analogous, 4 PRI or 16 BRI	96	+/+/+	+/+	+
833AS Remote Access Switch (Perle Systems)	+/−	12 analogous or 1 PRI	90 analogous or 4 PRI	92	+/+/+	−/+	+

Table 3.

Product, manufacturer	RADIUS/TACACS+	Basic number of access ports	Max. number of access ports	Max. number of connections	V.34/V.34 bis/V.90	BRI/PRI	T
RAServer 2100 (RAScom)	+/−	4 analogous or 4 BRI	60 analogous, 2 PRI or 24 BRI	60	+/+/+	+/+	+

3.7. *Overview of VPN Tools*

3.7.1. General Overview

Protocols used to create Virtual Private Networks (VPNs) can be implemented by network tools belonging to different categories:

- Remote-access servers (RAS), allowing the creation of protected tunnels at the channel level of the OSI model
- Routers able to support VPN at the channel and network levels of the OSI model
- Firewalls, possibly including RAS servers and allowing the creation of VPN at the channel, network and session levels of the OSI model
- Specialized software, capable of creating VPN at the network and session levels of the OSI model
- Specialized combinations of hardware and software, oriented towards creating protected tunnels at the channel and network levels of the OSI model

Remote-access servers can include functions for creating protected tunnels when organizing remote user access to local-area networks. Most often, such servers support tunneling protocols such as PPTP, L2F and L2TP, corresponding to the network level of the OSI model. It should be noted that not every manufacturer or vendor of RAS servers that are capable of creating protected tunnels to remote computers also supplies appropriate client software. Because of this reason, and due to the fact that, most often, remote computers run various versions of the Windows operating-system family, it makes sense to choose RAS servers supporting PPTP or L2TP protocols. Implementation of PPTP is included with Windows 9*x*/NT/2000. A more advanced tunneling protocol, L2TP, is implemented in Microsoft Windows

2000/XP. Because quite a significant amount of focus was put on autonomous RAS servers in the previous section, we will not spend much more time discussing them.

Routers can also support functions for creating protected tunnels by default or as an additional feature, provided separately. These devices are frequently oriented towards the creation of VPNs using the L2F, L2TP and IPSec protocols, corresponding to the channel and network levels of the OSI model. When choosing the router as a tool for creating VPN, one should pay attention to its performance and workload. If the router processor works with 80% workload without performing VPN functions, then adding a large number of protected tunnels will degrade traffic performance.

Since they can include an RAS, firewalls can be used as efficient tools for creating VPNs. Taking into account the fact that firewalls are specially intended for information interaction protection between public networks, one can draw the conclusion that, when implementing VPN functions using these devices, complex protection of the information exchange will be ensured. Firewalls can support any existing protocols for creating protected tunnels. At the channel layer of the OSI model, it is possible to implement PPTP, L2F and L2TP protocols; at the network layer, IPSec and SKIP; while, at the session layer, SSL/TLS and SOCKS.

The ability to control both plain-text and encrypted traffic is an important feature of firewalls. Access control from the firewall to all the traffic, including that which is tunneled, provides a higher protection level of network interaction. Such control is especially efficient when there is an object at the other side of the tunnel whose security strategy is unknown, or is not particularly reliable. When it is necessary to control the tunneled traffic and protect the message flow up to the recipient within the LAN, the endpoint of the main tunnel must reside at the firewall, which, after the control traffic decryption, must perform re-encryption of transmitted message packets. Thus, one of the advantages of using tunneling products closely integrated with the firewall relates to the fact that one can open a tunnel, apply firewall security rules to it, encrypt the data once again, and redirect the traffic to the recipients located within the LAN protected by the firewall. However, if the firewall has a low bandwidth, VPN implementation will only make the situation worse, because of the additional calculations that will be necessary to carry out.

In order to implement protocols for creating protected tunnels specialized software and hardware devices are available. Software tools, as compared with the complexities of hardware and software tools, are more flexible, since, for relatively low expense, they are able to update and correct bugs quickly. Some software products running on specific platforms can also perform some firewall functions and even cache Web pages. Combinations of specialized hardware and software, on the other hand, are distinguished by superior performance. Notice that these devices can also be flexible and support multiple functions. Combined multifunctional devices normally include router, firewall, bandwidth management tools and VPN tools. Such devices can be classified as multifunctional firewalls, and they are usually very easy to maintain. Of course, it is much easier to configure and maintain a single integrated user interface than perform the same operations separately for each component, including the router, firewall, VPN and bandwidth management module. However, in multifunctional devices, high performance of specific components is often achieved at the expense of the other devices.

When choosing VPN tools, it is important to take into account such characteristics as functionality, reliability, flexibility, performance, manageability and compatibility. Table 3.2 provides a generalized overview of advantages and drawbacks of various categories of VPN tools.

Table 3.2: Advantages and Drawbacks of Different VPN tool Categories

Category	Advantages	Drawbacks
Router-based VPN	VPN support functions can be built into routers, which will not require additional expenses for purchasing the separate tools implementing these features. VPN administration becomes easier.	VPN operation might have negative impact on other traffic. The channel between the information recipient within the LAN and the router might become vulnerable.
VPN software for firewalls	It is possible to control tunneled traffic. Efficient VPN administration. Complex protection of information exchange. No hardware redundancy for network security tools.	Operations related to data encryption might result in high processor workload and, consequently, degrade the firewall's performance. If the protected tunnel ends at the firewall, the channel between the information recipient within the LAN and the firewall might become a weak link in the network security system. After updating server products, hardware will need upgrading.
VPN based on special software	Capabilities of updating. Errors can be eliminated quickly. No special hardware devices are required.	VPN might require a separate application or even dedicated directory. Hardware might require upgrading to increase performance.
VPN on the basis of specialized software and hardware	Superior performance. Multifunctional combinations of software and hardware simplify configuration and	In multifunctional devices, high performance of one component is often achieved at the expense of others, degrading performance. Specialized devices can require special tools for administration and

Table 3.2: Advantages and Drawbacks of Different VPN tool Categories

Category	Advantages	Drawbacks
	support.	directories.
	Specialized software and hardware devices provide advanced customization capabilities for maximum performance.	Often, performance improvements for such devices are expensive or even impossible. The channel between the information recipient within the LAN and traffic encryption device can become a weak link in the security system.

One of the most promising VPN tools is the IPSec protocol. With its implementation, it is desirable to choose VPN tools supporting this protocol. However, one should keep in mind that the IPSec standard is relatively new, and, as a result, does not guarantee compatibility of solutions from different manufacturers. Developers supporting IPSec have only just started to work on compatibility problems. Thus, when selecting the tool, one should contact its manufacturer and request information on compatible products, especially if one is going to connect LANs where VPN tools from different manufacturers are used. Non-standard tunneling schemes and encryption algorithms are only good in one case, namely, if there is no need to communicate with other systems. However, as the LANs continue to grow, so the various aspects of scalability and compatibility are becoming matters of primary importance for any organization.

The technology of building virtual private networks is based on user authentication and cryptographic protection of the information. The highest efficiency of these functions is provided by the combined usage of symmetric and asymmetric cryptographic systems. Combined usage of single key and multiple key ciphers is provided in IPSec and SSL/TLS tunneling protocols that suppose the presence of Public Key Infrastructure (PKI). This infrastructure, providing open key management and user authentication based on digital certificates, will gradually become more and more important. The best practice is storing PKI information in a global directory that can be accessed using Lightweight Directory Access Protocol (LDAP). The creation of additional directories for storing such information will also require additional overhead. Because of this, most manufacturers prefer to include PKI contents into existing directory implementations, such as Windows 2000/XP Active Directory or NetWare NDS. Using a unified directory service managing all organizational information within a computer network results in significant improvement of the efficiency network administration, as well as increased productivity of end-users. A centralized directory, capable of interacting with other directories and applications, must store security-policy data, as well as performance management information.

When selecting VPN tools, one should remember that encryption is resource-consuming. For example, Pentium-class servers provide satisfactory encryption performance for a 10 Mbit/sec communication channel, but, at 100 Mbit/sec, performance is not sufficient. To provide a high speed of encryption, some manufacturers provide specialized hardware accelerators for standard platforms. However, usage of specialized hardware accelerators reduces flexibility of the tools for creating protected tunnels. Because of this, the development and implementation of fast cryptographic algorithms is a more attractive solution.

3.7.2. Creating Router-Based VPNs

Because all packets transmitted from the local-area network pass through the router, it can also used for encrypting these packets. Furthermore, the router can also decrypt the incoming traffic. Currently, most leading manufacturers and vendors of router equipment include VPN support with their products. On the other hand, most manufacturers of specialized VPN devices also complement them with support for routing protocols, which allows such devices to be used as routers.

Cisco Systems has complemented its Internetwork Operating System (IOS), developed for Cisco routers with support for L2TP and IPSec protocols (starting with version 11.3). The L2F protocol became an IOS component even earlier, and is now supported in all Cisco routing and remote-access devices. The VPN technology developed by Cisco Systems is distinguished by its flexibility and high performance. Cisco products provide the capabilities of tunneling with encryption for any IP traffic, transmitted in "pure" or encapsulated form. A protected tunnel is created on the basis of a specified source and target addresses, TCP/UDP port numbers and IP Quality of Service parameters. If the LAN contains a router with the IOS version, supporting L2TP and IPSec protocols, it is possible to install add-on software for data encryption. If encryption performance is critical, it is recommended that one use the Cisco ESA adapter (Encryption Service Adapter). This adapter contains a specialized encryption coprocessor.

Similar to other data-encryption devices, the ESA adapter not only encrypts information but also prevents and detects intrusions by reacting to all suspicious situations. If one simply removes the adapter from the router (even if one powers it down before doing so), the "Tamper" LED will turn on, and only support personnel will be able to restart the router. To restart the router, it is necessary to know the password installed when the router is first brought into operation or to be prepared for the adapter RAM to be cleared. If the ESA module is opened, a special switch will be activated, and RAM contents will be wiped out. Data encryption at the hardware level allows a significant improvement in performance, and reduces the negative influence of protected tunnels on the router's throughput.

Cisco Systems also manufactures Cisco VPN Concentrator, a specialized VPN concentrator which supports routing protocols. This product ensures the creation of encrypted tunnels between LANs and between LANs and remote users. To allow a remote user to participate in VPN, Cisco VPN client software must be installed on their computer.

The most common VPN concentrators from Cisco Systems are Cisco VPN 3005 Concentrator and Cisco VPN 3030 Concentrator. Cisco VPN 3005 Concentrator supports 100 encrypted tunnels with software encryption. The device has two network interfaces, and is not subject to a hardware upgrade.

Cisco VPN 3030 Concentrator uses a scalable hardware module, Scalable Encryption Processor (SEP), which is able of simultaneously supporting 1500 protected connections. This device is supplied with or without functional backup. The configuration with functional backup includes two power supplies and two SEP modules. Besides normal input and output ports, there is a third interface that can be placed within the network segment, isolated from any other traffic, or used for workload balancing.

Cisco VPN concentrators have advanced management functions, flexible monitoring capabilities, and high encryption speed (about 1,5 Mbit/sec). These hardware devices can register events in a log file and generate SNMP traps. The administrator can control sessions in real-time mode and filter events by groups. If software residing in ROM gets corrupted, the concentrator reboots from additional flash memory, thus reducing the time required for system recovery.

Cisco VPN client software enables administrators to automate most stages of the deployment process, by customizing the template files for a specific environment. One can even create a script for Microsoft Dial-Up Networking, after which the users will be able to connect to the LAN by simply clicking specific button. The client software is quite clever: for example, it detects the backup concentrator if the primary device fails.

3.7.3. Creating Protected Tunnels Using Firewalls

All network traffic passes through the firewall, as well as through the router. Consequently, the firewall can also be delegated the functions of encrypting the outbound and decrypting the inbound traffic. Currently, all leading manufacturers of firewalls include VPN functions with their products. However, it should be noted that VPN functionality can be implemented either in add-on modules or in the form of specialized products purchased separately.

For example, Symantec Enterprise Firewall does not in itself provide VPN functions, since Symantec distributes another product for this purpose: Symantec Enterprise VPN, which enables the creation of a VPN and integration with Symantec Enterprise Firewall. On the other hand, several firewalls typically include VPN modules. This group of firewalls includes Firebox 1000 from WatchGuard, and Secure PIX Firewall 520/525 from Cisco Systems.

To create a VPN using Firewall-1, it is necessary to purchase a specialized VPN module, also developed by Check Point. Besides this, the company supplies VPN-1 combining Firewall-1 and VPN tools. FireWall-1/VPN-1 ensures control over plaintext and encrypted traffic. Using a VPN module, the firewall decrypts the incoming data, then applies access-control rules specified by the system administrator, and, once again, encrypts the data packets that must be transmitted further. Besides traffic encryption, FireWall-1/VPN-1 also authenticates message packets. For distributed keys, it is possible to use the IPSec standard, as well as the SKIP protocol developed by Sun Microsystems. The product implements symmetric cryptographic systems such as DES, RC4 and FWZ1. The FWZ1 cryptographic system was originally developed by Check Point. For message authentication, it is possible to use MD5, SHA-1, CBC DES and MAC algorithms. Two encryption modes are supported:

- Encryption of the traffic between FireWall-1/VPN-1 modules transmitted via the Internet
- Encryption of remote network connections to the LAN protected by FireWall-1/VPN-1

In the first case, all encryption functions are transparently implemented by the communicating Firewall-1/VPN-1 systems. In the second case, encryption is performed by the FireWall-1/VPN-1 firewall protecting the LAN, to which the user connects remotely, and the special SecuRemote component that must be installed on the remote computer. In computers with the PCI bus, it is possible to use the add-on hardware accelerator supplied by Check Point.

Finally, there is also hardware implementation of the Check Point FireWall-1/VPN-1 system on the Nokia hardware platform. This combined device-Nokia Firewall/VPN appliances-joins firewall and VPN tools developed with the cooperation of Check Point and Nokia.

The Symantec Enterprise VPN product integrated with the Symantec Enterprise Firewall, is equipped to create virtual private networks with the IPSec protocol. Like FireWall-1/VPN-1, the Symantec Enterprise Firewall/VPN can apply predefined access control rules to the tunneled traffic. Symantec also provides a family of mobile VPN clients.

Besides this, Symantec, like Check Point, has started to manufacture combined hardware devices implementing Symantec Firewall and Symantec VPN. Its three new products, known as Symantec Firewall/VPN Appliances 100, 200 and 200R, have allowed these companies to increase the security of network communications between remote affiliations and to encrypt all data transmitted between the company and small offices connected by the VPN.

The Symantec Firewall/VPN Appliances 100 device, intended for small affiliations, has 1 WAN and 4 LAN ports. It provides 15 protected connections at a time. The 200 and 200R devices include 2 WAN and 8 LAN ports to support 30 encrypted tunnels. Only the 200R device supports remote clients to VPN gateway. Management of the 100, 200 and 200R devices is implemented via a local or remote Web console.

The Firebox 1000 firewall, supplied by WatchGuard, supports encrypted tunnels both between LANs and between LANs and remote users. The IPSec protocol is used for creating VPN between LANs. For organizing protected remote-access connection, the product supports the PPTP protocol in addition to IPSec. Usage of the IPSec for protected remote access requires purchasing additional user licenses and VPN client software-Mobile User VPN Client. FireBox 1000 simultaneously supports up to 1000 VPN clients.

3.7.4. Creating Software-Based VPNs

Specialized software tools are widely used for creating virtual private networks. Recently, quite a wide range of such software products has appeared on the market. All VPN software products allow users to create protected tunnels without additional hardware, and thus turn the server where they run into the TCP/IP router that receives the encrypted packets. It then decrypts them and transmits them further via the LAN to their destination points. Let us briefly discuss some of these products.

The easiest way to create a software-based VPN is to use the built-in capabilities of Microsoft Windows 2000, which supports VPN solutions both for remote connections to the protected LAN and for communications between LANs. To create a VPN, one can use the PPTP, L2TP or IPSec protocols directly in tunneling mode. The VPN concentrator is dedicated to a computer on Windows 2000 Server with two network interfaces and running the Routing and Remote Access Service (RRAS), regardless of the protocol used. One of the network interfaces must provide a physical connection to the Internet via an Internet Service Provider, for example, by means of T1 or T3 communication channel or via a modem. The OS (Windows 2000 Server) characteristics define the reliability and availability of this VPN solution. The administrator will have to customize quite a lot of parameters, search and eliminate all vulnerabilities, and update system software when new Service Packs are released.

3.7.5. Tunneling Based on Specialized Combinations of Hardware and Software

Hardware-based encrypted tunneling tools can provide the best performance. Table 3.3 outlines a general overview of characteristics of specific VPN products.

Table 3.3: General Overview of VPN Products

Characteristic/Tool	**Cisco VPN 3005 concentrator (Cisco Systems)**	**Cisco VPN 3030 concentrator (Cisco Systems)**	**FireWall-1/VPN-1 gateway (Check Point)**	**Firebox 1000 (Watch-Guard)**	**Windows 2000 server (Microsoft)**
Implementation tools					
Router	+	+			
Firewall			+	+	
Specialized software					+
Combination of specialized software and hardware					
Types of protected channel					
LAN-LAN	+	+	+	+	+
Remote workstation-LAN	+	+	+	+	+
Workstation-					

Table 3.3: General Overview of VPN Products

Characteristic/Tool	**Cisco VPN 3005 concentrator (Cisco Systems)**	**Cisco VPN 3030 concentrator (Cisco Systems)**	**FireWall-1/VPN-1 gateway (Check Point)**	**Firebox 1000 (Watch-Guard)**	**Windows 2000 server (Microsoft)**
Workstation					
Workstation-Server					
VPN protocols					
PPTP	+	+	+	+	+
L2TP	+	+	+		+
IPSec	+	+	+	+	+
SSL					
Proprietary					
Encryption algorithms					
DES/3DES	+	+	+	+	+
RC4				+	
RC5			+		
Blowfish			+		
CAST			+		
Capability of centralized distribution of the cryptographic keys	+	+	+	+	+
Routing support	+	+	+	+	+
Traffic filtering at the network level	+	+	+	+	
VPN-clients					
Windows 9*x*	+	+	+	+	+
Windows NT/2000	+	+	+	+	+
Custom VPN client for Windows	+	+	+	+	
Centralized administration capabilities	+	+	+	+	+
Certification					

Table 3.3: General Overview of VPN Products

Characteristic/Tool	Cisco VPN 3005 concentrator (Cisco Systems)	Cisco VPN 3030 concentrator (Cisco Systems)	FireWall-1/VPN-1 gateway (Check Point)	Firebox 1000 (Watch-Guard)	Windows 2000 server (Microsoft)
ICSA			+	+	

Bibliography

1. Steven Brown. Implement Virtual Private Networks. Osborne McGraw-Hill; ISBN: 007135185X; 1st edition (May 1, 1999).
2. John R. Vacca. *Intranet Security*. Charles River Media; ISBN: 1886801568; Bk&Cd-Rom edition (July 1997).
3. Sergei Dunaev. *Advanced Internet Programming: Technologies & Applications*. Charles River Media; ISBN: 1584500603; Bk&Cd-Rom edition (September 17, 2001).
4. A. Lukatsky. Protect Your Information with Intrusion Detection. A-LIST Publishing; ISBN: 1931769117; (November 2002).
5. Stuart McClure, Joel Scambray, George Kurtz. *Hacking Exposed: Network Security Secrets & Solutions*. Osborne McGraw-Hill; ISBN: 0072193816; 3rd edition (September 26, 2001).
6. Sead Muftic. *Security Mechanisms for Computer Networks*. Ellis Horwood Ltd; ASIN: 0137991800; (January 1989).
7. Stephen Northcutt. *Computer Security Incident Handling Step by Step*. SANS Institute; ISBN: 0967299217; reprint edition (May 30, 1998).
8. Stephen Northcutt, Donald McLachlan, Judy Novak. *Network Intrusion Detection: An Analyst's Handbook*. New Riders Publishing; ISBN: 0735710082; 2nd edition (September 22, 2000).
9. Terry William Ogletree. Practical Firewalls. Que; ASIN: 0789724162; 1st edition (June 12, 2000).
10. Timothy Parker. *Teach Yourself TCP/IP in 14 Days*. Sams Publishing; ASIN: 0672308851; 2nd edition (August 1, 1996).
11. William Stallings. *Cryptography and Network Security: Principles and Practice*. Prentice Hall; ISBN: 0138690170; 2nd edition (July 15, 1998).
12. Mark Joseph Edwards. *Internet Security With Windows NT*. 29th Street Pr; ASIN: 1882419626; Bk&Cd Rom edition (November 1997.
13. Michael Howard, Mark Levy, Richard Waymire. Designing Secure Web-Based Applications for Microsoft Windows 2000. Microsoft Press; ISBN: 0735609950, July 2000.
14. Jamie Jaworski, Paul Perrone. Java Security Handbook. Sams; ISBN: 0672316021; 1 edition (September 21, 2000).