Ministry of Education and Science of the Republic of Kazakhstan

M.Z. Yakubova, B.M. Yakubov, Y.R. Gabdulina

PROTECTION OF INFORMATION IN TELECOMMUNICATION SYSTEMS

Tutorial

Almaty

2017

UDC 621.39 (075)
LBC 32.84ya73
Ya 49

Reviewers:
Candidate of Technical Sciences, Professor of the Department "Computer and Program Engineering", "Turan" university, B.S. Kubekov
Candidate of Physical and Mathematical Sciences, Assistant Professor of the Department "Radio engineering, electronics and telecommunications", M. Tynyshpaev Kazakh Academy of Transport and Communications, M.A. Seidakhmetov
Candidate of Technical Sciences, Assistant Professor of the Department "Radio engineering and informational security", Almaty University of Power Engineering and Telecommunications, V.V. Artyukhin

It is recommended for publication by the Academic Council of the Almaty University of Power Engineering and Telecommunications (Minutes No. 4 of December 27, 2016). It is printed on
the thematic plan for the issuance of departmental literature for AUPET for 2017, position 48.

M.Z. Yakubova, B.M. Yakubov, Y.R. Gabdulina

Ya 49 Information security in telecommunication systems: Textbook (for students of the specialty 5B100200 – Systems of information security and 5B071900 – Radio engineering, electronics and telecommunications) / M.Z. Yakubova, B.M. Yakubov, Y.R. Gabdulina. – Almaty. Almanakh, 2017. – 75 pages: Table. 2, ill. 31, bibliograms. – 12 titles.

ISBN 978-601-7900-73-1

The proposed training manual is designed to study emerging problems when considering tasks that consider network security, and the use of encryption algorithms.

This tutorial is of interest to students and undergraduates studying in universities, users and administrators of computer systems and networks, business leaders interested in the security of their corporate information systems and networks and university professors of relevant specialties.

Ya1.1:140M.B. **UDC 621.39 (075)**
LBC 32.84ya73

ISBN 978-601-7900-73-1

Content

INTRODUCTION

Protection of information in telecommunication systems

The manual is devoted to methods and means of multilevel information protection in telecommunication networks. The basic concepts of the telecommunications network and the protection of information in it are formulated.

At present, to ensure the protection of information, it is not only necessary to develop private protection mechanisms, but to implement a systematic approach that includes a set of interrelated measures (use of special technical and software tools, organizational measures, regulations, moral and ethical countermeasures, etc.). The complex nature of protection stems from the complex actions of intruders, seeking to extract important information for them by any means.

The urgency and importance of the problem of information security and the security of telecommunications systems (TCS), classification of threats to information security of TCS, possible channels for its leakage, model of a likely violator, goals and possible scenarios of unauthorized access to TCS are described. The threats to information security in computer information systems and telecommunication networks are analyzed.

Attention is paid to the network components being attacked and their protection, classification of cryptoalgorithms and various types of encryption, management of cryptographic keys, key generation, protection of networks from remote attacks, methods and means for creating virtual secure channels and networks.

This tutorial is of interest to students and undergraduates studying in universities, users and administrators of computer systems and networks, business leaders interested in the security of their corporate information systems and networks and university professors of relevant specialties.

Preface

The problem of secret messaging exists for as long as there is written language, moreover, originally the writing itself was a method of secret information transfer, since it was available only to the select ones. To implement the secret transfer of a message from one addressee to another there are two directions: firstly, you can try to hide the fact of the message transmission, for example, by methods of cryptography; secondly, you can transform the message so that the information contained in the message is not available to the unauthorized person. The first direction is steganography, the second one is cryptography.

Let us note that no sphere of life of modern society can function without a developed information structure. The national information resource is today one of the main sources of economic and military power of the state. For years, cryptography was served exclusively military purposes. Today, ordinary users have the opportunity to access tools that enable them to protect themselves from unauthorized access to confidential information using computer-based cryptography techniques. Penetrating into all spheres of state activity, information acquires a concrete political, economic and material expression. Against this background, the task of ensuring the protection of information of the Republic of Kazakhstan as an integral element of its national security has become more and more relevant in recent decades and, especially now, and the protection of information is becoming one of the priority state tasks.

Issues of information protection have always occupied a special place in any society and state. At present, when the avalanche-like distribution of computer systems and their interaction through telecommunication networks is preserved, the protection of user information and service information is one of the first places.

In this regard, the manual is aimed to provide the theoretical and practical basis necessary to understand the principles of using "Information Security in TCS", used in telecommunication systems and networks in the transmission of information.

1. General Provisions

1.1 Basic Definitions

We introduce some concepts and definitions that are needed in what follows.

Figure 1.1 shows the classification of the main definitions and concepts of the subject area "Information Protection" [1, 2].

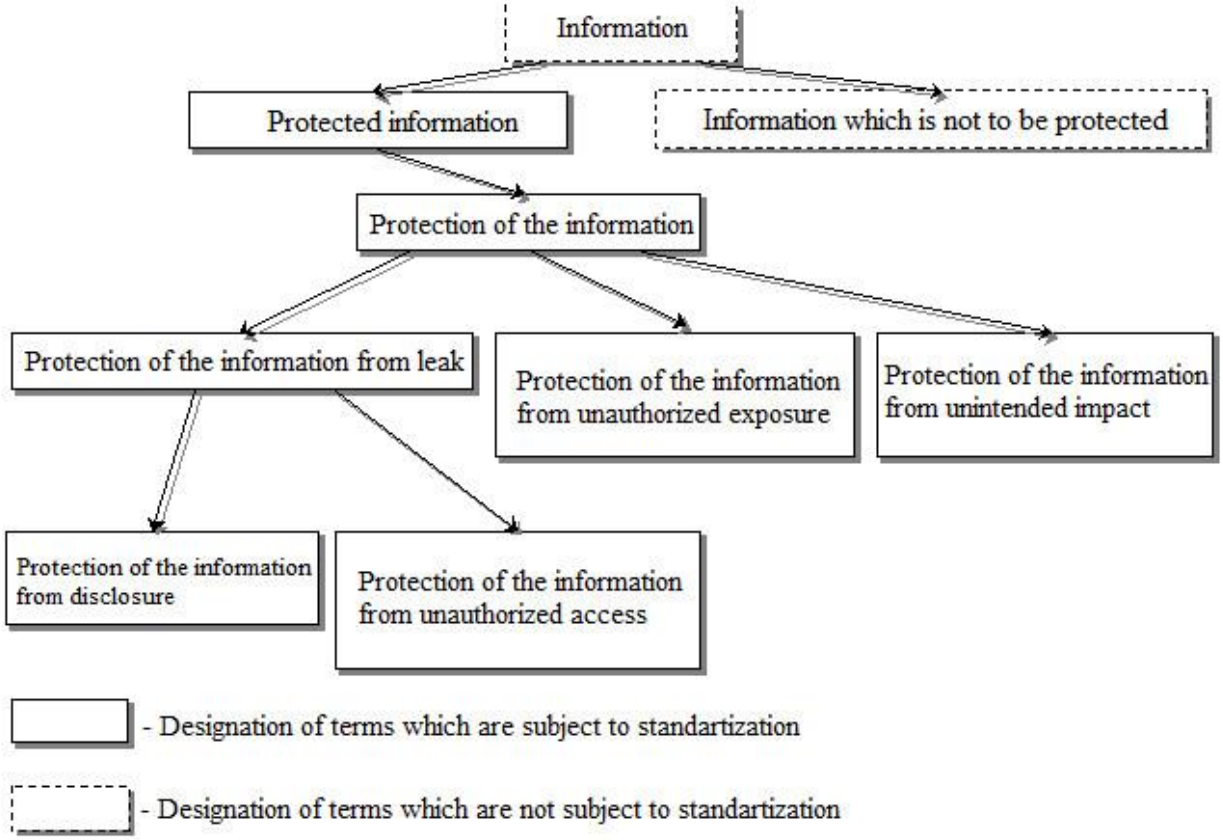


Figure 1.1 – Classification of definitions and concepts of “Protection of the information” subject field

Information [1] is information about persons, objects, facts, events, phenomena and processes, regardless of the form of their presentation.

Protected information is the information that is the subject of property and is to be protected in accordance with the requirements of legal documents or requirements established by the owner of information. The owner of the information can be the following: a state, a legal entity, a group of individuals, an individual.

Information security is activities aimed at preventing leakage of protected information, unauthorized and unintentional impacts on the protected information.

Protection of information from leaks - activities aimed at preventing uncontrolled dissemination of protected information as a result of its disclosure, unauthorized access to information and obtaining of protected information by intelligence services.

Protection of the information from unauthorized exposure is activities aimed at preventing the impact on protected information with violation of the established rights and (or) rules for changing information, leading to its distortion, destruction, blocking access to information, as well as loss, destruction or malfunction of the media information carrier.

Protection of information from unintended impact - activities aimed at preventing the influence of the user's errors, the malfunctioning of the hardware and software of information systems, natural phenomena or other actions that are not targeted for changing information, leading to distortion, destruction, copying, blocking access to information, as well as to loss, destruction or malfunction of the information carrier on the protected information.

Protection of the information from disclosure is activities aimed at preventing unauthorized disclosure of protected information to consumers who do not have the right of access to this information.

Protection of the information from unauthorized access is activities aimed at preventing the receipt of protected information by an interested entity in violation of the rights or rules for access to protected information established by legal documents or by the owner, the owner of information.

1.2 Requirements for telecommunication systems

Here are the basic requirements of users for telecommunications systems from the point of view of ensuring the protection of transmitted information. Telecommunication systems should provide:

1. Confidentiality of the information - ensuring that the information is viewed in an acceptable format only for users who have the right of access to this information;
2. Integrity of the information – ensuring that the information remains unchanged when it is transmitted;
3. Authenticity of the information – ensuring reliable identification of the source of the message, as well as ensuring that the source is not counterfeit;
4. Accessibility of the information – guaranteeing access of authorized users to information.

1.3 Classification of violations of information transmission

Normal transmission of the information (Figure 1.2a)) in networks with guaranteed quality of user service implies the implementation of three stages (Figure 1.2a)) [3, 4].

1. In the management plane - the formation and correction of databases (DB) on the status of network elements. The end result of functioning of this stage is the formation of an information distribution plan on the network - the calculation of the routing tables (RT) in all nodes for each telecommunication service.
2. In the control plane (a stack of signaling protocols) - the organization of the route between the source node (SN) and the destination node (DN) in the form

of a virtual switched or permanent connection (channel or path). The end result of functioning of this stage is the filling and zeroing of the switching tables (SwT).

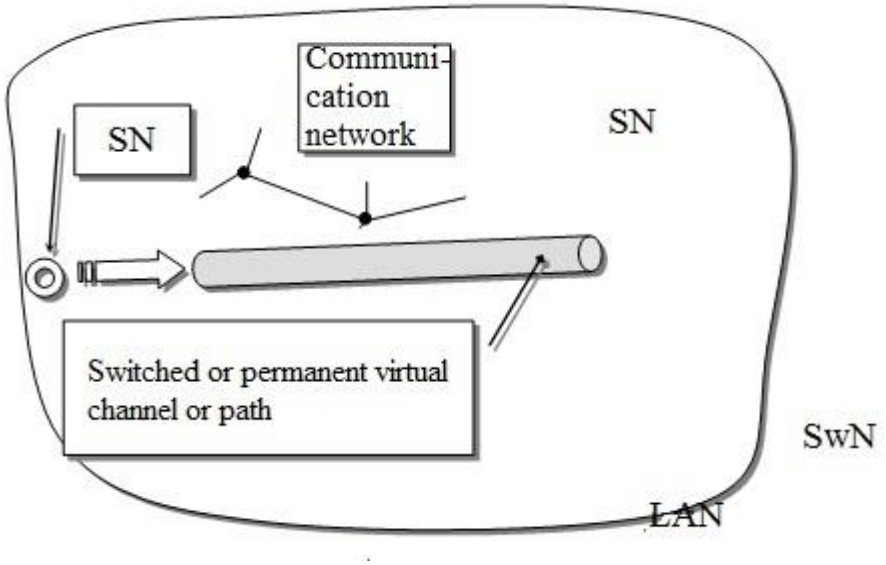
3. In the user plane - direct transmission of the user information.

In this case, the transfer of all types of the information in the network (service one - for the formation of the database and the SwT, user one) is carried out on its own separate virtual connections (channels and paths).

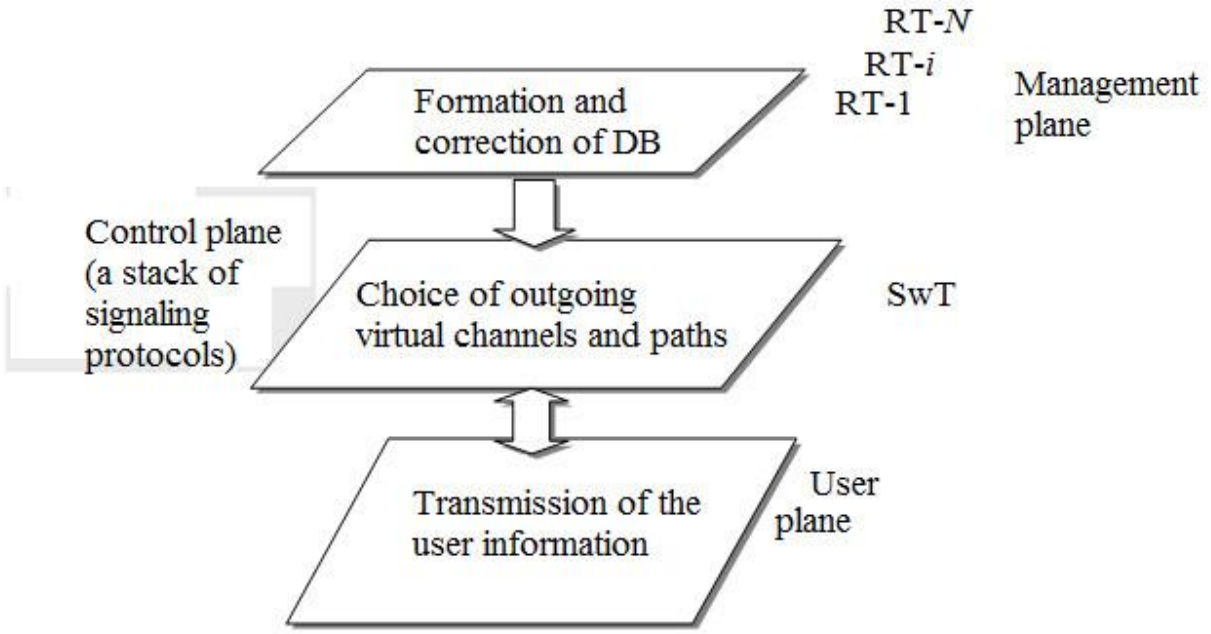
As the violation of the transfer of information, we will understand one of the situations that can be organized by the offender (Figure 1.3).

- Interruption or disconnection (Figure 1. 3. a)). Information is destroyed or becomes inaccessible or unusable. In this case, the availability of the information is violated. An example of such violations may be the influence of the intruder on the elements of the network (communication lines (LANs), switching nodes (SwNs), control devices, databases, etc.) with the aim of destroying them or putting them into a non-operational state.

- Interception (Figure 1. 3. b)). Unauthorized access is provided to the information. Confidentiality of the transmitted information is violated. An example of this type of violation is unauthorized connection to the communication channel.



a)

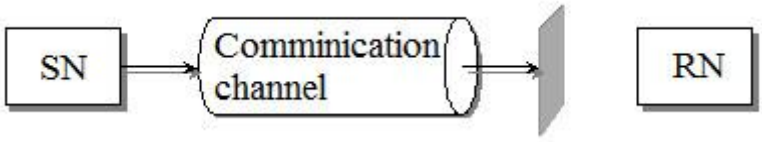


b)

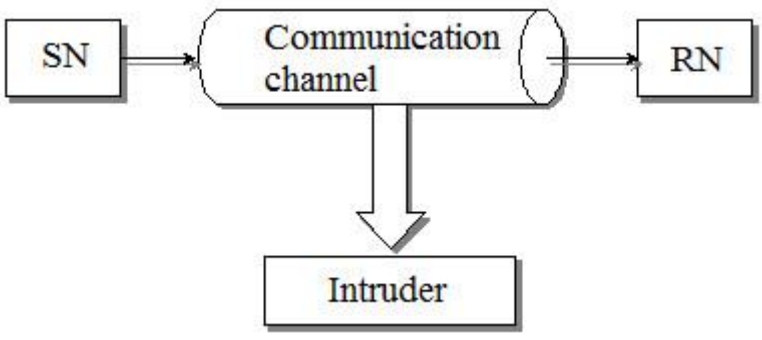
Figure 1.2 – Normal transmission of the information in the network with guaranteed quality of user service

- Modification (Figure 1. 3. c)). The unauthorized access to the information is opened with the aim of changing the information. At the same time, confidentiality of transmitted information and its integrity are violated. The purpose of this type of violation is to change the information transmitted over the network.

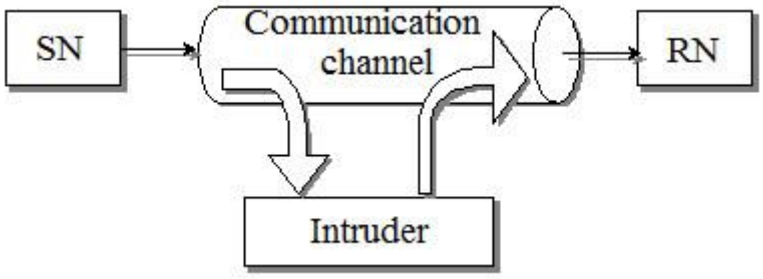
- Falsification (Figure 1. 3. d)). The intruder poses as a source of the information. This violates the authenticity of information. An example of this type of violation is sending fake messages over the network.



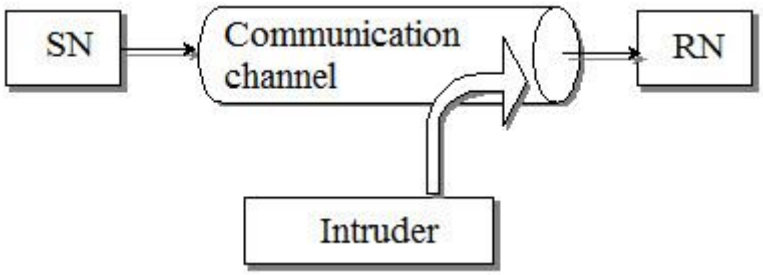
a) Interruption of the information transmission



b) Interception of the transmitted information



c) Modification of the transmitted information

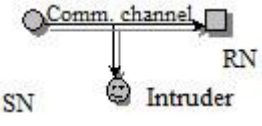
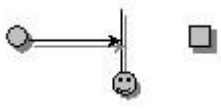
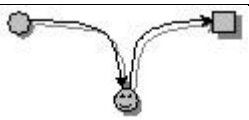
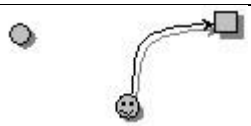


d) Falsification of the transmitted information

Figure 1.3 – Types of the information transmission violations

The above types of violations can be divided into two groups:
1 active;
2 passive.
The first group includes:
- interruption – violation of accessibility and confidentiality;
- modification – violation of integrity;
- falsification – violation of authenticity.

Table 1.1 – Classification of the information protection violations

Types of violations	Activity of violations	Graphic presentation of the information	Violation of the information properties
Interception of the transmitted information	Passive		Confidentiality of the transmitted information
Interruption of the information transmission	Active		Accessibility of the transmitted information
Modification of the transmitted information			Confidentiality and integrity of the transmitted information
Falsification of the transmitted information			Authenticity of the transmitted information

This type of violations has an active nature of the impact on the elements of the network and the information transmitted. The main purpose of these violations is to change or destroy the flows of information on the network Passive violations include interception to obtain the transmitted information, analyze it, and use it for specific purposes.

It can be fairly confidently asserted that passive violations pose as their ultimate goal the transition to a group of active violations.

The above classification of the information security breaches is presented in Table 1.1.The listed types of violations can take place, both in the user's plane, and in the control and management planes (Figure 1.2b)).

Moreover, active types of violations (interruption, modification and falsification) in the management plane lead to violations or destruction of the information stored in the databases of the Criminal Code. As a result, the routing tables are violated and as a result the impossibility of the normal operation of the control planes (signaling) and the user occurs.

1.4 Service Providers, Security Profile and Information Security Connections

Information security services (Figure 1.4) are responsible for ensuring the basic requirements of users for telecommunications systems (in terms of its reliability). And these services should function in all three planes: management, control and user ones.

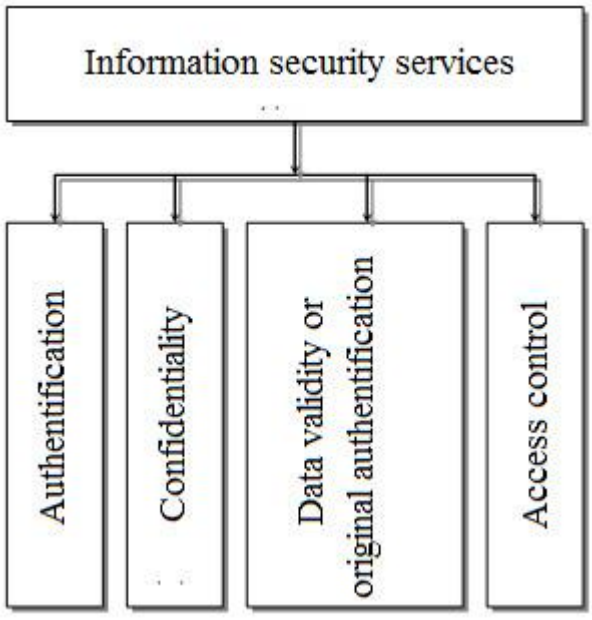


Figure 1.4 – Information security services

A set of information security services that provide user requirements form a security profile.

Security agents (SAs) are responsible for the installation and termination of a particular service.

The reconciliation of security services between agents occurs through security connections. These connections are used to exchange security information.

Figure 1.5 demonstrates the simplest way to organize a security connection - security agents are placed within the end user systems.

In this case, the end systems and security agents interact with the network through the user-to-network + security interface (UNI + Sec).

Protection agents for a virtual connection (channel or path) that is installed between end user systems consistently perform the following actions:

1 determine the type of security services that must be applied to this virtual connection;

2 coordinate protection services among themselves;

3 apply the required protection services to this virtual connection.

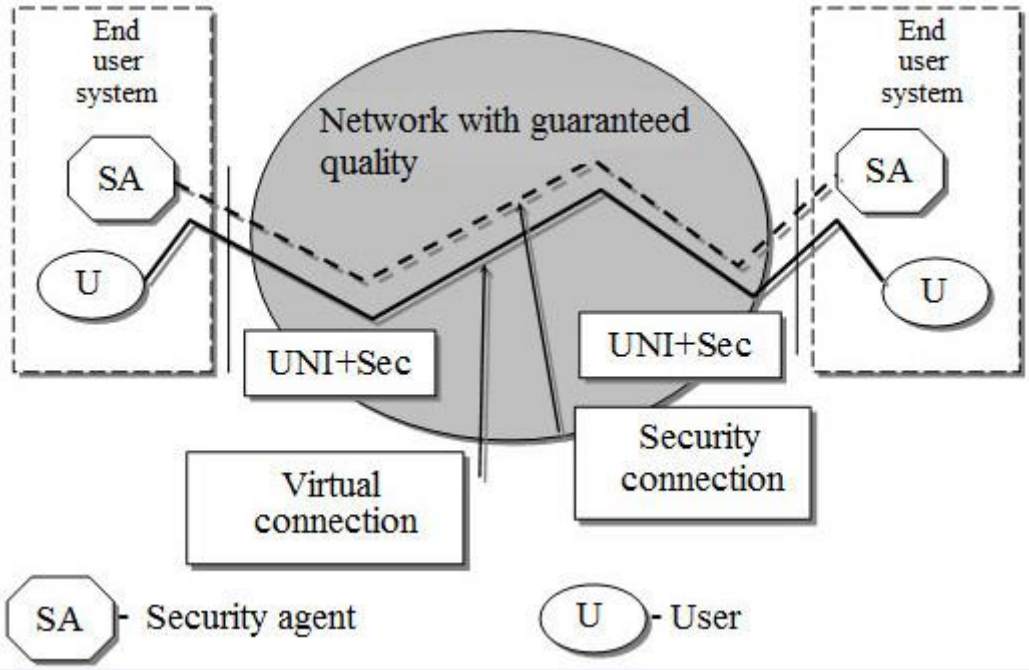


Figure 1.5 – Version of organizing a security connection between security agents

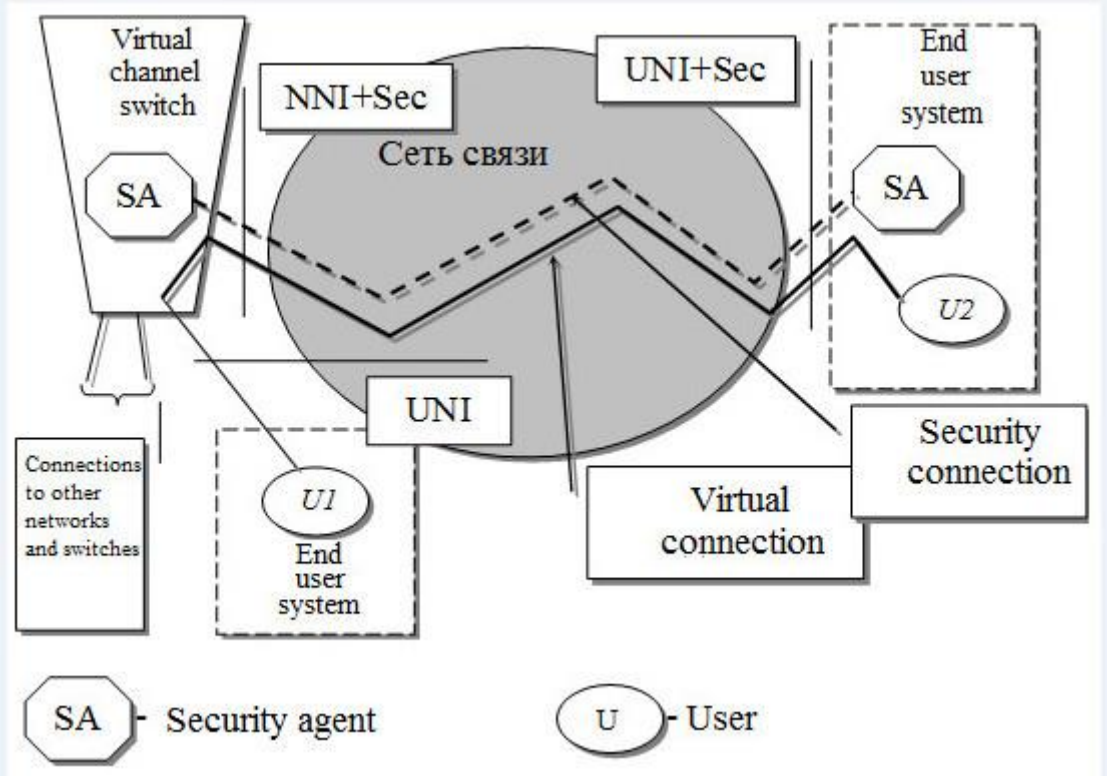


Figure 1.6 – Version of organizing a security connection between security agents

The number of security connections must be equal to the number of security services installed. That is, if for a given virtual connection authentication, confidentiality and reliability of data are simultaneously required, three independent security connections are established.

Figure 1.6 shows another version of organizing the protection connection. In this case, one security agent is placed on the end user system, and the other one is on the virtual channel switch. Accordingly, users and security agents interact with the communication network through (UNI) or UNI + Sec user-to-network interfaces; a virtual channel switch through a node-to-network + security interface (NNI + Sec). In this case, the protection agent located within the virtual channel switch is able to provide protection services not only to the user *U2*, but also to other nodes and networks that are connected to this virtual channel switch. These security agents are often called firewalls. In fact, a firewall is a gateway that performs the functions of protecting the network from unauthorized access from outside (for example, from another network).

There are three types of firewalls (Figure 1.7). The application layer gateway is often called a proxy server – it provides data relay functions for a limited number of user applications.

That is, if the gateway does not have support for an application, then the corresponding service is not provided, and data of the appropriate type cannot pass through the firewall.

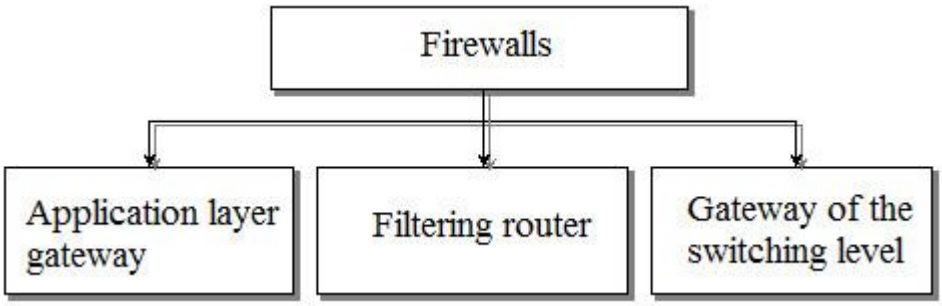


Figure 1.7 – Types of firewalls

Filtering router. More precisely, it is a router, whose additional functions include packet-filtering router. It is used on networks with packet switching in the mode of datagrams. That is, in those technologies for transmitting information on communication networks in which the signaling plane (there is no preliminary connection between the SN and the DN) is absent (for example, IP V 4). In this case, the decision to transfer the received data packet over the network is based on the values of its transport layer header fields. Therefore, firewalls of this type are usually implemented in the form of a list of rules applied to the values of the header fields of the transport layer.

Gateway of the switching level: protection is implemented in the control plane (at the signaling level) by enabling or disabling certain connections.

To increase the reliability of protection of virtual connections (channels and paths) it is possible to use more than one pair of security agents and more than one protection connection. In this case, a security connection topology is formed, which is based on the principle of embedding and not crossing security connections along the entire route between the SN and the DN (or the end user systems). An example of the principle of attachment and non-intersection of security connections is shown in Figure 1.8. In this case, the protection of the virtual channel organized between the end systems is performed by four security connections and eight security agents (SA1-SA8). Moreover, each connection does not know about the existence of other connections and does not care about the fact which security services the latter provides. That is, the security connections are absolutely independent of each other. This approach makes it possible to apply numerous strategies and tactics to protect different parts of the network. For example, a security connection between agents SA1 and SA8 provides authentication between the end systems. Regardless of this connection, the connection between SA2 and SA7 ensures confidentiality, and SA2, SA3, SA4 and SA4, SA5, SA6 provide data validity.

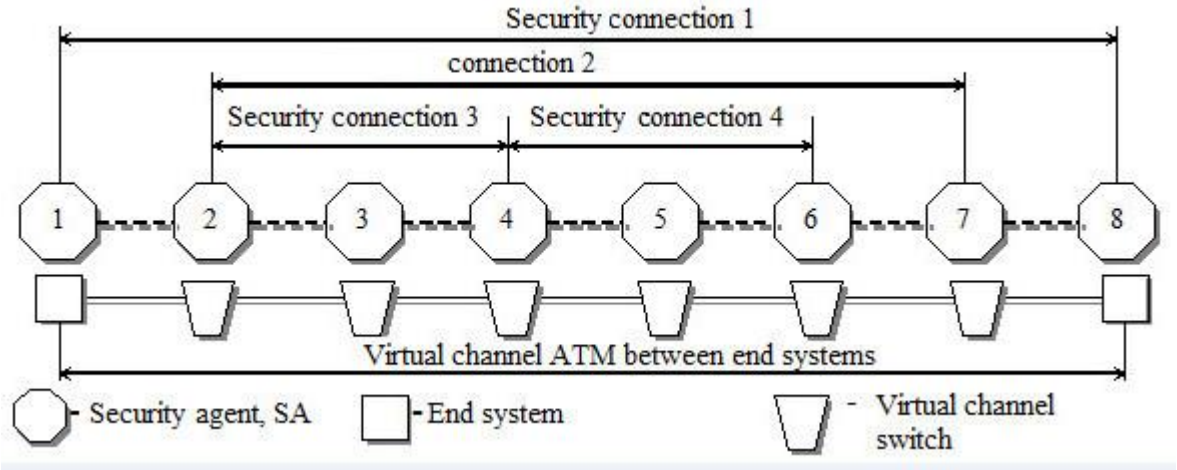


Figure 1.8 – Example of organizing topology of security connections using the principle of attachment and non-intersection

Each security connection can be represented as a segment

$$S_k = \lfloor SA_i, SA_j \rfloor,$$

Where k is the serial number of the protection connection; I, j are the serial numbers of the protection agents.

For Figure 1.8, the security connections can be written by the corresponding segments:

$$S_1 = [SA_1, SA_8];$$
$$S_2 = [SA_2, SA_7];$$
$$S_3 = [SA_2, S_3, SA_4];$$
$$S_4 = [SA_4, S_5, SA_6].$$

In turn, the second segment is nested in the first one. That is, in the symbolic form it looks like the following:

$$S_2 \subset S_1.$$

Not-intersection of segments can be represented as the following:

$$S_3 \not\subset S_4.$$

Taking into account that they are embedded in, we get:

$$[[S_3 \not\subset S_4] \subset S_2].$$

The final character record of the protection connection topology, shown in Figure 1.8, is as follows:

$$[[S_3 \not\subset S_4] \subset S_2] \subset S_1.$$

From figure 1.8 and the obtained expression it is seen that this topology of virtual channel protection connections between the end systems has three levels of nesting.

In this way:

1 the principle of embedding and not crossing protection joints, and

2 The maximum number of embedding levels (for ATM technology up to 16 levels) are the only constraints to the organization of the security connection topology for one virtual connection (channel or path).

At the same time, for the protection topology depicted in Figure 1.8, the connection between agents SA3 and SA5 is not possible, since the principle of non-intersection is violated.

Thus, the protection connection topology implements the user's security profile, which is distributed over the network.

The choice of the security connection topology is largely determined by the user requirements for the degree of protection of the transmitted information and the resource capabilities of the network to provide these requirements itself.

Control questions

1. Give the characteristics of the main infringements of information transmission through telecommunication systems.

2. Why do violations in the service planes (management and management) make it difficult, and sometimes impossible, to operate the user plane?

3. Give the following information properties:

- confidentiality;
- availability;
- Integrity;
- authenticity.

4. What violations belong to the active group and what properties of information they v

5 What is the main purpose of the service and security information connections?

6. What are the functions of the firewalls?

7. What are the functions of the proxy server?

8. What functions are performed by the following devices: filtering router; gateway switching level?

9. What restrictions are imposed on the organization of the protection connection topology?

10. List the main functions of the security message exchange protocol.

2. Cryptographic protection of information

2.1 Basic concepts of cryptographic protection of information

Cryptology is the methodological basis of modern information security systems in TCS. Historically, cryptography originated as a method of hidden message passing.

Cryptography is a collection of data transformation methods designed to protect this data, making it useless for illegal users. Such transformations are aimed at solving three main problems of data protection: ensuring confidentiality, integrity and authenticity of transmitted or stored data.

The following areas of theoretical and applied research are of greatest interest today: the creation and analysis of the reliability of cryptographic algorithms and protocols; adapting algorithms to various hardware and software platforms; use of existing cryptography technologies in new application systems; the possibility of using cryptography technologies to protect intellectual property.

Interest in research on the adaptation of algorithms to various hardware and software platforms is caused by the creation of cross-platform telecommunications systems based on unified standards for algorithms. The same algorithm must be effectively executed on a wide variety of hardware and software platforms from a mobile phone to a router, from a smart card to a desktop computer.

The existing means of data protection in telecommunication networks can be divided into two groups according to the principle of building a key system and an authentication system. We will refer the means using symmetric crypto algorithms for the construction of a key system and an authentication system to the first group, and asymmetric ones to the second group.

Let's make a comparative analysis of these systems. An information message ready for transmission, initially open and unprotected, is encrypted and thus converted into a ciphertext, i.e., into a closed text or graphic image of the document. In this form, the message is transmitted over a communication channel, even if the latter is not protected. The authorized user, after receiving the message, decrypts it (i.e., opens it) by means of the inverse transformation of the cryptogram, resulting in an initial, open message view that is accessible to the authorized users.

The use of a special algorithm corresponds to the method of transformation in a cryptographic system. The action of such an algorithm is triggered by a unique number (a sequence of bits), usually called an encryption key. For most systems, the key generator circuit may be a set of instructions and instructions, either an equipment node or a computer program, or all of this together, but in any case, the process of encryption (decryption) is implemented only by this special key. To exchange encrypted data successfully, both the sender and the recipient need to know the correct key setting and keep it in a secret.

The strength of any closed communication system is determined by the degree of secrecy of the key used in it. However, this key must be known to other network users so that they can freely exchange encrypted messages. In this sense, cryptographic systems also help to solve the problem of authentication

(establishing authentication) of received information. The cracker in case of interception of the message will deal only with encrypted text, and the true receiver, receiving messages closed by the key known to him and the sender, will be reliably protected from possible misinformation.

In addition, there is the possibility of encryption of information by a simpler way - using a pseudo-random number generator. Using a pseudo-random number generator is to generate a cipher scale using a pseudo-random number generator with a certain key and applying the resulting gamma to the open data in a reversible manner. This method of cryptographic protection is quite easy to implement and provides a fairly high encryption speed, but it is not very robust to decryption. Classical cryptography is characterized by the use of one secret unit - a key that allows the sender to encrypt the message, and the recipient to decrypt it. In the case of encryption of data stored on magnetic or other information carriers, the key allows to encrypt information when writing to a medium and decrypt when reading from it.

Let's define a number of terms used in cryptology.

A cipher is a set of reversible transformations of a set of open data into a set of encrypted data specified by a cryptographic transformation algorithm.

A cipher is a collection of injective mappings of a set of plaintexts into a set of encrypted texts, indexed by elements of the set of keys: $\{Fk: X \rightarrow S, K \in K\}$.

A cryptographic system, or cipher, is a family T of reversible transformations of plaintext into an encrypted one. k number called as a key can be unambiguously compared to members of this family.

Tk transformation is determined by the corresponding algorithm and the k key value.

Key is the specific secret state of certain parameters of cryptographic data transformation algorithm providing the selection of one of the options among the aggregate of all possible ones for this algorithm. The key secrecy should provide the impossibility of the original text recovery over the encrypted one.

The space of K keys is the set of possible key values. Usually the key is the consecutive row of alphabet letters. But the terms “key” and “password” should be distinguished.

The password is also a secret sequence of alphabet letters, however used not for encryption (as the key), but for peer entity authentication.

The cryptosystems are divided into symmetrical and asymmetrical ones [or with clear (public) key]. In symmetrical cryptosystems one and the same key is used for encryption and decryption.

In the systems with the public key two keys are used: clear (public) and private (secret), which are mathematically linked to each other. Information is encrypted with the public key available for everyone interested, and is decrypted with the private key known only to the receiver of the message.

The terms “distribution of keys” and “keys management” refer to the procedures of information processing, the content of which are the development and distribution of keys among the users.

Electronic (digital) signature is the cryptographic transformation of the text attached to it, which allows other users checking the authorship and integrity of the message when receiving it.

Encryption of data is the process of clear data transformation into the encrypted data with the help of cipher, and the decryption of data is the process of private data transformation into the clear data with the help of cipher. Instead of the term “clear data” often the terms “clear text” and “original text” are used, and instead of the term “encrypted data” – “encrypted text” is used. Decryption is the process of transformation of private data to clear ones, with the unknown key and possibly unknown algorithm, i.e. cryptanalysis methods.

Encryption is the process of encrypting or decrypting the data. The term “encryption” is also used as the synonym for encoding. However, it is incorrect to use the term “coding” as the synonym of encryption (and “code” instead of “cipher”), as coding usually means presenting the information in the form of symbols (alphabet letters). Cryptostrength is the type of cipher determining its strength to decryption. Usually this feature is determined by the period of time necessary for decryption.

Gammation is the process of superimposition of cipher to clear data according to certain gamma law. Cipher gamma is the pseudorandom binary sequence developed under the assigned algorithm for encryption of clear data and decryption of encrypted data.

Protection against falsified data entry is the protection from imposing of false data. To protect against the falsified data entry the message authentication code being the sequence of fixed length data is received as per the certain rule from the clear data and the key added to the encrypted data. Cryptographic protection is the protection of data with the help of cryptographic transformation, which means the transformation of data with encryption and (or) the development of message authentication code.

Initialization vector is the initial parameters of cryptographic transformation algorithm. The encryption (decryption) equation is the correlation describing the process of formation of encrypted (clear) data from clear (encrypted) data as a result of transformations assigned by the cryptographic transformation algorithm.

Requirements to the cryptographic systems:

The process of data enciphering may be carried out both, with software and hardware. The hardware implementation differs with the larger cost; however it has the following advantages: high performance, simplicity, protectiveness, etc. The software implementation is more practicable, allowing the certain flexibility in use. Irrespective of the type of implementation the following generally accepted requirements are formulated for modern cryptographic systems of information protection:

− cipher strength to stand against the cryptanalysis should be in such a way in order its disclosure could be done only by resolving the task of full search of keys and should go beyond the possibilities of modern computers (with consideration of possibility of network computing organization;

− cryptostrength provided not with the algorithm secrecy, but with the key secrecy (distinguishes the cryptosystems of general use (algorithm is available to a potential attacker) and limited use (algorithm is kept secret));

− encrypted message should be readable only with the key;

− cipher should be strong even in case the attacker knows sufficiently large amount of master data and corresponding encrypted data;

− slight change of key or original text should lead to significant change of the type of encrypted text;

− structural elements of encryption algorithm should be fixed;

− cipher text should not significantly outperform the background information in the volume;

− additional bytes entered into the message during the encryption process should be fully and securely hidden in the encrypted text;

− errors aroused during the encryption should not lead to tampering and loss of information;

− there should not be any simple and easily installed dependence between the keys consequently used during the encryption process;

− any key among the set of possible ones should provide the equal cryptostrength (providing linear (uniform) key space);

− encryption time should not be too long;

− the cost of encryption should be agreed with the cost of closing information.

The first class of cryptosystems the practical use of which became possible with the availability of powerful and compact computing means became the block ciphers. The DES (American encryption standard) has been developed in 70s (adopted in 1978). Let us consider the technology of the work of the American encryption standard DES.

Information about the cryptanalysis. The knowledge of some provisions of cryptanalysis is necessary for deep understanding of cryptography. The central figure in the cryptanalysis is the attacker (or cryptanalytic). It means the person (a group of people), the purpose of which is reading or forgery of messages protected with cryptographic methods.

A set of assumptions which, as a rule, are taken as a basis of mathematic or other models are used in respect to the attacker:

1. The attacker knows the encryption algorithm (or development of EDS) and special aspects of its implementation in the certain case, but does not know the secret key.

2. All encrypted texts are available to the attacker. The attacker may have access to some original texts, for which the corresponding encrypted texts are known.

3. The attacker has got in place the computing, human, timing and other resources, the amount of which is justified by the potential value of information which shall be received as a result of cryptanalysis. The attempt of reading or forgery of the encrypted message, key computation by cryptanalysis methods is

called the crypto-attack or the cipher attack. The successful crypto-attack is called hacking.

The cryptostrength is the cipher capability determining its strength to decryption without the knowledge of the key (i.e. the crypto-attack). The cryptostrength indicator is the main parameter of any cryptosystem. The following may be selected as a cryptostrength indicator:

− quantity of all possible keys or possibility of choosing the key during the assigned timeframe with the assigned resources;

− quantity of operations or time (with assigned resources) necessary for hacking the cipher with the assigned probability;

− the cost of key information or original text computation. All these indicators should also consider the level of possible crypto-attack. However, it should be understood that the effectiveness of information protection by the cryptographic methods depends not only on the cipher cryptostrength, but also on the other multiple factors, including the issues of cryptosystem implementation in the form of devices or programs.

During the cipher cryptostrength analysis the human factor should be considered. For instance, bribery of the certain person who holds the necessary information may be substantially cheaper than creation of a supercomputer for hacking the cipher.

The modern cryptanalysis relies on such mathematic sciences, like the theory of probability and mathematical statistics, algebra, theory of numbers, theory of algorithms and some others.

All cryptanalysis methods are put into the four directions in the whole:

1. Statistical cryptanalysis – investigates the possibilities of cryptosystems hacking on the basis of studying the statistic regularities of the original and encrypted messages. Its usage is complicated due to the fact that in real cryptosystems the information is compressed before the encryption (transferring the original text into the arbitrary sequence of symbols) or in case of gammation it uses the pseudorandom sequence of long length.

2. Algebraic cryptanalysis – searches the mathematically weak components of crypto-algorithms. For instance, in 1997 the class of keys significantly simplifying the cryptanalysis has been revealed in elliptic systems.

3. Differentiated (or incremental) cryptanalysis is based on the analysis of dependence of encrypted text changes from the change of the original text. For the first time it had been used by Murphy, improved by Bieham and Shamir for DES attack.

4. Linear cryptanalysis – method based on the search of linear approximation between the original and the encrypted texts. Proposed by Matsui, it also had been used for the first time during the DES hacking. As the differentiated analysis, it may be used in real cryptosystems only for analysis of certain blocks of cryptographic transformations.

Experience in hacking the cryptosystems shows that the main method is still the brute force – testing the keys. Also, as the experience shows, the cryptosystems mostly suffer from negligence in implementation.

It is customary to differentiate several levels of crypto-attack depending on the amount of information available to the cryptanalytic.

The three levels of crypto-attack may be determined based on the increase of complexity.

1. Attack on the encrypted text (KA1 level) – all or several encrypted messages are available to the attacker.

2. Attack on the pair "original text – encrypted text" (KA2 level) – all or several encrypted messages and corresponding original messages are available to the attacker.

3. Attack on the selected pair "original text – encrypted text" (KA3 level) – the attacker has got the possibility to select the original text, receive the encrypted text for it and compute the key based on the analysis of dependencies between them. All modern cryptosystems hold enough strength even to the KA3 level attacks, i.e. when actually the encryption device is available to the attacker.

2.2 Classification of cryptographic information closure methods

Currently, a large number of methods for cryptographic information closure is known. Classification of encryption methods (cryptoalgorithms) can be carried out on the following grounds:

- by the key type: symmetric crypto algorithms; asymmetric crypto algorithms;
- by the size of the information block: stream ciphers; block ciphers;
- by the nature of the effects produced on the data: the replacement method (permutation), the substitution method; analytical methods, additive methods (gammation), combined methods.

Encryption can be semantic, symbolic, combined. Closure of information in other ways can be achieved by steganography, compression / expansion, dissection / diversity.

2.3 Symmetric encryption cryptosystems

2.3.1 Symmetric encryption cryptosystems

Symmetric algorithms are algorithms in which the encryption key can be calculated by the decryption key and vice versa. In most symmetric systems, the encryption and decryption keys are the same. These algorithms are also called algorithms with a secret key or algorithms with one key. For such a system to work, it is required that the sender and receiver agree on the key used before beginning the secure transmission of the message (they had a secure channel for the transfer of the key). The security of a symmetric algorithm is determined by the key; The disclosure of the key allows an attacker to encrypt and decrypt all messages.

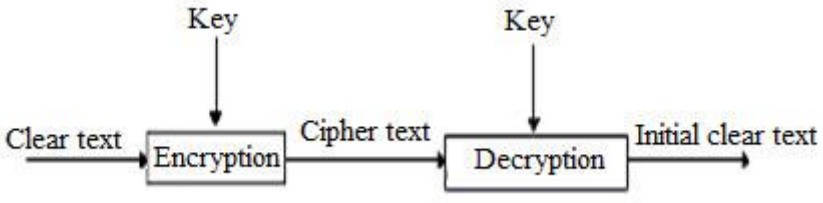


Figure 2.1 – Scheme for encrypting and decrypting the message

Because of the large redundancy of natural languages, it is difficult to make meaningful changes to an encrypted message, therefore, in addition to protecting information, protection against the imposition of false data is provided. If the natural redundancy is not enough, then a special control combination is used - imitavka. Since one key is used, each of the exchange participants can encrypt and decrypt messages, so this encryption scheme works on mutual trust. If it is not, then there can be various collisions, because if there is any dispute about the reliability of the message, the independent observer cannot tell which of the participants the message was sent. Symmetric algorithms fall into two categories. Some of them process the text bitwise (sometimes byte) and are called streaming algorithms or stream ciphers. The same ones that work with groups of plaintext bits are called block algorithms (ciphers). There are the following permutation ciphers: a simple, columned, commutative cipher, when in this type of cipher text is written on a horizontally sheet of paper of a fixed width, and ciphertext is read vertically. The decryption consists in recording the ciphertext vertically on a sheet of paper of fixed width and then reading the plaintext horizontally,

В О Л О Г О
Д С К И Й
Г О С У Д А
Р С Т В Е Н
Н Ы Й П Е
Д А Г О Г И
Ч Е С К И Й
 У Н И В Е
Р С И Т Е Т

Encrypted text: ВДГРНДЧ РОСОСЫАЕУСЛКСТЙГСНИОИУВ ОКИТГЙДЕПГИВЕО АНЕИЙЕТ

- a permutation cipher with a keyword, when letters of plain text are written into cells of a rectangular table by its lines. The letters of the keyword are written above the columns and indicate the order of these columns (by increasing the number of letters in the alphabet). To get the encrypted text, you need to write out the letters on the columns, taking into account their numbering.

Clear text: Applied mathematics Key: Cipher (Шифр)

Ш и ф р
4 1 3 2
П р и к

Figure 2.2 – Example of using simple encryption

Cryptogram: Раяеикнаааидммкплатт

The keyword (column sequence) is known to the recipient who can easily decrypt the message.

Since the symbols of cryptotext are the same as in plaintext, frequency analysis will show that each letter occurs approximately at the same frequency as usual. This gives cryptanalyst information that this is a permutation cipher. Applying the second permutation filter to the cryptographic text will significantly increase security. There are even more complex permutation ciphers, but with the use of a computer you can open almost all of them.

Although many modern algorithms use permutation, this involves the problem of using a large amount of memory, and sometimes you need to work with messages of a certain size. Therefore, the use of substitution ciphers is more common.Wildcards. In substitution ciphers, the letters of the original message are replaced with substitutions, for example, in the Caesar cipher, each letter is replaced by a letter that is k symbols to the right of the modulus equal to the number of letters in the alphabet. With its simplicity in implementation, single-alphabetical systems are easily vulnerable.

2.3.2 Gammation

Gammation is also a widely used cryptographic transformation. In fact, the boundary between the gammation and using the infinite keys and ciphers of Vizhiner, which was discussed above, is very conditional.

The principle of encryption by gammation consists in generating a cipher scale using a pseudorandom number sensor and imposing the resulting gamma on the open data in a reversible manner (for example, using addition modulo 2).

The process of decrypting data reduces to the re-generation of the cipher scale with a known key and the imposition of such a gamut on the encrypted data. The resulting encrypted text is quite difficult to disclose if the gamma cipher does not contain repeated bit sequences. In fact, the gamma of the cipher should change at random for each encrypted word. In fact, if the gamut period exceeds the length of the entire ciphertext and no part of the source text is known, then the cipher can only be opened by direct search (key-breakdown). The cryptographic stability in this case is determined by the size of the key.

Encryption by the method of gammation.

Gammation means the imposition on the open data according to a certain law of the cipher scale. Gamma cipher is a pseudo-random sequence generated by a certain algorithm, used to encrypt open data and decrypt ciphertext.
Visual representation of the transmitter circuit:

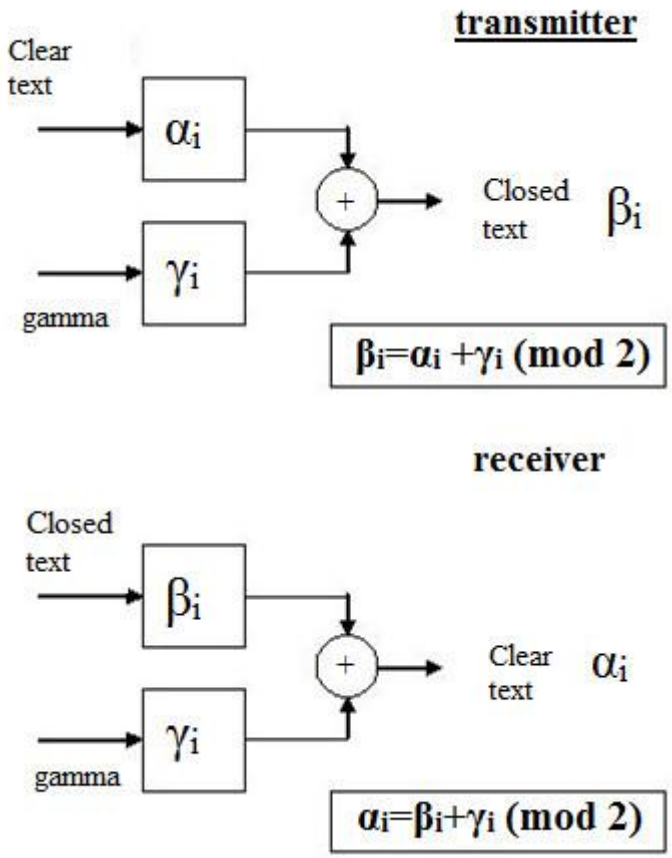


Figure 2.3 – The method of gammation

The resulting encrypted text is quite difficult to disclose if the gamma cipher does not contain repeated bit sequences. In fact, the gamma of the cipher should change at random for each encrypted word. In fact, if the period of the gamma exceeds the length of the entire ciphered text and no part of the source text is known, then the code can only be opened by direct search. The cryptographic stability in this case is determined by the size of the key.

The method of gammation becomes powerless if an attacker gets a fragment of the source text and the corresponding ciphertext. By simple subtraction modulo, a segment of the pseudorandom sequence (PSP) is obtained and the whole sequence is reconstructed from it. However, if the majority of sent messages begin with the words "SOV.SEKRETNO", then cryptanalysis of the entire text is greatly facilitated. This should be taken into account when creating real information security systems.

2.3.3 Encryption using a pseudorandom value sensor

Pseudo-random number sensors.

To obtain linear sequences of gamma elements whose length exceeds the size of the encrypted data, FPGA sensors are used. Based on the theory of groups, several types of such sensors have been developed.

Congruent Sensors

At present, the most accessible and effective are the congruent generators of the memory bandwidth. For this class of generators, we can make a mathematically rigorous conclusion about the properties of the output signals of these generators in terms of periodicity and randomness.

One of the good congruent generators is the linear congruential PSC sensor. It generates sequences of pseudo-random numbers T (i), described by the relation

T (i + 1) = (A * T (i) + C) mod m, where A and C are constants, T (0) is the initial value chosen as the generating number. Obviously, these three quantities form the key. Such a PRN sensor generates pseudo-random numbers with a certain repetition period, depending on the selected values of A and C. The value of m is usually set to 2n, where n is the length of the computer word in bits. The sensor has a maximum period M before the generated sequence begins to repeat. For the reason noted earlier, it is necessary to choose the numbers A and C such that the period M is maximal. As shown by D. Knuth, a linear congruent PRN sensor has a maximum length M if and only if C is odd and A mod 4 = 1.

A key of any size can be selected to encrypt the data using the PRN sensor. For example, let the key consist of a set of numbers x (j) of dimension b, where j = 1, 2, ..., n. Then the created gamma of the cipher G can be represented as the union of disjoint sets H (j).

If you have encrypted text on your hands

NUMBER = 10011000100010001001010010001000

We get plaintext

KEY = 1000 1010 1000 1011 1001 1110 1001 0111

Using the key for decryption

00010111000011010001100100001100

We get plaintext

FOAM = 10001111100001011000110110011100

The resulting encrypted text is quite difficult to disclose if the gamma cipher does not contain repeated bit sequences. In fact, the gamma of the cipher should change at random for each encrypted word. In fact, if the period of the gamma exceeds the length of the entire ciphered text and no part of the source text is known, then the code can only be opened by direct search. The cryptographic stability in this case is determined by the size of the key.

The method of gammation becomes powerless if an attacker gets a fragment of the source text and the corresponding ciphertext. By simple subtraction modulo, we obtain a segment of the PRS and restores the whole sequence from it. Attackers can do this on the basis of guesswork about the content of the source text. So, if the majority of messages sent begins with the words "SO.SECRETE", then

cryptanalysis of the entire text is greatly facilitated. This should be taken into account when creating real information security systems.

Therefore, further development was given to multi-alphabetic systems to which the Vernam cipher relates to.

In the classical sense, a one-time notepad is a large non-repeating sequence of key symbols randomly distributed. Originally it was a one-time tape for teletypewriters. The sender used each key symbol to encrypt only one plain text character. Encryption is an addition modulo n (power of the alphabet) of a plain text symbol and a key symbol from a one-time notepad. Each key symbol is used only once and for a single message, otherwise, even if you use a several gigabyte notebook, when the cryptanalyst receives several texts with overlapping keys, he can restore the source text. He will move each pair of ciphertexts relative to each other and calculate the number of matches in each position. If the ciphertexts are shifted correctly, the ratio of coincidences will increase sharply. From this point of view cryptanalysis is not difficult. If the key is not repeated and random, the cryptanalyst, whether he intercepts the texts or not, always has the same knowledge. A random key sequence, combined with non-random plain text, gives a completely random cryptotext, and no computing power can change it.

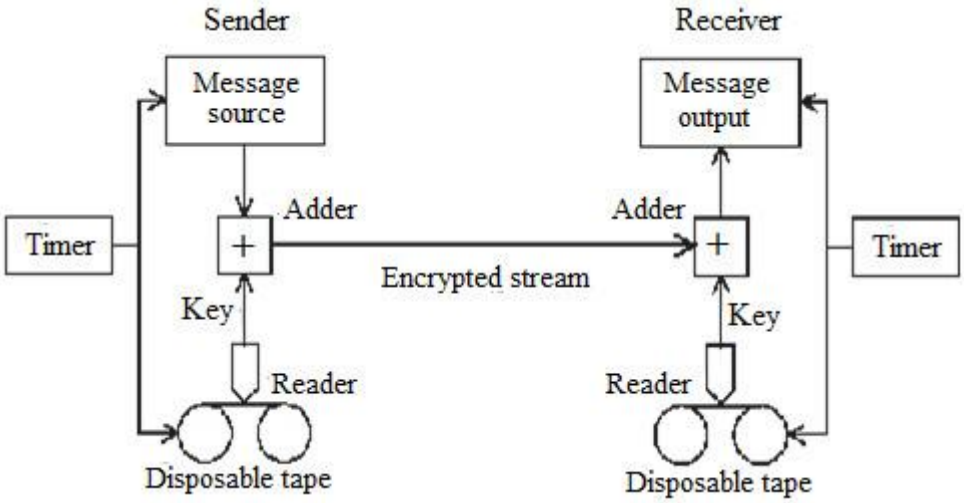


Figure 2.4 – Encryption using a one-time notebook

In real systems, two identical tapes with random key digits are first prepared. One remains with the sender, and the other is transmitted in a "non-interceptible" way, for example, by a courier with a guard, a legitimate recipient. When the sender wants to send a message, he first converts it into a binary form and places it in the device, which adds two digits read from the key tape to each digit of the message. On the receiving side, the encoded message is recorded and transmitted through a machine similar to the device used for encryption, which adds to each binary digit of the message (subtracts, since addition and subtraction modulo two are equivalent) modulo two digits read from the key tape, Thus an open text. In this case, of course, the key tape should move absolutely synchronously with its

duplicate used for encryption.

The main disadvantage of this system is that for each bit of transmitted information, a bit of key information must be prepared in advance, and these bits must be random. When encrypting a large amount of data, this is a serious limitation. Therefore, this system is used only for the transmission of messages of the highest secrecy. According to rumors, the "hot line" between the US and the USSR was encrypted using a one-off notebook. Many reports of Soviet spies were encrypted using one-time notepads. These messages are not disclosed today, and will never be disclosed (unless there is a way to return to the past and get these notebooks)

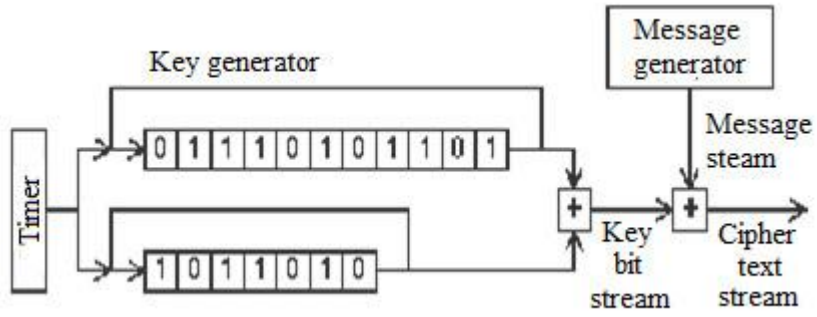


Figure 2.5 – One of the methods for building generators

To circumvent the problem of the preliminary transfer of a large-volume private key, engineers and inventors came up with many ingenious schemes for generating very long streams of pseudo-random digits from several short streams in accordance with some algorithm. The recipient of the encrypted message must be provided with exactly the same generator as the sender. But such algorithms add regularity to the ciphertext, the detection of which can help the analyst decrypt the message. One of the main methods for constructing such generators is to use two or more bit tapes, read from which the data is added folded to obtain a "mixed" stream. For example, a simple disposable tape can be replaced by two cyclic ribbons whose lengths are simple or relatively prime numbers. Since in this case the lengths of the tapes do not have common multipliers, the stream obtained from them has a repetition period equal to the product of their lengths: two tapes having lengths of 1000 and 1001, respectively, result in a composite stream with a period of 1000x1001 = 1001000 digits. The tapes circulate through the adder, which adds two digits of the digits that are calculated from them. The output of the adder serves as the key used to encrypt the message. Therefore, it is important that the composite flow exceeds the length of all the messages taken together, which can be transmitted over a reasonable period of time. Since the bit adder is a linear device, it is inherently cryptographically weak, but can be amplified in a large number of different ways. Another way is to indicate the location of the key as a place in the book, for example, Donald E. Knuth Art of Programming Volume 2. The resulting algorithms. Third edition. Page 83, 3rd paragraph. All characters included in the alphabet, starting from this place, are used as a one-time key for any message. But

in this case, the key will not be random and information about the frequency distribution of letters can be used.

It is not surprising, but the Vernam cipher class is the only class of ciphers for which non-disclosure in the absolute sense of the term can be proved (and was proved by Shannon).

2.3.4 Vigenère's Digest

This method is a simple form of a multi-alphanumeric substitution. Cipher Vigenera invented many times. This method was first described by Giovanni-Battista Bellaso in the book La cifra del. Sig. Giovan Battista Bellaso in 1553, but in the 19th century was named Blaise Vigenère, a Swiss diplomat. The method is simple to understand and implement, it is not available for simple cryptanalysis methods.

One of the oldest and most well-known multi-alphabetic cryptosystems is the Vigenère system, named after the French cryptographer Blaise Vigenere (Vigenere), in M.N. Arshinov, L.E. Sadovsky Codes and mathematics this code is called the Tritemius cipher. This method was first published in 1586. In this cipher, the key is given by a set of d letters. Such sets are signed with repetition under the message, and, then, the resulting sequence is added with plain text modulo n (the power of the alphabet). those. The following formula is obtained:

Vigd(mi)=(mi+ki mod d)(mod n)

Also, the ciphertext letter can be found from the following table, as the intersection of the column defined by the plaintext letter and the line defined by the key letter:

А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я
Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А
В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б
Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В
Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г
Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д
Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е
З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж
И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З
Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И
К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й
Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К
М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л
Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М
О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н
П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О
Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П
С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р

Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С
У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т
Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У
Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф
Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х
Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц
Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч
Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш
Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ
Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ
Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы
Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь
Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э
Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю

Figure 2.6 - Vigenera table for the Russian alphabet

1. Assume that. Chose the encryption key. Word – ГУЛЬШАТ
2. 2. From figure 1, we write down the lines corresponding to the key-ГУЛЬШАТ. It will look like this:

А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я
Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В
У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т
Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К
Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы
Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч
А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я
Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я А Б В Г Д Е Ж З И Й К Л М Н О П Р С

Figure 2.7 – Lines of the alphabet corresponding to the key ГУЛЬШАТ

3. The following sentence should be encrypted:

БЕЙСЕНБАЕВААЙНУРКЕНЖЕБАЕВНА

3. Under it, write the key ГУЛЬШАТ, until the encrypted sentence ends, as shown in Figure 2.8

БЕЙСЕНБАЕВААЙНУРКЕНЖЕБАЕВНА
ГУЛЬШАТГУЛЬШАТГУЛЬШАТГУЛЬШАТ

Figure 2.8 – Preparing for the encryption process

5. Next, the first letter of Г and the letter Б above it are found at the intersection of the letter Д, the letter Д is the first letter of the encryption, the next

letter of У to the intersection with the letter Е at the intersection, we have the letter Ш, and so on.

The result is the following sequence of letters. Which we consider to be the encrypted text (see Figure 2.9)

БЕЙСЕНБАЕВААЙНУРКЕНЖЕБАЕВНА
ГУЛЬШАТГУЛЬШАТГУЛЬШАТГУЛЬШАТ
ДШФНЭНУГШНЬШЙЯЦГХБЕАЧДУРЭЕА

Figure 2.9 – Received encrypted text (Above the third line above)

6. The mechanism of deciphering. Next, over the received encrypted text, write down the key-word ГУЛЬШАТ, as shown in Figure 2.10

ГУЛЬШАТГУЛЬШАТГУЛЬШАТГУЛЬШАТ
ДШФНЭНУГШНЬШЙЯЦГХБЕАЧДУРЭЕА

Figure 2.10 – Preparation for decryption

7. The process of deciphering. From the letter Г of the encrypted text in Figure 2, we reach the letter Д, in the same line and choose the letter Б that stands directly above it, then У-Ш-Е, the deciphered letter Ш, etc., we get the third line:

ГУЛЬШАТГУЛЬШАТГУЛЬШАТГУЛЬШАТ
ДШФНЭНУГШНЬШЙЯЦГХБЕАЧДУРЭЕА
БЕЙСЕНБАЕВААЙНУРКЕНЖЕБАЕВНА

Figure 2.11 – Decryption process

2.3.5 Block ciphers

Block ciphers are a sequence (with possible repetition and alternation) of the basic transformation methods applied to the block (part) of the encrypted text. Block ciphers are more common in practice than "pure" transformations of a class due to their higher cryptographic strength. Russian (GOST 28147-89) and American (Rijndael) encryption standards are based on this class of ciphers.

Consider the technology of the US standard DES encryption.

DES (Data Encryption Standard) is a symmetric encryption algorithm developed by IBM and approved by the US government in 1977 as the official standard (FIPS 46-3).

DES has blocks of 64 bits and 16 frame structure of the Feistel network, for encryption uses a key with a length of 56 bits.

The algorithm uses a combination of nonlinear (S-blocks) and linear (permutations of E, IP, IP-1) transformations. For DES, several modes are recommended.

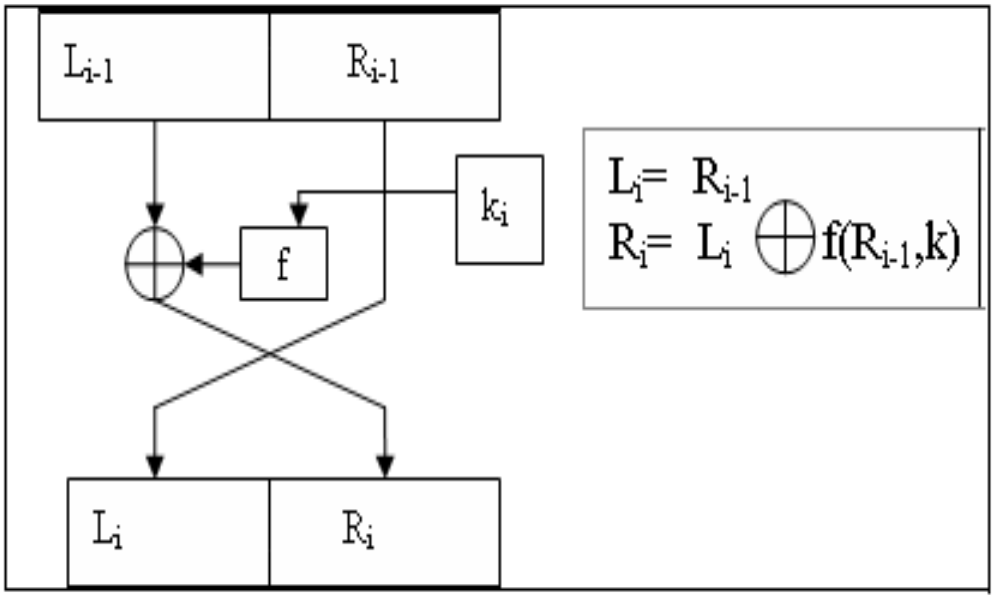
DES is a block cipher. To understand how DES works, you need to consider the principle of the block cipher, the Feistel network.

The input data for a block cipher is a block of size n bits and a k-bit key. At the output, after applying the encrypting transformation, we obtain an n-bit encrypted block, and insignificant differences in the input data, as a rule, lead to a significant change in the result. Block ciphers are implemented by repeatedly applying to the blocks of the source code some basic transformations.

Basic conversions:

1. Complex conversion on one local part of the block.
2. Simple conversion between parts of the block.

Since the conversion is done in blocks, as a separate step, the separation of the original data into blocks of the necessary size is required. In this case, regardless of the format of the source data, be it text documents, images or other files, they must be interpreted in a binary form and only then divided into blocks. All of the above can be implemented both software and hardware.



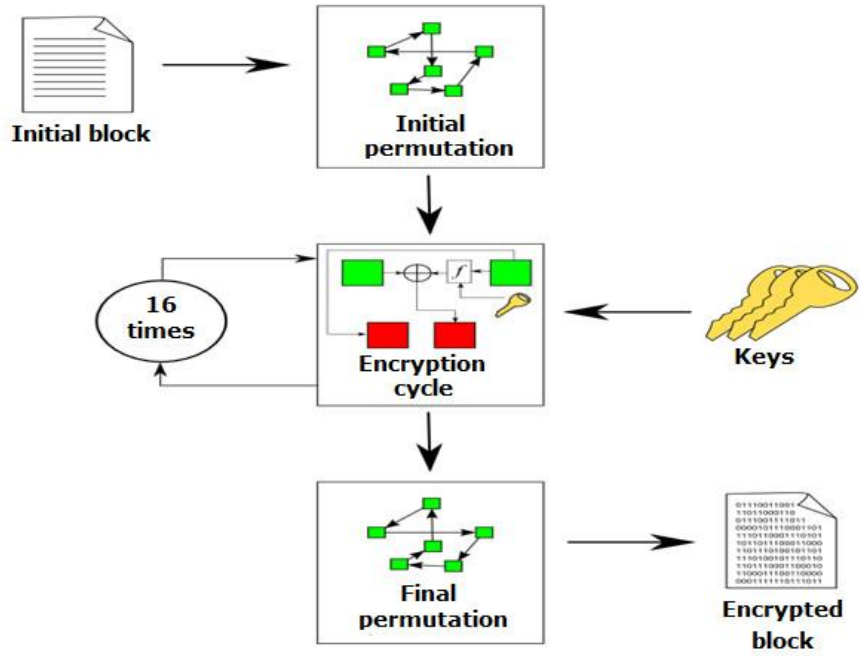


Figure 2.12 - The encryption process in the DES algorithm

The main advantages of the DES algorithm:

1. Only one 56-bit key is used;

2. Encrypting the message using one package, you can use any other for decryption;

3. Relative simplicity of the algorithm ensures high speed of information processing;

4. Sufficiently high stability of the algorithm.

The encryption process consists in the initial permutation of the bits of the 64-bit block, the sixteen encryption cycles and, finally, the reverse bit rearrangement (Figure 2.12).

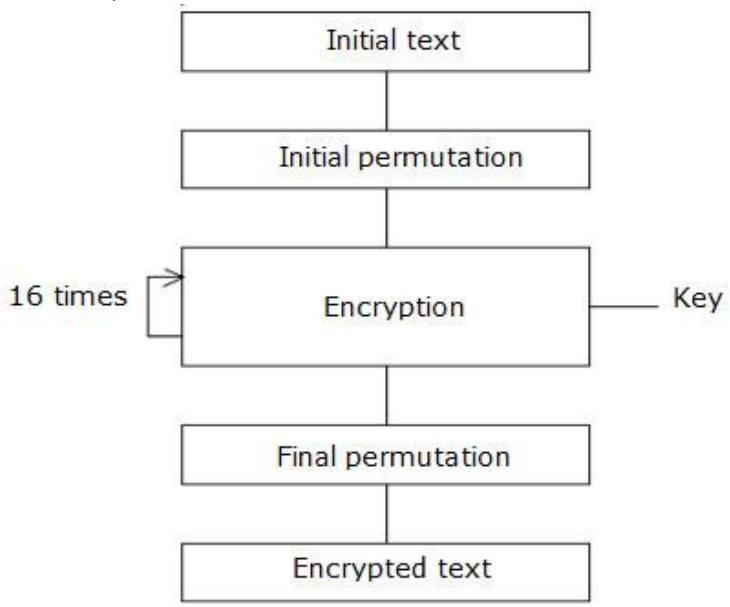


Figure 2.13 – Generalized scheme of encryption in DES algorithm

Now consider the encryption function f (R (i-1), K (i)). Schematically it is shown in Figure 2.14.

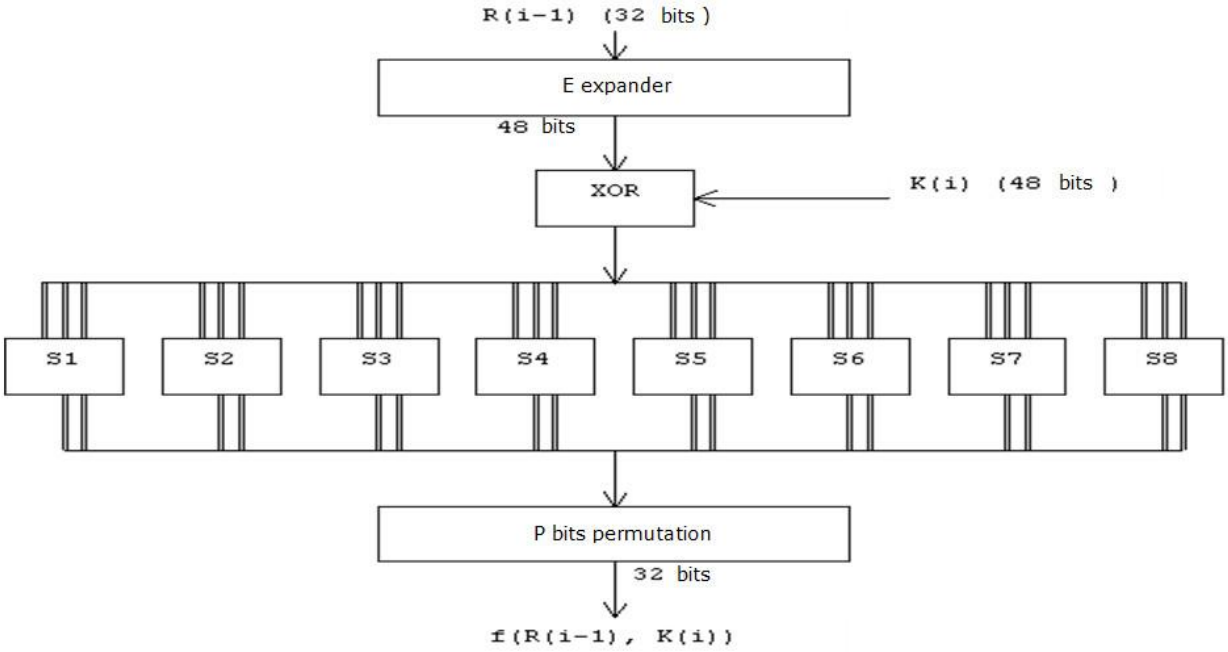


Figure 2.14 – Calculation of the function f (R (i-1), K (i))

The following matrix functions are used to calculate the value of the function f:

1. E – expansion of the 32-bit sequence to 48-bit,

S1, S2, ..., S8 – conversion of the 6-bit block into 4-bit,

2. P – permutetion of bits in the 32-bit sequence.

2.3.6 Encryption Algorithm GOST 28147-89

Brief description of the algorithm

The algorithm encrypts the data with 64-bit blocks using a 256-bit encryption key. There are 32 rounds of transformations, each of which provides the following operations (see Figure 2.15):

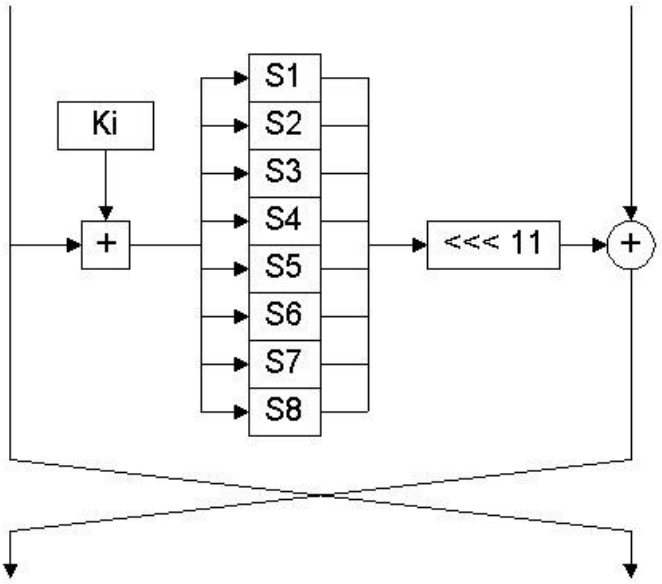


Figure 2.15 – Encryption round in the algorithm GOST 28147-89

1. One of the 32-bit data sub-blocks is added to the 32-bit key value of the Ki round modulo 232.

2. The result of the previous operation is divided into 8 fragments of 4 bits, which are "run" in parallel through 8 replacement tables S1 ... S8. The replacement tables in the standard [14] are not defined. Examples of possible replacement tables can be found, for example, in [17] or [18].

3. 4-bit fragments (after substitutions) are combined back into the 32-bit subunit, whose value is cyclically shifted to the left by 11 bits.

4. The subblock processed by the previous operations is superimposed on the unprocessed one using the bitwise logical operation "exclusive or" (XOR).

5. The subblocks change places.

In fact, there is no procedure for extending the key in the GOST 28147-89 algorithm: in rounds of encryption, 32-bit fragments K1 ... K8 of the original 256-bit encryption key are successively used in the following order: K1, K2, K3, K4, K5, K6, K7, K8, except for the last 8 rounds - in rounds 25 through 31 the fragments are used in the reverse order.

Decryption is completely analogous to encryption, but with a different order of use of key fragments:

In direct order – in the first 8 rounds;

In the remaining rounds – in the reverse order.

The modes of gammation and gammation with feedback, providing for the calculation of the pseudo-random sequence-the cipher scale-with its above-described transformations-and its superimposition on the encrypted text;

Simulation mode is a cryptographic checksum used to verify the integrity of the data; In this mode, 16 rounds of transformations are performed instead of 32 rounds.

Algorithm GOST 28147-89 can be used in various widely used encryption modes (provided by the standard [4]). As can be seen from the description, the algorithm GOST 28147-89 is very simple to implement, which is its undoubted advantage.

The main differences between DES and GOST.

The main differences between DES and GOST are as follows:

1. DES uses a complex procedure to generate subkeys from keys. In GOST this procedure is very simple;

2. In DES 56-bit key, and in GOST - 256-bit. If you add secret permutations of S-blocks, then the total volume of GOST secret information will be approximately 610 bits;

3. S-blocks DES have 6-bit inputs and 4-bit outputs, and S-blocks of GOST have 4-bit inputs and outputs. In both algorithms, eight S-blocks are used, but the size of the S-block of the GOST is equal to a quarter of the size of the S-block DES;

4. In DES, irregular permutations, called the P-block, are used, and the GOST uses an 11-bit cyclic shift to the left;

5. In DES 16 cycles, and in GOST - 32.

The power attack on the GOST is absolutely unpromising. GOST uses a 256-bit key, and if you take into account the secret S-blocks, then the key length will be even greater. GOST, apparently, is more resistant to differential and linear cryptanalysis than DES. Although random S-blocks of GOST at some choice do not guarantee high cryptographic strength in comparison with fixed S-blocks DES, their secrecy increases the stability of GOST to differential and linear cryptanalysis. In addition, the effectiveness of these cryptanalytical methods depends on the number of conversion cycles – the more cycles, the more difficult cryptanalysis. GOST uses twice as many cycles as DES, which, possibly, leads to the inconsistency of differential and linear cryptanalysis.

GOST does not use the existing permutation in DES with the extension. Removing this permutation from DES weakens it due to a decrease in the avalanche effect; It is reasonable to assume that the absence of such an operation in the GOST adversely affects its cryptographic strength. From the point of view of cryptostability, the operation of arithmetic addition, used in the GOST, is not worse than the XOR operation in DES.

The main difference is the use of a cyclic shift in GOST instead of a permutation. The permutation of DES increases the avalanche effect. In GOST, changing one input bit affects one S-block of one conversion cycle, which then affects two S-blocks of the next cycle, then three blocks of the next cycle, and so on. It will take eight cycles before changing one input bit will affect each bit of the result; In DES, you need only five cycles. However, GOST consists of 32 cycles, and DES only out of 16.

The developers of GOST tried to achieve a balance between crypto-stability and efficiency. Taking as a basis the design of Feistel, they developed a cryptalgorithm that is better than DES, suitable for software implementation. To increase the cryptographic strength, an extra-long key is introduced and the number of cycles is doubled. However, the question whether the efforts of developers have resulted in the creation of a more crypto-resistant crypto algorithm than DES remains open.

2.3.7 RSA cipher system

The RSA system is currently the most common public key encryption system.

One of the most common methods of asymmetric encryption - decryption is the public key encryption method, which uses the RSA algorithm.

Despite the rather large number of different SOKs (public key system), the most popular is the RSA cryptosystem, developed in 1977 and named after its creators: Ron Rivest, Adi Shamir and Leonard Adelman.

They took advantage of the fact that finding large prime numbers is computationally easy, but the factorization of the product of two such numbers is practically impracticable. It is proved (Rabin's theorem) that the disclosure of the RSA cipher is equivalent to such a decomposition. Therefore, for any key length it is possible to give a lower estimate of the number of operations to open the cipher, and taking into account the performance of modern computers, it is also necessary to estimate what is needed at that time.

The ability, guaranteed to assess the security of the RSA algorithm, was one of the reasons for the popularity of this SOK against dozens of other schemes. Therefore, the RSA algorithm is used in banking computer networks, especially for working with remote clients (credit card service).

Consider the mathematical results that form the basis of this algorithm.

Theorem 1. (Minor Fermat's theorem.)

If p is a prime number, then xp-1 = 1 (mod p) (1)

For any x simple relative to p, and xp = x (mod p) (2)

For any x.

Evidence. It suffices to prove the validity of equations (1) and (2) for xp. We carry out the proof by induction.

It is obvious that equation (3) is satisfied for x = 0 and 1.

Further, xp = (x -1 + 1) p = C (p, j) (x -1) j = (x -1) p + 1 (mod 2),

Since C (p, j) = 0 (mod p) for 0 <j <p.

Taking this inequality into account and proposing the method of proof by induction, the theorem is proved.

Definition. Euler function φ (n) is the number of positive integers less than n and simple with respect to n.

Theorem 2. If n = pq, (p and q are prime numbers different from each other), then

$$\varphi(n)=(p-1)(q-1).$$

Theorem 3. If n = pq, (p and q are prime numbers different from each other) and x is prime with respect to p and q, then $x\varphi(n) = 1 \pmod n$.

Consequence. If n = pq, (p and q are prime numbers different from each other) and e is prime relative to (n), then the mapping Eb, n: xxb (mod n)

Is one-to-one on Zn.

It is also obvious that if e is simple with respect to f (n), then there exists an integer d such that $ed \equiv 1 \pmod{\varphi(n)}$. (3)

On these mathematical facts, the popular RSA algorithm is based.

Let n = pq, where p and q are distinct prime numbers. If e and d satisfy the equation, then the mappings Ee, n and Ed, n are inversions on Zn. Both Ee, n, and Ed, n are easily calculated when e, d, p, q are known.

If e and n are known, but p and q are not known, then Ee, n is a one-way function; Finding Ed, n for a given n is equivalent to the decomposition of n. If p and q are sufficiently large simple, then the decomposition of n is practically impracticable. This is the basis of the RSA encryption system.

Let n = pq be an integer represented as a product of two large prime numbers p, q. The numbers e and d, which determine the encryption and decryption algorithms, respectively, must satisfy the condition

$$ed \equiv \pmod{\varphi(n)}, \qquad (1)$$

Where $\varphi(n) = (p-1)(q-1)$ is the value of the Euler function of the number n.

Let k = (n, p, q, e, d) be the selected key consisting of the public key k1 = (n, e) and the secret key k2 = (n, p, q, d).

Let M be a block of plaintext and C the corresponding block of ciphertext. Then the rules for encryption and decryption are defined by the formulas:

$$C = E_k(M) = M^e \pmod n, \qquad D_k(C) = C^d \pmod n. \qquad (2)$$

We note that, in accordance with (2), Dk (C) = M.

When finding the values of e and d satisfying condition (1), the value of e is usually set so that it is coprime with $\varphi(n)$, and the value of d is determined from equation

$$\varphi(n)x + ed = 1 \qquad (3)$$

In the general case, equation (3) has the form ax + by = c (here $a = \varphi(n)$, b = e, y = d) and is called the Diophantine equation.

The solution of this equation

$$y = (-1)^{\mu} \cdot a_{\mu-1} \cdot x = (-1)^{\mu+1} \cdot b_{\mu-1} \qquad (4)$$

Can be obtained by expanding the ratio a / b into a continued fraction:

$$\frac{a}{b} = r_0 + \cfrac{1}{r_1 + \cfrac{1}{r_2 + \cfrac{1}{r_3 + \ldots}}}, \quad \ldots \cfrac{1}{r_\mu + 0}$$

Where μ is the order of the continued fraction, i.e. The index of the fraction coefficient, for which the remainder is zero,

$$\begin{cases} a_0 = r_0 \\ b_0 = 1 \end{cases}, \quad \begin{cases} a_0 = r_0 \cdot r_1 + 1 \\ b_1 = r_1 \end{cases}, \tag{5}$$

And for all members, starting with the third one, it is fair

$$\begin{cases} a_i = r_i \cdot a_{i-1} + a_{i-2} \\ b_i = r_i \cdot b_{i-1} + b_{i-2} \end{cases}. \tag{6}$$

Thus, to solve equation (3), it is necessary to represent the ratio a / b in the form of a continued fraction, while determining the values r0, r1 ... rμ and μ. Then, in accordance with (4) - (6), the values of ai, bi, and also x and y are determined.

Example.

We encrypt the abbreviation RSA using p = 17, q = 31. To do this, we compute n = pq = 527 and μ (n) = (p-1) (q-1) = 480. Next, we choose e as a number that is relatively prime to M (n), for example, e = 7. Using the continued fractions, we find integers x and y satisfying the relation μ (n) x + ey = 1. We write 480x + 7y = 1

$$\frac{480}{7} = 68 + \frac{4}{7}, \quad \frac{7}{4} = 1 + \frac{1}{3}, \quad \frac{4}{3} = 1 + \frac{1}{3}, \quad \frac{3}{1} = 3 + 0.$$

Consequently,

M = 3, a0 = 68, b0 = 1, a1 = 69, a2 = 1 • 69 + 68 = 137, b2 = 1 • 1 + 1 = 2.

Thus, x = -2, y = -137.

Since -137 (mod 480) = 343, then d = 343.

Checking 7 • 343 = 2401 = 1 (mod 480).

Now we will present this message in the form of a sequence of numbers contained in the interval 0 ... 526. For this, the letters R, S and A are coded with five-dimensional binary vectors, using the binary notation of their ordinal numbers in the alphabet:

R = 18 = (10010), S = 19 (10011), A = 1 (00001).

Then RSA = (100101001100001). Filling into the given interval 0 ... 526, we obtain the following representation:

RSA = (100101001), (100001) = (M1 = 297, M2 = 33).

Then sequentially encrypt M1 and M2:

C1 = Ek (M1) = M1B = 2977 (mod 527) = 474.

In doing so, we took advantage of the fact that

$$\left[a^{b}\right](\mathrm{mod}\, n) = \left[\left(a^{b}(\mathrm{mod}\, n)\right)\right]^{b}(\mathrm{mod}\, n),$$
$$\left[a^{b+c+1}\right](\mathrm{mod}\, n) = \left[a^{b}(\mathrm{mod}\, n)\right]\cdot\left[a^{c}(\mathrm{mod}\, n)\right]\cdot\left[a^{1}(\mathrm{mod}\, n)\right],$$

those

2977 = ((2972) 3297) (mod 527) = ((2003 (mod 527) 297) mod 527),

C2 = Ek (M2) = M2B = 337 (mod 527) = 407.

As a result we get the ciphertext: C1 = 474, C2 = 407.

When decrypting, you need to do the following.

First, calculate

Dk (C1) = C1d = 337 (mod 527) = 474343 (mod 527).

Note that when raising to power, it is convenient to use the fact that 343 = 256 + 64 + 16 + 4 + 2 + 1.

On the basis of this representation we obtain:

4742 (mod 527) = 174, 4744 (mod 527) = 237,

4748 (mod 527) = 307, 47416 (mod 527) = 443,

47432 (mod 527) = 205, 47464 (mod 527) = 392,

474128 (mod 527) = 307, 474256 (mod 527) = 443,

Whereby 474343 (mod 527) = (443□392□443□237□174□474) (mod 527) = 297,

Similarly, 407343 (mod 527) = 33.

Returning to the alphabetic record, we get after deciphering the RSA.

2.3.8 Hashing and digital signature of RSA documents

Previous data are used to obtain the hash code m for the message M using the hash function H, taken from the CCITT recommendations X.509. The initialization vector H0 is set equal to zero.

The digital signature is calculated by the RSA method under the electronic document M, using the calculated hash code m and the secret key d.

The hash function of CCITT X.509 is written as follows:

Hi = [(Hi-1 □ Mi) 2] (mod n), where i = l, n, H0 is the initialization vector, Mi = M1, M2, M3 ..., Mn is the block length.

All blocks are divided in half and an equivalent number of units is added to each half. With the blocks transformed in this way, they perform integration actions.

The procedure for calculating the hash code:

1) Get the value of the module: n = pq;

2) Present a message in the form of the numbers of the letters of the Russian alphabet in decimal and binary forms:

3) Split the byte in half by adding one unit of the nibble to the beginning, and get hashed blocks of Mi:

4) Take interactive steps;

Thus, the original message in the final integration will have a hash code m = the last computation.

To calculate the digital signature, use the following formula:

$$S=m^d \pmod n.$$

The pair (M, S) is transmitted to the recipient as an electronic document M, signed with a digital signature S, and the signature S is generated by the owner of the secret key d.

After receiving the pair (M, S), the receiver computes the hash code of the message M in two ways:

1) Restores the hash code m 'by applying a cryptographic signature conversion S using the public key e:

$$M' = Se \pmod n.$$

2) Find the result of hashing the received message using the same hash function: m = H (M).

When the calculated values of m 'and m are equal, the recipient recognizes the pair (M, S) as genuine.

Control questions

1. Give the characteristics of the Wizner encryption system, as a cipher for a complex replacement.
2. Modern symmetric cryptosystems.
3. American standard of data encryption DES.
4. The Russian standard of data encryption GOST 28147-89.
5. The concept of cryptosystem with a public key.
6. Cryptosystem RSA. The procedure for encryption and decryption in RSA.
7. Cryptographic methods of information protection.
8. Encryption by the method of gammation.

3. Protecting networks from remote attacks using firewalls

Intensive development of global computer networks. The emergence of new information retrieval technologies attracts more and more attention to the Internet from private individuals and various organizations. Many organizations decide to integrate their local and corporate networks into a global network. Due to the openness of its ideology, the Internet provides much greater opportunities for intruders than traditional information systems. Therefore, the issue of the protection of networks and its components becomes quite important and relevant.

A number of tasks to address the most likely threats to internal networks are able to solve firewalls. In the domestic literature, until recently, other terms of foreign origin were used instead of this term: a firewall and a firewall. Outside the computer sphere, a firewall is a wall made of non-combustible materials and

preventing the spread of fire. In the field of computer networks, a firewall is a barrier that protects against a Figuretive fire - attempts by intruders to intrude into the internal network in order to copy, modify or erase information, or take advantage of the memory or processing power of computers operating on the network. The firewall is designed to provide secure access to an external network and limit the access of external users to the internal network.

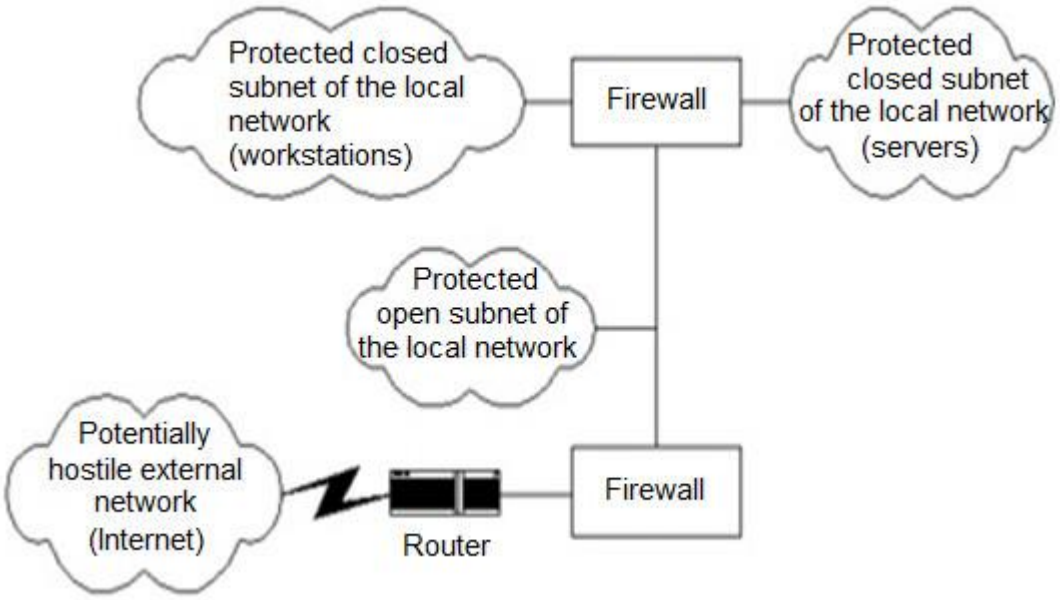


Figure 3.1 – Example of the connection scheme for firewalls

Firewalls (firewall, firewall) make it possible to filter incoming and outgoing traffic going through the system. The firewall uses one or more sets of "rules" to check network packets when they enter or leave a network connection, or it allows traffic to pass or blocks it. Firewall rules can verify one or more of the characteristics of a packet, including, but not limited to, the protocol type, the source or destination host address, and the source or destination port.

Firewalls can seriously increase the level of security of the host or network. They can be used to perform one or more of the following tasks:

• To protect and isolate applications, services and machines on the internal network from unwanted traffic coming from an external Internet network.

• To restrict or deny access to the hosts of the internal network to the services of the external Internet network.

• To support network address translation (NAT), which allows the use of private IP addresses on the internal network (either through one dedicated IP address or through an address from the pool of automatically assigned public addresses).

A firewall (FW) is a firewall system that allows you to divide a common network into two parts or more and implement a set of rules that determine the

conditions for passing packets with data across the border from one part of the common network to another. As a rule, this boundary is between the corporate (local) network of the enterprise and the global Internet network, although it can be carried out within the enterprise corporate network. The FW passes all traffic through itself, taking for each passing packet a decision - to skip it or discard it. In order for the FW to do this, it needs to define a set of filter rules.

The main argument in favor of using the firewall is that without it, internal network systems are endangered by poorly protected Internet services, as well as by probing and attacking from any other host computers on the external network.

Most firewall components can be classified in one of three categories:

1) filtering routers;
2) network-level gateways;
3) application-level gateways.

These categories can be considered as the basic components of real firewalls. Only a few firewalls include only one of the listed categories. However, these categories reflect the key capabilities that distinguish firewalls from each other.

To protect the corporate or local network, the following main firewall organization schemes are used:

1) firewall – filtering router;
2) a firewall based on a two-port gateway;
3) firewall based on a shielded gateway;
4) The firewall is a shielded subnet.

A firewall based on packet filtering is the most common and easiest to implement. It consists of a filtering router located between the protected network and the Internet. The filtering router is configured to block or filter incoming and outgoing packets based on the analysis of their addresses and ports. Computers in the protected network have direct access to the Internet, while most of the access to them from the Internet is blocked. Such dangerous services as X Windows, NIS and NFS are often blocked. In principle, the filtering router can implement any of the security policies.

A firewall based on a two-port application gateway includes a bi-homed host with two network interfaces. When information is transferred between these interfaces, the main filtering is performed. To provide additional protection between the application gateway and the Internet, a filtering router is usually placed. As a result, an internal shielded subnet is formed between the application gateway and the router. This subnet can be used to place available information servers from outside.

Unlike the firewall with a filtering router, the application gateway completely blocks IP traffic between the internet network and the protected

network. Only authorized proxy servers located on the application gateway can provide services and access to users.

This version of the firewall implements a security policy based on the principle "all that is not allowed in explicit form" is prohibited, while the user is not available to all services except those for which the appropriate authority is defined. This approach provides a high level of security, only routes to the protected subnet are known only to the firewall and are hidden from external systems. The considered scheme of firewall organization is quite simple and quite effective. It should be noted that the security of a dioecious host computer used as an application gateway must be maintained at a high level. Any breach in its protection can seriously weaken the security of the protected network. If the gateway is compromised, the attacker will have the opportunity to enter the protected network. This firewall can require users to use strong authentication, as well as registration of access, attempts to probe and attack the system of the intruder.

A firewall based on a shielded gateway combines a filtering router and an application gateway that is allowed by the internal network. The application gateway is implemented on the host computer and has only one network interface. In this scheme, the primary security is provided by the filtering router.

A firewall consisting of a shielded subnet represents the development of a firewall scheme based on a shielded gateway. Two shielding routers are used to create a shielded subnet. The external router is located between the internet network and the screened subnet, and the internal router is located between the shielded subnet and the protected internal network. A shielded subnet contains an application gateway, and can also include information servers and other systems that require controlled access. This firewall scheme provides good security by organizing a shielded subnet that better isolates the internal protected network from the Internet.

Some firewalls allow you to organize virtual corporate networks. Several local networks connected to the global network are united into one virtual corporate network. Data transfer between these local networks is done transparently for users of local networks. Confidentiality and integrity of transmitted information should be provided by means of encryption, the use of digital signatures. When transferring data, not only the contents of the packet can be encrypted, but also some header fields.

3.1 Batch filtering. Use of routers as a firewall

Filtering is carried out at the transport level: all packets or data frames passing through the firewall are analyzed, and those that have specified ("unresolved") values in certain fields are discarded.

Skipping the internal network of network layer packets or link-layer frames by addresses (MAC addresses, IP addresses, IPX addresses) or TCP port numbers corresponding to applications. For example, in order for telnet traffic to not cross

the internal network boundary, the firewall must filter all packets whose TCP header contains the destination port address of the receiving process, equal to 23 (this number is reserved for the telnet service). It is more difficult to monitor FTP traffic, which works with a large range of possible port numbers, which requires more complex filtering rules.

Of course, a normal router can be used to filter packets, and indeed, on the Internet 80% of packet filters work on the basis of routers. However, routers cannot provide the degree of data protection that firewalls guarantee.

The main advantages of filtering a firewall in comparison with filtering by a router are as follows:

1) The firewall has much more advanced logical capabilities, so unlike a router, it can easily, for example, detect fraud by IP address.

2) The firewall has great audit capabilities for all security-related events.

3.2 Features of the functioning of the FW at different levels of the OSI model

The FWs support interworking security at various levels of the OSI model. In this case, the protection functions performed at different levels of the reference model are significantly different from each other. Therefore, a complex FW is conveniently represented as a set of indivisible screens, each of which is oriented to a separate level of the OSI model.

Most often, a complex screen operates at the network, session and application levels of the reference model. Accordingly, there are distinguished such indivisible FWs (Figure 3.2), as:

- Shielding router;
- Session-level gateway (shielding transport);
- Application-level gateway (shielding gateway).

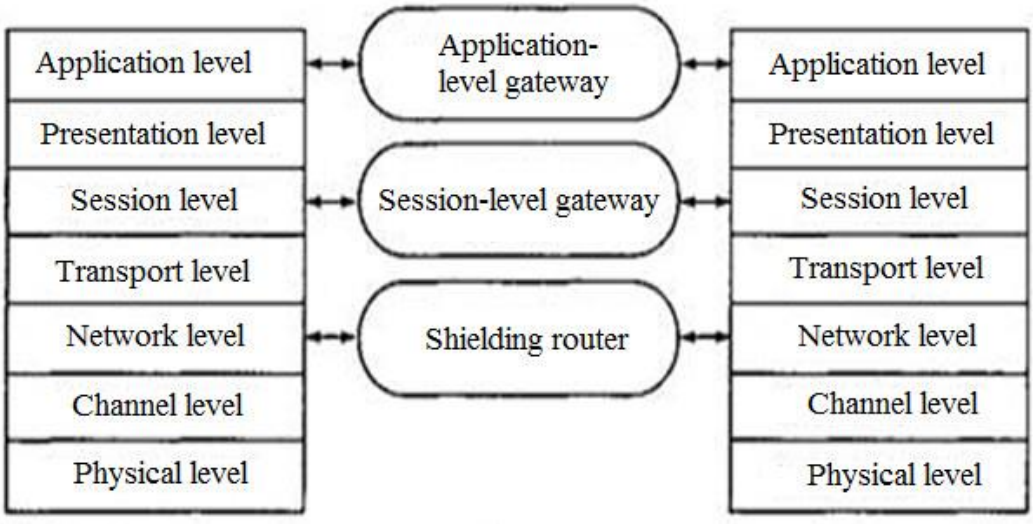


Figure 3.2 – Types of firewalls functioning at singl levels of OSI model

The protocols used in the networks (TCP / IP, SPX / IPX) do not fully correspond to the OSI reference model, so the screens of the listed types can cover neighboring levels of the reference model when performing their functions. For example, an application screen can automatically encrypt messages when they are transferred to an external network, and also automatically decrypt cryptographically closed received data. In this case, such a screen functions not only on the application layer of the OSI model, but also at the presentation level.
The gateway of the session layer, in its functioning, covers the transport and network layers of the OSI model. A shielding router, when analyzing message packets, checks their headers not only in the network, but also in the transport layer.

FW of these types have their advantages and disadvantages. Many of the used FWs are either application gateways or shielding routers, not providing full interconnect security.

There are two main versions of the design of the FW: software and firmware. In turn, the firmware version has two versions - in the form of a specialized device and as a module in a router or switch.

Currently, a software solution is more often used, which at first glance looks more attractive. This is due to the fact that for its application it is enough, it would seem, only to purchase FW software and install it on any computer available in the organization. However, in practice, not always in the organization is a free computer that satisfies sufficiently high requirements for system resources. Therefore, simultaneously with the acquisition of software, a computer is also purchased to install it.

Control questions

1. Give the features of the functioning of firewalls.
2. The main components of firewalls.
3. Filtration routers.
4. Network-level gateways.
5. Application-level gateways.
6. Shielding router.
7. Firewall system.

4. Management of cryptographic keys

One of the most important aspects of cryptography is the key. In cryptography, the key is a variable that is introduced into the algorithm used to encrypt the data. Usually the key is a secret value or carries a secret component. It is important to ensure that the key remains a secret.

It may be difficult to develop key generation rules without taking into account the entire cryptographic environment and the software used to generate the keys. The rules can provide for the development of working instructions on which to work, leaving to the discretion of administrators the development of appropriate technology. The following questions should be included in the working instructions.

• The format allowed for generated keys is either binary or plain text.

• The way keys are stored. This can include online storage devices, removable storage devices, as well as devices that store keys inside themselves.

• Determine the expiration date of the key. For algorithms that use a public key, the certificate expiration date may contain the expiration date of the key. For symmetric keys, it is necessary to have administrators who must work with users to regenerate the keys when they expire.

• Require that the key generation algorithms and software used for this are not publicly available.

Another consideration regarding the generation of keys is related to the processing of materials used in the generation of keys. The rules that prescribe the destruction of the materials used to generate the keys involve ensuring that the memory used to generate the keys should not contain any residual information that can be read using another program. In addition, other tools, such as floppy disks that can be used to transfer keys from the computer on which the keys were generated, must also be taken into account in these rules. The formulation of the rules can look like this.

All materials used in the generation of cryptographic keys must be destroyed after using them. All memory and external storage devices must be carefully wiped out or physically destroyed.

When questions arise about which technology to use, the answer is usually the use of standards. However, if an organization uses an open cryptographic key and tries to create a public key infrastructure (PKI), the standards are constantly changing, and it is difficult to answer this question. Manufacturers can provide instructions, but care must be taken to ensure that these instructions do not conflict with the rules of the organization, because this can lead to the blocking of their own decisions.

Proceeding from the tasks set by the security policy, it is possible to distinguish three areas that need to be considered in the rules of key management: the disclosure and removal of keys, the storage of keys and the forwarding of keys. This, of course, is not a complete list, but these are the main questions from which it is necessary to begin the development of rules.

4.1 Disclosure of keys

Regardless of the type of encryption system used, at some stage the keys must be opened. If the organization is connected to a virtual private network, then

the network devices on which encryption is performed, the keys are generated for those who start work, or are replaced, if the expiration date has expired. This will happen regardless of whether the organization itself maintains the environment or the environment is supported by the service provider.

The keys can be disclosed by order of law enforcement agencies. Law enforcement authorities can receive orders to control the transfer of data to your organization's networks. If they are encrypted, the court can request the provision of all the features of the encryption algorithm used, as well as the keys that encrypt the data. Despite the fact that it can embarrass anyone, you have to put up with it.

If an organization uses external services that use an encryption system, then providers often manage keys using key recovery systems. Providers will argue that this simplifies the key replacement process. But it also simplifies the disclosure of keys, and the organization will not know who it was done. If it is a criminal investigation concerning an organization in some way, law enforcement agencies can present a warrant to the service provider, and the organization will not even know about this case. Despite the fact that these phrases can be regarded as if the author is advocating the concealment of illegal activities, the author believes that in this case compliance with the organization of laws, and even more assistance for law enforcement will be difficult.

Providing key management is very important to ensure confidentiality of encrypted data. Despite the fact that the rules look very elaborate, some rules must be added to avoid confusion. The formulation of the key management rules can look like this.

Cryptographic keys can be disclosed only at the request of legal authorities.

This wording does not affect the withdrawal of keys, the management of keys by third parties or the disclosure of keys of employees when they are dismissed. These are real aspects of the rules, which cannot be viewed in a general way. When working with a service provider, the organization should obtain from the provider a wording of the rules explaining its approach to the rules of key disclosure.

4.2 Key storage

Certain aspects of key storage cannot be controlled. Hardware encryption has the memory resources necessary for their proper operation. The software must have online memory resources, including those that are in RAM. The scope of the rules governing the storage of keys includes the creation of backup copies and other options for storing keys.

Key storage rules can dictate how to store keys, how to make backup copies, or ensure that they are forwarded. But it is especially important to consider the case of storing keys on the same device or storage medium where protected data is stored. In one of the discussions, someone noticed that storing keys on the

same disk as the protected data is like leaving the key under the carpet in front of the door. The formulation of the rules is very simple.

Keys should not be stored on the same disk as the protected data.

As for the rules regarding other aspects of key storage, such as the destruction of keys on a medium, most organizations prefer not to include these requirements in the rules, but include them in the procedures.

4.3 Transferring keys

Any key encryption algorithm has a key replacement function. The public key or asymmetric encryption technologies involve fewer questions, since the public key can be forwarded openly without worrying about hacking (for an explanation of what public key cryptography is, see Chapter 6). Public keys are used as part of the PKI, and they can also be replaced based on certification authority, which allows you not only to store keys, but also to digitally sign them to verify their ownership.

When using symmetric encryption, you need to find alternative ways to forward keys. When initializing a connection that has cryptographic support for symmetric encryption to protect the forwarding, an out-of-band method of forwarding the key to the remote workstation must be found. The word "out-of-band" implies some method of forwarding keys not in the way in which the data is sent. For example, using standalone methods such as a messenger that transfers a floppy disk or tape is considered a method of out-of-band forwarding. In some organizations, procedures are introduced to initialize the encryption device (or VPN) before sending the key to the remote workstation. After initialization, the old key can be used to forward a new key. However, if the old key has been compromised, the electronic transfer of a new key in this way becomes meaningless from a security point of view.

If an organization uses external VPN services, these issues will be resolved by the service provider. However, an organization may ask the provider how it manages and forwards these keys through a variety of network connections. Despite the fact that these issues are never reflected in the rules, you can develop rules for reviewing this information in conjunction with the service provider.

Many of those who manage the forwarding of their own keys, forward keys, using the same methods that are used to send conventional data. One organization installed a PKI with certified authority checks to manage its keys via a modem installed in a system that is almost completely isolated from the rest of the organization's network. The organization was guided by a simple rule that prescribes out-of-band forwarding. Here it is.

Under any management, the public key / asymmetric cryptographic keys should not be forwarded using the same network through which the encrypted data is sent. All symmetric cryptographic keys must be physically replaced, and not sent over any network.

Note that the rules do not define the transfer of symmetric keys. This organization understood that if old keys are compromised, then sending new keys, at which old keys are used for encryption, becomes meaningless.

4.4 Distribution with symmetric keys

To encrypt large messages, cryptography with symmetric keys is more efficient than cryptography with asymmetric keys. Cryptography with symmetric keys, however, needs a secret key that is used by the two sides.

If Alice should exchange confidential messages with N people, she needs N different keys. What if N people have to communicate with each other? Then the required total number of keys is N (N - 1). If we let Alice and Bob use two identical keys for bidirectional communication for both directions, then only N (N - 1) / 2 keys are needed. This would mean that if one million people are connected to each other, each person has almost one million different keys. In total, almost one trillion keys are needed. This is called an N-problem, because the number of keys required for N objects is N2.

The number of keys is not the only problem; the distribution of keys is another misfortune. Alice (Алиса) and Bob (Боб) want to contact each other. They need a way to exchange secret keys. If Alice wants to contact one million people, how can she exchange one million keys with one million people? Using the Internet is clearly not a safe method. Obviously, we need an effective way to maintain and distribute the keys of secrecy.

4.5 The Key Distribution Center: KDC

A practical solution is to involve a third person who is trusted. It is called here the KDC (Key Distribution Center). To reduce the number of keys, each person sets a public encryption key with the KDC, as shown in Figure 5.1.

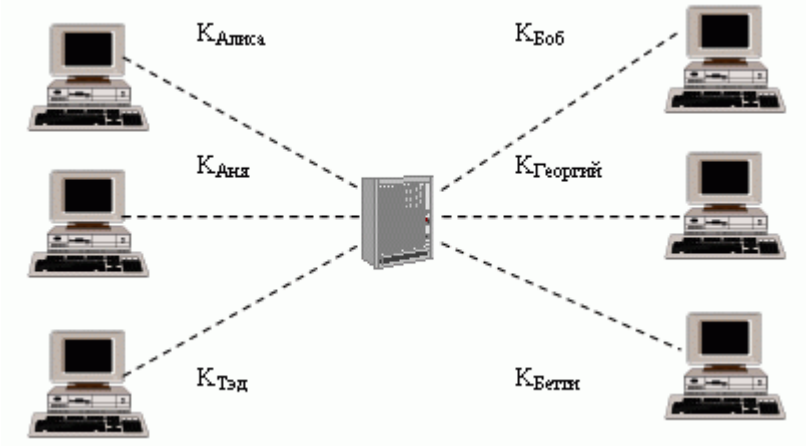


Figure 4.1 – Key Distribution Center (KDC)

The secret key is set between the KDC and each member of the community. Alice has a secret key with the KDC, which we call KAlice. Bob has a secret key

with the KDC, which we call KBob. Now the question is how Alice can send a confidential message to Bob. The process is as follows:

1. Alice sends a KDC request - a statement that she needs a session (temporarily) and a secret key between herself and Bob.
2. The KDC informs Bob about Alice's request.
3. If Bob agrees, a session key is created between them.

The secret key between Alice and Bob, which is installed with the KDC, is used to confirm the authenticity of Alice and Bob to the KDC and prevent Eve from playing the role of any of them. We will discuss later in this material how the session key is set between Alice and Bob.

When the number of people using the KDC (Key Distribution Center) increases, the system becomes unmanageable and its bottleneck triggers - the number of keys can end. To solve the problem, we must have a lot of KDCs. We can divide the world into domains. Each domain can have one or more KDCs (for redundancy in the event of a failure). Now, if Alice wants to send a confidential message to Bob, who belongs to a different domain, she comes into contact with her KDC, which, in turn, gets in touch with the KDC in Bob's domain. Two KDCs can create a secret key between Alice and Bob. Figure 4.2 shows KDCs, where the centers are of the same level. We call such an organization of centers - a "flat (non-hierarchical) set of centers (KDC)".

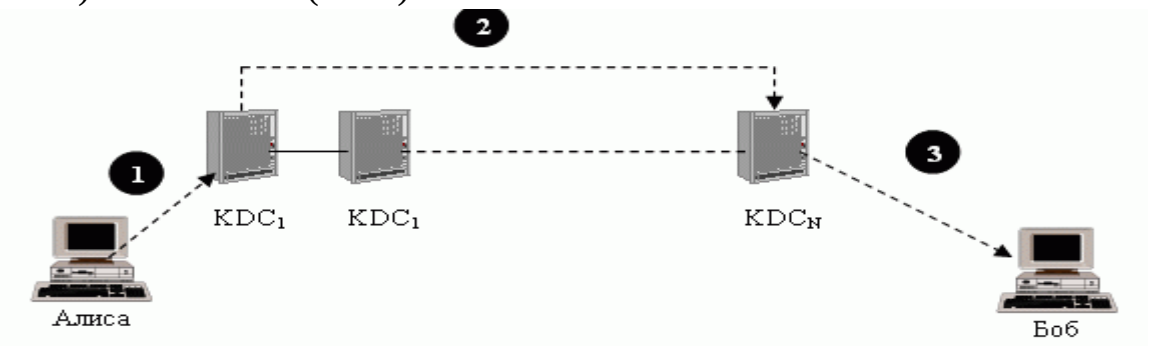


Figure 4.2 – A flat (non-hierarchical) set of KDC

A hierarchical set of key distribution centers

The concept of a flat set of KDCs can be extended to a hierarchical KDCs system, with one or more KDCs at the top level of the hierarchy. For example, local KDCs, national KDCs and international KDCs may exist. When Alice should contact Bob, who lives in another country, she sends her request to the local KDC; Local KDC retransmits the request to the national KDC; The national KDC relays the request to the international KDC. The request is then broadcast full path down to the local KDC where Bob lives. Figure 4.3 shows the conFiguretion of the hierarchical set of KDCs.

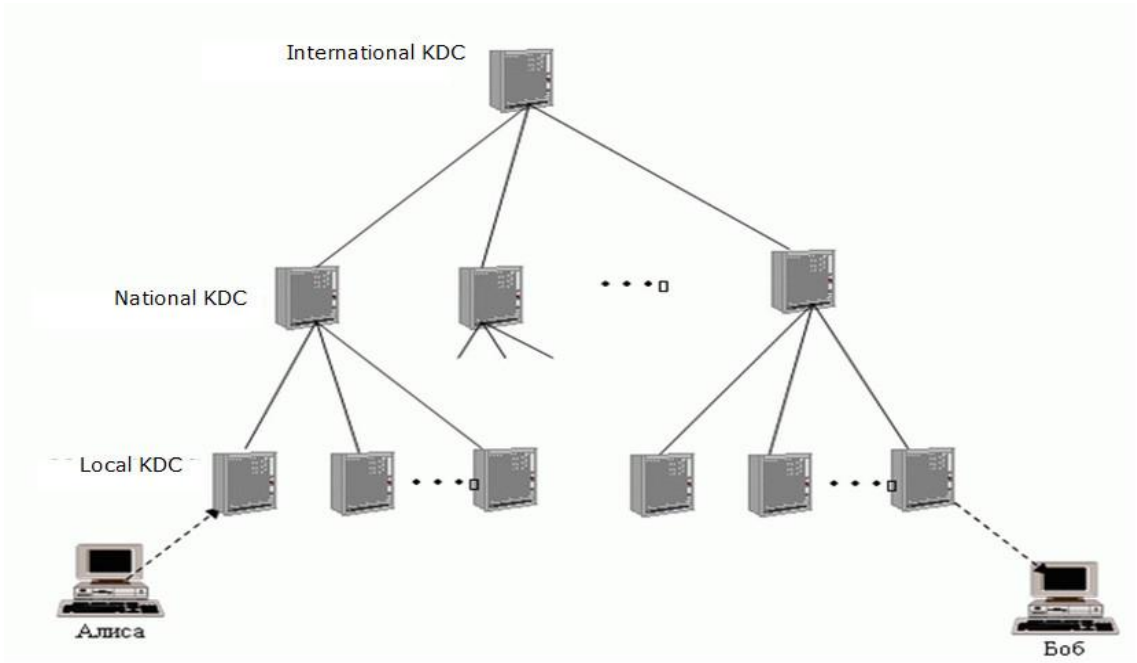


Figure 4.3 – A hierarchical set of key distribution centers

4.6 Session keys

The KDC creates a secret key for each subscriber. This secret key can only be used between the subscriber and the KDC, and not between two members of the community. If Alice should contact secretly with Bob, she needs a secret key between herself and Bob. The KDC can create a session key between Alice and Bob using their keys to the center. Alice and Bob's keys are used to confirm the availability and authority of Alice and Bob to the center and to each other before the session key is set. After the connection is completed, the session key is no longer needed.

A symmetric session key between two sides is used only once.

A simple protocol that uses the KDC

Let's see how KDS can create a session key KAB between Alice and Bob. Figure 4.4 shows the steps being taken.

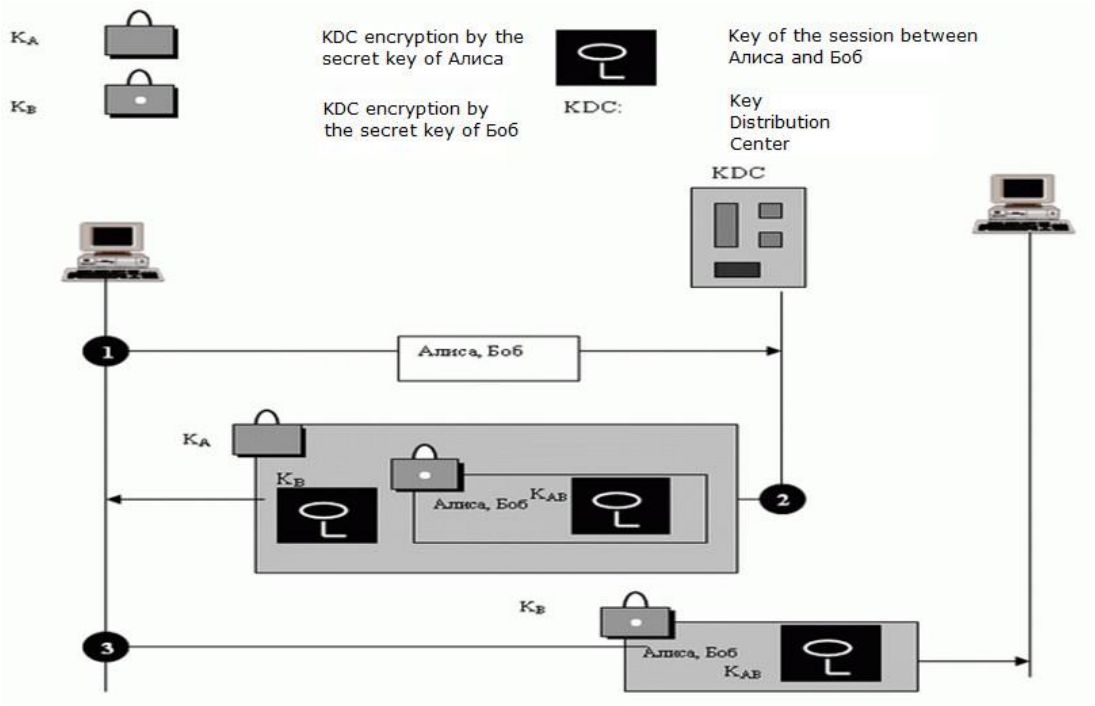


Figure 4.4 – The first method using KDC

1. Alice sends a KDC source text message to get a symmetric session key between himself and Bob. The message contains its registered identification code (Alice's word or figure) and Bob's identification code (the word Bob or the figure). The message is encrypted or publicly available - the KDC does not bother.

2. The KDC receives a message and creates what is called a ticket. The ticket is encrypted using Bob's key (KB). The ticket contains the Alice and Bob identifiers and the session key (KAB). A ticket with a copy of the session key is passed to Alice. Alice receives the message, decrypts it and retrieves the session key. She cannot decipher Bob's ticket; A ticket intended for Bob is not available to Alice. Note that the message contains double encryption: the ticket is encrypted, and the full message is encrypted. In the second message, Alice is actually registered with the KDC, so only Alice can open the whole message using her secret key with the KDC.

3. Alice passes the ticket to Bob. Bob opens the ticket and knows that Alice must send him a message that uses KAB as the session key. Please note that in this message, Bob is registered with the KDC, so only Bob can open a ticket. Because Bob is registered with the KDC, he is also registered with Alice, who trusts the KDC. In the same way, Alice is also registered with Bob, because Bob trusts the KDC, and the KDC has given Bob a ticket that includes Alice's identification code.

Unfortunately, this simple protocol has a drawback. Eve can apply the response attack considered earlier - that is, she can save the message of step 3 and use it later.

4.7 The first public key system is the Diffie-Hellman system

This cryptosystem was discovered in the mid – 1970s by American scientists Diffie and Hell, and led to a real revolution in cryptography and its practical applications. This is the first system that allows you to protect information without using secret keys transmitted over secure channels. In order to demonstrate one of the schemes of application of such systems, consider a communication network with N users, where N is a large number. Let us want to organize a secret connection for each pair of them. If we use a conventional secret key distribution system, then each pair of subscribers must be provided with its secret key, i. E.

Total

$$C^2_n = N(N-1)/2 = N2/2$$

If the subscribers are 100, then 5000 keys are required, if the subscribers are 104, then the keys should be $5\text{-}10^7$. We see that with a large number of subscribers, the system of supplying their secret keys becomes very cumbersome and expensive.

Diffie and Hellman solved this problem by openly distributing and calculating keys. We now turn to the description of the system proposed by them.

Let the communication system for subscribers A, B, C, ... is constructed. Each subscriber has his own secret and open information. To organize this system, we choose a large prime number p and some number g, 1 <g <p - 1, such that all numbers in the set {1, 2, ..., , P - 1} can be represented as different degrees of g mod p (different approaches are known for finding such numbers g, one of them will be presented below). The numbers p and g are known to all subscribers.

Subscribers choose large numbers Xa, Xh, Xc, which are kept secret (usually it is recommended to do this randomly using random number sensors). Each subscriber calculates a corresponding number Y, which is openly transmitted to other subscribers,

$$\begin{aligned} Y_A &= g^{Xa} \text{mod} p, \\ Y_B &= g^{Xb} \text{mod} p.. \\ Ye &= g^{Xc} \text{ mod p.} \end{aligned} \qquad (l)$$

The result is the following table.

Table 4.1. Diffie-Hellman User Keys

Subscriber	Secret number	Public key	Private key
A	X_A	Y_A - .--	Z_{AB}> Z_AC
B	$X_в$	Y_B	ZBA, ZBC
C	Xc	Y_c	ZCA, ZCB

Let's say subscriber A has decided to organize a communication session with B, while open information from Table 1 is available for both subscribers. 2. Subscriber A tells B on the open channel that he wants to send him a message. Subsequently, subscriber A calculates a value

$$Z_{AB} = (Y_B)^{X}{}_A \text{ mod} p \quad (2)$$

No one else but A can do this, since the number of HA is classified.
In turn, subscriber B calculates the number Xv

$$Z_{BA} = (Y_A)^{X}{}_B \text{mod} p \quad (3)$$

Figure 4.5 shows the key exchange scheme in the Diffie-Hellman system.

Let us now consider the problem of choosing the number p mentioned above. For an arbitrary given g, it can turn out to be a difficult problem connected with the factorization of g - 1

The point is that to ensure the high stability of the considered system, the number g - 1 must necessarily contain a large prime factor. Therefore, it is often recommended to use the following approach.
The number p is chosen such that the equality p = 2q + 1, (where q is also a prime number) is fulfilled, and the inequalities 1 <g <p -1 and gq mod p ≠ 1 are valid.
In order for the system to be stable to cryptanalysis, it is necessary to choose the number p to be very large.

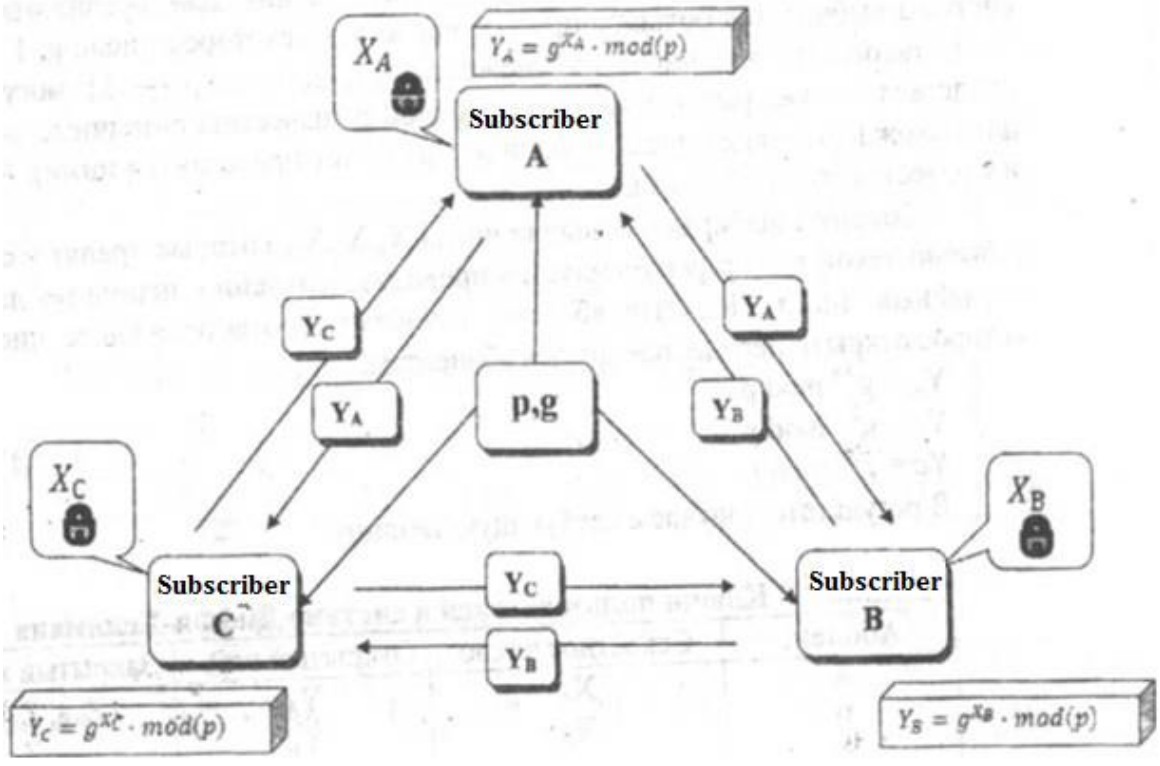


Figure 4.5 – The key exchange scheme in the Diffie-Hellman system

Example 2.1. Let g = 43. Choose the parameter p. Let's try to take q = 17401.

Correspondingly, p = 2 * 17 401 + 1 = 34,803. Let's check: 4317401 mod 34 803 = 17 746. The necessary conditions are met, hence such a p is suitable.

So, we have chosen the parameters p = 34 803, g = 43. Now each subscriber selects a secret number and calculates the corresponding open number. Let XA = 7, XB = 13 be chosen. We compute YA = 437 mod 34 803 = 11 689, YB = 4313 mod 34 803 = 14 479.

Let A and B decide to form a shared secret key. For this, A calculates ZAB = 144797 mod 34 803 = 6 415, and B calculates ZBA = 11 68913 mod 34 803 = 6 415. Now they have a common key 6 415, which was not transmitted over the communication channel.

Control questions

1. Give an algorithm for the open distribution of Diffie-Hellman keys.
2. Distribution of keys with the participation of the distribution center.
3. Management of cryptographic keys.
4. Generation, storage and distribution of keys.
5. Scheme of key exchange in the Diffie-Hellman system.
6. Method of storing keys.
7. A simple protocol that uses the KDC.

5. Virtual Private Network – VPN

Recently, in the world of telecommunications, there has been an increased interest in virtual private networks (VPNs). This is due to the need to reduce the cost of maintaining corporate networks due to cheaper connection of remote offices and remote users through the Internet. Indeed, when comparing the cost of services for connecting several networks via the Internet, for example, with Frame Relay networks, you cannotice a significant difference in cost. However, it should be noted that when combining networks via the Internet, the question of the security of data transfer immediately arises, so it became necessary to create mechanisms to ensure the confidentiality and integrity of the transmitted information. Networks built on the basis of such mechanisms, and received the name VPN.

Methods for implementing VPN networks

Virtual private network is based on three methods of implementation:

- Tunneling;
- Encryption;
- Authentication

5.1 Tunneling

Tunneling provides data transfer between two points - the end of the tunnel - in such a way that the entire network infrastructure lying between them is hidden for the source and the data receiver.

Transport environment of the tunnel, like a ferry, picks up the packets of the network protocol used at the entrance to the tunnel and without any changes delivers them to the exit. Tunneling is sufficient to connect two network nodes so that, from the point of view of the software running on them, they look connected to one (local) network. However, we must not forget that in fact a "ferry" with data passes through many intermediate nodes (routers) of an open public network.

This state of affairs is fraught with two problems. The first is that information transmitted through the tunnel can be intercepted by intruders. If it is confidential (bank card numbers, financial reports, personal information), then the threat of its compromise is quite real, which in itself is unpleasant. Worse, attackers are able to modify the data transmitted through the tunnel so that the recipient cannot verify their validity. The consequences can be the most deplorable. Given the above, we come to the conclusion that the tunnel in its pure form is suitable except for some types of networked computer games and cannot claim a more serious application.

Both problems are solved by modern means of cryptographic protection of information. To prevent unauthorized changes to the data packet on the way to its passage through the tunnel, the electronic digital signature (EDS) method is used. The essence of the method is that each transmitted packet is supplied with an additional information block that is generated in accordance with an asymmetric cryptographic algorithm and is unique for the content of the packet and the secret key of the sender's EDS. This information block is an EDS package and allows you to authenticate the data to the recipient who knows the public EDS key of the sender. Protection of data transmitted through the tunnel from unauthorized viewing is achieved by using strong encryption algorithms.

5.2 Authentication

Security is the primary function of VPN. All data from client computers pass through the Internet to the VPN server. Such a server can be located at a great distance from the client computer, and the data on the way to the organization's network passes through the equipment of many providers. How can I ensure that the data has not been read or changed? For this, various authentication and encryption methods are used.

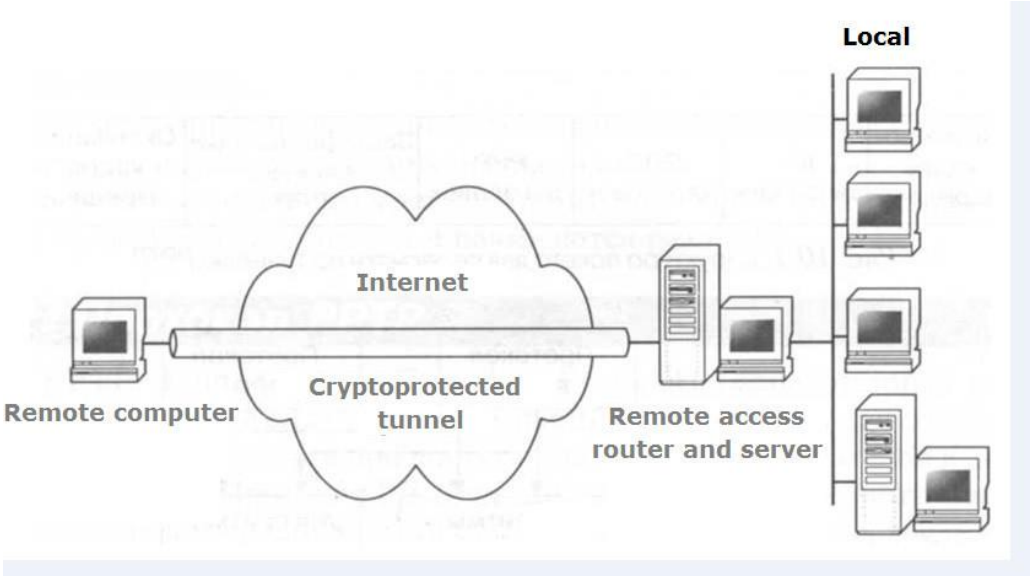


Figure 5.1 – Tunneling scheme for direct connection of the remote computer to the Internet

IPSec (short for IP Security) – a set of protocols to ensure the protection of data transmitted over the Internet IP protocol, allows authentication and / or encryption of IP packets. IPsec also includes protocols for secure key exchange on the Internet. Basically, it is used to organize vpn-connections.

5.3 IPsec architecture

IPsec protocols, unlike other well-known SSL and TLS protocols, operate at the network layer (layer 3 of the OSI model). This makes IPsec more flexible, so it can be used to protect any protocols based on TCP and UDP. IPsec can be used to secure between two IP nodes, between two security gateways or between an IP node and a security gateway. The protocol is an "add-on" over the IP protocol, and processes the generated IP packets in the manner described below. IPsec can ensure the integrity and / or confidentiality of data transmitted over the network. IP-sec uses the following protocols to perform various functions:

1) Authentication Header (AH) ensures the integrity of the virtual connection (transmitted data), the authentication of the information source and the additional function to prevent the retransmission of packets

2) Encapsulating Security Payload (ESP) can provide confidentiality (encryption) of transmitted information, limiting the flow of confidential traffic. In addition, it can ensure the integrity of the virtual connection (transmitted data), the authentication of the information source and the additional function to prevent the retransmission of packets (Whenever an ESP is applied, one or another set of security services must be used)

3) The Security Association (SA) provides a bunch of algorithms and data that provide the parameters needed to operate the AH and / or ESP. The Internet

security association and key management protocol (ISAKMP) provides the basis for authentication and key exchange, key authentication.

The IPsec protocol is used mainly for the organization of VPN tunnels. In this case, the ESP and AH protocols work in tunneling mode. In addition, by configuring security policies in a specific way, the protocol can be used to create a firewall. The meaning of the firewall is that it controls and filters the packets passing through it in accordance with the specified rules. A set of rules is established, and the screen scans through all the packets passing through it. If the transmitted packets fall under these rules, the firewall processes them accordingly. For example, it can reject certain packets, thereby stopping unsafe connections. By configuring the security policy appropriately, you can, for example, prohibit Internet traffic. For this, it is enough to forbid the sending of packets into which the HTTP and HTTPS protocols are embedded. IP-sec can also be used to protect servers-all packets are discarded, except packets that are required for the correct execution of server functions. For example, for a Web server, you can block all traffic, except for connections through the 80th TCP port, or via TCP port 443 in cases where HTTPS is used.

Security Association

To start exchanging data between two parties, you need to establish a connection called SA (Security Association). The SA concept is fundamental to Ipsec, in fact, its essence. It describes how the parties will use the services to provide secure communication. The connection SA is simplex (unidirectional), therefore for interaction of the parties it is necessary to establish two connections. It is also worth noting that Ipsec standards allow endpoints of a secure channel to use as one SA to transmit traffic of all hosts that communicate through this channel, and to create for this purpose an arbitrary number of secure associations, for example, one for each TCP connection. This makes it possible to select the desired level of detail for protection. OSI. Connection setup begins with mutual authentication of the parties. Next, the parameters are selected (whether authentication, encryption, data integrity check) and the necessary protocol (AH or ESP) of the data transfer. After that, specific algorithms are selected (for example, encryption, hash function) from several possible schemes, some of which are defined by the standard (DES for encryption, MD5 or SHA-1 for hash functions), others are added by manufacturers of products using Ipsec (for example Triple DES, Blowfish, CAST).

Control questions

1. Virtual private network as a means of protecting information.
2. Tunneling in virtual private networks.
3. The IPSec protocol.
4. The main protocols in the VPN.

5. Classification of VPN.
6. Transport and tunnel modes in IPSec.
7. Protection at the link layer protocols: PPTP, L2F, L2TP.
8. The concept of SA.
9. Transport environment of the tunnel.

6 . Computer virus

A computer virus is a type of malicious software that can create copies of itself and be embedded in the code of other programs, system memory areas, boot sectors, and distribute copies of itself across a variety of communication channels.

Typically, the purpose of the virus is to disrupt the operation of software and hardware systems: deleting files, rendering data structures unprofitable, blocking user work, etc. Even if the author of the virus has not programmed malicious effects, the virus can lead to computer failures due to errors , Unaccounted for the intricacies of interaction with the operating system and other programs. In addition, viruses tend to take up space on storage devices and consume some other system resources.

The present concept of building secure computer systems (CS) implies the use of software for various purposes in a single complex. An important point in the operation of application programs, especially information protection, is the need for potential non-interference of other applied or system programs present in the CS in the information processing process.

Recall that under unauthorized access (UA) to the resources of the protected CS understand the actions for the use, modification and destruction of programs and data produced by the entity that does not have the right to such actions. We will call this subject an attacker (violator). The remaining subjects are called legal users. This apriori division involves several essential points:

1) the system has a mechanism for distinguishing malefactors and legal users;

2) the system has passive and active components (executable modules and data), the use of which by an attacker is undesirable;

3) the system has a mechanism for establishing the compliance of the subject and the information to which he has access.

At present, the term "security policy" is used for the integrated designation of information security procedures, and all possible violations of the a priori of the prescribed rules are called security breaches.

Considering that the attacker perfectly possesses all software and hardware of the system, it can be assumed that unauthorized access can be caused by the following reasons:

At present, the term "security policy" is used for the integrated designation of information security procedures, and all possible violations of the a priori of the prescribed rules are called security breaches.

Considering that the attacker perfectly possesses all software and hardware of the system, it can be assumed that unauthorized access can be caused by the following reasons:

1) disabling or modifying protective mechanisms by an attacker. The intruder must change part of the protection system so that it ceases to perform its functions. For example, modify the access authorization system so that it skips any user, or change the encryption program so that it stops encrypting or has changed the encryption algorithm to a simpler one;

2) an attacker's entry into the system under the name and authority of a legitimate user. The intruder must in some way find out or forge the real user ID. For example, an attacker will use information that is extracted from some data set created when the attacker's software and the access control system are working together.

In both cases, the UA can be represented as a model of indirect unauthorized access, when the penetration into the system is carried out on the basis of some impact produced by the program previously introduced into the system.

A program with potentially dangerous consequences is a program that can perform any of the following functions:

1) to hide the signs of its presence in the program environment of the CS;

2) realize the self-duplication, associating oneself with other programs and (or) transferring their fragments to other (not occupied by the initially specified program) areas of operational or external memory;

3) destroy (distort in an arbitrary manner) the code of programs in the operating memory of the CS;

4) transfer (save) fragments of information from RAM into some areas of operational or external memory;

5) change arbitrarily, block and (or) replace the information output to the external memory or communication channel or change its parameters.

Programs with potentially dangerous consequences can be conditionally divided into three classes.

1. Classic virus programs. The peculiarity of this class of harmful programs lies in the non-directionality of their impact on specific types of application programs, and also that the duplication of the virus is at the heart of the matter.

2. Programs such as "software worm" or "trojan horse" and fragments of programs such as "logical hatch." In this case, there is a reverse situation: duplication itself is not always inherent in such programs or program fragments, but they have the ability to intercept confidential information, or extract information from segments of security systems, or delineate access.

3. Program bookmarks, or destructive program impacts (DPI) – a generalized class of programs (in the sense of the absence of specific features) with potentially dangerous consequences, necessarily implementing at least one of the points 3 to 5 of the definition of a program with potentially dangerous consequences.

Next, along with the term "program with potentially dangerous consequences", the terms "bookmark", "program bookmark" or reduction of the RPW will be used.

In addition, program bookmarks can be classified according to the method and place of their implementation and application:

1) bookmarks associated with the hardware / software environment of the computer system (main BIOS or extended BIOS);

2) bookmarks associated with the primary boot programs (located in the Master Boot Record or Boot sectors of active partitions), - boot bookmarks;

3) bookmarks associated with the loading of the operating environment;

4) bookmarks associated with general-purpose application software (built-in keyboard and screen drivers, computer testing programs, utilities);

5) executable modules containing only the bookmark code (usually embedded in batch processing files such as .BAT);

6) simulation modules that coincide in appearance with some programs that require the input of confidential information (most typical for Unix-systems);

7) bookmarks masqueraded as software for gammation and entertainment purposes (usually used for initial introduction of bookmarks).

As you can see, program bookmarks have much in common with classic viruses, especially in terms of associating themselves with executable code. In addition, the program bookmarks, like many well-known viruses of the classical type, have advanced tools for combating debuggers and disassemblers.

In order for a bookmark to be able to perform any actions with respect to an application program or data, it must receive control, ie the processor must start executing instructions related to the bookmark code. This is possible only if both conditions are met:

1) the bookmark must be in the RAM before the program starts, which is the target of the bookmark, therefore, it must be loaded earlier or simultaneously with this program;

2) the bookmark should be activated by some common for both the bookmark and the program event, i.e., under certain conditions in the hardware / software environment, the control must be transferred to the bookmarking program. This event will be called activating. Usually, the fulfillment of the specified conditions is achieved by analyzing and processing the effects of common (with respect to the bookmark and application) effects (usually interrupts) or events (depending on the type and architecture of the operating environment). These conditions are necessary, i.e. if they are not executed, the activation of the

bookmark code will not occur and the code will not be able to have any effect on the operation of the application program.

6.1 Classification of viruses

Known software viruses can be classified according to the following characteristics:

- habitat
- the way in which the habitat is contaminated
- Effects
- features of the algorithm

Depending on the environment, viruses can be divided into:

- Network
- file
- Bootable
- file-boot.

Network viruses spread through various computer networks.

File viruses are implemented primarily in executable modules, ie, in files that have COM and EXE extensions. They can be embedded in other types of files, but, as a rule, recorded in such files, they never get control and, therefore, lose the ability to reproduce.

Boot viruses are introduced into the boot sector of the disk (Boot-sector) or to the sector containing the boot disk (Master Boot Record).

File-boot viruses infect both files and boot sectors of disks.

By the method of infection, viruses are divided into:

- Resident
- Non-resident.

A resident virus when infected (infected) with a computer leaves its resident part in RAM, which then intercepts the invocation of the operating system to infected objects (files, boot sectors of disks, etc.) and is embedded in them. Resident viruses are in memory and are active until the computer turns off or reboots.

Non-resident viruses do not infect the computer's memory and are active for a limited time.

By the degree of exposure, viruses can be divided into the following types:

- non-dangerous, not interfering with the computer, but reducing the amount of free RAM and memory on disks, the actions of such viruses appear in any graphic or sound effects
- Dangerous viruses that can lead to various disruptions in the operation of the computer
- Very dangerous, the impact of which can lead to loss of programs, destruction of data, erasure of information in the system areas of the disk.

According to the peculiarities of the algorithm, viruses are difficult to classify due to a wide variety. The simplest viruses are parasitic, they change the contents of files and sectors of the disk and can be easily detected and destroyed. Viruses-replicators, called worms, that spread on computer networks, calculate the addresses of network computers and write down their copies to these addresses. Invisible viruses called stealth viruses, which are very difficult to detect and disinfect, since they intercept the invocations of the operating system to infected files and sectors of disks and substitute for their body uninfected areas of the disk. It is most difficult to detect mutant viruses (polymorphic viruses) containing encryption-decryption algorithms, through which copies of the same virus do not have any repetitive chain of bytes. There are also so-called quasivirus or Trojan programs, which, although not capable of self-propagation, are very dangerous, because they masquerade as a useful program and destroy the boot sector and the disk file system.

Taking into account the observation that the bookmark should be loaded into the OS earlier than the purpose of its effects, two types of program bookmarks can be distinguished.

1. Bookmarks of the resident type, stored in memory from a certain moment of time until the end of the computer. The bookmark can be loaded into memory when the computer is booted up, the operating system is loaded, or some program is launched, and also launched separately.

2. Non-resident type bookmarks that begin work, as well as bookmarks of a resident type, but finish it yourself after a certain period of time or for some event, while unloading itself from memory entirely.

Define the areas of influence of program bookmarks on computer systems. To do this, consider certain bookmark models.

1. The "interception" model. The software bookmark is embedded in ROM, OS or application software and stores all or selected fragments of the processed information in a hidden area of local or remote external memory. This model can be two-stage: initially only attribute attributes are preserved (for example, the names or the beginning of the files), then the accumulated information is removed, and the attacker decides on the specific objects of the further attack. For this model, the presence in the external memory of the storage location of information is essential, which must be organized in such a way as to ensure its safety for a specified period of time and the possibility of subsequent removal.

2. The Trojan horse model. The bookmark is built into the constantly used software and for some activating event simulates a bad situation on the means of information storage or in the equipment of the computer (network). Thus, two goals can be achieved: firstly, the normal operation of the computer system is paralyzed and, secondly, the attacker (under the guise of repair) can become acquainted with the information available in the system or the "interception" stored in the model. Events that activate the bookmark can be a certain time point, either a

signal from the modem link, or the state of some counters (for example, the number of program starts).

3. The "observer" model. The bookmark is built into network or telecommunication software. Taking advantage of the fact that this software is usually always active, the software bookmark monitors the processing of information on this computer, the installation and removal of bookmarks, and the collection of accumulated information. The bookmark can trigger events for previously embedded bookmarks that operate on the Trojan horse model.

4. The "compromise" model. The bookmark either transfers the information specified by the attacker to the communication channel, or saves it, not relying on the guaranteed possibility of subsequent reception or withdrawal.

5. The model of the "initiator of errors". The software bookmark distorts the data streams that occur during the operation of application programs, either it distorts the input information streams, or initiates errors that occur during the operation of application programs.

6. The model of "garbage collection". In this case, the direct impact of DPI may not be, as the rest of the information is being studied. In the case of the application of the program book, such an order of work is imposed to maximize the number of remaining fragments of valuable information. The attacker gets either these fragments, using the bookmarks of models 2 and 3, or direct access to the computer under the guise of repair or prevention.

The considered bookmark models have an important common feature - the existence of a write operation made by a bookmark (in the operative or external memory). In the absence of this operation, no negative impact is possible. It is understandable that for a directed action, the bookmark must also perform read operations, otherwise only the destructive functions can be realized.

The execution of the bookmark code can be accompanied by operations of unauthorized recording (URC) (for example, to save certain pieces of information) and unauthorized reading (URD), which can occur separately from the operations of reading the application program or in conjunction with them. In this case, read and write operations may not be related to obtaining information.

Unauthorized recording by bookmark can occur:

1) in an array of data that does not coincide with user information, - storing information;

2) in the data array, which coincides with the user information or its subset, - distortion, destruction or imposition of information by a bookmark.

Therefore, we can consider three main groups of destructive functions that can be performed by tabs:

1) preservation of fragments of information that occurs during user operation, applications, data input, output, external memory, including various passwords, keys and access codes, confidential documents in electronic form, or an

unaddressed compromise of fragments of valuable information (models "Interception", "compromise");

2) change of algorithms of functioning of applied programs. There is a change in the actual source algorithms of the programs (for example, the access control program will let users through any password (models "distortion", "Trojan horse"));

3) the imposition of a certain mode of operation (for example, when information is destroyed, write-to-disk is blocked, and information is not, of course, destroyed) or the replacement of recorded information imposed by a bookmark.

So, we can distinguish the following components of the software environment in which the bookmark exists – a lot of code fragments of application programs, a lot of fragments of the bookmark code and a lot of events as control transfer sequences from one code fragment to another in the software environment. In the last set, events are marked out, according to which control is transferred to a subset of fragments of the bookmark code (hereinafter referred to as their activating events).

6.2 Antivirus programs

The most effective antivirus programs in the fight against computer viruses. However, I would like to note at once that there are no antiviruses that guarantee absolute protection against viruses, and statements about the existence of such systems can be regarded as either unscrupulous advertising or unprofessionalism. Such systems do not exist, since it is always possible to propose a counter algorithm for any antivirus algorithm invisible to this antivirus (otherwise, fortunately, it is also true: an antivirus can always be created for any virus algorithm).

The problem of developing a comparative analysis of modern antivirus tools is very urgent. This is due to the fact that at present a very large number of new viruses have appeared, with which not all antivirus programs can fight. And among the anti-virus programs there are those that do not even detect the simplest viruses.

Requirements for antivirus programs are quite contradictory. On the one hand, users want to have reliable, powerful anti-virus protection. On the other hand, they want this protection does not require much time and effort from the user. And these are quite natural requirements.

The signs of the appearance of the virus include:

• slowdown of the computer;

• impossibility to load the operating system;

• frequent "hanging up" and malfunctioning of the computer;

• termination of work or malfunction of previously successfully functioning programs;

• increase the number of files on the disk;

• resizing files;

• Periodical appearance on the monitor screen of inappropriate system messages;

• Reducing the amount of free RAM;

• a noticeable increase in the access time to the hard disk;

• change the date and time of creating files;

• destruction of the file structure (the disappearance of files, distortion of catalogs, etc.);

• The drive's warning light comes on when there is no treatment.

It should be noted that these symptoms are not necessarily caused by computer viruses, they can be a consequence of other causes, so the computer should be periodically diagnosed.

The number and variety of viruses is large, and to quickly and effectively detect them, the antivirus program must meet certain parameters.

Stability and reliability of work. This parameter is undoubtedly decisive - even the best antivirus will be completely useless if it cannot function properly on your computer if, as a result of any failure in the program, the computer scan process does not go to the end. Then there is always the possibility that some infected files have gone unnoticed.

The size of the program's virus database (the number of viruses that are correctly detected by the program). Given the constant appearance of new viruses, the database should be regularly updated - what's the use of a program that does not see half of the new viruses and, as a result, creates an erroneous sense of "cleanliness" of the computer. This includes the ability of the program to identify various types of viruses, and the ability to work with files of various types (archives, documents). Another important thing is the presence of a resident monitor that checks all new files "on the fly" (that is automatically, as they are written to disk).

The speed of the program, the availability of additional features such as algorithms for detecting even viruses unknown to the program (heuristic scanning). This includes the ability to restore infected files without erasing them from the hard drive, but only removing them from viruses. Important is also the percentage of false positives program (erroneous definition of the virus in a "clean" file).

Multiplatform (the availability of versions of the program for various operating systems). Of course, if the antivirus is used only at home, on one computer, then this parameter does not really matter. But the antivirus for a large organization is simply required to support all common operating systems. In addition, when working in a network, it is important to have server functions designed for administrative work, as well as the ability to work with different types of servers.

6.3 Characteristics of anti-virus programs

Antivirus programs are divided into:

- Detector software
- Doctor programs
- audit programs
- Filter programs
- Vaccine programs.

Detector programs provide search and detection of viruses in RAM and on external media, and when detected, an appropriate message is issued. Detectors are universal and specialized.

Universal detectors in their work use the verification of the immutability of files by counting and comparing with the standard of the checksum. The disadvantage of universal detectors is connected with the impossibility of determining the causes of the distortion of files.

Specialized detectors search for known viruses by their signature (a repeating section of the code). The disadvantage of such detectors is that they are unable to detect all known viruses.

A detector that detects several viruses is called a poly-detector.

The disadvantage of such anti-virus programs is that they can find only those viruses that are known to developers of such programs.

Doctor programs (phages) not only find virus-infected files, but also "treat" them, i.e. Delete the body of the virus program from the file, returning the files to their original state. At the beginning of their work, the phages look for viruses in the RAM, destroying them, and only then go to the "treatment" of the files. Among the phages, polyphages are isolated; Doctor programs are designed to search for and destroy a large number of viruses.

Given that new viruses are constantly appearing, detection programs and doctor programs are rapidly becoming obsolete, and their versions are regularly updated.

Program-auditors are among the most reliable means of protection against viruses. The auditors remember the initial state of programs, directories and system areas of the disk when the computer is not infected with the virus, and then periodically or at the user's request compare the current state with the original one. The detected changes are displayed on the video monitor screen. As a rule, state comparisons are performed immediately after the operating system is loaded. When comparing, the file length, cyclic control code (file checksum), date and time of modification, other parameters are checked.

Program-auditors have sufficiently developed algorithms, detect stealth viruses and can even distinguish changes in the version of the program being tested from changes made by the virus.

Filter programs (watchmen) are small memory resident programs designed to detect suspicious actions when the computer is running, specific to viruses. Such actions may include:

- attempts to correct files with COM and EXE extensions;
- changing file attributes;
- direct write to the disk to the absolute address;
- write to the boot sectors of the disk.
- Download the resident program.

When any program attempts to perform these actions, the watchman sends a message to the user and proposes to prohibit or allow the corresponding action. Filter programs are very useful, because they can detect a virus at the earliest stage of its existence before reproduction. However, they do not "treat" files and disks. To destroy viruses, you need to use other programs, for example phages. Disadvantages of watchdog programs include their intrusiveness (for example, they constantly issue a warning about any attempt to copy an executable file), as well as possible conflicts with other software.

Vaccines (immunizers) are resident programs that prevent infection of files. Vaccines are used if there are no doctor programs "treating" this virus. Vaccination is possible only from known viruses. The vaccine modifies the program or disk in such a way that it does not affect their work, and the virus will perceive them as infected and therefore will not penetrate. At present, vaccine programs have limited application.

An essential shortcoming of such programs is their limited ability to prevent infection from a large number of diverse viruses.

There are several basic methods for searching for viruses that are used by antivirus programs:

- Scanning
- Heuristic analysis
- Detecting changes
- Resident monitors

Antivirus programs can implement all of the above methods, or only some of them.

Scanning

Scanning is the most traditional method of searching for viruses. It consists in finding the signatures extracted from previously detected viruses. Anti-virus scanners that can remove detected viruses are usually called polyphages.

The disadvantage of simple scanners is their inability to detect polymorphic viruses that completely change their code. To do this, you need to use more sophisticated search algorithms, including heuristic analysis of the programs being tested.

In addition, scanners can detect only known and previously studied viruses for which a signature has been determined. Therefore, scanner programs will not

protect your computer from the penetration of new viruses, which, by the way, appears on several pieces per day. As a result, the scanners become obsolete at the time of the release of the new version.

Heuristic analysis

Heuristic analysis is often used in conjunction with scanning to search for encrypted and polymorphic viruses. In most cases, heuristic analysis also detects previously unknown viruses. In this case, most likely their treatment will be impossible.

If the heuristic analyzer reports that the file or boot sector is possibly infected with a virus, you should treat it with great attention. It is necessary to additionally check such files with the help of the latest versions of the antivirus programs of the scanners or send them for investigation to the authors of the anti-virus programs.

Detecting changes

While infecting a computer, the virus makes changes on the hard disk: it appends its code to the infected file, changes the system areas of the disk, etc. The detection of such changes is based on the work of the antivirus program-auditors.

Antivirus program-auditors remember the characteristics of all areas of the disk that can be attacked by the virus, and then periodically check them. In case of detection of changes, a message is issued that it is possible that a virus has attacked the computer.

It should be borne in mind that not all changes are caused by the invasion of viruses. So, the boot record can change when the operating system version is updated, and some programs write data inside their executable file.

Resident monitors

Antivirus programs that reside in the computer's memory and track all suspicious activities performed by other programs are called resident monitors or watchmen. Unfortunately, resident monitors have many drawbacks that make this class of programs unsuitable for use. They irritate users with a large number of messages, for the most part not related to virus infection, as a result of which they are disconnected.

When choosing an anti-virus program, it is necessary to take into account not only the percentage of virus detection, but also the ability to detect new viruses, the number of viruses in the anti-virus database, the frequency of its updating, and the availability of additional functions.

Currently, a serious antivirus should be able to recognize at least 25,000 viruses. This does not mean that they are all "free". In fact, most of them either have already ceased to exist or are in laboratories and are not distributed. In reality, you can meet 200-300 viruses, and only a few dozen of them are dangerous.

Control questions

1. Computer viruses.
2. Means of antivirus protection.
3. The effects of program bookmarks.
4. Classification of viruses.
5. Characteristics of antivirus programs.
6. Signs of the appearance of the virus.
7. The size of the virus database program.
8. Requirements for anti-virus programs.

List of abbreviations

TCS – telecommunication systems
DB – databases
RT – routing tables
SN – source node
RN – receiver node
S_wT – Switching tables
LAN – communication lines
S_wN – switching nodes
SA – Security Agent
UNI + Sec – user–network+protection
NNI + Sec – node–network+protection
Proxy server – proxy server
T – reversible transformations of plaintext into encrypted
K – key
DES – American standard of encryption
EDS – electronic digital signature
PRS is a pseudo-random sequence
PRN – pseudo-random numbers
GOST 28147-89 – Russian Encryption Standard
RSA – cryptosystem received the name in honor of its creators
SOK – system with a public key
NAT – network address translation (support for network address translation)
FW – firewall
NIS – Network Information System
IP – Internet Protocol Address – a unique identifier (address) of a device (usually a computer) connected to a local network or the Internet.
MAC – Media Access Control – media access control
IPX – serves as a network adapter driver
TCP – transmission control protocol
FTP – File Transfer Protocol – File Transfer Protocol
OSI – open systems interconnection basic reference model – basic reference model of open systems interaction, – network model of OSI / ISO network protocols stack
TCP / IP – Transmission Control Protocol (TCP) and Internet Protocol (IP) set of network communication protocols
SPX / IPX – internetwork packet exchange / sequenced packet exchange – inter-packet exchange / sequential packet exchange
Software – Software
PKI – public key infrastructure
VPN – Virtual Private Network

KDC – Key-Distribution Center

IPSec (short for IP Security) – a set of protocols to ensure data protection

SSL – secure sockets layer – level of protected cokes

TLS – Transport Layer Security – transport layer security

AH – Authentication Header – authentication of information source

ESP – Encapsulating Security Payload – to provide confidentiality (encryption)

SA – Security Association – provides a bunch of algorithms and data

HTTP – HyperText Transfer Protocol – HyperText Transfer Protocol

HTTPS– HyperText Transfer Protocol Secure) – extension of the HTTP protocol

CS – computer system

UA – unauthorized access

DPI – destructive program impacts

BIOS – basic input / output system – basic input / output system

RAM – random access memory

ROM – Read-Only Memory

OS – Operating system

URC – unauthorized recording

URD – unauthorized reading

References

1. S. Simon. The book of codes. - Moscow: AST, 2007. - 448 c.

2. N. Smart. Cryptography M .: Technosphere 2005. - 528 c.

3. S.B. Makarov. Telecommunication technologies: introduction to GSM technologies: a manual for higher education. Training. Institutions. / S.B. Makarov, N.V. Pevtsov, E.A. Popov, M.A. Sivers. - Moscow: Publishing Center "Academy", 2006. - 256 p.

4. B. Ya. Ryabko. Cryptographic methods of information protection: a textbook for high schools. / B. Ya. Ryabko, A.N. Fionov. - Moscow: Hot Line - Telecom, 2005. - 229 p.

5. A.N. Volkov. UMTS. The standard of cellular communication of the third generation. / A.N. Volkov, A.E. Ryzhkov, M.A. Sivers. - St. Petersburg .: Publishing house "Link", 2008. - 224 p.

6. M.Z. Yakubova. And others. Methodical instructions for the performance of laboratory works on the discipline: "Protection of information on TCS" Electronic version. 2016.- 56 pp.

7. M. Werner. The fundamentals of coding. Textbook for high schools. - Moscow: Technosphere, 2006.

8. Protection of information in mobile communication systems: A manual for higher education institutions / A.A. Chekalin, and others, under the general scientific editorship of A.V. Zaryaeva and S.V. Concealing. - 2nd ed. - M .: Hot line-Telecom, 2005.

9. W. Mao. Modern cryptography: theory and practice / V. Mao. - St. Petersburg. : Williams, 2005.

10. V.P. Melnikov. Information security and information security. -M., 2008.

11. T.P. Partyka. Information security. -M., 2008.

12. V.G. Gribunin. Integrated system of information security in the enterprise: Proc. The manual - M .: Academy, 2009.

13. S.K.Varlataya, M.V. Shahanova. Software and hardware protection of information: teaching.- Vladivostok: FENU Publishing House, 2010.

14. F. Shanguin. Protection of computer information. Effective methods and means. - M: DMK Press, 2010.

15. V. Shangin, Comprehensive protection of information in corporate systems. Tutorial. M .: Forum, 2010.

16. P.B. Khoreev. Methods and means of information protection in computer systems. - M: Academy, 2007.

17. Belov et al. Fundamentals of Information Security. Training. M: Hot line-Telecom, 2006.

18. V.I. Yarochkin. Information Security: Textbook.-M: Academic Project, Triksta, 2005.

19. O.R. Laponina. Fundamentals of network security: cryptographic algorithms and protocols of interaction: A course of lectures: Ucheb.posobie / Pod red.V.A.Suhomlina.-M.: Internet-Un. Inform. Technologies, 2005.

20. N.A. Moldovyan, A.A. Moldovyan. Introduction to cryptosystems with a public key: ucheb.posobi.- SPb.