SECURITY PROTOCOLS

[*]Randjelović D.[*], Petrović L.[1], Radovanović R.[1], Popović

Criminal Justice and Police Academy, Belgrade, Serbia

Abstract: The Internet, as a computer network, connects millions of people all around the world and gives them a possibility to access a big quantity of data. Throughout the Internet users exchange data using certain protocols and a part of this communication is private or secret. TCP (Transmission Control Protocol) and IP (Internet Protocol) protocols are the kernel of Internet protocol. Everything that is transmitted through the Internet uses these protocols, but they cannot provide security of data transfer. For example, IP packages can be easily changed and their content can be seen by everybody in every moment, even by an unauthorized person. Today the world is already globally connected and the individuals and institutions need privacy and also the protection from identity theft that is today a very frequent aspect of misuse of the Internet. So, we need transparent and flexible tools to fulfill demands of different users and at the same time capable to achieve the assigned degree of security. Security protocols, as the most prominent SSL (Secure Sockets Layers) and TLS (Transport Layer Security), solve a good part of given problems.

Key words: Security protocols, TCP/IP kernel of protocols, SSL, TLS, computer networks, cyber criminal

1. Introduction

Users of computer systems, computers in network and independent computers, first of all want to be sure that only those who are allowed will have access to their data.

Therefore analogue to the safety of one's property, users of computer systems want the so-called computer security. The concept of computer security can be divided into four fewer parts: security made by bringing a user face to face, security from external influence, interior security mechanisms and communication security mechanisms, so that in this way it is possible to consider the four basic categories of computer security:

[*] Corresponding author: e-mail: dragan.randjelovic@kpa.edu.rs

- Authentication is the process which includes the process of identification (gives the answer to the question of the person in question) and the process of verification (only confirms the identity of person in question), that explicitly identifies the user of computer system and enables him to use data and resources in accordance with his rights;
- Cryptography is the process of data protection against unauthorized access using data coding;
- Control of the property ownership over the files (users and groups) is achievable thanks to nowdays dominant multiuser operating systems;
- Protocols of security communications.

In view of the fact of permanent threats against computer systems, in the second part - security from external influence, we can put the following category:

- Malicious programs and their two basic subcategories:
  - Intrusion detection system (IDS),
  - Systems for anti-malicious working and fire-wall for filtering of malicious programs.

1.1 Protocols

In this paper the attention is focused on the communication security mechanisms defined by security protocols, above all SSL and TLS protocols, where all other mentioned properties of computer security are considered realized: authentication as security achieved by bringing a user face to face, security from external influence, including also fire walls in addition to cryptography and already mentioned IDS, and in the end the systems for control of the ownership over the files (Stallings, 1998). A simple explantation of protocols is that they are rules and procedures based on them which enable communication. The word „protocol“ is of Greek origin and it means a seal which is put on documents as a proof of their authenticity and today this word is used in different contexts. For example, diplomatic protocol is the set of rules and customs of behaviors in inter-states relations.

In computer environment a protocol is the set of rules and conventions which define communication frame between two or more participants whereby the participants in communication can be users, processes or computer systems. If at least one part of a message is coded, the protocol can be considered cryptographic and it is used to establish the secure communication via unreliable global networks and distributed computer systems and naturally there exist also protocols for effective transfer of data which do not belong to the group of security protocols as are for example the well-known http, ftp.

1.2 Security services and threats

Security protocols should enable the implementation of security services which considers the usage of security mechanisms, i.e. mechanisms which should prevent the attacks on security or recover the system from the attacks. Security mechanisms are technologies which can be implemented in the system and they change with the development of technologies but the first three of the below listed - CIA triad (from the first letters of their English names) stay constant:

• Confidentiality, privacy – international standardization organization, ISO, defined privacy as service which provides access to information only for users who are authorized to access this information. Generally this idea is defined as capability of authors to hide all that does not have to be publicly accessible, i.e. this is the service which provides the information to be accessible only to those users it is designed for. Data must be protected when they are put into storage, during the data processing and during the transfer.

• Integrity – a service which provides totality of data, i.e. provides that the attacker cannot change the data without being observed. Consequently, integrity is security service from unauthorized, unpredicted or unintentional modification. Data must be protected when they are put into storage, during the data processing and during the transfer.

• Availability – a service that provides accessibility of data and availability of system which provides service. Examples of such service are protections against infection with the viruses which erase or damage files and avert execution of services, i.e. programs.

• Authentication – a service that demands from each user to be presented to the system before he does something and which also provides that everyone who claims to have a certain identity (for example user name), must also prove it.

• Non-repudiation – a service that provides that the user who sends a message or changes some data cannot claim later that he has not done it. For example, the user who has signed a document digitally with his private key cannot claim later that he has not made and has not signed this document because this signature can be easily checked.

• Access control – a service that prevents misuse of resources. With access control it is permitted to the user with the verified identity and suitable authorities to use some services or operations of system which are defined in the so-called matrices of access.

In order to achieve security services the following mechanisms and their combinations can be used:

- Coding;
- Digital signature;
- Mechanisms for access control;
- Mechanisms for control of data integrity (integrity of the field of information and of the flow of information) are used for time stamp, cryptographical connection;
- Mechanisms of authentication (password, smart card, biometrical devices);
- Mechanisms for traffic supplement;
- Mechanisms for direction of routing (static, dynamic), and
- Mechanisms for registering (they are usually based on digital signature).

Security protocols provide communication secure from possible threats which are manifested as active or passive and which are given in four categories:

• Interruption represents attack on availability. With one interruption the flow of information is disconnected, i.e. it is impossible to provide some service or functioning of some system. This attack belongs to the group of active attacks.

• Interception represents an attack on confidentiality. Interception in practice can be carried out as traffic eavesdropping. As a passive attack, it can hardly be discovered because it does not change data, i.e. it does not affect the functioning of the system. It is often a preliminary phase for some other type of attack.

• Modification represents an attack on the integrity. This is an active attack. If it happens on the communications path, it can be demonstrated, for example, as a man-in-the-middle attack. An attack can also happen inside some computer system and in this case there is a change of data, access rights, and the way of program or system functioning or something similar.

• Fabrication represents an attack on the authentication. This active attack is performed by an attacker generating false data, false traffic or issuing unauthorized commands. There is often a false presentation of user, service, server, web site or some other part of system.

2. Security protocols on different TCP/IP layers

OSI (Open Systems Interconnection Basic Reference Model) reference model for connection is the most used abstract description of architecture of

computer network thereby dividing it into seven logical levels from the lowest physical level, data level, network level, transport level, session level, presentation level, to application level which are grouped in two bigger groups – the first four make Transport set and they define how the information is transmitted from some location to the other and the last three make Application set and they describe the process of the application intercommunication, user's work with application and interaction user - computer.
Many protocols on the set of TCP/IP protocol can be found on the Internet, some are given in Table1.

Table 1 - OSI reference model

Level of OSI model	Unit	Protocols
Application: Network processes connected with application	Data	HTTP, FTP, Telnet, DNS, POP/SMTP
Presentation: Encryption and coding of data	Data	SSL,TLS.
Session: Establishing of session of ultimate users	Data	NetBIOS, SSH
Transport: Connection, confidence, transport	Segment Datagram	TCP, UDP
Network: Logical addressing and routing	Package	IP,IPsec, ICMP, ARP,
Connection level: Physical addressing,medium access	Frame	PPP, PPTP
Physical level: Transmission of signals	Bit	RS 232,RS422, STP

The choice of place in the stack of TCP/IP protocols where security will be implemented depends on the security and other application requirements. It is possible to provide all or only some of the stated services depending on the place in stack where the safety is implemented. It is also possible that some services are provided on one level and other services on other levels.

Application level - Protocols which provide safety and function on application level must be implemented in final points of communication, i.e. on

final computers. Advantage of this way of safety implementation is that the application can be expanded without the support of security services which the operating system provides. The second advantage is the complete access to data which user wants to protect. This advantage makes reservation of security services easier (for example non-repudiation), it also provides easy access of user authentication. Bad side is that these security mechanisms must be projected for each application and that has as consequence that existing application must be expanded. Because different applications have different needs, the consequence is in the design of many different systems the bigger probability of errors, and thus the greater possibility of security failure.

Presentation level is the sixth level of OSI model. It makes possible the work of entities of Application set, i.e. the entities of higher levels can use different syntax and semantics. Units of data are encapsulated in SPDU (Session Protocol Data Units) blocks and are sent to lower level. This level enables independence during the presentation of data thanks to the translating from application in network form and the other way round. Presentation level transforms data in the form which is accepted by the application level. Applications which work at this level form and encode data so that they can be sent through non secure networks, giving them independence from problem of coordination. Authentic structure uses the rules of coding ANS.1 (Abstract Syntax Notation One) from the set of cryptographic rules.

Session level – As the fifth level of OSI model, it controls the connection between users. It establishes, directs and defines the connection between local and distant applications. It supplies two-direction (full-duplex) and one-direction (half-duplex) operations, establishment of checkpoint and delaying and repeated start of procedures. This level is responsible for closing and repeating of session. It is usually implemented explicitly in the application environment using RPC (Remote Procedure Call) calls.

Transport level – Providing security at this level has advantages over providing security on application level because it is not necessary to expand each application. All existing applications receive equal degree of security which depends on security mechanisms implemented on transport level and it is obtained, as in the case of application level, at the end computers. And this type of security implementation characterizes dependency on protocols and, for example, TLS protocol provides security services of checking identity, integrity and confidentiality over TCP protocol. Since security services depend on transport protocol, the services such as key directing must be duplicated for each transport protocol. The fault of this level also is that applications must be changed so that they can require secure services from transport level.

Network level – The implementation of security on this level has many advantages. For example, surpassing caused by a change of keys is significantly reduced since all transport protocols and applications now divide infrastructure

key directing which is now provided by the network level. It is also important that if security is provided from lower levels, it is necessary to have lower changes of application. One of the most useful possibilities which protocols from network level offer is the capability of virtual private Network (VPN) and Intranet building. Problems in security protocols using on network level are difficult supplying of no repudiation service and difficult realization of control on user level at the multi-user computer; these problems must be solved with the introduction of additional mechanisms at the end computers. For example, IPsec obtains security on network level and is only of protocols which provides happening of all types of traffics.

Connection level – If the intended connection between two computers or routers exists and if all traffic between them must be coded, so that all attacks of types of catching or changing of data are denied, in that case it is possible to use a device for coding. An advantage of this solution is speed. The fault of such a solution is that this is useful only in intended connections, i.e. if the sites which communicate are in physical connection. This method is used, for example in bank automats.

Physical level is the lowest level of OSI model. It defines electronic and physical specifications of devices, i.e. it determines a connection between devices and physical medium. At this level voltage levels are defined, as well as the number of pins, i.e. the number of pairs in cables or coaxial cable if it is a transferable medium. For example, devices of networks card, hubs and repeaters are such. Basic functions of physical level are connection and disconnection with communication medium.

3. Secure Sockets Layer (SSL) Protocol

SSL protocol provides mechanisms for both the identification of two participants connected by computer network and secure transmission between them. SSL protocol practically provides the transmission of unsecured data over secure communication channel and fulfills the following aims:

- Cryptographical protection which implies providing of mechanisms for coding of data, i.e. for realization of the secure connection between two participants in communication.
- Independence from software and hardware which enables to programmers to write software in which SSL is implemented so that two different programs – for example, Web server and reader of the Web can exchange parameters of coding and within that do not recognize code of the other one.
- Expandability implies making of frames within which, if necessary, it is possible to embed new symmetrical algorithms and algorithms with public

key, by which the need to design new protocols is avoided.

- Efficiency influences that coding as operation uses computer processor less independent from complexity of algorithm, which is especially expressed in the case of algorithms with public keys. SSL memorizes the communication parameters of the established connections in order to reduce the number of connections which it must reestablish and in that way provides smaller load to both the processor and the network.
- The task of Secure Sockets Layer (SSL) protocol is to accomplish the secured data transfer trough the network. SSL provides mechanisms for the identification of server and client as well as the coded data exchange between them, which makes the complete system of secured communication between two network entities. The protection of communication which makes protocol SSL has three basic characteristics:

Privacy, therefore the exchanged data are coded with symmetrical algorithms for coding (DES, RCA).

- Possibility to check the identity of a client and server with public key, for this possibility SSL is using RSA and DSS algorithms.
- Reliability, therefore SSL is using SHA and MD5 hash functions to check the integrity of the received messages.
- SSL protocol forms special communication level placed over the transport level (Figure 1).

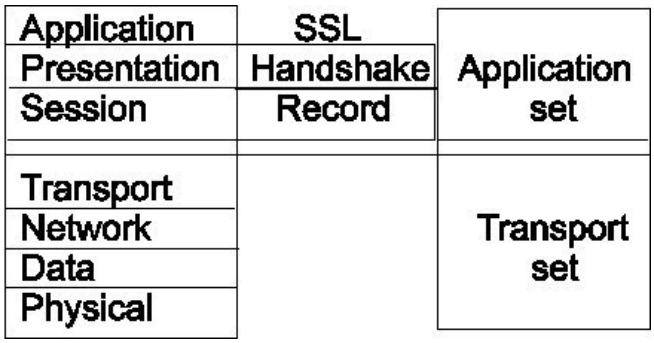

Figure 1 - SSL in the set of protocols

Application level is placed over SSL. At the side of a sender, the SSL receives message from application level which it divides into the parts suitable for coding and adds the control number to them; then it codes and possibly compresses this parts of the message. In this way the sender sends coded parts of message. The receiver receives this parts that he possibly decompresses, decodes, checks control numbers, composes the parts of messages and gives them to application level. SSL is transparent and independent from application level and

establishes security communication before application level receives or sends first byte of data. Also before the beginning of sending the coded data through network, the SSL client identifies server with which he communicates. The SSL is practically composed of two protocols (Pleskonjić, Maček, Đorđević & Carić, 2007):

- SSL Handshake provides reciprocal identification and exchange of parameters for transfer to client and server, i.e. the choice of algorithms and keys.
- SSL Record is responsible for coding and transfer of messages.

The SSL reqires at least identification of server in order to establish a secured transfer. SSL makes this during the handshake stage, sending the certificate to a client. The SSL uses public key and digital signature of server for identification. After server identification, client and server exchange mutual messages which are coded with symmetrical algorithm. The client identification is identical to server identification. After the process of identification, the exchange of data can start. The communication between the server and the client with certificate publisher is not the part of the SSL.

The SSL can establish a session between a client and a server without the identification of either the server or the client, but this means that the security level of data transfer is very low because the data are protected only with symmetrical coding with key which is in unsecured communication agreed between a server and a client.

3.1 SSL handshake protocol

The SSL handshake protocol which works over the SSL level of record makes attributes which describe a session. The handshake protocol delivers messages to the SSL record protocol, which codes and sends them, in the same way as all the others. Before the phase of session establishing, the attributes of communication are not defined and therefore the first messages are sent unsecured. When the SSL client and server start to communicate, they make the agreement about the version of protocol, choice of algorithm for symmetrical coding. Optionally, they make identification and use algorithm of public key to generate shared secret (the key for symmetrical coding). This completely described process happens in the SSL handshake protocol.

As the first client sends salutation message to server – Client hello (Figure 2) on which the server must answer with its salutation - Server hello.

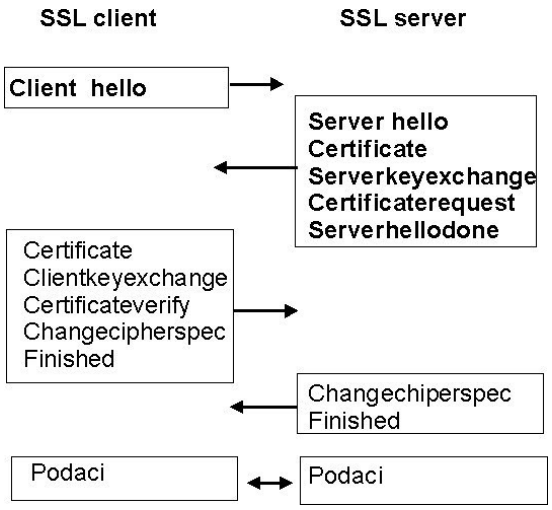

Figure 2 – The SSL handshake protocol

And if that procedure does not happen, communication is stopped. Salutation messages of client and server are used to establish the next attributes of session: version of protocols, session identification, algorithm of coding, algorithm of compression and unexpected values which set client and server. In his salutation, the client delivers the list of possible manners of coding and compression (beginning from the best for him) to server. The server chooses the best combination which it can accept from that list. After the salutation message, the server sends its certificate because the server must identify itself. If the server is positively identified, it can demand the certificate from a client if that is in harmony with the agreed algorithm of coding. After that the server sends a message to the client about the end of salutation – Server hello end. If server has demanded the certificate from the client, it expects the answer which contains certification confirmation or report that the client does not have a certificate.

Then the client sends new attributes with which it will send coded messages and this attributes set as active. After that client sends report about the end of send, coded with active attributes – End. As the answer, the server sends its attributes, and after that reports about the end of sending, coded with a new attributes.

Therefore, the phase of session establishing is ended and both the client and the server can start with the exchange of data from the application level. During the session establishing the order of messages must be strongly respected. Otherwise, the error is reported caused by an unexpected message which stops the session establishing.

3.2 Attributes of the SSL session and connection

When the SSL handshake protocol identifies the server and/or the client, and agrees on the manners of coding, then the session is established. Often, the client and the server want to establish more sessions parallely, for example, the transfer of files and reading of contents of a web-site. Therefore, it is possible to establish more connections inside one session. The session is described with attributes about which the client and the server make agreement durign the stage of handshake protocol. These attributes are basic for the establishing of each new connection. The SSL allows more connections inside one session as well as parallel executing of more sessions between the same client and server. Each SSL session is described with the following attributes:

- Secret. Before transfer of data the client and the server exchange mutual secret. This secret, i.e. sequence from 48 bits, is used to generate symmetrical keys and calculation of MAC (Message Authentication Code) values.
- Widening. Notation which shows if it is possible to establish a new connection inside the given session.
- Session identificator. The sequence of bytes which is agreed by the client and server and which unitedly identifies this session.
- Entity confirmation. The client and the server fill in this attribute during the identification process, which is otherwise empty (NULL value).
- Method of compression. Algorithm of data compression before coding (NULL value without compression).
- Coding. Two algorithms are cited: one for symmetrical coding (for example, DES), NULL value means that data are not coded and second - hash function, concrete algorithm MAC, (for example, MD5, SHA). Also other pieces of information needed for coding are defined with algorithms such as the length of control number, if both the server and the client will identify themselves, or only the server or no one.

The attributes of the SSL connection are random variables of the client and the server. The attributes are used for coding and they must be different. Server's and client's MAC secret is used for the identification of messages which server sends, i.e. client and symmetrical key of client and server (to whom server codes and client decodes and opposite), and ordinal number of messages. Both the client and the server must consider ordinal number of messages which were sent and received for each connection. If the manner of coding is changed during the connection, ordinal numbers will be set to zero. These attributes are known to the client and the server and each of them saves a copy of their values.

The task of the SSL handshake protocol is to coordinate, i.e. to equalize their values. The SSL permits the change of session attributes and the connection

while they last, and by that reaching the higher level of protection. How the process of attributes changes should not affect the communication process, both the client and the server will have to save two copies: active attributes and new attributes. The attributes for sending and for receiving of messages are saved separately. When either the client or the server receive new attributes for decoding, new attributes become active attribute which are used for decoding from that moment on. They cannot be written as active immediately because the message which contains new attributes is not coded with active attributes yet. It is same when the client or the server change a manner of coding for sending.

3.3 Resume of the SSL session

The client and the server can continue connection if they have already communicated with the SSL protocol and with this they skip the identity check and make agreement only about the necessary new attributes.

The session between the client and the server happens in the following order: first, the client sends salutation message using the session identifier which it wants to begin again.

The server looks over the list of his sessions and checks if this identifier exists. If the server finds the identifier, it answers with his salutation which contains exactly this identifier. The server and the client further exchange new attributes of coding, and then send the message for the end of the restored session. After that the data from application level can be sent over the SSL record protocol. If the server does not find the session identifier in its list, it answers with a new identifier, and the server and the client again pass trough the complete process of session establishing.

3.4 The SSL record protocol

The SSL record protocol receives the data from a higher level in the arbitrary size blocks, it does not interpret them but it separates them into parts of suitable size. The SSL record protocol then protects the data cryptographically and sends them to the conversationalist, where the reverse process takes place. At the side of the sender the following processes are happening:

- Before processing continues, the received data are separated into the blocks of fixed length without call attention on the length of client messages and by that more messages can be merged into one or one message divided into more fragments.
- All fragments of the SSL record protocol are compressed with

algorithm, which is defined by the attributes of the session. During the compression the algorithm must not lose data.

- Messages are protected by symmetrical algorithm for coding and thus enable privacy and by the MAC algorithm, which provides for the message integrity, and these are determined by the session attributes.
- After coding of compressed fragment and adding of MAC value, the result is ready for sending like the other data which are necessary for the transfer of message (for example, header), but they are not specific for the SSL protocol. The receiver decodes the accepted fragment, calculates MAC value and compares with value which the sender has generated. If both MAC values are equal, the message is accepted. Otherwise the error report is returned.

3.5 The application of the SSL

The SSL is often used for paying of goods with credit cards, when only the transfer of credit card number is protected. For that and many other cases of secured transfers, such as authenticated access to web site, distant access, exchange of electronic mail, the SSL is proper solution. It is because the SSL contains all available security methods which can be used in the case of communication channels establishing over the network. The SSL provides check of credibility with the help of certificates, using different keys for individual sessions and of the end coding and checking of integrity. If the client and the server are not active for longer period of the session, if they have equal attributes for longer period, attributes are changed.

Basic defect of the SSL protocol is the increased work of processor, which is the basic limit of its implementation. This is the consequence because the functions such as crypting and especially the operation of distribution of public keys demand the additional work of processor.

The additional work of administrator is also defect of the SSL protocol. This defect is the consequence of complicated enviroment which demands maintaince so that administrators must configure system and manage with certificates.

The size of package by the SSL protocol certainly is one of defects because the defined pieces of information are added in packages which are exchanged through network. In this way the size of package is increased and the consequence of this increasing is increasing time necessary for processing also increasing the time necessary for transfer of data and finally the late data transfer.

One of the defects of the SSL protocol is that it demands from programmer of application software a good knowledge of operating system for which he writes

this software. Namely, if the operating system directly accesses to TCP/IP protocol, it needs to be directed to work with the SSL protocol.

For the successful work of the SSL protocol the donors of certificates are important (Thawte, VeriSign, …).

4. Transport Layer Security (TLS)

Version 2.0 of the SSL developed in 1995 has contained many defects and because of that the version 3.0 was developed and published 1996. This version is used later as a basis for further development of the TLS protocol version 1.0 as IETF (Internet Engineering Task Force) standard protocol which is defined in RFC (Request for Comments) 2246 recommendation in 1999.

The TLS, like the SSL protocol, according to Table 1, works on the levels under application protocols such as HTTP, FTP, SMTP, NNTP and XMPP, but also over the reliable protocols of transport level such as TCP protocol. Therefore, the TLS protocol can supplement the security of any other protocol which uses the reliable connections and the TLS protocol is often used:

- In combination with the HTTP protocol and so receives the HTTPS protocol which is used for the security of Web sites on which the applications for electronic commerce are placed;
- In combination with the FTP protocol and so receives the FTPS protocol which is used in two modes of work: explicit – for the secure transfer of data exclusively on demand of the client and implicit – when the server without negotiation on demand of the clent enables the secure connection for the client;
- As the so-called STRATTLS, this gives the manner of annex of unsafely connection to safely connection, instead of using special connections for cryptographic communication.

The TLS protocol enables to build a tunnel through the Internet and therefore creates VPN (Virtual Private Network) network. This brings some advantages in security barrier (Fire-wall) and to NAT (network address translation) components (possibilities of coding of all data which are transferred through tunnel).

The TLS protocol is used more and more as the standard method for security of application signalization SIP (Session Initiation Protocol). It can be used for authentication and coding of SIP signalization connected for VoIP (Voice over Internet Protocol) and other applications based on the SIP protocol.

4.1 Protocols and attributes of the TLS session and connection

The TLS, like the SSL practicaly consists of the two protocols:

– The TLS Handshake - provides to client and server reciprocal

identification and exchange of parameters for transfer, i.e. choice of algorithm and keys.

– The TLS Record - is in charge for coding and transfer of messages.

For establishing the protected transfer, the TLS demands at least server identification. This is performed in the phase of session establishing (handshake) in all according to Figure 2, and by that server sends its certificate to the client. The public key and digital signature of server are used for identification. After the server identification, the client and the server mutually exchange messages coded with symmetrical algorithm using record protocol. The identification of the client is identical. After the identification of both of them, they can start exchanging data. Practically the TLS includes three basic phases:

1. Equalizing mediation for algorithm support
2. Exchange of key and authenticity
3. Symmetrical coding of encryption and establishing of messages authenticity

In first phase, the client and the server negotiate and define which coding they will use, exchange key and establishing of credibility of algorithms also authentic codes of messages MAC (Message Authentication Code). The key of exchange and establishing of credibility of algorithm are typical public keys of algorithms, or in the TLS-PSK algorithm the pre-set keys which can be used in common. The codes for authenticy of messages are composed from cryptographic collection of functions using HMAC construction. Typical algorithms can be:

–For key of exchange: RSA, Diffie-Hellman, DSA, SRP, PSK

–For symmetrical coding: RC4, Triple DES, AES or Camellia

–For criptography collection of functions: HMAC-MD5 or HMAC-SHA.

The session, which is established after the defined way of coding, is described by the attributes about which the client and the server negotiate in the stage of session establishing. These attributes are basic for the establishing of each new connection. The TLS permits more connections within one session, but also the parallel executing of more sessions between same client and the server. Each TLS session describes the following attributes which are described in detail in Section 3.2 – The Attributes of the SSL session and connection, and because of that they will only be listed here:

- Secret. (The client and the server exchange mutual secret before the transfer of data. This secret is used to generate symmetrical keys and extraction of HMAC values);

- Widening. (Notation which shows if it is possible to establish a new connection within the given session);

- Session identifier (The sequence of bytes which is agreed between the client and the server and which identifies this session);

- Entity confirmation. (The client and the server fill this attribute during the identification process, otherwise it is empty (NULL value)),
- Method of compression. (Tje algorithm of data compression before coding (NULL value without compression)),
- Coding. (Two algorithms are cited: one for symmetrical coding (for example, DES), NULL value means that data are not coded and the second - hash function, concrete algorithm HMAC (MD5 or SHA)).

4.2 The application of the TLS

The basic application of the TLS protocol is to make a safe system during the viewing of a web-site and information in HTTPS communication. The protocol can be used for many other purposes. Some examples of the TLS protocol applications are:

- The safe transfer of data for the needs of e-commerce – the protocol is applied between the client and the server. The best example is the use of credit cards for payment of products and services through the Internet. The TLS must have a possibility to be presented on web-site where circulation of data is.
- The authenticated access to web-site – in order for the authentication to be achieved, the user and the server need certificates from the CA entity. Certificates can be copied on user accounts on the two basis:
- One on one – it is used when the server has a copy of user certificate. During each registration the server checks the identity of a user. It is usually applied for handshake of private data like banking services through the Internet.
- More on one – it is used when someone wants to give access to secure materials to some group of users. Then the group is created and it is joined a defined certificate with permissions.
- Distant access – enables using of resources and services on distant computers and during that the TLS protocol can be used for the authentication and protection of data (by the user registration). Thus the users can access the e-mail messages or applications with decreasing of risk of disclosing information to other users of the Internet services.
- SQL access – Microsoft SQL Server, or a suitable operating system, makes possible for a user to ask for client authentication while connecting to the server where the SQL server is started up and it is possible to define requests for encrypting of data that are exchanged.
- Electronic mail messages – using Exchange servers. It is possible to use the TLS protocol for the security of data which are transmitted among servers or networks where it is necessary to use S/MIME (Secure/Multipurpose

Internet Mail Extensions) protocol for the ensurance of total protection of message transmission.

When the TLS protocol is enabled on the server for electronic mail message exchange of the one that sends and the one that receives, the information exchanged among them are encrypted. Those servers use the SMTP protocol for sending and receiving of messages.

There are also other uses of the TLS protocol in almost every application thanks to the possibility of the access to the protocol via the SSPI (Security Service Provider Interface) system. The primary defects and limits of the TLS protocol are the same as in the SSL:

- The programmer of the applicable software must have good knowledge of the operating system for which the software is written, therefore if the operating system directly accesses TCP protocol it should be rerouted to do it through the SSL protocol.
- The increased work of processor, which is the main limit in the implementation of the TLS protocol, because the functions as encrypting, and especially the operations connected with the distribution of the public key, demand the additional work of the processor and it is not possible to exactly define decreasing of the performance of the system which fluctuates from the frequency of the network setup and in its duration.

The greatest number of resources is spent during connecting.

- Additional work of the administrator – the TLS environment is quite complicated and it requires maintenance, so the administrators must configure the system and supervise the certificates.
- The quantity of the package – As the TLS adds certain information to the packages which are exchanged via network, the size of the package increases and the outcome of the increasing is the increasing of the time needed for the processing and the transmission of data, which results in data delay.

4.3 Similarities and differences between the TLS and the SSL

As already mentioned, the TLS derives from the SSL version 2, so it is very similar to it, and the fundamental difference is that in the TLS protocol KHMAC (keyed-Hashing for Message Authentication Code) algorithm substitutes MAC (Message Authentication Code) algorithm which is used in the SSL protocol. KHMAC provides more security than MAC algorithm. In addition to this, it creates the integrity check value, as MAC algorithm, but with using of hash functions which makes it more complex and difficult for the attacker. It is not always necessary to set up certificates from the CA root entities in TLS protocol, instead it is sufficient to use the middle CA entities.

The TLS protocol defines the values for the block increasing (padding block values) which are used in the blocks of the algorithm coding. In addition, in the specification of TLS protocol many new messages for the report warning are added.

Namely, for the insurance of the correct flow of the session, both the TLS and the SSL protocols use reports as a special kind of messages. They are also compressed and coded, and instead of the data from the higher level they consist of the type of report and description. There are two types of the report: report on the end of connection and report on the error.

Before the end of connection, both the client and the server must agree about its end, and they do it with the help of reporting the end of the connection, where the end can be initiated by any participant. That message helps the recipient to understand that the sender will not send messages inside that connection anymore. If the recipient receives messages after the report of the end, he will ignore them. Every participant is obliged to send a warning about the end of sending, so that he can continue with receiving of the messages until he receives report on the end of sending from the other participant. The obligation of the other participant is to close connection declaring its attributes invalid. After closing the connection, the client and the server must erase its attribute values. If one of the participants finds out the mistake in the communication, he informs the other participant using error report, where if it is an error about mistake which endangeres the transmission security, both participants terminate connection. The communication via other connections in the session can be continued, but it is necessary to change the identifier of the session, so that the future usage of the same identifier is prevented.

In the TLS and the SSL protocols the following errors are possible:

- An unexpected message. The error causes the end of connection (the suspicion of the data fabrication type of the attack)
- Malfunctioned MAC value. The error causes the end of connection (the suspicion of the exchange of data attack)
- The error upon an occasion of decompression. The input parameter of the decompressed algorithm does not have the expected result.
- The error in the phase of the session establishing. It shows that the sender is not capable to adjust to the sugested attributes of protection.This error makes the session end.
- Certificates errors. No certificates (It appears if there is a request for a certificate and an answer is negative), Unsuitable certificate (The protocol does not support a concrete type of certificate), Invalid certificate (The certificate validity has expired or a certificate has not become valid), Cancelled certificate (The owner has cancelled the certificate), Bad certificate (A certificate is inconsistent, the existing signature does not confirm identity, etc.) and

Unacceptable certificate (if during the certificate processing something unexpected appears, a certificate is declared unacceptable).

- Invalid parameter. Some attribute values are out of permitted parameters. This error makes the connection end.

4.4. The TLS and the SSL implementations

In this paper the three most known TLS/SSL implementations are considered: OpenSSL,

GnuTLS and NSS, furthermore JSSE (Java Secure Socket Extension) programming package is mentioned.

JSSE consists of a group of programs (API tools, algorithm implementation etc.) that enable safe communication via network implementing Java version SSL and TLS protocols.

JSSE also includes the following functionalities:

• Data encrypting
• Authenticity of the server and authenticity of a client (optionally),
• Message integrity,
• Cryptography and
• PKI (Public Key Infrastructure)

The package was an optional addition to Java program versions 1.3. (While implemented in version 1.4)

4.4.1. OpenSSL

OpenSSL is a free cryptographic tool which implements security protocols version 2 and 3 and TLS version l, and other cryptographic standards which are connected with these protocols (e.g. 3DES, AES and RSA). The program is accessible for almost all UNIX (Solaris, Linux) and Mac OS X and for BSD operating systems with open code, as for OpenVMS and Microsoft Windows operating system.

OpenSSL enables various cryptographic functions implemented in OpenSSL folders to be requested. OpenSSL enables: defining parameters for RSA and DSA keys, creating X.509 digital certificates, CRL lists and requests for signing of certificates, calculating of hash messages, coding and decoding, SSL/TLS communication support and operating with e-mail messages which are signed or coded in accordance with S - MIME (Secure Multipurpose Internet Mail Extensions) standard.

OpenSSL has a great number of instructions for managing certificate center, managing lists of taken digital certificates, calculating of hash, generating pseudo-sonic parameters and managing certificates. Package also has pseudo-

instructions list (standard–commands), 1ist of message (digest-commands) and 1ist of cipher-commands (gives all standard instructions), for hash calculation and cryptographic instructions.

4.4.2. GNU Transport Layer Security Library (GnuTLS)

GnuTLS program is free implementation of the SSL and TLS protocols, whose purpose is to enable API (Application Programming Interface) support to applications so the safe communication is enabled. It is edited under GNU LGPL (Lesser General Public License) and some parts under GNU GPL (GNU General Public License) license.

The above mentioned licenses provide free copying and distribution of tools. The main difference is that LGPL license enables connection with free programs which are not under the same license, also some additional rights of program changes.

In the beginning it is developed for GNU projects, and it is used in programs such as GNOME, CenterIM, Exim, Mutt, Slrn, Lynx and CUPS.

GnuTL is possible for majority of UNIX operating systems also for Microsoft Windows, and it could be downloaded from the site: http://www.gnu.org/software/gnutls/download.html.

Revision 1.04 CCERT-PUBDOC-2009-03-257 pages 22/29 GnuTLS includes the following characteristics:

• SSL protocol support for version 3.0 and TLS protocol version 1.0 and 1.1;

• PSK (pre-shared key) algorithm support during authenticity;

• Mechanism of TLS protocols enlarging;

• Support for strong encrypting algorithms (SHA-256/384/512 and Camellia);

• Compression, and

• Handling X.509 and OpenPGP certificates.

• It supports many algorithms for key exchange: Anon-RSA, RSA, DHE-RSA, DHE-DSS, SRP-DSS, SRP-RSA, SRP, PSK, DHE-PSK;

• It supports Cryptographic algorithms: AES-256, AES-128, 3DES, DES, RC4-128, C4-40, Camellia.

4.4.3. Network Security Services (NSS)

The NSS program is a group of libraries that serve to the SSL and S/MIME protocols. It is developed by the Netscape organization, and it is used by AOL, Red Hat, Sun Microsystems operating systems in various applications (Mozilla Firefox, Thunderbird and SeaMonkey, AOL Instant Messenger, Evolution, Pidgin, OpenOffice.org 2.0., Red Hat Directory Server etc.). It is

licensed with three licenses: „Mozilla Public License“, „GNU General Public License“ and „GNU Lesser General Public License“. The actual version wasedited in 2008, and it is the version 3.12.

The program supports various security standards:

• SSL protocol versions 2.0 i 3.0;

• TLS protocol versions 1.0;

• PKCS standards;

• PKCS #1- #12.RSA standards that define the implementation of cryptography beside RSA algorithm;

• CMS (Cryptographic Message Syntax) used in S/MIME protocol;

• X.509 certificates;

• OCSP (Online Certificate Status Protocol) certificates;

• PKIX certificates;

• Algorithms: RSA, DSA, ECDSA, Diffie-Hellman, EC Diffie-Hellman, AES, Triple DES, DES, RC2, RC4, SHA-1, SHA-256, SHA-384, SHA-512, MD2, MD5, HMAC, and

• FIPS generator pseudo –random numbers.

4.4.4 Comparing TLS and SSL implementations

In this part of the paper the three already described tools are compared:

- OpenSSL, GnuTLS and NSS from the aspect of different versions of the TLS and SSL support;

- The support of different algorithms for public key exchange;

- Cryptographic algorithms and different compression procedure support.

1. Table 2 shows the comparison of support for various versions of the SSL and TLS protocols (The Croatian Academic and Research Network (CARnet), 2009). GnuTLS tool consists of the support for every mentioned versions of protocol, and other tools support only some versions.

Table 2 - Tools support for different versions of the SSL and TLS

	SSLv2.0[1]	SSLv3.0	TLSv1.1	TLSv1.2
GnuTLS	Yes	Yes	Yes	Yes
OpenSSL	No	Yes	No	No
NSS	Yes	Yes	No	No

2. In Table 3 there is a description for key exchange. The majority of tools are supported by GnuTLS tool.

Table 3 - Tools support for various algorithms for key exchange of the SSL and TLS

	Anon-RSA	RSA	DHE-RSA	DHE-DSS	SRP-DSS	SRP-RSA	SRP	PSK	DHE-PSK
GnuTLS	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
OpenSSL	Yes	Yes	Yes	Yes	No	No	No	No	No
NSS	Yes	Yes	Yes	Yes	No	No	No	No	No

3. In Table 4 there is a description of cryptographic algorithms that are used in implementations. The majority of algorithms are supported by GnuTLS tool.

Table 4 - Support of tools for cryptographic algorithms for the implementation of the SSL and TLS

	AES-256	AES-128	3DES	DES	RC4-128	RC4-40[1]	Camellia
GnuTLS	Yes	Yes	Yes	Yes	Yes	Yes	Yes
OpenSSL	Yes	Yes	Yes	Yes	No	No	Yes
NSS	Yes	Yes	Yes	Yes	No	No	Yes

4. Support of the compression procedures can be found in Table 5 where there is the evidence that GnuTLS supports both of the mentioned procedures, OpenSSl only ZLIB procedure and NSS none.

Table 5 - Support of tools for various compression procedures in SSL and TLS implementations

	ZLIB	LZO[1]
GnuTLS	Yes	Yes
OpenSSL	Yes	No
NSS	No	No

From the above mentioned data given in Tables, it is clear that GnuTLS tool has the greatest possibilities.

5. Security protocols competitive to the TLS and SSL protocols

In application group of OSI model of protocols some of competitive solutions which use architecture and principles similar to those that exist in the SSL and TLS protocols are: S/MIME (Secure-MIME), SSH (Secure Shell), PCT (Private Communication Technology) and OpenPGP.

In transport group of OSI model of protocols there is a well-known IPsec (Internet Protocol Security) protocol which is competitive to the TLS and SSL.

- Protocol S-MIME is made by RSA as a supplement to existing protocol MIME. It uses a system of public keys as a base for the check of integrity and coding.
- SSH is used for connecting to distant computers by means of protected channel which is provided by SSH. The user is identified thanks to the

password that is coded before sending through the network.

- OpenPGP is a protocol for coding of electronic mail by means of cryptography with public keys, based on the original PGP distribution of Phillip Zimmermann. Protocol OpenPGP defines standard forms of coded messages, signatures and certificates for the exchange of public keys. At the moment OpenPGP is the leading standard in cryptography with public keys.
- PCT is a product of Microsoft Company that is made as a reaction to the errors made in version 2 of the SSL protocol. Although Microsoft solved this problems, PCT is practically not used after the appearance of version 3 of the SSL protocol, and that is why we will not consider it in this paper (Stinson, 1995), (Schneider, B., 1996).

5.1 S/MIME protocol

Today, one of the protocols that are probably most widely used in the applicative level is S/MIME (Secure Multipurpose Internet Mail Extension). S/MIME applications are installed in software packages that are today the most dominant on the market, for example Netscape Communicator, Microsoft Outlook, Mozila Firefox, etc.

S/MIME is based on popular internet MIME (*Multipurpose Internet Mail Extension*) standard and it enables following cryptographic services that have to do with security of applications such as electronic exchange of messages: authentication, message integrity and certainty (using digital signature), and data secrecy (digital envelope).

S/MIME can be used by traditional *Mail User Agents*, MUA, so the cryptographic security services can be added to the sent mail and to interpret cryptographic security services in the received mail.

S/MIME is not made only for electronic mail; it can be used with any transporting mechanism which transmits MIME data, as it is HTTP (*HyperText Transfer Protocol*). Beside that, S/MIME can be applied in the agents of automatic transmission of messages which use cryptographic security services and which do not need any intervention done by man (such as signing of the software generated documents and coding of fax messages that are sent via the Internet).

MIME standard supplies general structure of the Internet messages content and it allows extensions for the applications of new content.

5.2 OpenPGP protocol

OpenPGP is developed from the commercial program version 5.0 PGP and in 1998 RFC2440 document is published under the authority of the Internet Engineering Task Force which totally defines OpenPGP standard and all other

information needed for the development of the applications that are compatible with it. It exchanges data via standardized packages (key, digital signature, etc.) and in its work it uses many cryptographic algorithms. It takes the best characteristics from the world of symmetric and asymmetric cryptographic systems combining them into powerful protocol.

Encryption via OpenPGP starts with generating of the disposable key which is used in encrypting of the message with symmetric algorithm. That is usually randomly generated number. Randomly generating of symmetrical key gives the highest degree of security, because it prevents random „discovery“ of the key. After the message is encrypted with generated symmetrical key, a symmetrical key itself is encrypted with public key of the receiver for the sake of preserving the speed. The message encrypted by symmetrical algorithm and a symmetrical key encrypted with the public key of the receiver are the parts of the final message which can be safely sent via the unsecured chanell to the receiver.

The procedure of decrypting is oposite. After receiving the message encrypted with OpenPGP from the sender, the receiver first of all decodes a symmetrical key with his private key, and after that decodes messages with the received key. If the message was compressed it must be decompressed in order to get the original. The security of the message is guaranteed under the suggestion, if the private key is really private.

The encryption itself can be sufficient for the security of the message, but it cannot prove the correctness of the received message, i.e. if the message which is received is really the message which is sent. Also the legitimacy of the the sender of the message is in question. How can we be sure that the sender of a message is really the person for which he presents himself? Both of these questions are solved by the system of digital signature. What does a digital signature mean? Digital signature is the same as personal signature in the real world, it insures that the message is not falsified. Namely, the actual algorithms in digital world make it very difficult or almost impossible and certainly unprofitable to forge a digital signature. The system of digital signature is realized so that the sender of message makes abstract of message with some algorithm for abstract calculating, then codes this abstract with his private key and such abstract is sent together with or separately from message. With this all demans are effectively insured. Such a coded abstract of message corresponds to suitable received message only if the digital signature or message are not changed.

5.3 Secure Shell (SSH) protocol

SSH is the popular protocol for coding of communication channels, which is mostly used to supply security sessions of distant registration on the system. The architecture of SSH protocol is two-tier client/server architecture. SSH server is the software which accepts or rejects connections which arrive to computer.

SSH client software is installed on distant computers; clients send to SSH server demands of type „please, report me on the system ", „please, send me a file " or „please, execute this command ". In spite of this, SSH codes all data which are transferred through network and the coding of this process is transparent to the user. At this moment two incompatible versions of this protocol are used - version 1 and version 2.

SSH protocol provides security mechanisms of identity check, coding of data and also supplies integrity of data which are sent through the network. The SSH also provides for the keys to be used for registrations to distant computers instead of passwords. The SSH agent for the identity check which works on local computer is used for that purpose. This SSH functionality is especially suitable when users have distant access to many computers whose users accounts are protected with various long passwords.

5.4 Internet Protocol Security (IPsec) protocol

IPsec represents a set of protocols intended for security communication over the Internet. It belongs to Transport of OSI model of protocols which works on its network level. IPsec offers simple and effective protection for the TCP and UDP protocols of communication through computer network. IPsec ensures the realization of the following security demands:

- Confidentiality; only authorized person can access data;
- Integrity; impossible change of data by unauthorized person;
- Authentication; verification of identity of sender;
- Availability; availability of data in spite of the unexpected events.

In the base IPsec consists of two subsets of protocol:

• Cryptographic protocols – ESP protocol (Encapsulating SecurityPayload), AH protocol
(Authentication Header)

• Protocols for key exchange – IKE protocol (Internet Key Exchange)

IPsec is designed to satisfy two basic functions:

• Tunneling of packages, and

• Transport modeof work.

By tunneling few computers (or one local computer network) Ipsec hides them behind one knot and like that they are invisible to the rest of the network (so they are protected from attacks).

In the second case packages are sent between two end computers on the network, and the computer which receives package executes security checks before the delivery of packages to higher levels.

The basic idea of IPsec protocol protection is building of Virtual Private Network – VPN.

5.5 Comparison of security protocols which are competitive to TLS and SSL

The comparison of the four previously described competitive protocols, S/MIME, OpenPGP, SSH and IPsec with the TLS and SSL protocols from the viewpoint of support to different algorithms for coding of data, identification, control numbers, and also their application and implementation is given in Table 6. (Tanenbaum & Woodhull, 1997).

Table 6 - Comparison of protocols S/MIME, OpenPGP, SSH and IPsec with the TLS and SSL

	S/MIME	OpenPGP	SSH	IPsec	SSL/TLS
Coding of data	Triple DES	TripleDES	RC4,Triple DES, AES	DES, Triple DES	DES i RC4/ RC4, Triple DES, AES or Camellia
Identification algorithm	Diffie-Hellman, DSA,RSA	ElGamal, DSA,RSA	Diffie Hellman, DSA,RSA	DSA,RSA	RSA , DSS/ RSA, DSA,SRP, Diffie-Hellman
Algorithm of control numbers (hash)	SHA-1	MD5, SHA-1	MAC-SHA or MD5	SHA-1,MD5	MAC-SHA,MD5/ HMAC-MD5,SHA
Application	e-mail, web access	Data exchange	Data exchange, distant access	Data exchange	Data exchange, web access, distant access
Implementation	GpgSM,S/ MIME	GnuPGGNU PrivacyGuard	OpenSSH	FreeS/WA Npluto	OpenSSL,GnuTTL , NSS,JSSE

6. Conclusion

The TLS and SSL protocols are cryptographic protocols which provide safe communication through the Internet for the jobs of electronic banking and commerce, electronic mailing, access to distant computers and other ways of data transfer. Based on the data presented in Chapter 3, Secure Sockets Layer (SSL) Protocol, and in Chapter 4, Transport Layer Security (TLS), it is undisputable that the insignificant differences between the TLS and SSL exist, because the TLS as a derived solution uses only the safest and the most modern algorithms in the action of communication establishing but these protocols are basically the same.

1. Based on the systematized data given in Chapter 3, Secure Sockets Layer (SSL) Protocol, Chapter 4, Transport Layer Security(TLS), 4.4 TLS and SSL implementations and Chapter 5, Security protocols competitive to TLS and SSl protocols, it can be conclused that the TLS and SSL present standard and the

best solution for safe communication because in relation to competitive protocols from Application set of OSI model of protocols they have better protection, they cover wider area of applications, they have a larger number of implementations and majority of their implementations are program of open code. In the paper we note that the TLS protocol enables building of tunnel through the Internet so that the VPN can be created and which in relation to IPsec protocol, which also enables VPN, is manifested with capability for easier administration of distant access.

2. Based on comparisons given in chapter 4.4.4 Comparing TLS and SSL implementations, the best implementation is GnuTLS because it is open codeed program and also it permits to support the majority of versions of the SSL and TLS protocols, the majority of different algorithms for exchange of public key, the majority cryptographic algorithms and the majority of different actions of compression.

8. References

1.Croatian Academic and Research Network(CARnet),(2009),*TLS protocol* (CCERT-PUBDOC-2009-03-257), Retrieved July 01.2009, from http://www.cert.hr/documents/ CCERT-PUBDOC-2009-03-257.pdf
2.Pleskonjić, D., Maček, N., Đorđević, B., & Carić, M. (2007), *Sigurnost računarskih sistema i mreža*, Beograd, Srbija: Mikro knjiga.
3. Schneider, B. (1996), *Applied Cryptography*, New Jersay, USA: John Wiley & Sons.
4. Stallings, W. (1998), *Cryptography and Network Security*, New Jersay, USA: Prentice Hall.
5. Stinson, D. (1995), *Cryptography - Theory and Practice*, Florida, USA: CRC Press.
6. Tanenbaum, A. S., & Woodhull, A. S. (1997), *Operating System Design and Implementation*, New Jersay,USA: Prentice Hall.

REZIME

Internet, kao računarska mreža, povezuje milione ljudi širom sveta i obezbeduje im pristup velikoj količini informacija. Korisnici preko Interneta razmenjuju podatke na osnovu određenih protokola, a deo te komunikacije je privatnog ili službeno tajnog karaktera. Pri ovoj razmeni, korisnici resursa računarskih sistema, računara u mrežama i samostalnih računara, pre svega žele da budu sigurni da će pristup njihovim podacima i resursima uopšte imati samo oni kojima se pristup dozvoli. Dakle, analogno sigurnosti fizičke imovine korisnici računarskih sistema žele takozvanu računarsku sigurnost.

Jezgro Internet protokola predstavljaju TCP (Transmission Control Protocol) i IP (Internet Protocol) protokoli. Sve što putuje Internetom koristi ove protokole, ali oni ne obezbeđuje sigurnost prenosa podataka. IP paketi se, na primer, mogu lako izmeniti a njihov sadržaj može u bilo kom trenutku da pregleda

ma ko, pa i neovlašćena osoba. U svetu koji je danas već globalno povezan, pojedinci i razne institucije imaju potrebe za privatnošću, kao i za zaštitom od krađe identiteta, koja postaje sve češći vid zloupotrebe globalne mreže. Dakle, potrebna su sredstva koja su transparentna i dovoljno fleksibilna da zadovolje zahteve raznih korisnika, a istovremeno ostvare zadati stepen sigurnosti.

U ovom radu, pažnja je usmerena na komunikacijske zaštitne mehanizme definisane sigurnosnim protokolima, pri čemu se smatra da su ispunjene ostale kategorije računarske sigurnosti. Protokoli TLS (Transport Layer Security) i SSL (Secure Sockets Layers) su kriptografski protokoli koji omogućavaju sigurnu komunikaciju na Internetu za poslove elektronskog bankarstva i trgovine, e-mail, fax, pristup udaljenim računarima, a korisnicima rešavaju dobar deo navedenih problema.

SUMMARY

The Internet, as a computer network, connects millions of people all around the world and gives them a possibility to access a big quantity of data. Throughout the Internet users exchange data based on certain protocols and a part of this communication is of private or secret character. During this exchange, the users of computer systems, computers in network or independent computers, primarily want to be sure that only those who are allowed will have access to their data. Therefore, analogue to the safety of one's property, computer systems users want the so-called computer security.

TCP (Transmission Control Protocol) and IP (Internet Protocol) protocols are the kernel of Internet protocol. Everything that is transmitted through the Internet uses these protocols, but they cannot provide security of data transfer. For example, IP packages can be easily changed and their content can be seen by everybody at any moment, even by an unauthorized persons. Today the world is already globally connected and the individuals and various institutions need privacy, as well as the protection from identity theft, which becomes a very frequent aspect of misuse of the Internet. So, we need transparent and tools sufficiently flexible to fulfill the demands of different users and at the same time capable to achieve the assigned degree of security.

In this paper the attention is focused on the communication security mechanisms defined by security protocols, whereas all other properties of computer security should already be implemented. The TLS (Transport Layer Security) and SSL (Secure Sockets Layers) protocols are cryptographic protocols which provide safe communication over the Internet for the jobs of electronic banking and commerce, electronic mailing, access to distant computers and other ways of data transfer, while solving a good part of the mentioned problems for the users.