INTEGRATING QUANTUM TECHNIQUES INTO SECURE SOCKET LAYER PROTOCOL

Gabriela MOGOŞ

University of Oradea, Oradea, Romania

***Abstract**: The Secure Sockets Layer protocol is a protocol layer which may be placed between a reliable connection-oriented network layer protocol and the application protocol layer. Secure Sockets Layer provides for secure communication between client and server by allowing mutual authentication, the use of digital signatures for integrity, and encryption for privacy. The protocol is designed to support a range of choices for specific algorithms used for cryptography, digests, and signatures. This allows algorithm selection for specific servers to be made based on legal, export or other concerns, and also enables the protocol to take advantage of new algorithms. This work proposes the replacement of classical techniques of client-server authentication by a quantum one, which is not vulnerable to the cybernetic attacks, and which solves the problem of Secure Sockets Layer protocol security*

***Keywords**: Secure Socket Layer protocol, qutrit, quantum cryptography.*

1. INTRODUCTION

In the last years, the Internet is more and more used in business activities. As it is commonly known, both the users' authentication and the access authorization are realized based on username and password. However, there are two weak points concerning the security:

- The data transmitted between the web server and the client's browser are not protected at interception, and it is possible for a person to be able to intercept confidential information, as the passwords or the data about credit cards, bank accounts etc., which circulate between the client's browser and the web server;

- While the web server presents a reasonable security level reported to the client user, the client has no possibility to establish if the web server is the correct one.

The Secure Sockets Layer (SSL) protocol has the intention to assure a private communication channel between the web server and the client's browser, and in the same time to assure the clients that the server to which they are connected is the real one. For this it is used the SSL certificate, which is a digitally signed certificate. The Secure Socket Layer protocol is a client/server protocol that provides the following basic security services to the communicating peers:

- Confidentiality - by the use of an encrypting algorithm;

- Authentication - by the use of digital certificates;

- The control of the integrity (without recovery) - by the use of some algorithms for the integrity of the messages.

Secure Socket Layer works by combining public key cryptography and secret key encryption to ensure data confidentiality. In the classical version, the Rivest-Shamir-Adleman public key algorithm is used to generate the certificates and the public and private key used pairs utilized in Secure Socket Layer. When a client connects to a server that is configured for Secure Socket Layer, a Secure Socket Layer handshake process is initiated with the server. The server at this stage has already obtained a server certificate from a Certificate Authority (CA). A Certificate Authority (CA) can be defined as an entity that generates and validates digital certificates. The Certificate Authority adds its own signature to the public key of the client.

This essentially indicates that the public key can be considered valid, by those parties that trust the Certificate Authority.

This work proposes the replacement of the existent Secure Socket Layer handshake protocol techniques with quantum versions. The quantum versions proposed substitute the authentication procedure, as well as the method of secret key distribution. The quantum versions use tri-dimensional quantum systems - qutrits, which are not vulnerable to the cybernetic attacks, assure the correct authentication, and determine giving up the long row of authentication certifications used in the classical case for removing any suspicions.

2. AN OVERVIEW OF THE SECURE SOCKET LAYER - HANDSHAKE

The Secure Socket Layer protocol itself is made up from two sub protocols: the Secure Socket Layer - Record protocol defines the method employed to transmit data and the Secure Socket Layer - Handshake protocol uses the record protocol to perform a two-way handshake. The Secure Socket Layer - Handshake Protocol is layered on top of the Secure Socket Layer - Record Protocol. It allows a client and server to authenticate each other and to negotiate items like cipher suites and compression methods. Each time a Secure Socket Layer session is initiated an exchange of messages, known as the handshake, must be performed. This handshake allows the server to authenticate itself to the client and optionally allows the client to authenticate itself to the server.

In the classical case, after authentication the client and server cooperate to generate symmetrical session keys which will be used for encryption/decryption and tamper detection throughout the session. The handshake may also be initiated at any time during a given session to re authenticate the two hosts and generate new cryptographic settings. The handshake uses public key encryption to communicate securely.

The most important part of the handshake is the authentication of the server. If this is corrupt all further generation of session keys will be corrupt and the entire Secure Socket Layer session will be insecure. The server is authenticated via a digital certificate which it sends to the client and the clients will proceed then to validate the identity of the host that the certificate claims to represent. A digital certificate contains information such as the certificate version, serial number, signature, issuer, and validity period, among other information.

3. QUANTUM VERSION OF THE SECURE SOCKET LAYER - HANDSHAKE

The Secure Socket Layer - handshake process occurs between a client and a server to negotiate the secret key encryption algorithm which the client and the server will utilize to encrypt the data which is transmitted in the Secure Socket Layer session. The most important part of the Secure Socket Layer - handshake is the authentication of the server. If this is corrupt all further generation of session keys will be corrupt and the entire Secure Socket Layer session will be insecure. The server is authenticated via a digital certificate that it sends the client and the clients will proceed then to validate the identity of the host that the certificate claims to represent. A digital certificate contains information such as the certificate version, serial number, signature, issuer, and validity period, among other information.

The procedure of generation of a quantum certificate by the provider of certificates is based on the method of encoding the state of two non-entangled qubits in a qutrit introduced by Grudka and Wójcik (2003). We start from the idea that the information contained by the digital certificate (the owner's public key; the owner's Distinguished Name; the Distinguished Name of the Certificate Authority (CA) that is issuing the certificate; the date from which the certificate is valid; the expiry date of the certificate; a version number; a serial number) are encoded in elementary units of the quantum information, i.e. qubits.

The encoding of two non-entangled qubits in a qutrit is presented as follows. Suppose there are two qubits with the states:

$$\begin{aligned} |\Psi\rangle_1 &= a_1|0\rangle_1 + b_1|1\rangle_1 \\ |\Psi\rangle_2 &= a_2|0\rangle_2 + b_2|1\rangle_2 \end{aligned} \tag{1}$$

The total state of these two qubits is:

$$\begin{aligned} &|\Psi\rangle = |\Psi\rangle_1|\Psi\rangle_2 = (a_1|0\rangle_1 + b_1|1\rangle_1) \\ &(a_2|0\rangle_2 + b_2|1\rangle_2) = a_1a_2|0\rangle_1|0\rangle_2 + \\ &+ a_1b_2|0\rangle_1|1\rangle_2 + b_1a_2|1\rangle_1|0\rangle_2 + b_1b_2|1\rangle_1|1\rangle_2 \end{aligned} \tag{2}$$

The encoding operation then consists of the following mapping:

$$|0\rangle = |0\rangle_1|0\rangle_2;\ |1\rangle = |0\rangle_1|1\rangle_2;\ |2\rangle = |1\rangle_1|1\rangle_2$$

The state $|1\rangle_1|0\rangle_2$ is filtered out in the mapping which is necessary in order to accommodate the two qubits into a single qutrit.

The normalized state of a qutrit (Melikidze *et al.*, 2004: 014435) after encoding is written as state (3):

$$|\Psi\rangle = \frac{1}{\sqrt{1-|b_1|^2|a_2|^2}}\left(a_1a_2|0\rangle + a_1b_2|1\rangle + b_1b_2|2\rangle\right) \tag{3}$$

In the process of generation of a quantum certificate we will use both the qubits belonging to the server, and the qubits belonging to the client, who will be encoded two by two (a server-qubit and a client-qubit) in qutrits which will realize the certificate. The certificate thus obtained could be used both for the authentication of the server to the client, and for the client's authentication, without the need of further exchange of certificates.

A Secure Socket Layer session always begins with an exchange of messages called the Secure Socket Layer - handshake. The handshake allows the server to authenticate itself to the client using quantum techniques.

The steps which need to be followed in *the client-server authentication* in the quantum version included in the Secure Socket Layer handshake protocol can be resumed as follows:

1. The owner of a site who wishes to use the Secure Socket Layer protocol sends an application for a certificate to a certificate provider.

2. The certificate provider works together with a Certificate Authority to whom he sends the application received from the client. After checking if the certificate application is available and if it comes from the source which he pretends he is representing, he creates a certificate which he then encodes with his private key. It is as though he "signed" the certificate. This signature consists of a row of qutrits placed at the beginning of the row of qutrits composing the certificate. The certificate is then sent to the site owner who asked for it.

One thing should be mentioned here, which is the fact that each of the qutrits of the "signature" are obtained through encoding the state of two non-entangled qubits, one qubit of every qutrit contributing to the construction of the public key, and the other qubit which was left - to the construction of the private key. In the case when the belonging of the certificate is contested, or when an intruder interposes between the applicant and the authority and tries the interception, the extraction, and the substitution of the "signature", the Certificate Authority can use the qubits of the private key to reconstruct the qutrits of the "signature" demonstrating the validity of the certificate.

The certificate contains qutrits with the state obtained through the encoding of the states of two non-entangled qubits, one belonging to the server (used in the procedure of server-client authentication), the other belonging to the client (used in the procedure of client - server authentication). Using such certificates, the client-server authentication can be realized for both parties without the need of further exchange of certificates.

3. After the server receives the digital certificate from the Certificate Authority (CA), from now on, every time a client browser compatible with Secure Socket Layer is connected to the server, this certificate will be send to the client browser. In addition, the client receives information related to the bi-dimensional subspace (base vectors) necessary for decoding the qutrits of the certificate, and for the extraction of the qubits that are checking the authenticity of the server. Similarly, for the client's authentication, the server will receive information related to the

bi-dimensional subspace (base vectors) which must be used in the decoding of the qutrits of the certificate, and in the extraction of the client's qubits. This exchange of information between the two parties is realized after the authenticity of the certificate was checked.

4. The client browser, who trusts the Certificate Authority (CA) emitting the certificate, validates the certificate with the public key of the Certificate Authority. The public key contains information related to the bi-dimensional subspace (base vectors) necessary in the procedure of decoding the qutrits from the "signature" of the Certificate Authority. If the qubits of the "signature" applied by the Certificate Authority were obtained, then the validity of the certificate is certified.

After the client assured himself of the validity of the certificate, we will continue with the procedure of client-server authentication.

The authentication mechanism is based on the study realized by Bartuśková *et al.* (2003) according to which from a qutrit with the state (3) one qubit can be perfectly extracted. For the successive extraction of the two qubits it is necessary to project the qutrit on bi-dimensional subspaces, where the projections obtained are:

$$\begin{aligned} &P_{1+} = |1\rangle\langle 1| + |2\rangle\langle 2|; \;\; P_{1-} = |0\rangle\langle 0|; \\ &P_{2+} = |0\rangle\langle 0| + |1\rangle\langle 1|; \;\; P_{2-} = |2\rangle\langle 2| \end{aligned} \tag{4}$$

As a result, each of the parties will use the information related to the bi-dimensional subspaces (base vectors) where the qutrits of the certificate must be projected, consequently realizing the client-server authentication.

After finishing the authentication process, the parties will continue with the generation of encrypting keys. For this they will use the H. Bechmann-Pasquinucci and A. Peres protocol [2] which generates a symmetrical key which will be used afterwards in the encryption/decryption of the information exchanged between client and server.

The H. Bechmann-Pasquinucci and A. Peres protocol (2000:3313) extended the distribution protocol of the quantum key for systems with three states, the so called qutrits. For the preparation of the states of the qutrits which will be sent to the client, the server chooses between any base vectors $|l\rangle$ and $|m\rangle$ belonging to different bases satisfying the condition: $|\langle l|m\rangle|^2 = 1/3$. As a result, the server will use some bases called mutually unbiased bases (Ivanovic, 1981:3241; Wooters, 1986: 391). Suppose the first base chosen arbitrarily is: $\{|\alpha\rangle, |\beta\rangle, |\gamma\rangle\}$. The other bases are obtained by the application of the discrete Fourier transformations.

The first base:

$$\begin{cases} |\alpha'\rangle = (|\alpha\rangle + |\beta\rangle + |\gamma\rangle)/\sqrt{3} \\ |\beta'\rangle = (|\alpha\rangle + e^{2\pi\pi i/}|\beta\rangle + e^{4\pi\pi i/}|\gamma\rangle)/\sqrt{3} \\ |\gamma'\rangle = (|\alpha\rangle + e^{4\pi\pi i/}|\beta\rangle + e^{2\pi\pi i/}|\gamma\rangle)/\sqrt{3} \end{cases} \tag{5}$$

The second base is obtained through cyclical permutations:

$$\begin{cases} |\alpha''\rangle = (e^{2\pi\pi i/}|\alpha\rangle + |\beta\rangle + |\gamma\rangle)/\sqrt{3} \\ |\beta''\rangle = (|\alpha\rangle + e^{2\pi\pi i/}|\beta\rangle + |\gamma\rangle)/\sqrt{3} \\ |\gamma''\rangle = (|\alpha\rangle + |\beta\rangle + e^{2\pi\pi i/}|\gamma\rangle)/\sqrt{3} \end{cases} \tag{6}$$

The third base is obtained through cyclical permutations:

$$\begin{cases} |\alpha'''\rangle = (e^{4\pi\pi i/}|\alpha\rangle + |\beta\rangle + |\gamma\rangle)/\sqrt{3} \\ |\beta'''\rangle = (|\alpha\rangle + e^{4\pi\pi i/}|\beta\rangle + |\gamma\rangle)/\sqrt{3} \\ |\gamma'''\rangle = (|\alpha\rangle + e^{4\pi\pi i/}|\beta\rangle + e^{4\pi\pi i/}|\gamma\rangle)/\sqrt{3} \end{cases} \tag{7}$$

The server randomly chooses one of the 12 states and sends it to the client. He randomly chooses one of the four bases and measures the state, then announces publicly what base he used, without telling the result he obtained. The server checks if the choice is correct. If it is, then both are in the possession of the same bits of information; if not, they give it up.

The procedure is repeated until the server and the client obtain a sufficiently big number of bits, the following steps being to correct the errors and to remove any residual information which an intruder could have introduced.

At this moment, both parties are in the possession of a symmetrical key, the Secure Socket Layer - handshake protocol ends, thus assuring a secure connection between a client and a server through which any quantity of data could be sent securely.

4. CONCLUSIONS

The use the method of the encoding the state of two non-entangled qubits in a qutrit method in the authentication procedure determines giving up the long row of authentication certifications used in the classical case in order to remove the suspicions existent before starting the communication process. The main purpose of the protocol based on encoding the state of two non-entangled qubits in a qutrit is the fact that it offers a different conceptual way to solve some of the problems related to client-server authentication. The integration of quantum techniques bring an advantage in what concerns the security of the method, the no-cloning theorem, and the principle of irreversibility of quantum systems measurement, guaranteeing for it. The advantages consist in the improvement of the efficiency of the classical protocols, the detection of the intruders implying the comparison of a smaller number of bits as compared to the high probability that the intruder modifies the result expected by the parties involved in communication.

4. ACKNOWLEDGMENT

This work was co financed from the European Social Fund through Sectoral Operational Programme Human Resources Development 2007-2013, project number POSDRU/89/1.5/S/56287 "Postdoctoral research programs at the forefront of excellence in Information Society technologies and developing products and innovative processes", partner University of Oradea.

BIBLIOGRAPHY

1. Bartuśková L. *et al.* (2003). *Optical implementation of the encoding of two qubits to a single qutrit.* quant-ph/0603174v1.
2. Bechmann-Pasquinucci H., Peres A. (2000). Quantum Cryptography with 3-state systems., *Phys. Rev. Lett.* 85.
3. Grudka A. and Wójcik A. (2003), *How to encode the states of two non-entangled qubits in one qutrit*, http://quant-ph/0303168.
4. Ivanovic I.D. (1981), J. Phys. A: *Math. Gen.*, 14.
5. Melikidze A., Dobrovitski V.V., De Raedt H.A., Katsnelson M.I. and Harmon B.N. (2004). Parity effects in spin decoherence. *Phys. Rev.* B 70.
6. Wootters W.K. (1986). *Found. Phys.*, 16.