"HENRI COANDA"
AIR FORCE ACADEMY
ROMANIA

"GENERAL M.R. STEFANIK"
ARMED FORCES ACADEMY
SLOVAK REPUBLIC

INTERNATIONAL CONFERENCE of SCIENTIFIC PAPER
AFASES 2014
Brasov, 22-24 May 2014

BUILDING A HYBRID SECURE SOCKET LAYER PROTOCOL

Gabriela MOGOS*

* Facultad de Informatica y Electronica, Escuela Superior Politecnica de Chimborazo, Riobamba, Ecuador

Abstract: *Randomness is an essential resource for cryptography and random number generators are, at the same time, very important for the most cryptographic systems. Using these weak random values may lead to vulnerabilities of the systems, causing the adversary to easily break the system, as demonstrated by breaking the implementation of Secure Socket Layer.*

This paper presents a hybrid Secure Socket Layer that includes together a Quantum Random Number Generator (QRNG), to generate the random numbers, and, an Elliptical Curve Cryptography (ECC) algorithm, to encrypt the data.

Keywords: *asymmetric cryptosystems, quantum random numbers, Secure Socket Layer.*
MSC2010**:** 81P45, 94A15.

1. INTRODUCTION

Secure Socket Layer uses a combination of public key and symmetric-key encryption. Symmetric-key encryption is much faster than public-key encryption, but public-key encryption provides better authentication techniques.

The Secure Socket Layer handshake allows the server to authenticate itself to the client using public-key techniques, then allows the client and the server to cooperate in the creation of symmetric keys used for rapid encryption, decryption, and tamper detection during the session that follows.

In our paper, we present a theoretical demonstration of how to improve the security of Secure Socket Layer protocol by replacing Random Number Generator with Quantum Random Number Generator and the *Rivest, Shamir, Adleman* **(**RSA) cryptographic algorithm, currently used, with *Elliptic Curve Cryptography* (ECC) algorithm.

In first part of paper, we present *Random Number Generator* (RNG) and *Quantum Random Number Generator* (QRNG), and, advantages of using QRNG in generating a sequence of true random bits.

To motivate the need for replacing the *RSA cryptosystem* (existing in SSL 3.0) with *ECC cryptosystem*, in the second part of the paper, we present a comparative analysis of those cryptosystems and in the last part, we present the hybrid Secure Socket Layer protocol.

2. HYBRID SECURE SOCKET LAYER PROTOCOL

2.1 Random Number Generators vs. Quantum Random Number Generators.

Random numbers seem to be of an ever increasing importance in cryptography,

various stochastic numerical simulations and calculations, statistical research, various randomized or stochastic algorithms, etc. and the need for them is spanning a wide range of fields from engineering to physics to bioinformatics.

The applications usually put constraints on properties of input random numbers (probability distribution, bias, correlation, entropy, determinism, sequence repeatability, etc.).

Random number generators based on quantum physics are true random number generators as quantum physical phenomena are intrinsically random.

Our paper is based on the study done by researchers [11] from the Joint Quantum Institute (JQI), in partnership with European quantum information scientists, who have found a method of producing a certifiably random string of numbers based on fundamental principles of quantum mechanics.

They report their results in the 15 April 2010 issue of *Nature* [12].

What is the rationale for that quantum randomness is a better form of randomness than, classic randomness?

Quantum hardware random number generators produce sequences of numbers that are not predictable, and therefore provide the greatest security when used to encrypt data.

Three principal types of quantum indeterminism underlying to realize a QRNG:

(i) The indeterminacy of individual outcomes of single events as proposed by Born and Dirac;

(ii) Quantum complementarity (due to the use of conjugate variables), as put forward by Heisenberg, Pauli and Bohr;

(iii)Value indefiniteness due to Bell, Kochen & Specker, and Greenberger, Horne & Zeilinger).

A simple design for a QRNG is based on a construction with a photon source, a 50/50 beam-splitter and two identical single photon detectors. This construction ensures a random and balanced probability of detection of the photon at each detector.

Meanwhile, commercial Quantum Random Number Generators have already appeared on the market. In 2001, *ID Quantique* [3] introduced the first commercial quantum random number generator, which generated a strong interest.

Seeing that, the Quantum Random Number Generator is ideal for applications requiring very high rates of true random numbers, our aim is to replace existing the classic Random Number Generator (RNG) used by Secure Socket Layer protocols, with quantum one (QRNG).

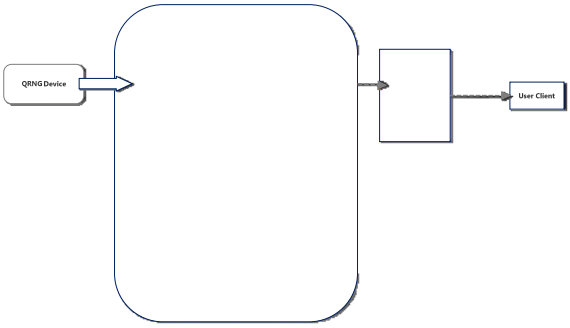


Figure 1. Connect a Quantum Random Number Generator to a computer

The Quantum Random Number Generator is used for cryptographic purposes and other applications requiring the highest levels of security and randomness properties.

The Quantum Random Number Generator device connects to a computer via PCI or USB interface.

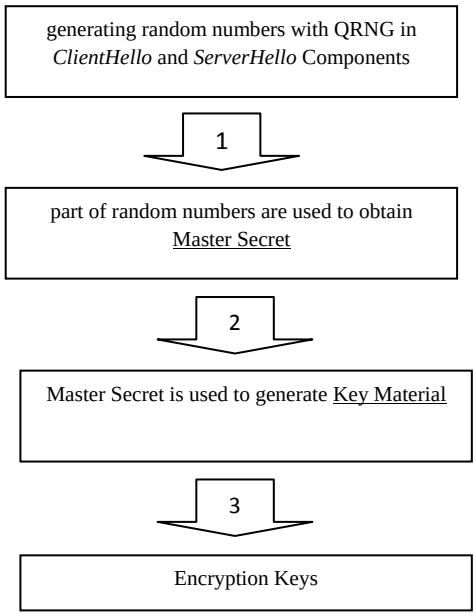
Advantages of Quantum Random Number Generator on Secure Socket Layer:

The *Quantum Random Number Generator* offers the best option for strong security in the long term, especially if the technologies are based on the use of algorithms. Photons and entangled photons of the sort used in quantum cryptography and communication have a form of tamper evidence and protection built into the technology at the level of quantum-mechanical processes. At the moment, it is widely believed that this technology cannot be defeated.

Quantum Random Number Generators (QRNGs) have the advantage over conventional randomness sources of being invulnerable to environmental perturbations and of allowing live status verification.

Quantum Random Number Generator (QRNG) can generate truly random numbers from the fundamentally probabilistic nature of quantum processes.

Generating large numbers using QRNG, lead to a "chain reaction" of improving the security SSL protocol.



2.2 Elliptical Curve Cryptography (ECC) vs. Rivest, Shamir, Adleman (RSA) cryptosystem.

Every Secure Socket Layer connection begins with a handshake, during which the two parties communicate their capabilities to the other side, perform authentication, and agree on their session keys. The session keys are then used to encrypt the rest of the conversation, possibly spanning multiple connections. They are deleted afterwards. The goal of the key exchange phase is to enable the two parties to negotiate the keys securely, to prevent anyone else from learning these keys.

Several key exchange mechanisms exist, but, at the moment, by far the most commonly used one is based on RSA [9], where the server's private key is used to protect the session keys.

Elliptic Curve Cryptography (ECC) algorithm [5] relies on the difficulty of solving the Elliptic Curve Discrete Logarithm Problem (ECDLP) in much the same way that RSA depends on the difficulty of factoring the product of two large primes. The best known method for solving ECDLP is fully exponential, whereas the factoring problem is sub-exponential.

Starting from the data set [1], we have realized a comparative analysis between these two algorithms:

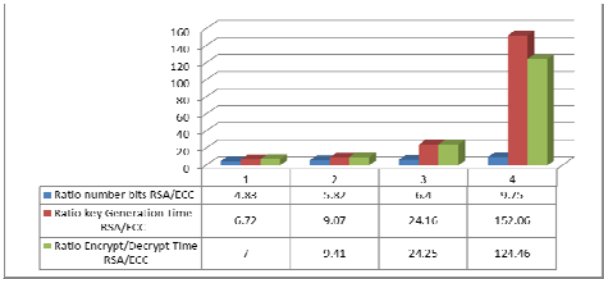


Figure 2. Comparative analysis RSA/ECC

Advantage of Elliptic Curve Cryptography on Secure Socket Layer:

- ECC is an emerging public-key cryptosystem that offers equivalent security with smaller keys sizes.
- Augments end to end encryption for data in flight by helping to maintain data privacy and prevent data leakage of sensitive information particularly when providing the next generation of security level requirements.
- We can reduce the transmission cost during handshake.
- Can be used to implement encryption on small, mobile devices with limited resources in terms of power, CPU and memory.
- Can be used for large web servers to handling many encrypted sessions.

2.3 Secure Socket Layer a hybrid protocol

Speculating weaknesses of the current SSL version 3.0, we propose an improvement of it in terms of performance, security and space requirements.

In the following we present our reasons to replace the classical random number generator currently used by Secure Socket Layer protocol with quantum random generator, and, RSA algorithm with ECC.

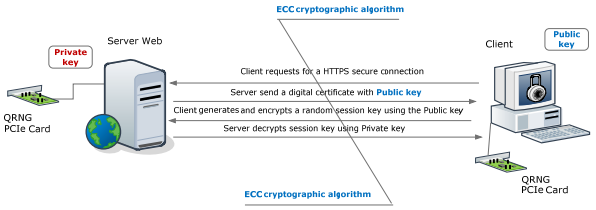


Figure 4. Schematic diagram of the protocol

I. Replace the Random Number Generator (RNG) with quantum one (QRNG) - the obtained key is very safe and we can get an increase of the number of bits that should be a "cryptographically secure" random number.

Due to indeterminism (Bohr and Heisenberg) and exclusion principles (Pauli) from quantum physics, the opportunity of attackers to "guess" the numbers is zero, which makes the generated numbers to be truly random. Since the data will be encrypted using asymmetric cryptographic algorithms, key size is essential in ensuring the security and integrity of encrypted information.

II. Replace RSA cryptographic algorithm with ECC algorithm - we will provide the same security protocol, however, the smaller key sizes lead to faster processing, which is very useful to implementing encryption on small, mobile devices with limited resources in terms of power, CPU and memory.

3. CONCLUSIONS & ACKNOWLEDGMENT

We can conclude that the *Hybrid Secure Socket Layer* can offer security advantages over conventional alternatives, can be used for large web servers that will be handling many encrypted sessions and its performances are better than the classic Secure Socket Layer protocol.

This theoretical study is part of a larger project [13] that we will be developed by *Facultad de Informatica y Electronica, Riobamba, Ecuador*.

This work was financed from Secretaría de Educación Superior, Ciencia, Tecnología e Innovación (SENESCYT) of Ecuador, through Prometeo project.

REFERENCES

1. Kancheti S., *Comparative Study of Elliptic Curve Cryptography and RSA in Constrained Environment*, (2010).
2. Lofberg J., *Yalmip: A toolbox for modeling and optimization in MATLAB.*
3. http://www.idquantique.com
4. Navascues M., Pironio S., and Acin A.. *Bounding the set of quantum correlations*, Physical Review Letters, 98:010401, (2007).
5. NIST, Recommendation for Key Management – Part 1: General (Revised). *National Institute of Standards and Technology*. NIST Special Publication 800-57, (2007).
6. Pironio S., Navascues M., Acin A., *Convergent relaxations of polynomial optimization problems with non-commuting variables*, arXiv:0903.4368, (2009).
7. Pironio S., Acin A., Brunner N., Gisin M., Massar S., and Scarani V., *Device-independent quantum key distribution secure against collective attacks*, New Journal of Physics, 11:045021, (2009).
8. In the second edition of *Numerical Recipes in C: The Art of Scientific Computing* (Cambridge University Press, 1992), p. 277. Chapter 7 includes a thorough (and sobering) discussion of random number generation.
9. Rivest R.L., Shamir A., & Adleman, L., *A Method for Obtaining Digital Signatures and Public-Key Cryptosystem*, Retrieved from http://people.csail.mit.edu/rivest/Rsapaper.pdf , (1978).

10. Sturm J.F., SeDuMi, *a MATLAB toolbox for optimization over symmetric cones*, http://sedumi.mcmaster.ca.
11. http://jqi.umd.edu/sites/default/files/newsletters/feb_2010_newsletter.pdf.
12. Pironio S., Acín A., Massar S., Boyer A. de la Giroday, Matsukevich D. N., Maunz P., Olmschenk S., Hayes D., Luo L., Manning T. A. & Monroe C., *Random numbers certified by Bell's theorem*, Nature, 2010; 464 (7291): 1021 DOI: 10.1038/nature09008.
13. Mogos G., Radu Gh., *Hybrid Secure Socket Layer protocol*, Review of the Air Force Academy, Vol. XII, No. 2(26), pp. /2014, http://www.afahc.ro/revista/Nr_2_2014/91_MOGOS_RADU.pdf.