"HENRI COANDA"
AIR FORCE ACADEMY
ROMANIA

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

INTERNATIONAL CONFERENCE of SCIENTIFIC PAPER
AFASES 2013
Brasov, 23-25 May 2013

QUANTITATIVE EVALUATION OF IDENTITY BASED CRYPTOGRAPHY IN AN AUTHENTICATION SCENARIO

Mihai-Lica PURA*

*Faculty of Military Electronic and Information Systems, Military Technical Academy, Bucharest, Romania

Abstract: *Identity based cryptography is a particular case of asymmetric cryptography in which the public key is chosen so that it uniquely identifies its owner. This information can be any identifier of a person or a system, like the e-mail address, the IP address, and so on. This way there is no need for certificates, because such public keys can be managed very easy. This certificate less operation of identity based cryptography has obvious advantages over classical asymmetric cryptography because it eliminates all the validations required by the use of certificates. But can this advantage be quantified? Asymmetric cryptography is very popular, and, even with all the research conducted in order to promote the use of identity based cryptography, it is still used only in a limited number of applications. Our purpose is to quantitatively evaluate the difference between these two types of asymmetric cryptography in order to have a formal way of comparison between them. For these we have chosen an authentication scenario, giving the fact that authentication is used in the vast majority of security applications. In order to obtain the quantitative results we have implemented Needham-Schroeder protocol using both classical asymmetric cryptography and identity based cryptography. Then we measured the time needed for the authentication of the two parties in each of the case. The results suggest that, at the setup phase, identity based cryptography is slower than traditional asymmetric cryptography, but that after it, it is faster.*

Keywords: *security, identity based cryptography, simulation, ns2, Needham-Schroeder*

1. INTRODUCTION

Identity based cryptography is a special case of asymmetric cryptography. Its particularity is given by how the public-private key pair is computed. In the classical asymmetric cryptography the key pair is generate by the Hardware Secure Module (HSM) of the client. The two keys two very large numbers with no special meaning. The public key is then sent to the Certification Authority (CA), where the certificate is build and signed, thus connecting the key pair owner with the public key, through the CA.

But in the case of the identity based cryptography, the aim is to eliminate the need of using certificates. This is accomplished by choosing the public key so that it is bounded uniquely to the user ([10]). Then, the private key is computed based on the chosen public key.

Because the certificates are no longer needed, identity based cryptography eliminates all the operations required for them: management (certificate store and lookup), validation (time, revocation, and issuer), and renewal. This would suggest that using identity based cryptography would be faster than using traditional asymmetric cryptography. But this informal way of drawing that conclusion could me misleading. So our purpose was to evaluate the two types of cryptography from a quantitative point of

view so that the comparison would be more reliable.

In order to make this evaluation we used a simulation environment, namely Network Simulator 2 (ns2). We have implemented Needham-Schroder authentication protocol ([5]) in two ways: using classical asymmetric cryptography and using identity based cryptography. Then, in a very simple communication scenario, we measured the time taken by the authentication of two nodes using the two versions of the protocol. The results allowed us to make a more reliable comparison between the two types of cryptography.

The rest of the paper is organized as follows: section 2 presents identity based cryptography in a briefly manner and its main differences from classical asymmetric cryptography, as well as Needham-Schroeder authentication protocol. Section 3 describes the two implementations made for ns2. In section 4 we gave the simulation scenarios and the obtained results. Section 5 contains some conclusions and future research directions.

2. THEORETICAL ASPECTS

2.1 Identity based cryptography. Identity based cryptography (IBC) is a type of asymmetric cryptography in which the public key is computed starting from an arbitrary string of characters ([10]). In its main application, as the name suggests, this string represents the identity of the owner of the key, thus being bound directly to it. This string must be unique for each of the users of the cryptographic system. In the simplest implementation of IBC, this string represents a single identifier of the owner, but in complex applications it is better to be formed from more than one identifier. As we already highlighted, the public key is directly connected to its owner ([7]), so there is no longer the need for a trusted third party to certify that a certain key belongs to a certain entity ([13]). And this is the main advantage of IBC systems.

Still there is a need for a trusted third party, but it has another function. In traditional PKI an entity computes its own key pair and uses the CA to certify that the public key really belongs to it. But in IBC system this is no longer possible. Each user of the system computes its public key starting from the identity or the identities that are used. But this means that it can also compute the keys of all the other users. This is normal, because the public key is public. But the users must not be capable of computing the private keys. If a user could compute its private key, it could compute the private key of any user. So a trusted third party is needed for the purpose of computing the private keys of each of the users of the system so that they are really private. And this is the main disadvantage of IBC: the fact that the private keys are known by a central authority ([3]).

Given this explanation, IBC operates as described below ([1], [6]). The trusted third party responsible for generating the private keys is called Key Generation Center (KGC). It computes the public parameters of the system that must be known by all the users in order to compute the public keys and to perform the cryptographic operations. Each of the users of the system receives these public parameters from the KGC and computes its public key and the public keys of the users it wants to communicate with. Then, it requests its private key from the KGC. The KGC computes the private key for each of the users starting from the public key of a user and using the private parameters that correspond to the public ones made available to all the nodes. Each node must receive the private key from the KGC on a secure channel, so that no eavesdropping is possible ([3]).

This way each of the users has the following elements: the public parameters of the KGC, the public key and the private key. Also it can compute the public key of any other user. The cryptographic operations are conducted using this elements in a similar way to those in a PKI system, so we will not present them here ([8]).

To summarize, IBC has two main advantages: there is no need for certificates, and the KGC is no longer needed after all the users obtained their private key, so it can be eliminated, thus eliminating the central point of failure of this cryptographic system.

Because there is no need for certificates, all the validations necessary for their use in a PKI

are eliminated: time validity, revocation status and signature of the CA over the certificate ([11], [3]). Also certificate management for the users is eliminated too. From a practical point of view these eliminations mean that when a user wants to perform a cryptographic operation (signature, encryption, decryption, and signature verification) it can perform the operation right away, without the need to do additional computations. So it should be obvious that IBC assures a faster operation than PKI.

What we wanted to research in our work was to compare IBC and PKI from the point of view of these cryptographic operations, ignoring the advantage given by the elimination of the certificates and to view which type of cryptographic system is faster. For this evaluation we chose RSA asymmetric key algorithm for classic asymmetric cryptography, and Boneh-Franklin identity based encryption scheme for IBC ([2]).

2.2 Needham-Schroeder protocol. Needham-Schroeder is a mutual authentication protocol between two nodes ([5]). In order to explain how the protocol works, let us consider two nodes A and B that want to authenticate each other. Each of the nodes has a key pair formed by a public key pub(node) and a private key priv(node). In the first step of the protocol, node A generates a nonce N(A), encrypts it with the public key of B and then sends it, together with its own public key to node B. The public key of A is also encrypted with the public key of B. Of course this first step assumes that A has somehow received the public key of B. How it is obtained it is not important for the run of the protocol.

In the next step of the protocol, node B receives the encrypted message from A and decrypts it using its private key. Then B generates a nonce of its own, N(B), and sends back to A the received nonce N(A) and the generated nonce N(B), both encrypted with the public key received from A.

In the final step of the protocol, node A will decrypt the two received nonces and will compare the nonce N(A) received from B with the nonce it has generated for the first step of the protocol. If the two match, it means that B was authenticated by A. Then A sends the nonce N(B) received from B back to it, encrypted with B's public key. B receives the nonce, decrypts it and then compares it with the one it has generated in the second step of the protocol. If the two match, A was authenticated by B. And thus the mutual authentication was successful.

This is the original version of the protocol. Over the years that have passed from its proposal, it was proved that it is insecure and other secure versions of it were proposed. But for the purpose that we have set, this has no importance at all.

3. PROTOCOL IMPLEMENTATIONS

The purpose of our work was to make a quantitative comparison between the two types of asymmetric cryptography, as we talked above. We chose to do the evaluation in a simulation environment. And we used Network Simulator 2 (ns2). Ns2 allows the implementation of a communication protocol at any layer of the TCP/IP protocol stack. Needham-Schroeder is an application layer protocol and was implemented in order to be used for the mutual authentication of two nodes that belong to the same network in an ns2 scenario. For details on how to implement a protocol in ns2, please see ([12], [9]).

We have developed two implementations: one version that uses classical asymmetric

cryptography, and another version that uses identity-based cryptography. We will further present the key aspects of these implementations.

For the implementation of the cryptographic operations necessary in the protocol we used a library called MIRACL ([4]) which implements both RSA and Boneh-Franklin IBC encryption scheme ([2]).

3.1 Needham-Schroeder protocol classical asymmetric cryptography implementation. In this version of the implementation we used RSA asymmetric cryptography algorithm in order to implement the encryptions necessary in the protocol. Because we did not want to include in our evaluation the operations performed for validating the certificates, we did not implement a CA. Each node, at the initialization of the protocol, generates for itself a public-private key pair needed for RSA. Then, at each step of the protocol, each node uses RSA to perform the necessary encryptions.

3.2 Needham-Schroeder identity-based cryptography implementation. Because in IBC a node cannot generate its own private key, in this second implementation it was mandatory to implement a KGC responsible for generating the private keys for each of the two nodes. So, at the initialization of the protocol, the node that represents the KGC generates the public and private parameters necessary for the encryption scheme that will be used. Then, each of the nodes requests from it a private key. The public key used is the IP address of the node. The KGC responds to each of the nodes with the public parameters of the scheme and with the private key that corresponds to the IP address of the node. We have presumed that this communication between a node and the KGC is done over a secure channel. This presumption does not influence the actual run of the Needham-Schroeder protocol, neither is affects our evaluation.

After each of the two nodes has obtained its private key, the KGC is disposed and the protocol starts running. As we stated above, the algorithm used for the necessary encryption is based on the Boneh-Franklin IBC encryption scheme ([2]).

4. SIMULATION AND RESULTS

The scenarios in which the tests were performed are very simple. The network is composed of only two nodes in the case of classical asymmetric cryptography, and three nodes in the case of IBC (the two communicating nodes, and the KGC). After the initialization phase takes place, we start measuring the time needed until the protocol is completed and the authentication succeeds. The results obtained are these: the protocol run takes 13.1 milliseconds in the case of the version that uses classical asymmetric cryptography, and 16.6 milliseconds in the case of the version that uses IBC. The computer that we used for the tests had a Intel Dual Core processor at 2 GHz, 2 GB of RAM and ran openSuse 11.

5. CONCLUSIONS & ACKNOWLEDGMENT

We have proposed to offer a quantitative comparison between asymmetric cryptography and identity-based cryptography from the point of view of the cryptographic operations. We targeted RSA for classical asymmetric cryptography and Boneh-Franklin IBC encryption scheme. The scenario used for the simulations in which we did the actual evaluation was an authentication scenario based on Needham-Schroeder protocol. We measured for the two versions of the protocol that were implemented how much time is needed for a protocol run.

The obtained results showed that RSA is faster than the Boneh-Franklin scheme. And this is due to the fact that the generation of the public parameters and the generation of the private keys for IBC takes more time than the generation of the public-key pair that is done at each node in case of classical asymmetric cryptography.

The conclusion is that if the certificates are eliminated, IBC is slower than classical asymmetric cryptography so we have to be very careful when choosing between one and the other. The scenario in which cryptography will be used is very important in making this choice: if the certificates can be eliminated

than PKI is a better choice from performance point of view. But if the certificates are mandatory, than IBC will be the best choice.

As future work, we want to extend this evaluation by comparing other asymmetric cryptography algorithms and other IBC schemes ([COCL01]). Also it would be interesting to make this evaluation not in a simulation scenario, but using a real implementation.

We would like to thank to engineer Florin Vladescu who made the ns2 implementations in C++ and Tcl for the two versions of the protocol in the work for his diploma paper.

REFERENCES

1. Balfe, S., Boklan, K. D., Klagsbrun, Z., Paterson, K. G., *Key Refreshing in Identity-Based Cryptography and its Applications in MANETS*, *Proceedings of the Military Communications Conference MILCOM '07* (2007).
2. Boneh, D., Franklin, M. K., *Identity-Based Encryption from the Weil Pairing*, *Advances in Cryptology: Proceedings of CRYPTO '01* (2001).
3. Joux, A., *Introduction to Identity-Based Cryptography*, *Identity- Based Cryptography*, IOS Press (2009).
4. MIRACL, online. Available: https://certivox.com/solutions/miracl-crypto-sdk/ (February 2013).
5. Needham, R.M., Schroeder, M.D., *Using encryption for authentication in large networks of computers*, *Communications of the ACM*, vol. 21, issue 12 (1978).
6. Ng, Y., Mu, Y., Susilo, W., *An Identity-Based Broadcast Encryption Scheme for Mobile Ad Hoc Networks*, *Journal of Telecommunications and Information Technology*, issue 1, pp. 24-29 (2006).
7. Oliviera, L., Aranha, D., Morais, E., Daguano, F., Lopez, J., Dahab, R., *TinyTate: Identity-Based Encryption for Sensor Networks*, unpublished, online. Available: http://eprint.iacr.org/2007/020.pdf, (February, 2013).
8. RFC 5091, online. Available: http://www.rfc-editor.org/pipermail/rfc-dist/2007-December/001834.html (February 2013).
9. Ros, F. J., Ruiz, P. M., *Implementing a New Manet Unicast Routing Protocol in NS2*, online. Available: http://imtl.skku.ac.kr/~hjlim99/ns2/%5B%BB%F5%B7%CE%BF%EE%20%B6%F3%BF%EC%C6%C3%20%C7%C1%B7%CE%C5%E4%C4%DD%20%B1%B8%C7%F6%5D/Implementing%20a%20New%20Manet%20Unicast%20Routing%20Protocol%20in%20ns2.pdf (February 2013).
10. Shamir, A., *Identity-Based Cryptosystems and Signatures Schemes*, *Advances in Cryptology: Proceedings of CRYPTO'84*, *Lecture Notes in Computer Science*, 7:47-53 (1984).
11. Sherman S.M. Chow, Certificateless Encryption, *Identity-Based Cryptography*, IOS Press, pp. 207-225 (2009).
12. Xu, L., How to Add a New Protocol in NS2, online. Available: https://docs.google.com/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbWFpbnxuYWVlbWtoYWRlbWl8Z3g6NTA4YmZmNmNhOTk3MmRl (February 2013).
13. Yang, G., Rong, C., Veigner, C., Wang, J., Cheng, H., *Identity-Based Key Agreement and Encryption for Wireless Sensor Networks*, *International Journal of Computer Science and Network Security*, vol. 6, issue 5B, pp. 182-189 (2006).