

Security of information systems: Implementation of encryption

Mohammed ERRITALI, Mohamed Fakir, Belaid Bouikhalene

Département d'informatique

Faculté des sciences et techniques Benimellal

Maroc

mederritali@yahoo.fr

Abstract— In this work we provide a panorama on the use of cryptography and digital signature to secure an information system, we start with a state of the art about cryptographic algorithms, and digital signature algorithms and we finish by some applications.

Keywords-- Symmetric encryption, asymmetric encryption, digital signature, SSL.

I. INTRODUCTION

The security of computer systems is generally limited to guaranteeing rights of access to data and system resources by implementing authentication mechanisms and monitoring to ensure that users of these resources have only those rights that they were granted. The security mechanisms in place can still cause discomfort to users and guidelines and rules are becoming increasingly complicated as they as the network expands. Thus, IT (Information Technology) security must be studied in such a way that does not prevent users to develop uses that are necessary, and ensure that they can use the information system with confidence.

Indeed the concept of cryptography is born from the moment we wanted to provide safe from eavesdroppers. From Julius Caesar and his army, to Romeo and Juliet, through all the treasure maps, it took encrypt certain information. The contemporary period has not improved in this area. Instead, the consumer society has created new cryptographic needs. Of course these are military applications remained secret communications, and banking.

But it must also ensure the secrecy of communications on networks remote computer, and prevent the modern-day pirates to infiltrate these networks of computers. Cryptography is used increasingly in various fields.

Until recently, the security of these systems was based on secret information that is shared by users, and allowed to communicate confidentially. For this reason, all of these systems is called secret key cryptography. As secret key systems, it uses algorithms using the same key for encryption and decryption and for this, are called symmetric encryption algorithms. DES, AES, IDEA are the most famous examples. Although these algorithms are still used for encrypting messages because of very high speed, they no longer meet the new needs.

The public key cryptography has been formalized and helped meet these needs. These are all ways that can provide solutions to new problems which are identification, authentication and confidentiality of messages.

The public key cryptography is involved in many everyday applications, the use of smart cards through mobile phones, until a user logs in to a computer. However, the security of

these applications depends mainly on two issues considered difficult number theory: the problem of factoring and discrete logarithm problem. Although these two problems are still resisting the cryptographers, they are not immune from a theoretical breakthrough or even quantum computers that would endanger the difficulty of solving them.

II. SYMMETRIC SECRET KEY ENCRYPTION

From the time of Julius Caesar in the late 1970s, many cryptosystems have been invented (DES, AES,...) [2], consisting in subjecting a clear text processing more or less complex to derive a text, said encrypted. The transformation is based on two elements: a mathematical function and a secret key. Only a person familiar with the function and the key can perform the inverse transformation, which transforms the cipher text into plaintext. The same key used for encryption and decryption, and for this reason it must remain secret.

A. Data Encryption Standard (DES)

The first data encryption standard was developed by German-born American cryptographer Horst Feistel in 1934 [2].

DES is based on the following principles: the plaintext is encoded in binary and cut into blocks of 64 bits. Each block is cut in half blocks whose bits undergo complex permutations, then the half-blocks are added together and subjected to other transformations. The operation is repeated sixteen times. The transformation function has variations depending on the key, which is an arbitrary number chosen by the user code. The number of possible values for the key determines the number of ways in which a message can be encrypted. The sender of the message secret number according to the DES algorithm using the key, the receiver applies the inverse function with the same key to decrypt it.

In 1976 [2] the standardization of DES has a limit on the key size to 56 bits. Today value is notoriously weak, and it uses the triple DES with a key length of 112 bits.

B. Advanced Encryption Standard (AES)

It comes from an international call for applications launched in January 1997 and has received 15 proposals. Of these 15 algorithms, five were selected for further evaluation in April 1999: MARS, RC6, Rijndael, Serpent, and Twofish. After this assessment, it was finally the candidate Rijndael, named after its two designers Joan Daemen and Vincent Rijmen (both Belgian nationality) who has been chosen [9,10]. These two experts in cryptography were already authors of another algorithm: Square. AES is a subset of Rijndael: it only works with blocks of 128 bits, whereas Rijndael offers block

sizes and keys that are multiples of 32 (between 128 and 256 bits).

In so doing the AES replaces the DES (chosen as standard in the 1970s) which today became obsolete, because it used only 56-bit keys. The AES has been adopted by NIST (National Institute of Standards and Technology) in 2001 [9,10]. Moreover, its use is very convenient because it uses little memory and is not based on a Feistel scheme, its complexity is lower and it is easier to implement.

The algorithm takes as input a block of 128 bits (16 bytes), the key is 128, 192 or 256 bits. The 16 input bytes are swapped according to a predefined table. These bytes are then placed in a 4x4 matrix components and lines are rotated to the right. The increment for the rotation varies with the number line. A linear transformation is then applied to the matrix, it consists of a binary multiplication of each element of the matrix with polynomials from an auxiliary matrix, this increase is subject to special rules as GF (2^8) (Galois group finite) [9,10]. The linear transformation ensures a better distribution (propagation of bits in the structure) on several laps.

Finally, an XOR between the matrix and another matrix provides an intermediate matrix. These different operations are repeated several times and set a "tower". For a key of 128, 192 or 256, AES requires respectively 10, 12 or 14 towers.

C. Protocol Diffie and Hellman

If two network users Ayoub and Mohammed wants to keep a secret correspondence, they may agree to encrypt their messages with an algorithm such as Triple DES or AES, we have presented. This algorithm has all the guarantees of robustness, but it will take them to be agreeing on a secret key: for this they must meet, which may be impossible, or to communicate the key by mail. In both cases, the moment of exchange is that a spy can take advantage to steal their secret and thus nullifying the security of their communications. This is the problem of key exchange.

C. 1. The problem of key exchange

For centuries the problem of key exchange was seen as a natural disadvantage of encryption.

With the use of computer and tele-transmission, and the dematerialization of information they allow the problem is different. In 1970 an independent researcher, Whitfield Diffie, reflected by two of ARPANET users to exchange encrypted emails without physically meet beforehand to agree on the encryption key that they use it[2]. In 1974 he gave a lecture on the research center Thomas J. Watson of IBM in Yorktown Heights (already at work of Horst Feistel), and there he learned that Martin Hellman, a professor at Stanford University in Palo Alto, gave a lecture on the same subject. He immediately took his car and crossed the continent to meet Hellman [2].

Diffie and Hellman were looking for a way to agree on a shared secret without being circulated among the participants, in other words, a mathematical function such that participants can exchange information alone could deduce the secret. The desired characteristics of such a function are the relative ease of calculation in the forward direction, and almost impossible to calculate the inverse function. Thus, if s is the secret to clear the encryption function F , c secret encrypted, the decryption function D , it is necessary that $c = F(s)$ is easy to calculate, but if $D = (c)$ impossible to calculate for any other participants.

C.2. Implementation of Diffie-Hellman

The protocol for key exchange Diffie-Hellman is based on a function of the form, first with P & $W < P$.

This function is very easy to calculate, but the knowledge of K does not imply easily X . This function is public, and the values of W and P .

1. Ayoub chooses a number that will remain his secret, say A .
 2. Mohammed chooses a number that will remain his secret, say B .
 3. Ayoub and Mohammed want to exchange the secret key, which is actually, but they do not know yet, since everyone knows that A or B , but not both.
 4. Ayoub applies to A the one-way function, α is the result: $\alpha = W^A \bmod P$
 5. Mohammed applies to B -way function, β is the result: $\beta = W^B \bmod P$
 6. Ayoub sends α to Mohammed, and Mohammed sends β , as shown by, they may be known to the whole world without the secret of Ayoub and Mohammed is disclosed.
 7. Ayoub received β and calculates $\beta^A \bmod P$ (that is to say in passing by, $(W^B)^A \bmod P$, but he does not know B): $S = \beta^A \bmod P$.
 8. Mohammed received α and computes $\alpha^B \bmod P$ (that is to say in passing by $(W^A)^B \bmod P$, but he does not know A): $S = \alpha^B \bmod P$.
- Mohammed and Ayoub get to the end of their respective calculations the same number that has never been exposed to the sight of prying: the S key.

III. THE ASYMMETRIC PUBLIC KEY ENCRYPTION

The method of Diffie and Hellman allows the exchange of keys, but it imposes a preliminary dialogue between the actors. Sometimes it is not practical: if Ayoub wants to send to Mohammed an encrypted email while it is on holiday, it will be obliged to await its return to establish the key with him.

Whitfield Diffie had another idea, which he did not find appropriate mathematical solution: a system which would use a key to encrypt and another to decrypt. Thus, Mohammed propose to Ayoub an encryption key, with which it would encrypt the message, and Mohammed decrypt it with a different key, the decryption key. The encryption key only allows you to encrypt, even Ayoub would be unable to decipher his own message with this key, only Mohammed can with his decryption key. As the encryption key only works in one direction, it creates secrets but not to disclose, and may be public, appearing in a directory or on a website.

Anyone who wants to send an encrypted message to Mohammed can take and use.

It must only be sure that nobody can calculate the decryption key from the encryption key. And that mathematical intuition is decisive.

If the idea of asymmetric encryption using public keys back to Diffie and Hellman, the realization of this idea came to Rivest, Shamir and Adleman. They found a mathematical solution to the RSA implementation.

A person wishing to communicate using this method must do the following:

1. Take two primes p and q .
2. Calculate $n = pq$.

3. Calculate $z = (p - 1)(q - 1)$. (This number is the value of the function $\phi(n)$, called Euler function, and we note that it gives the size of the multiplicative group modulo n , Z_n^*).
4. Take a small integer e , odd and prime to z .
5. Calculate the inverse of $e \pmod{z}$, that is to say d such that $ed = 1 \pmod{z}$. The modular arithmetic theorems assure us that, in our case, d exists and is unique.
6. A pair $P = (e, n)$ is a public key.
7. The triple $S = (d, p, q)$ is the private key.

Ayoub wants to send a message to Mohammed. it gets the public key of Mohammed on her Web site and proceeds through encryption of the message M to obtain the encrypted C as follows:

$$C = P(M) = M^e \pmod{n}$$

To obtain the plaintext T , Mohammed decrypt with the secret key as follows:

$$T = S(C) = C^d \pmod{n}$$

In fact it is quite logical:

$$\begin{aligned} S(C) &= C^d \pmod{n} \\ &= (M^e)^d \pmod{n} \\ &= M^{e \cdot d} \pmod{n} \\ &= M \pmod{n} \end{aligned}$$

The latter result, $= M \pmod{n}$ arises because e and d are inverse modulo n , it is demonstrated through the Fermat's little theorem.

IV. DIGITAL SIGNATURE

A. Introduction

The digital signature is a very concrete application of the asymmetrical cryptography which was invented in the middle of the Seventies.

Indeed modern cryptography is no longer limited to ensure confidentiality of information, but it can also authenticate them through the digital signature.

The digital signature is a mechanism to authenticate a message, i.e. to prove that a message really comes from a specific sender.

According to ISO 7498-2 on the security architecture for open systems, the definition of digital signature: “**data appended to a data unit, or cryptographic transformations of a data unit, enabling a recipient of prove the source and integrity of the data unit and protects against counterfeiting by the recipient**” [7].

The signature is made using the signer's private key, so all partners can check the signature using the public key. In all operational protocols, it is actually a hash, not the whole document is signed, for performance reasons; asymmetric algorithms are very resource-intensive.

The technique used to calculate the hash is the hash. The technique produces a message digest which is a small representation of the unique and complete message. Hash algorithms are one-way encryption algorithms, so it is impossible to find the original message from the digest. The main reason why it produced a digest of the message are:

1. The integrity of the message sent is preserved, and any alteration of the message will be immediately detected;
2. The digital signature will be applied to condense whose size is usually much smaller than the message itself;
3. Hash algorithms are much faster than any encryption algorithm (either public key or symmetric key).

The message digest is very probably unique in the sense that it is almost impossible to find two meaningful messages that occur simultaneously on the same digest. Therefore, the

probability that a message tampered produce the same digest as the original is virtually zero.

B. Principle of the digital signature

- M = set of messages to sign,
- S = set of signatures,
- K = set of keys

For a given key $k \in K$, a signature function: $M \rightarrow S$ and verification function:

$M \times S \rightarrow \{true, false\}$ such that for every message $m \in M$ and each signature $s \in S$ we have $V_k(m, s) = true \Leftrightarrow S_k(m) = s$.

B.1 RSA Signature

- $M = S = Z_n$, where n is the product of two primes p and q .
- $K = \{(n, e, d) \mid ed \equiv 1 \pmod{\phi(n)}\}$ n and e are public, d are secret.

The signature function is calculated by

$$s = S_k(m) = m^d \pmod{n}$$

Verification by computing $m' = s^e \pmod{n}$ and $V_k(m, s) = true \Leftrightarrow m = m'$

B.2 El Gamal Signature

Signature process

Choosing a prime number p .

Generator g of the multiplicative group Z_p^*

Choose an integer x between 0 and $p-1$

It calculates $y = g^x \pmod{p}$

The public key is (p, g, y)

The private key is x

To sign a message m :

Choose $k < p-1$ and the first random $p-1$.

Compute $r = g^k \pmod{p}$ and $s = k^{-1}(H(m) - xr) \pmod{p-1}$.

The signature of m is (r, s) .

To verify a signature:

Test if $0 < r < p$

Calculate $u = y^r r^{-s} \pmod{p}$ and $v = g^{H(m)} \pmod{p}$

Accept if $u = v$

B.3 DSA Signature (Digital Signature Algorithm)

An American standard (FIPS 186) dating from 1995 signature based on the principle of the ElGamal signature.

Process signature:

We selected the following parameters:

A prime number p

A second prime q dividing $p-1$

A generator g of the cyclic group Z_p^* of order q

An integer $x < q$, we calculates $y = g^x \pmod{p}$

The public key is (p, q, g, y) .

The private key is x

To sign a message m :

Choose an random integer $k < q$

Calculate:

$$r = (g^k \pmod{p}) \pmod{q}$$

$$s = k^{-1}(H(m) + xr) \pmod{q}$$

The signature of m is (r, s)

To verify the signature:

Verify that r and s are in the interval $[1, q-1]$

Calculate:

$$w = s^{-1} \pmod{q}$$

$$u = wH(m) \pmod{q}$$

$$v = rw \pmod{q}$$

$$z = (g^u y^v \pmod{p}) \pmod{q}$$

Accept if $z = r$

V. SECURITY OF EXCHANGE ON INTERNET

A. Introduction

The security protocol SSL / TLS is currently the main protocol used worldwide for secure exchange and online transactions (e-commerce, bank accounts, online auctions, electronic voting ...).

B. SSL Client Authentication with digital certificate X.509

An SSL session always begins with an exchange of messages called SSL handshake. The negotiation allows a server to authenticate the client using public key techniques, then allows the client and server to cooperate in creating symmetric keys used for rapid encryption, decryption, and detection of alteration of data during the following session. Eventually, the SSL handshake can also allow the client to authenticate to the server. Figure 1 illustrates this authentication.

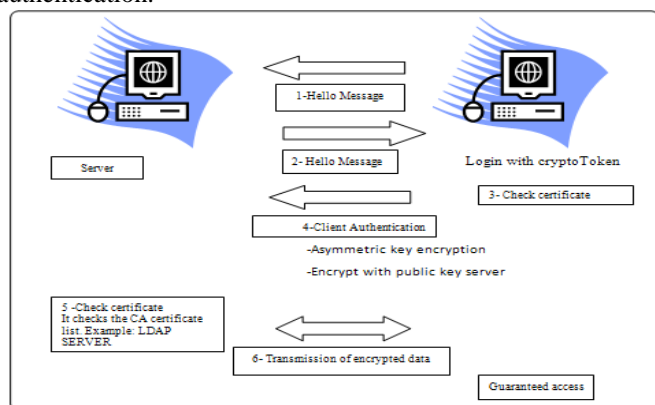


Figure 1: Authentication Process with an X.509 certificate

This authentication is performed by using an X.509 digital certificate issued by a certification authority (CA). But more and more web applications are now using authentication client by exploiting TLS. It is then possible to provide mutual authentication between the client and the server. The client certificate can be stored as software on the client or the physical format (smart card, USB token) to increase the security of TLS link. This solution can provide strong authentication.

VI. APPLICATION

In this work our aim is to test certain cryptographic algorithms as well as digital signature.

To make these algorithms accessible to users in a first step, we have developed a java application which provided the users the possibility of giving texts clear to encrypt or decrypt using four cryptographic algorithms DES, AES, Blowfish, and RSA, to verify the integrity of files with MD5 hash functions (message digest 5), SHA1 and SHA256 or signing it with DSA (Digital Signature Algorithm).

Then we discussed the comparison of execution time of software implementations.

Finally, to test the functioning of X.509 certificates used in SSL we wrote a program that just lets you use the SSL functionality.

VII. CONCLUSION AND PRESPECTIVE

Cryptography is an area that attracts increasing attention of research groups.

Indeed, the public key cryptography is very attractive and rich in perspectives, incorporating both encryption and digital signature. It is a real breakthrough compared to symmetric key cryptosystems.

Beyond the technical aspect, we must see the need to develop architecture or a PKI, which includes the tools needed to effectively manage and use keys and certificates. In this work, we first presented some ideas about cryptography and digital signatures and their uses to secure exchanges on internet. I wish in conclusion to mention a few related lines of work that I have unfortunately not had time to dig in my work: cryptography with elliptic curves and cryptanalysis.

For elliptic curves I think they are beginning to be known to a wider audience. Perhaps in a world dominated by the RSA public key cryptography, the latter eventually become a credible alternative.

Some ideas and prototype of the cryptosystem presented in this work remain to be completed. But the cryptographic concepts that we are developed permit to see more clearly the importance of encryption and digital signature in trade security in information systems.

Today, two types of encryption allow to secure digital exchange, however, is not to our knowledge of systems that combine these two techniques.

We conclude that both techniques are complementary and can be combined into a single system if we wish to obtain an encryption system not only efficient, but also respond to the needs and expectations of users.

VIII. REFERENCES

- [1] Ewelle Ewelle Richard ,TPE : Connectivité et sécurité des réseaux sans fils, Institut de la francophonie pour l'informatique, rapport final, Hanoi, Juillet – 2009
- [2] Laurent Bloch et Christophe Wolfhugel ,Sécurité informatique Principes et méthode, éditions Eyrolles 2007.
- [3] Cédric Llorens , Laurent Levier et Denis Valois ,Tableaux de bord de la sécurité réseau, éditions Eyrolles ,2ème édition 2006.
- [4] CGI, Étude technique : Cryptographie à clé publique et signature numérique Principes de fonctionnement, Septembre 2002 .
- [5] Mohammed C Kocher, Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems, Advances in Cryptology - CRYPTO'96,LectureNotes in Computer Sciences, Springer, 1996.
- [6] David Brumley et Dan Boneh, Remote Timing Attacks Are Practical, 12th USENIX Security Symposium, 2003.
- [7] La Lettre d'ADELI n°46 ,Signature cryptographique : du numérique à l'électronique , Janvier 2002.
- [8]Sammy POPOTTE-Laboratoire SUPINFO des technologies Microsoft ,Présentation d'IPSEC dans un environnement Windows 2000,
- [9] National institute of standards and technology (NIST),Advanced Encryption standard (AES) Conference, (Rome, Italy), March 1999.
- [10] National institute of standards and technology (NIST),Advanced Encryption standard (AES), Federal Information Processing Standards (FIPS) publication197,2001.