Universe Scientific Publishing

RSA Signature

Shengkai Shen

University of Toronto 190 **Borough Dr., Scarborough, Ontario, Canada M1P 0B6**

Email: shengkai.shen@mail.utoronto.ca

Abstract:RSA method has been a staple of cryptology for centuries. It is the corner stone of modern encryption method. This study explores how RSA encryption method has affected people's life, the use of RSA and the fundamental theory of RSA encryption. The report includes the history, development, application and future application and development of the RSA encryption method. In recent years, online banking, shopping, cloud technology has become wildly excepted and grow in popularity. In particular, the past two years had forced large population to incorporate internet activities to their norm. Most people have been using encryption technology without realizing or understanding it, which sometimes leads to information leaking or financial invader. By understanding the inner workings of the encryption process, people have a better chance of protecting themselves.

Key words:RSA, cryptology, electronic signature, RSA application, encryption, cyber security

1.Introduction

Since humans stepped into the information age around a hundred years ago, dissemination of information becomes much easier especially after the computer microminiaturization [1], the modernized information and communication processes became the driving force of our social evolution. As technology evolves, attacks on digitalized information are growing right along the side of it. Compared to the traditional way of disseminating information, digital information is more venerable to attack, forgery, and lacking authority. Along with the advancement of information technology, people have developed many methods to protect and validate important information. Some ways to ensure security like password protection or VPN are more well-known than other methods, like digital signatures. Overtime standards for security methods have become stricter. At present, one of the most used methods to ensure information security and validity are digital signatures generated by NIST- approved digital signature algorithms: DSA, RSA, and ECDSA [2].

2.Background

In this section, first I will briefly introduce digital signatures. Then, I will describe the history of RSA signature, the working process of RSA, Attack on and Security of RSA, and briefly discuss RSA applications. The knowledge of these concepts is necessary to understand the technical aspects of this project.

2.1 Digital Signature

In the most general term, a digital signature act as an electronic analog of a written signature, which provides assurance that the claimed signatory signed the information, and the information was not modified after signature generation [2]. In addition, digital signatures can be used to detect unauthorized modifications to data and to authenticate the identity of the signatory when the validity of the signature is questioned. In fact, digital signatures provide non-repudiation, which ensures the signatory cannot easily repudiate the signature after signing [3].

Digital signature is one of the Applications for Public-Key Cryptosystems. A digital signature scheme and key

generation algorithm are used to generate digital signatures. Through the digital signature signing process, signature generation algorithm [3] and private key are applied to the data that is formatted into a signable message to produce a signature. This signature is a data string for the message. This data string is a number dependent on the private key known only to the signatory, and the message being signed. The associated verification algorithm [3] is used in the verification process along with a method for recovering data from the message. The verification process and the public key are used by a verifier to verify the authenticity of the signature [3].

In practice, digital signature schemes are generalized into two classes:

1. Digital signature schemes with appendix require the original message as input to the verification algorithm.
2. Digital signature schemes with message recovery do not require the original message as input to the verification algorithm.

The first class is digital signature schemes with appendix relies on cryptographic hash functions, they are the most commonly used in practice because are less prone to existential forgery attacks. Some examples of mechanisms providing digital signatures with appendix are the DSA, El Gamal, and Schnorr signature schemes. The second class has the feature that the message signed can be recovered from the signature itself. In practice, this feature is of use for short messages. Examples of mechanisms providing digital signatures with message recovery are RSA, Rabin, and Nyberg-Rueppel public-key signature schemes.

2.2 RSA Signature

The RSA scheme is currently the most widely accepted and implemented a general-purpose approach to public-key encryption which developed by Ron Rivest, Adi Shamir, and Len Adleman at MIT in 1977 and first published in 1978. The Rivest- ShamirAdleman (RSA) scheme is the first cryptographic algorithm that responded to the new approach to cryptography that met the requirements for public-key systems. Despite RSA signature scheme is the first method discovered, it remains today one of the most practical and versatile techniques available.

Among the Public-Key algorithm families, RSA belongs to the Integer-Factorization Schemes algorithm family. In this family, public-key schemes are based on the fact that it is difficult to factor large integers. In particular, RSA relies on the practical difficulty of factoring the product of two large prime numbers, which is known as the “factoring problem”. When applies RSA encryption by itself, breaking the encryption is as difficult as the factoring problem. In fact, there are no published methods to defeat the system if a large enough key is used.

There are pros and cons to using RSA scheme. Since the mechanism of RSA scheme is to slice the original text into blocks that meet the required size, encrypt each block with the public key, and use the private key to decrypt the message. Therefore, an RSA signature has the feature that the message signed can be recovered from the signature itself. But notice that RSA is a relatively slow algorithm, which means RSA scheme is not commonly used to directly encrypt user data. Also, from the security point of view, although directly using RSA scheme is safe from algorithmic attacks with large enough keys; it is vulnerable to other attacks, such as existential forgery.

In practice, RSA is mainly used for encryption of small pieces of data, such as key transport, and digital signatures. To ensure the validity of a signature, two aspects of the process needs to be managed. First is during the key pair generation the length of the modulus for RSA should be: 1024, 2048, and 3072 bits [3]; afterward, the key pairs need to be protected and managed follow the key pair management [3] guideline. Then the signing process is that the message is first hashed to produce a short digest, which is then padded, then signed with RSA. When these two standers meet, the signature is safe against both algorithmic attacks and forgeries.

3.RSA scheme

This section is dedicated to explaining the operation process of RSA, And an example is provided for a clear understanding of RSA scheme.

In general, the RSA algorithm involves three steps: key generation, encryption, and decryption.

Key generation:

1. Select p, q, where p and q are both prime, $p \neq q$
2. Calculate $n = pq$
3. Calculate $\Phi(n) = (p-1)(q-1)$
4. Select integer e, where $\gcd(\Phi(n), e) = 1; 1 < e < \Phi(n)$
5. Calculate $de \equiv 1 \pmod{\Phi(n)}$
6. Public key, $PU = (e, n)$
7. Private key, $PR = (d, n)$

Encryption:

1. Plaintext: $m < n$
2. Ciphertext: $c = m^e \bmod n$

Decryption:

1. Ciphertext: c
2. Plaintext: $m = c^d \bmod n$

When using RSA scheme to generate a signature, one needs to consider the format of the message. Since RSA algorithm only takes integers between 0 and n-

1. Therefore, OS2IP – Octet-String-to-Integer primitive (OS2IP) is used to convert a string into an integer which can be processed by the encryption algorithm; and Integer-to-Octet-String primitive (I2OSP) is utilized to convert integer to string such at verifier can verify the signature in string form if needed.

I will use an example taken from The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography and show it in Figure 1 to explain the process.

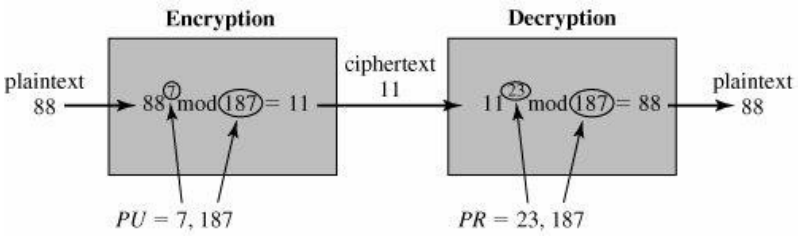

Figure 1. Example of RSA Algorithm

For this example, the keys were generated as follows:

1. Select two prime numbers, p = 17 and q = 11.
2. Calculate $n = p*q = 17 \times 11 = 187$.
3. Calculate $\Phi(n) = (p-1)*(q-1) = 16 \times 10 = 160$.
4. Select e such that e is relatively prime to $\Phi(n) = 160$ and less than $\Phi(n)$

we choose e = 7.

5. Determine d such that $de \equiv 1 \pmod{160}$ and $d < 160$. The correct value is d = 23, because $23 \times 7 = 161 = 10 \times 160 + 1$.

From above we get the resulting keys are public key PU = (e, n) = (7, 187), and private key PR = (d, n) = (23, 187). The example shows the use of these keys for a plaintext input of m = 88.

For encryption, we need to calculate $c = 887 \bmod 187$. Exploiting the properties of modular arithmetic, we can do this as follows:

887 mod 187 = [(884 mod 187) x (882 mod 187) x (881 mod 187)] mod187
881 mod 187 = 88
882 mod 187 = 7744 mod 187 = 77
884 mod 187 = 59,969,536 mod 187 = 132
887 mod 187 = (88 x 77 x 132) mod 187 = 894,432 mod 187 = 11
There for the encrypted data is c = 11.

For decryption, we calculate m = 1123 mod 187:
1123 mod 187 = [(111 mod 187) x (112 mod 187) x (114 mod 187) x (118 mod 187) x (118 mod 187)] mod 187
111 mod 187 = 11
112 mod 187 = 121
114 mod 187 = 14,641 mod 187 = 55
118 mod 187 = 214,358,881 mod 187 = 33
1123 mod 187 = (11 x 121 x 55 x 33 x 33) mod 187 = 79,720,245 mod 187 = 88
There for the original text m = 88.

3.1 Computational Aspects

We now turn to the issue of the complexity of the computation required to use RSA. There we mainly focus on the process of encryption and decryption. Both encryption and decryption in RSA involve raising an integer to an integer power, mod n. If the exponentiation is done over the integers and then reduced modulo n, the intermediate values would be gargantuan. Fortunately, by application of a property of modular arithmetic: [(a mod n) x (b mod n)] mod n = (a x b) mod n. We can reduce intermediate results modulo n, which makes the calculation practical.

To further speed up the encryption operation, a smaller e is usually selected. The most common choice is 65537 (216 1); two other popular choices are 3 and 17. Unlike the encryption process, we cannot choose a small constant value of d for efficient decryption operation. A small value of d is vulnerable to a brute-force attack and to other forms of cryptanalysis. However, one can use the Chinese remainder theorem (CRT) and Fermat's theorem to speed up the computation process. Without going into too much detail, the result is that the calculation is approximately four times as fast as evaluating M = Cd mod n directly.

3.2 Security Aspects

The security of the RSA cryptosystem is based on two mathematical problems: the problem of factoring large numbers and the RSA problem. We can identify three approaches to attack RSA mathematically: factor n into its two prime factors, determining Φ(n) directly, and determining d directly.

Currently, the most promising approach to solving the RSA problem is to factor the modulus n. Determining Φ(n) given n is equivalent to factoring n.

With presently known algorithms, determining d, given e and n appear to be at least as time-consuming as the factoring problem.

To factor the modulus n, an attacker factor n into p and q, and computes l cm (p − 1, q − 1) that allows the determination of d from e. Although it is thought to be infeasible on the assumption, no polynomial-time method for factoring large integers on a classical computer has yet been found.

Therefore, we can conclude RSA signature scheme meets all three requirements for being an effective public-key encryption.

4.Conclusion

As I have said in the introduction, I believe digital signatures already sipped into every aspect of people's daily life. Considering the case of the dematerialization process occurs in the Public Sector, where paper documents should disappear, and long-term traditional archives should be digitalized by ensuring authenticity and integrity of documents by means of qualified electronic signature. In general, people expect that in e-government applications, and also in transactions between citizens and companies, the use of qualified electronic signatures will always keep increasing in the future.

In this paper, I have described my study on RSA signature in-depth. Because of the properties of RSA scheme, RSA signatures are mostly applied with hash function and padding. The use of smart cards is a great example of RSA signature application. Now, when considering RSA signature by itself, I believe its future may be limited. One of my concerns is the security of the RSA signature; although I have stated previously it is infeasible to break RSA signature, the continuing increase in computing power, and the continuing refinement of factoring algorithms poses threats to large key size. Considering with the limitation of the RSA scheme, I conclude, compared to RSA scheme, elliptic curve cryptosystems may be more popular in the future.

References

[1] Kluver, R. Globalization, Informatization, and Intercultural Communication. un.org. Retrieved 18 April 2013.

[2] Computer Security Division, Information Technology Laboratory. "Digital Signatures: CSRC." CSRC, csrc.nist.gov/projects/digital-signature, 22 June 2020.

[3] National Institute of Standards and Technology. "Digital Signature Standard (DSS)." CSRC, 19 July 2013, csrc.nist.gov/publications/detail/fips/186/4/final.