IJARnD
INTERNATIONAL JOURNAL OF ADVANCE RESEARCH AND DEVELOPMENT

(Volume 4, Issue 9)
Available online at: www.ijarnd.com

Secure image transferring using asymmetric crypto system

Khet Khet Khaing Oo[1], Yan Naung Soe[2], Yi Mar Myint[3]
[1,2]University of Computer Studies, Myitkyina, Myanmar
[3]University of Computer Studies, Monywa, Myanmar

ABSTRACT

Nowadays, with the growth of technology, security is an important issue in communication. In many applications, the transmission of data (massage, image and so on) is needed to provide security against from preventing unauthorized users. Cryptography is a technique that provides secure communication. The two widely accepted and used cryptographic methods are symmetric and asymmetric. There are so many different techniques should be used to protect confidential image data from unauthorized access. This system provides secure image transmission between sender and receiver. This system uses Optimal Asymmetric Encryption Padding (OAEP) with RSA encryption algorithm to provide security during transmission. Only public-key cryptography is focused here. The public-key cryptographic algorithm relies on one key for encryption and a different but related key for decryption. In this system, different image formats (jpg, png, gif, and bmp) are used to test for encryption and decryption processes.

Keywords— *Cryptography, OAEP with RSA encryption algorithm, RHA-1*

1. INTRODUCTION

The requirements of information security within an organization have undergone tremendous changes. Advances in networking and communication technology bring the business organizations worldwide working together as one entity. Due to the impact of this globalization, vast amount of various digital documents such as texts, images, videos, and audio are sent from one destination to another via the Internet. However some of these documents might be sensitive and confidential and therefore it is needed to be protected. So, security is essential to transfer important information securely over the communication channels.

Cryptography plays a very vital role in keeping the message safe as the data is in transit. It ensures that the message being sent at one end remains confidential and should be received only by the intended receiver at the other end. Cryptography converts the original message into a non-readable format and sends the message over an insecure channel [6]. The people who are unauthorized to read the message try to break the non-readable message but it is hard to do it so. The authorized person has the capability to convert the non-readable message to readable one. The original message or the actual message that the person wishes to communicate with the other is defined as Plain Text. The message that cannot be understood by anyone or meaningless message is what we call as Cipher Text. Encryption is the process of converting plaintext into ciphertext with a key. A Key is a numeric or alphanumeric text or maybe a special symbol. Decryption is a reverse process of encryption in which original message is retrieved from the ciphertext. Encryption takes place at the sender end and Decryption takes place at the receiver end.

A system or product that provides encryption and decryption is called cryptosystem. Cryptosystem uses encryption algorithms that determine how simple or complex the encryption process will be, the necessary software component, and the key (usually a long string of bits), which works with the algorithm to encrypt and decrypt the data [7]. Thus, the security of an encryption system should rely on the secrecy of the encryption/decryption key instead of the encryption algorithm itself. The security level of an encryption algorithm is measured by the size of its keyspace [7]. Cryptographic algorithms are either symmetric algorithms, which use symmetric keys (also called secret keys), or asymmetric algorithms, which use asymmetric keys (also called public and private keys). There are several encryption algorithms used today. The most widespread algorithms today are RSA, ECC, TDES, AES, and RC6.

2. RSA PUBLIC KEY CRYPTOSYSTEM

RSA cryptosystem was developed by Ron Rivest, Adi Shamir, and Leonard Adleman at MIT in 1978 [4,6]. In cryptography, RSA is an algorithm for public-key cryptography. It is the first algorithm known to be suitable for signing as well as encryption, and one of the first great advances in public-key cryptography. RSA is widely used in electronic commerce protocols and is believed to be secure given sufficiently long keys and the use of up-to-date implementations. The RSA cryptosystem was first publicly presented in 1976 by Ron Rivest (whom we met before with the RC ciphers and MD series of hash function), Adi Shamir, and Leonard Adleman. They went on to patent the system (U.S. Patent 4,405,829) and to form a company of the same name—RSA Security [2].

2.1 RSA Algorithm

The RSA algorithm involves three steps: key generation, encryption and decryption [4].

2.1.1 Key Generation

RSA involves a public key and a private key. The public key can be known to everyone and is used for encrypting messages.

Messages encrypted with the public key can only be decrypted using the private key. The keys for the RSA algorithm are generated the following way: [1, 6]

Select **p, q**; where **p** and **q** are both prime, $\mathbf{p \neq q}$
Calculate $\mathbf{n=p*q}$
Calculate $\mathbf{\Phi\ (n) = (p-1)\ (q-1)}$
Select integer **e**; $\mathbf{gcd\ (\Phi\ (n), e) =1;\ 1<e< \Phi\ (n)}$
Calculate **d**; $\mathbf{d=e^{-1}\ mod\ \Phi\ (n)}$
Public Key; **KU = {e, n}**
Private Key; **KR = {d, n}**

2.1.2 Encryption

Plaintext**:** $\mathbf{M < n}$
Cipher text**:** $\mathbf{C = M^e\ mod\ n}$

2.1.3 Decryption

Cipher text**: C**
Plaintext**:** $\mathbf{M = C^d\ mod\ n}$

2.2 Pseudo-Random Number Generator (PRNG)

A Pseudo-Random Number Generator (PRNG), also known as a deterministic random bit generator (DRBG) [9], is an algorithm for generating a sequence of numbers that approximates the properties of random numbers. The sequence is not truly random in that it is completely determined by a relatively small set of initial values, called the PRNG's state. Although sequences that are closer to truly random can be generated using hardware random number generators, pseudorandom numbers are important in practice for simulations and are central in the practice of cryptography and procedural generation. Common classes of these algorithms are linear congruential generators, Lagged Fibonacci generators, linear feedback shift registers, feedback with carrying shift registers, and generalized feedback shift registers. Recent instances of pseudorandom algorithms include Blum Blum Shub, Fortuna, and the Mersenne twister. Careful mathematical analysis is required to have any confidence a PRNG generates numbers that are sufficiently "random" to suit the intended use [9].

3. OAEP

In cryptography, Optimal Asymmetric Encryption Padding (OAEP) is a padding scheme often used together with RSA encryption. OAEP was introduced by Bellare and Rogaway and subsequently standardized in PKCS #1v2 and RFC 2437 [8].

The OAEP algorithm is a form of Feistel network which uses a pair of random oracles G and H to process the plaintext prior to asymmetric encryption. When combined with any secure trapdoor one-way permutation f, this processing is proved in the random oracle model to result in a combined scheme which is semantically secure under chosen-plaintext attack (IND-CPA). When implemented with certain trapdoor permutations (e.g., RSA), OAEP is also proved secure against chosen ciphertext attack. OAEP can be used to build an all-or-nothing transform.

OAEP satisfies the following two goals:

- Add an element of randomness that can be used to convert a deterministic encryption scheme (e.g., traditional RSA) into a probabilistic scheme.
- Prevent partial decryption of ciphertexts (or other information leakage) by ensuring that an adversary cannot recover any portion of the plaintext without being able to invert the trapdoor one-way permutation f.

3.1 RSAES-OAEP

RSAES-OAEP is a public-key encryption scheme which combines the encoding method *Optimal Asymmetric Encryption Padding* (OAEP) with the RSA encryption primitive RSAEP. RSAES-OAEP takes a plaintext as input, transforms it into an encoded message via OAEP and applies RSAEP to the result which is interpreted as an integer using an RSA public key. The RSA was invented by Ronald L. Rivest, Adi Shamir, and Leonard Adleman, while the OAEP was invented by Mihir Bellare and Phillip Rogaway which was later enhanced by Don B. Johnson and Stephen M. Matya [3].

3.2 OAEP Encryption and Decryption

To achieve the best rendering both in the proceedings. The encryption operation produces a ciphertext from a message with a recipient's public key, and the decryption operation recovers the message from the ciphertext with the recipient's corresponding private key [3]. The following Figure 1 depicts OAEP encryption operation and decryption operation.
In the encryption operation:

- Pad the plaintext to make m-bit message M, if M is less than m-bit
- Choose a random number r of k-bits. (used only once)
- Use one-way function G that inputs an r-bit integer and outputs an m-bit integer. This is the mask.

$$P1 = M \oplus G(r)$$

- $P2 = H(P1) \oplus r$, function H inputs m-bit and outputs k-bit
- C = E(P1 || P2). Use RSA encryption here.
- In the decryption operation:

$$P = D\ (P1 \parallel P2)$$

- Receiver first calculate the value of r:

$$H(P1) \oplus P2 = H(P1) \oplus H(P1) \oplus r = r$$

- Receiver recovers the original message M:

$$G(r) \oplus P1 = G(r) \oplus G(r) \oplus M = M$$

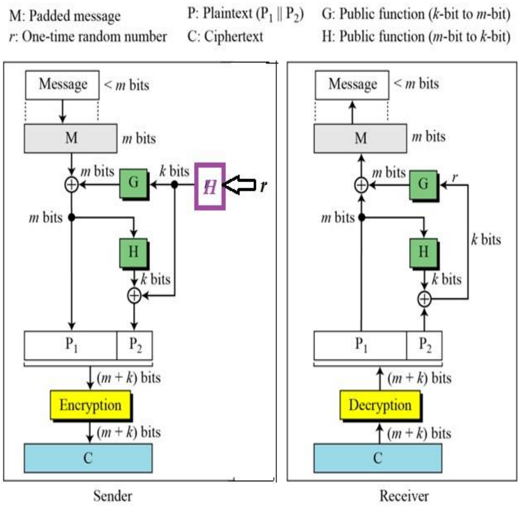

Fig. 1: Optimal Asymmetric Encryption Padding (OAEP)

4. SECURE HASH ALGORITHM (SHA-1)

Secure hash algorithm (SHA-1) was developed by NIST as a message digests function, which is necessary to ensure the security of the Elliptic Curve Digital Signature Algorithm (ECDSA). SHA-1 takes a message of length at most 2^{64} bits and produces a 160-bit output called message digest. The message digest is then inputting to the ECDSA, which computes the signature for the message. The same message digest should be obtained by the verifier of the signature when the received version of the message is used as input to SHA-1. So, both the sender and receiver of message computing and verifying a digital signature use the SHA-1 [1].

The SHA-1 is called secure because it is computationally infeasible to find a message corresponding to a given message digest or to find two different messages which produce the same message digest. Any change to a message in transit will, with a very high probability, result in a different the designing of the MD4 message-digest algorithm and is closely modeled after that algorithm.

5. DIGITAL IMAGE

A digital image is defined by an array of individual pixels and each pixel has its own value. The array, and thus the set of pixels, is called a bitmap. If an image is 512 pixels × 512 pixels, it means that the data for the image must contain information about 262144 pixels [5].

Digital images are produced through a process of two steps: sampling and quantization. Sampling is the process of dividing the original image into small regions called pixels, whereas quantization is the process of assigning an integer value (i.e. color) to each pixel [5].

5.1 Digital Image Format

Basically, there are three types of image files: bitmap, vector, and metafiles. When an image is stored as a bitmap file, its information is stored as a collection of pixels, manifest as colored or black-and-white dots. When an image is stored as a vector file, its information is stored as mathematical data. The metafile format can store image information as pixels (bitmap), mathematical data (vector), or both. There is no single format that is appropriate for all types of images. According to [5], larger files take longer to load, require more disk space and can take longer to print, whereas small file sizes mean greater performance. The most common file formats are JPEG, BMP, GIF, PICT and so on.

6. OVERVIEW OF THE SYSTEM

This system includes two parts the sender side and receiver side. In the key generation, the private key and public key are randomly generated by the RSA public key generation algorithm for each user. The public key is used in the encryption process and the private key is used in decryption process.

The sender side of the system is depicted in Figure 2. To create a secure image, the image is firstly converted into binary data. And then the binary data is divided into 1024-bit block. If the final block is less than 1024 bit, the 0's are padded this block until 1024-bit length. Then the seed value is hashed using SHA-1 hash algorithm which produces 160-bit length of the hash value (r) for Pseudo-Random Number Generation (PRNG). After generation, the random number is XORed with the padded blocks to produce plaintext P1. On the other hand, the P1 is hashed by SHA-1 to generate another hash value that is XORed with the hash value (r) to produce plaintext P2. Finally, P1 and P2 are concatenated and then the P1||P2 is encrypted by RSA public key algorithm with public key to generate ciphertext.

Figure 3 shows the receiver side of the decryption process flow diagram. In this process, firstly the ciphertext is decrypted using RSA with private to produce P1||P2. The P1||P2 is split to recover P1 and P2. Then P1 is hashed with SHA-1 and the hashed value is XORed with P2 to produce the hashed value (r) for PRNG. After random number has been generated, the P1 and the random number are xored to produce the padded message. Finally, the padded message is unpadded and then this message is recovered to the original image

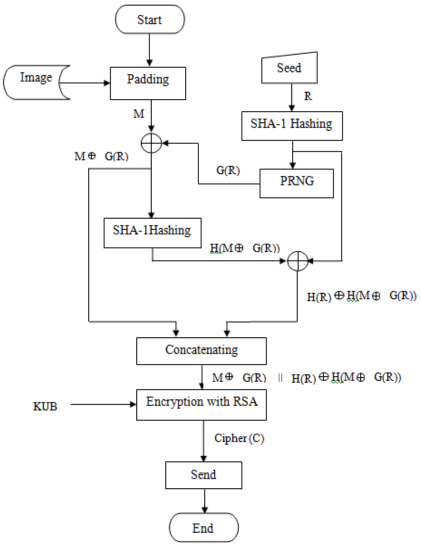

Fig. 2: Flow chart of sender-side

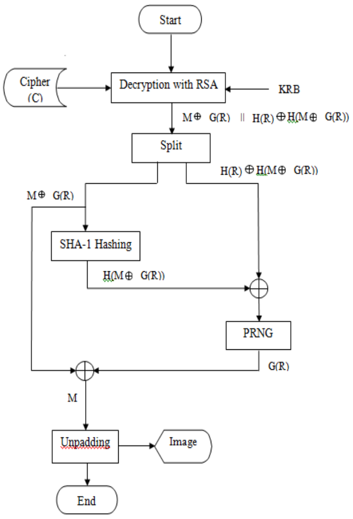

Fig. 3: Flow chart of the receiver side

7. EXPERIMENTAL RESULTS

The experimental results for encryption and decryption using OAEP algorithm are as shown in Figure 4. The different image file type and different file sizes are used to test for both processes. According to experimental results, the encryption processing time is faster than the decryption process in this system.

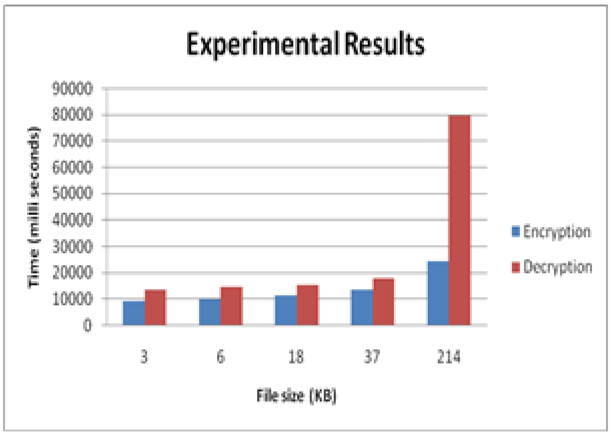

Fig. 4: Experimental results for the proposed system

8. CONCLUSION

The role of image encryption is more and more important in today's multimedia world. There are many encryption schemes for digital images. This system is mainly intended to encrypt/decrypt the images for secure image transferring system using the Optimal Asymmetric Encryption Padding (OAEP) with RSA. Encryption/decryption times may differ depending on the size of the images. This system provides secrecy and data confidentiality by using OAEP with a 1024-bit RSA public-key cryptosystem. It can be supported for different image files (jpg, png, gif, and bmp). This system limits the file size, especially under 250KB. So this implemented system is suitable for applications where it requires security based on the environment such as E-commerce, E-mail and so on.

9. REFERENCES

[1] Bruce Schneier, “Applied Cryptography, Second Edition: Protocols, Algorithms and Source Code in C”, ISBN: 0471128457.

[2] Narasimham Challa and Jayaram Pradhan, “Performance analysis of public-key cryptographic systems RSA and NTRU”.

[3] Neha Garg, “Comparison of Asymmetric Algorithms in Cryptography”, 2014.

[4] Nick Galbreath, “Cryptography for Internet and Database Application Developing Secret and Public Key Technique with Java TM”.

[5] Mohammad Alimoh’d bani Younes, “An Approach to Enhance Image Encryption Using Block-based Transformation Algorithm”, 2009.

[6] William Stalling, “Cryptography and Network Security”.

[7] White.B, Gregory Cisco Security Certification Exam Guide, McGrew-Hill, 2003.

[8] http://en.wikipedia.org/wiki/Optimal_asymmetric_encryption_padding

[9] https://www.princeton.edu/~achaney/tmve/wiki100k/docs/Pseudorandom_number_generator.html

BIOGRAPHY

Khet Khet Khaing Oo
Faculty of Computer System and Technologies,
2University of Computer Studies, Myitkyina, Myanmar

Yan Naung Soe
Faculty of Computer System and Technologies,
2University of Computer Studies, Myitkyina, Myanmar

Yi Mar Myint
Faulty of Information Science,
University of Computer Studies, Monywa, Myanmar