COMP 4109 - RSA Project Proposal

Danen Van De Ven
100820351

November 2015

1 Introduction

RSA is a popular public key cryptosystem. It uses three functions, key generation, encryption, and decryption, in conjunction with a public key and a private key to share a given message. RSA is one of the most widely used cryptosystems in the world and one of the best known [1, 6]. It can be used for both encryption/decryption or signature and verification. The system is built upon the difficulty of factoring large numbers with some similarities to the Diffie-Hellman cryptosystem [1]. For this project, RSA will be implemented using OAEP padding and Chinese Remainder Theorem for decryption. RSA variants will then be compared to the standard.

2 History

Ron Rivest, Adi Shamir, and Leonard Adleman, of MIT, are responsible for publicly describing the algorithm in 1977. However, Clifford Cocks had developed a similar algorithm in 1973 but it remained classified until 1997 [5]. Being a mathematician, Adleman was tasked to find weaknesses in Rivest and Shamir's one way functions. RSA was patented in the USA until the year 2000 [4].

Optimal Asymmetric Encryption Padding (OAEP) was introduced by Bellare and Rogaway. It is used as the standard in PKCS#1 v2. [2]. OAEP's original version was developed in 1994.

3 RSA

3.1 What is RSA?

The RSA scheme is a block cipher where the plaintext and ciphertext are integers between 0 and $n - 1$ for some n [1]. The typical size for n is 2^{1024} at minimum. RSA provides both encryption/decryption or signature and verification.

3.2 How does it work?

For some plaintext block M, and some ciphertext C, encryption and decryption can be defined as the following [1]:

$$C = M^e \bmod n$$
$$M = C^d \bmod n$$

Both sender and receiver must know the value of n.

RSA uses two key types, an RSA public key and an RSA private key, known as an RSA key pair. The RSA public key consists of two parts,

n	the RSA modulus, a positive integer
e	the RSA public exponent, a positive integer

For a valid RSA public key, n is a product of two or more odd primes r_i. The public exponent, e, is an integer where $3 < e < n-1$. e must satisfy $GCD(e,\ LCD(r_1 - 1, ..., r_u - 1))$, where u is the number of odd primes. r_1 and r_2 are also commonly referred to as p and q. The RSA private key can have two representations. A more complicated private key uses the Chinese Remainder Theorem (CRT) to represent the private key. A list of the variables used in this representation is provided excluding some of the details. All variables are positive integers [2].

p	the first factor
q	the second factor
dP	the first factor's CRT exponent
dQ	the second factor's CRT exponent
$qInv$	the first CRT coefficient
r_i	the i^{th} factor, where $i = 3, ..., u$
d_i	the i^{th} factor's CRT exponent
t_i	the i^{th} factor's CRT coefficient

In conjunction with the RSA key pair, multiple types of cryptographic primitives can be used depending on the purpose of RSA for encryption/decryption or signature and verification. This includes RSAEP/RSADP and RSASP1/RSAVP1 respectively. Combining these primitive, cryptographic schemes can be created to perform a specific security goal [2].

Part of combining these primitives and creating schemes is the use of padding rules for required real world security. We will be using RSAES-OAEP to encode a message and then encrypt using RSA. This method is probably secure in the random oracle model [3].

4 Standards Specifications

RSA meets PKCS#1, ANSI X9.31, and IEEE 1363 certification [3].

References

[1] N. Ferguson, B. Schneier, and T. Kohno. *Cryptography Engineering: Design Principles and Practical Applications*. Wiley, 2010. ISBN 9780470474242. URL `https://books.google.ca/books?id=GNbfjwEACAAJ`.

[2] R. Laboratories. Pkcs #1: Rsa cryptography standard, . URL `http://www.emc.com/emc-plus/rsa-labs/standards-initiatives/pkcs-rsa-cryptography-standard.htm`.

[3] R. Laboratories. Pkcs #11: Cryptographic token interface standard, . URL `http://www.emc.com/emc-plus/rsa-labs/standards-initiatives/pkcs-11-cryptographic-token-interface-standard.htm`.

[4] C. Paar and J. Pelzl. *Understanding cryptography: a textbook for students and practitioners*. Springer Science & Business Media, 2009.

[5] N. Smart. Dr clifford cocks cb. URL `http://www.bristol.ac.uk/pace/graduation/honorary-degrees/hondeg08/cocks.html`.

[6] S. William and W. Stallings. *Cryptography and Network Security, 4/E*. Pearson Education India, 2006.