African Journal of Mathematics and Computer Science Research Vol. 4 (6), pp. 213-216, June 2011
Available online at http://www.academicjournals.org/AJMCSR
ISSN 2006-9731 ©2011 Academic Journals

Review

One way functions and public key cryptography

Kenekayoro Patrick T.

Department of Mathematics/Computer Science, Niger Delta University, Wilberforce Island, P.M.B 071, Amassoma, Bayelsa State, Nigeria. E-mail: ktarila@gmail.com

Accepted 9 May, 2011

The problem faced with symmetric ciphers has always been key exchange. Hellman and Diffie (1976) proposed the idea for key exchange (public key cryptography). Their idea was based on the difficulty in inverting certain mathematical functions; one way functions. The RSA was one of the first practical solutions of key exchange. This paper identifies one way functions and their corresponding public key encryption system, with emphasis on the RSA.

Key words: Public key cryptography, RSA, one way functions, digital signatures, cryptanalysis, encryption, decryption.

INTRODUCTION

Security of data is important; encryption is used to do this. Early encryption systems like DES, AES (NIST, 2001) are examples of symmetric/secret key encryption systems. In these encryption systems, the key used to encrypt is the same as the key used to decrypt. The DES is still secure and used today (Kenekayoro, 2010), but the DES and all secret key cryptosystems from Enigma (German encryption system in WWII) to advanced encryption systems like AES have one weakness; key exchange. How keys are securely transmitted from sender to receiver was a major problem. Diffie and Hellman (1976) proposed the idea of public key cryptosystems, an encryption system where the key used to encrypt is different from the key used to decrypt. The paper suggested that public key cryptosystems can be achieved using one way functions, functions that are easy to solve but hard to reverse, hard in the sense that there are no efficient methods to solve them. Several one way functions have been found and are used in public key cryptography; this paper identifies these mathematical functions and their corresponding cryptosystems. Emphasis is laid on the RSA but other encryption systems are discussed.

PUBLIC KEY CRYPTOGRAPHY (PKC)

The concept of public key cryptography was invented by Whitfield and Hellman but there was no practical implementation until the RSA, Merkle also independently invented the concept of public key cryptography, although credit is given to Whitfield and Hellman. Using the traditional names used in cryptography “Bob”, “Alice” and “Eve” to describe the concept of public key cryptography, Bob places his encryption key in a public directory where everybody can see it and has a decryption key different from the public key that only he knows. If Alice wants to send a message to Bob, she checks the public directory for Bobs public key, encrypts the message; EB (M) = C and sends the cipher text C to Bob. If the message is intercepted by Eve she would not be able to decipher it (even Alice cannot decrypt the message) as only Bob knows the decryption key. When the message gets to Bob, he decrypts the cipher text with his decryption key (private key); DB (C) = M and reads the message. In public key cryptography as opposed to symmetric cryptography, Bob and Alice do not need any prior communication for key exchange, they do not need to share any secret (key), only Bob can decrypt messages encrypted with his public (encryption key) and vice versa. All public key cryptosystems security rests on a computational problem; the difficulty in inverting some mathematical functions. These functions are known as one way functions.

ONE WAY FUNCTIONS

It is a cliché that most things are easier to do than to undo, in mathematics, there is a formal term for this, one way functions (OWF). The famous axiom, the Axiom of Choice says that; every function f has an inverse g such that $f(g(x)) = x$ for x in the range of f. The axiom is accepted but proofs dependent on this axiom are being

singled out (Levin, 2003). Another set of mathematical functions, OWF; are those functions that are easy to solve in one direction but hard to invert. Easy is defined as functions that can be computed in polynomial-time, and hard defined as functions not computable in polynomial time. There is no proof that one way functions exist, proving this also proves the complexity class $P \neq NP$. Functions belonging to complexity type P, can be computed by a deterministic making machine in a time which is bounded above by some polynomial time, while functions belonging to complexity type NP (nondeterministic, polynomial) are problems computable in polynomial time on a "nondeterministic" computer (Diffie and Hellman, 1976). Examples of NP hard problems are the travelling salesman problem (Letchford, 2010), and other scheduling and optimization problems. These problems are hard to compute in worst case, but for a function to be used in cryptography, it should be hard to compute in average case.

ONE WAY TRAPDOOR FUNCTION

One way function, with certain unique information (trapdoor information), makes it easy to invert information. These functions are candidates for public key encryption systems; encryption systems where the key used to encrypt is different from the key used to decrypt. The forward operation of the mathematical function (encrypting) is easy but inverting this function (decrypting) is hard without knowledge of the trapdoor information. This trapdoor information can be seen as the private key. Finding such functions is difficult, several candidate functions have been identified, although not proven to be one way, no efficient way to invert has been found. If an efficient method to invert any of these functions is found, the encryption system that uses this function becomes useless. In the following sections, we discuss several one way trapdoor functions and their corresponding encryption systems.

RSA

The RSA, gotten from the names of the inventors "Rivest", "Shamir", Adelman" is one of the first public key encryption systems. The underlying hard problem known as the RSAP is based on the difficulty in factoring large integers.

The RSA algorithm

In the RSA algorithm, we have a public key (e, n) and a private key d. To encrypt a message M (create cipher text C), the message is raised to the e^{th} power modulon and to decrypt a cipher text C (go back to message M), the cipher text is raised to the d^{th} power modulo n:

$$C \equiv E(M) \equiv M^e \pmod{n}, \text{create cipher text}$$

$$M \equiv D(C) \equiv C^d \pmod{n}, \text{decipher cipher text}$$

(Rivest et al., 1978)

Prime numbers

A natural number is a prime number if it is divisible by only itself and one. A natural number that is not prime is called composite and 1(one) is neither prime nor composite. Prime numbers are important in cryptography for example, in the RSA you have to select two large primes in order to build a one way function and in the elliptic curve domain, a prime number p defines the field as well as the elliptic curve form.

It has been proven that there is an infinite number of prime numbers. In modern cryptography, large primes are often needed and as numbers get bigger it becomes harder to find prime numbers. Edson Smith, George Woltman and Scott Kurowski in 2008 found the largest known prime number. The project used volunteers' computers to hunt for prime numbers, the largest prime number found has 13 million digits ($2^{43,112,609} - 1$). It is a Mersenne prime that is, it can be written in the form 2^n-1.

To find prime numbers[1], we choose random odd numbers and perform a primality (is prime) or compositeness (not prime) test. All major primality testing algorithms for large numbers are probalistic. An example is the Fermat Little Theorem that is turned into a primality testing algorithm. The theorem states that "for any prime number p and any number a not divisible by p, the equivalence $a^{p-1} \equiv 1 \pmod{p}$ most hold". We can test the primality of a number n by randomly choosing a value not divisible by n for a, and computing $a^{n-1} \equiv 1 \pmod{n}$. If this value is not equal to 1, then n is definitive not a prime. However, there is a flaw in the algorithm, finding an 'a' for which $a^{n-1} \equiv 1 \pmod{n}$ does not imply that n is a prime number. Composite numbers for which $a^{n-1} \equiv 1 \pmod{n}$ is true for all 'a' are called Carmichael numbers. Because of this Fermat test is not widely used, other primality testing algorithms used are Solovay-Stassen test (Solovay and Strassen, 1977) and Miller-Rabin test (Rabin, 1980).

Generating RSA keys

To generate the public and private key pair for an RSA encryption system we need to:

[1] Eratosthenes of Cyrene 276BC-194BC discovered the first algorithm to find prime numbers, the Sieve of Eratosthenes.

1. Compute n as a product of two very large random prime numbers p and q. Although n is part of the public key, factors of n, p and q are kept secret.

$$n = p * q$$

2. Pick private (decryption) key d as a large random integer that is relatively prime to $(p-1)*(q-1)$, that is their greatest common divisor must be 1. The inventors of the RSA encryption system suggested that any prime number greater than max(p, q) would suffice for decryption key d, but it is important that d is chosen from a large enough set so that a cryptanalyst cannot find it by direct search:

$$\gcd(d, (p-1)*(q-1)) = 1$$

3. The encryption (public) key is computed to be the multiplicative inverse of d modulo $(p-1)*(q-1)$.

$$e * d \equiv 1 \left(mod\ (p-1)*(q-1)\right)$$

It is important to note that $if\ n = pq\ and\ p, q\ are\ prime\ numbers$, then Euler's totient function $\phi(n) = (p-1)*(q-1)$. $\phi(n)$ is the number of numbers that are smaller than n and are co prime with n. There is no efficient algorithm to factorize large integers and if the factors p and q of n are not known, there is also no efficient way to compute $\phi(n)$. The security of the RSA lies in this property.

Digital signatures

"Digital signature is a special application of cryptographic technology to assure the origin of a message and identity of the sender" (Brown, 1993). There are various cryptographic schemes to implement digital signatures and the RSA algorithm is first encryption system suitable for signing and encrypting documents.

For Bob to sign a message M he wants to send to Alice using the RSA public cryptosystem, he first computes the message signature "sign" using his decryption algorithm/key D_B, this makes sense because in RSA cryptosystem has a property that each message is a cipher text of some other message:

$$Sign = D_B(M)$$

After the message has been signed to assure the identity of the sender, he can now encrypt it with Alice's encryption algorithm/key for privacy:

$$Cipher = E_A(Sign)$$

When Alice gets the cipher text that she knows it's from Bob, she first decrypts the cipher text D_A to get "Sign" and encrypts the signature with Bob's algorithm E_B to get the message:

$$M = E_B\left(D_A(Cipher)\right)$$

With the message signature pair (M, Sign), Alice cannot modify the message because she will have to know Bob's private key to sign it and Bob cannot deny having sent Alice the message because only he knows his private key and can create:

$$Sign = D_B(M).$$

The RSA cryptosystem has been adapted to create several signature schemes, the most recent the Probalistic Signature Scheme (RSA-PSS) which follows the hash then sign paradigm (RSA Laboratories, 2009). It was invented by Phillip Rogaway and Bellare Mihir and patented in 2006.

Security of the RSA

There have been several attacks on the RSA encryption system and most of them are as a result of improper use of the encryption system. For example, the low private exponent where a small decryption key is chosen to reduce the decryption keys generation time or the low public exponent where a small public key is chosen to improve encryption or signature time.

When a low decryption key is used, the factorization of N can be found in polynomial time, and when a small encryption key is used, there are several attacks possible, for example the partial key exposure attacks (Boneh and Durfee, 1998).

These attacks can be avoided by using a very large random encryption/decryption keys. However, the adaptive chosen cipher text attack is not quite as a result of improper use of the RSA. Bleichenbacher was able to mount a practical attack on the RSA implementation of the Secure Socket Layer (SSL) (Bleichenbacher, 1998) using adaptive chosen cipher text attack. This attack however can be prevented using secure padding schemes like the Optimal Asymmetric Encryption padding (OAEP) (Bellare and Rogaway, 1994). The RSA-OAEP was proven to be secure under the RSA assumption (Eiichiro et al., 2004) and there have been several improvements thus strengthening the RSA-OAEP encryption system standard (Garg and Verma, 2009).

CONCLUSION

Key exchange has always been a problem for symmetric ciphers, the idea of public key cryptography by Whitfield and Hellman solved this problem. The idea is based on mathematical one way functions; that is functions that are easy to solve on one direction but hard to reverse. The RSA was the first encryption that gave a practical solution to this idea and the RSA can also be used to sign documents digitally (Digital Signatures).

The RSA when used with the Chinese remainder theorem RSA-CRT is theoretically known to be 4 times faster than the standard RSA and (Garg and Verma, 2009) proposed a scheme that is 56 times faster than the standard RSA. Although this is a significant improvement, symmetric encryption systems are faster than asymmetric encryption systems. In hardware, the DES is between 1000 and 10,000 times faster than the RSA (RSA Laboratories, 2009). As a result of this, if Bob wants to send a message to Alice, he encrypts the message with a symmetric encryption system like the AES and the symmetric encryption key with a public key cryptosystem. The encrypted message and encrypted key is sent to Alice. Alice can now decrypt the symmetric key with her private key and use the decrypted key to decrypt the message.

There are other public key encryption systems based on different computational problems. Rabin Crypto-system; based on the difficulty in finding square roots a modulo number (Rabin, 1979), Polly Cracker; the difficulty of Multivariate Polynomial Equation (Koblitz, 1998), Knapsack algorithm (Merkle and Hellman, 1982). The knapsack algorithm for PKC has now been broken. More popular encryption systems Elgamal Cryptosystem (Elgamal, 1985), Elliptic Curve Encryption System (Koblitz, 1987) are based on the discrete logarithmic problem.

The security of all public key encryptions systems rests on the underlying mathematical problem, if there is any leap in mathematics that finds an efficient way to solve any of these problems; the corresponding encryption system becomes useless.

REFERENCES

Bellare M, Rogaway P (1994). Optimal Asymmetric Encryption Padding -- How to Encrypt with RSA. In Advances in Cryptology-Eurocrypt ''94, pp. 92-112.

Bleichenbacher D (1998). Chosen Ciphertext Attacks against Protocols Based on RSA Encryption Standard PKCS#1. Advances in Cryptology -CRYPTO' 98 , pp. 1-12.

Boneh D, Durfee G (1998). Cryptanalysis of RSA with private key d<N.292. Proceedings of Eurocrypt' 98, pp. 1-11.

Brown PW (1993). Digital signatures: can they be accepted as legal signatures in EDI? Proceedings of the 1st ACM conference on Computer and communications security (pp. 86-92). New York: ACM.

Diffie W, Hellman M (1976).New Directions in Cryptography. IEEE Trans. Info. Theor. , pp. 644-654.

Eiichiro F, Tatsuaki O, David P, Jacques S (2004). RSA-OAEP is Secure under RSA Assumption. J. Cryptogr., pp. 81-104.

Elgamal T (1985). A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. Info. Theor., pp.469 - 472.

Garg D, Verma S (2009). Improvement over Public Key Cryptographic Algorithm. Patiala: Advance Computing Conference, IACC. IEEE International.

Kenekayoro PT (2010). The data encryption standard thirty four years later: An overview. Afr. J. Math. Comput. Sci. Res., 2(10): 267-269.

Koblitz N (1987). Elliptic Curves Cryptosystems. Math. Comput. 48: 203 - 209.

Koblitz N (1998). Algebraic Aspects of Cryptography. Berlin: Springer-Verlag.

Letchford AN (2010). The Travelling Salesman Problem. Swansea: Lancaster University.

Levin LA (2003). A tale of one way functions. Boston University, Boston.

Merkle RC, Hellman ME (1982). Hiding Information and Signature in Trapdoor Knapsacks. Secure Communications and Asymmetric Cryptosystems, pp. 197-215.

NIST (2001). Announcing the Advanced Encryption Standard (AES). Federal Information Processing Standards Publication.

Rabin MO (1979). Digitalized signatures and public-key function as intractable as factorization. MIT Laboratory for Computer Science.

Rabin MO (1980). Probabilistic algorithm for testing primality. J. Number Theor. 12 (1): 128-138.

Rivest R, Shamir A, Adleman L. (1978). A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Commun. ACM, 21: 120 - 126.

RSA Laboratories. (2009). RSA The Security Division of EMC. Retrieved April 20, 2011, from RSA The Security Division of EMC: http://www.rsa.com/rsalabs/.

Solovay R, Strassen V (1977). A Fast Monte-Carlo Test for Primality. SIAM J. Comput., 6(1): 84-85.