Public Key Cryptography

This document contains summary of some of the items discussed as part of the COMS 3110 Lectures. This document **does not replace the lecture materials**. This document may contain some topics that are not covered as part of the lecture; you will not be tested on those parts, they are made available to you for gaining further knowledge on topics/concepts that are related to class-lecture.

Public key cryptography is a cryptographic system that uses a pair of keys, a **public key** and a **private key**, to enable secure communication over untrusted networks.

- **Public Key**: Can be shared openly. Used to encrypt messages.
- **Private Key**: Kept secret by the owner. Used to decrypt messages.

A sender uses the recipient's public key to encrypt a message. Only the recipient, with the corresponding private key, can decrypt it.

RSA: Rivest-Shamir-Adleman (1977)

Key Idea: RSA relies on the difficulty of factoring large numbers, specifically the product of two large prime numbers.

Key Generation

1. Choose two large prime numbers P and Q.
2. Compute $N = PQ$. N is used as the modulus for both the public and private keys.
3. Compute Euler's totient function: $T = (p-1)(q-1)$.
4. Choose a public exponent e such that $1 < e < T$ and $\gcd(e, T) = 1$.
5. Compute the private exponent d such that $ed \mod T = 1$.

Public Key: (e, n)
Private Key: (d, n)

Encryption and Decryption

- **Encryption:** To send a message M, compute ciphertext $C \equiv M^e \mod N$.
- **Decryption:** To recover M, compute $M \equiv C^d \mod N$.

Correctness of RSA Algorithm: Modular Arithmetic

The correctness of RSA relies on the fact that decryption correctly recovers the original message, i.e., if a message M is encrypted as $C = M^e \mod n$, then the decryption yields

$$(C^d \mod N) = M.$$

We will use several foundational results from number theory and modular arithmetic to prove this claim. These results describe the behavior of integers under modular operations, particularly when working with prime moduli and multiplicative inverses.

Given integers x and $n > 0$, we write $x \mod n$ to denote the reminder obtained when x is divided by n.

In modular arithmetic, we say that two integers x and y are **congruent** modulo n if they have the same remainder when divided by n, i.e., $(x \mod n) = (y \mod n)$. In this case, we write

$$x \equiv y \pmod{n},$$

which means that $x - y$ is divisible by n. Congruences are preserved under exponentiation, i.e., if $x \equiv y \pmod{n}$, then $x^k \equiv y^k \pmod{n}$ for any integer k.

Given two integers x and y, $gcd(x, y)$ is the greatest common divisor of x and y. We say that two integers x and y are **coprime** if $gcd(x, y) = 1$.

The first result ensures the existence of a multiplicative inverse when the two integers are coprime.

Theorem 1. *If a and n are coprime, then there exist a unique integer b with $0 \leq b < n$ such that*

$$ab \equiv 1 \pmod{n}$$

*The integer b is called **multiplicative inverse** of a modulo n.*

Moreover, the multiplicative inverse can be efficiently computed using the Extended Euclid's Algorithm, which runs in time $O(n)$.

The following is known as **Fermat's little theorem** and plays a crucial role in RSA, particularly when dealing with prime moduli.

Theorem 2. *If p is a prime, then for every a,*

$$a^p \equiv a \pmod{p}$$

The following more general result builds on Fermat's Little Theorem.

Theorem 3. *If p is a prime, then for every L and a,*

$$a^{L(p-1)+1} \equiv a \pmod{p}$$

We now claim that RSA is correct.

Claim 1. *For any integers M, e, d, and any positive integer N, the following holds*

$$\left((M^e \mod N)^d\right) \mod N = M^{ed} \mod N$$

Proof. Let $C = M^e \mod N$. By definition, $C \equiv M^e \pmod{N}$. We now raise both sides of the congruence to the power d. Since congruences are preserved under exponentiation, we get

$$C^d \equiv (M^e)^d \pmod{N}.$$

Replacing C with $M^e \mod N$ and taking both sides modulo N again give

$$\left((M^e \mod N)^d\right) \mod N = M^{ed} \mod N$$

□

Claim 2. $C^d \mod N = M$

Proof. Since $C = M^e \mod N$, applying Claim 1, we get

$$C^d \mod N = M^{ed} \mod N.$$

Recall that $ed \equiv 1 \pmod{T}$, where $T = (P-1)(Q-1)$. This means that, $ed - 1$ is divisible by T, which implies that there exists $k > 0$ such that $ed = kT + 1$. Substituting this into the exponent gives

$$\begin{aligned} M^{ed} &= M^{kT+1} \\ &= M^{k(P-1)(Q-1)+1} \end{aligned}$$

.

Now, let $L = k(Q-1)$. Then, we get that $M^{ed} = M^{L(P-1)+1}$ and by Theorem 3,

$$M^{ed} \equiv M \pmod{P}$$

Similarly, by setting $L = k(P-1)$, we obtain

$$M^{ed} \equiv M \pmod{Q}$$

Thus, $M^{ed} - M$ is divisible by both P and Q. Since P and Q are distict primes, $M^{ed} - M$ must be divisible by $PQ = N$. Thus, we can conclude that

$$M^{ed} \equiv M \pmod{N}$$

Finally, since $M < N$, it follows that $M \mod N = M$. Thus, we can conclude that $M^{ed} \mod N = M$. This completes the proof that RSA decryption correctly recovers the original message. □

Note that the security of the RSA cryptosystem critically depends on the fact that a third party can not find d from e and N. Note that N is a product of two primes P and Q. If we can find P and Q, then it is easy to find d as d is a multiplicative inverse of e modulo $(P-1)(Q-1)$. Thus, if there is an efficient algorithm to factor N, we can break the RSA cryptosystem. To date, we do not know any efficient algorithm to compute factors of larger integers N. To ensure that RSA is secure, P and Q are picked to be large. This ensures that N is large.

Fun Fact

As we have seen, the RSA cryptosystem builds on clever usage of number theory. The famous mathematician G.H. Hardy once proudly described number theory as the "purest"" form of mathematics as it is almost wholly useless. In his 1940 essay *A Mathematician's Apology*, Hardy wrote

"I have never done anything 'useful'. No discovery of mine has made, or is likely to make, directly or indirectly, for good or ill, the least difference to the amenity of the world"

He viewed this "uselessness" as a mark of intellectual purity and superiority.

Little did Hardy foresee that, decades later, these once-"useless" number-theoretic principles would become the backbone of modern cryptography and secure online commerce!