The RSA Cryptosystem

Isabella Li
SUMaC Online
November 7, 2025

Contents

Historical Overview

Historical Overview

- In 1978, Ron Rivest, Adi Shamir and Leonard Adleman developed the RSA cryptosystem, hence the name RSA.
- It is a public-key asymmetric cryptosystem: the encryption key is public while the decryption key is only known to the decrypter.
- It can be used for digital signatures and key exchange; text is often converted to numbers in the form of ASCII codes.

Historical Overview

- RSA-250 (829 bits): Factored in roughly 2700 CPU years (i.e. 2700 years of computation on a single CPU) in 2020.
- RSA-2048 (2048 bits): Theoretically factorable, the time and resources required are currently astronomical for any practical purposes.
- Some systems that need higher security use 3072- or 4096-bit keys.

Mathematical Lemmas

Mathematical Lemmas

Lemma (Division Algorithm)

$\forall a, b \in \mathbb{N}, \exists\ q, r \in \mathbb{Z}$ *such that* $a = bq + r, 0 \leqslant r < b$.

Lemma (Euclidean Algorithm)

Let $a, b \in \mathbb{N}$ *with* $a > b$. *Then,* $\gcd(a, b) = \gcd(b, r)$ *where* r *is the remainder in* a *divided by* b. *Repeating the division algorithm, we get a sequence of remainders* $r_1, r_2, \ldots, r_n$ *such that* $r_n = 0$. *Then,* $\gcd(a, b) = r_{n-1}$.

Lemma (Multiplicative Inverse)

For $a, b \in \mathbb{N}$ *with* $a > b$,
$\gcd(a, b) = 1 \iff \exists\ a^{-1} \in \mathbb{Z}_b$ *such that* $aa^{-1} \equiv 1 \pmod{b}$.

Mathematical Lemmas

Lemma (Properties of φ)

For $x, y \in \mathbb{N}, \gcd(x, y) = 1$, $\varphi(xy) = \varphi(x)\varphi(y)$.

Moreover, for $n \in \mathbb{N}$, let the unique prime factorization of n be $n = \prod_{i=1}^{k} p_i^{e_i}$ for distinct primes p_i. Then, $\varphi(n) = n \prod_{i=1}^{k} \left(1 - \frac{1}{p_i}\right)$.

Lemma (Euler's Theorem)

For $a, n \in \mathbb{N}$ with $\gcd(a, n) = 1$, $a^{\varphi(n)} \equiv 1 \pmod{n}$.

Procedure

Procedure

In RSA, the sender and receiver will follow these steps:

1. Step 1: Key Creation. Nahida generates a public key (for encryption) and a private key (for decryption).
2. Step 2: Encryption. When Alhaitham wants to send a message to Nahida, he uses the public key to encrypt the message, and sends it over an insecure channel.
3. Step 3: Decryption. Nahida uses her private key to decrypt the received ciphertext.

$\longrightarrow$

$\longrightarrow$

Procedure

Step 1: Key Creation.

- Choose two distinct primes p, q.
- Compute $N = pq$ and $\varphi(N) = (p-1)(q-1)$.
- Choose e such that $\gcd(e, \varphi(N)) = 1$.
- N and e are the public key.
- Compute d such that $ed \equiv 1 \pmod{\varphi(N)}$. d is the private key.

Procedure

Step 2: Encryption.

- Suppose m is the message.
- $c = m^e \pmod{N}$ is the ciphertext.
- While it is easy to compute c from m, e, and N, it is computationally infeasible to recover m from c without knowing the private key d (i.e. multiplicative inverse of e modulo $\varphi(N)$).

Procedure

Step 3: Decryption.

- Compute $c^d = m \pmod{N}$ which is the plaintext.

Procedure

Numerical.

- When $N = 119, p = 7, q = 17$, $\varphi(N) = (p-1)(q-1) = 96$.
- Choose $e = 5$ because $\gcd(5, 96) = 1$.
- Compute d such that $5d \equiv 1 \pmod{96} \implies d = 77$.
- Suppose $m = 14$, then $c = m^e \equiv 14^5 \equiv 63 \pmod{119}$.
- We can verify that $c^d \equiv 63^{77} \equiv 14 = m \pmod{119}$.

Security and Examples

Security

- The security of RSA relies the factorization of large numbers. Currently there is no computer program that can factor large numbers in a reasonable amount of time (polynomial time).
- So, when creating N, we need to choose two large primes p, q such that N is hard to factor through existing algorithms.

Examples

We will demonstrate how different factorization methods for N can be used to break RSA making it exponentially quicker to compute d.

- Fermat's Factorization Method
- Basic Factorization Principle
- Exponentation Principle
- Pollard's $p - 1$ Method

Example 1: Fermat's Factorization Method

Definition (Nontrivial Factor)

A nontrivial factor of N is a divisor of N that is not 1 or N.

Consider $N = 5959$.

- Fermat's method works well when p and q are close. Try $a = \lceil\sqrt{5959}\rceil = 78$.
- Trying $78^2, 79^2, 80^2$, we find $80^2 - 5959 = 441 = 21^2$.
- So $N = (80 - 21)(80 + 21) = 59 \cdot 101$.
- $\varphi(N) = (59 - 1)(101 - 1) = 58 \cdot 100 = 5800$.
- Compute d such that $13d \equiv 1 \pmod{5800} \implies d = 2677$.
- Compute $m = c^d \bmod N = 4361^{2677} \bmod 5959$ with a calculator.

Example 2: Basic Factorization Principle

Let $N = 95$.

- Suppose we find $x = 12$, $y = 7$ which satisfy $x^2 \equiv y^2 \equiv 49 \pmod{95}$, but $x \not\equiv \pm y \pmod{95}$.
- $\gcd(12 - 7, 95) = 5$.
- So we can take $p = 5$ and $q = 19$ as two relatively prime nontrivial factors of N.

Example 3: Exponent Factorization

Let $N = 91$.

- Choose a base $b = 3$ (not a factor of N).
- Compute powers of b modulo N such that $\text{ord}_N(b) = y$ and make sure y is even.

$$
\begin{aligned}
3^1 &\equiv 3 \pmod{91} \\
&\vdots \\
3^4 &\equiv 81 \pmod{91} \\
3^5 &\equiv 61 \pmod{91} \\
3^6 &\equiv 1 \pmod{91}
\end{aligned}
$$

- So $y = 6$.

Example 3: Exponent Factorization (cont.)

- We want to rewrite $y = 2^k \cdot s$ for s odd.
- So $k = 1$, $s = 3$.
- Compute $b_0 = 3^3 \equiv 27 \pmod{91}$.
- Compute $b_1 = b_0^2 \equiv 27^2 = 729 \equiv 1 \pmod{91}$.
- So now the i such that $b_i \equiv 1$ for the first time is $i = 1$.
- So the algorithm tells us that a nontrivial factor of 91 is $\gcd(b_{i-1} - 1, 91) = \gcd(b_0 - 1, 91) = \gcd(26, 91) = 13$.

Example 4: Pollard's $p - 1$ Method

Let $N = 299$.

- We choose $b, C \in \mathbb{N}$ and hope $C!$ is a multiple of $p - 1$. Then, compute the a such that $a \equiv b^{C!} \pmod{n}$ and then compute $(a - 1, n)$, and repetitively compute a such that $(a - 1, n) = d \neq 1$ or n, then d is a nontrivial factor of n.
- Choose $b = 2,\ C = 5$. Compute $a = 2^{5!} \equiv 196 \pmod{299}$.
- Compute $\gcd(a - 1, N) = \gcd(195, 299) = 13$.
- So $p = 13$ and $q = 23$.

References

References

Kraft, James, and Lawrence Washington. An Introduction to Number Theory with Cryptography. CRC Press, 29 Jan. 2018.

Thank you!

Hope you enjoyed the presentation and thank you all for making SUMaC so memorable.