MA080G Cryptography Summary Block 2

Viktor Rosvall

April 22, 2019

Public-key cryptography

One of the problems public-key cryptography solves is the **key distribution problem** by using a distributed *public* key for encryption and a *private* key for decryption.

This works because encryption is done using a *One-way function*.

One-way function: a function $f()$ is a one-way function if:

1. $y = f(x)$ is computaitonally easy
2. $x = f^{-1}(y)$ is computaitonally impossible

This means that even if the public key used to encrypt a message is know, it can't be decrypted without the private key. [1]

Key Distribution Example [1]

Let's say Alice wants to send x to Bob. Both Alice and Bob have a public and private key-pair: $k = (k_{\text{pup}}, k_{\text{priv}})$.

Alice encrypts x using Bob's public key b_{pup}, as:

$$y = e_{b_{\text{pup}}}(x)$$

where e is a one-way function. Now Bob can decrypt the received message y using his private key b_{priv} and retrieve x, as:

$$x = d_{b_{\text{priv}}}(y)$$

We can send any data securely using this method. It's common to send key's for symmetric ciphers such as AES, since it's computationally heavy to use these computations.

RSA

Man-in-the-middle attack

Fermat's Little Theorem

Fermat's Little Theorem is useful in primality testing and in public-key cryptography. It can also be used for find the inverse of an integer a modulo a prime. [2]

Theorem: let a be an integer and p be a prime, then:

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

This can also be rewritten as:

$$a^{p-1} \equiv 1 \pmod{p}$$

If p is a prime then the inverse of a can be calculated as:

$$a^{-1} \equiv a^{p-2} \pmod{p}$$

Proof using modular arithmetic [3]

Let's assume a is a positive integer, not divisible by prime p. If we write down the sequence of numbers in modulo p

$$a, 2a, 3a, ..., (p-1)a$$

and after reducing each integer modulo p, we get the resulting sequence of numbers

$$1, 2, 3, ..., p-1.$$

Which means the two sequences are congruent modulo p

$$a, 2a, 3a, ..., (p-1) \equiv 1, 2, 3, ..., p-1 \pmod{p}$$

Which is the same as

$$a^{p-1}(p-1)! \equiv (p-1)! \pmod{p}.$$

After canceling out the sequence of both sides we get

$$a^{p-1} \equiv 1 \pmod{p}$$

Example

Let $a = 2$ and $p = 7$. The sequence of numbers thus is

$$2, 4, 6, 8, 10, 12$$

and after reducing each integer modulo p, we get

$$2, 4, 6, 1, 3, 5$$

reordered as

$$1, 2, 3, 4, 5, 6.$$

The two sequences are also congruent

$$\begin{aligned} 2, 4, 6, 1, 3, 5 &\equiv 1, 2, 3, 4, 5, 6 \pmod{p} \\ 2^6 6! &\equiv 6! \pmod{p} \\ 2^6 &\equiv 1 \pmod{p} \end{aligned}$$

Euler's generalization [2]

Euler's generalization of Fermat's Little Theorem allows any integer modulo m, instead of just modulo prime.

‘

> **Euler's Theorem:** let a and m be co-prime integers, i.e., $\gcd(a, m) = 1$, then:
>
> $$a^{\Phi(m)} \equiv 1 \pmod{m}$$

Example

Let $a = 3$ and $m = 8$. The $\gcd(3, 8) = 1$.
First we need to calculate $\Phi(8)$.

$$\Phi(8) = \Phi(2^3) = 2^3 - 2^2 = 4.$$

Now we can use Euler's theorem:

$$3^{\Phi(8)} = 3^4 = 81 \equiv 1 \pmod{8}$$

Compute the order of elements in Zp

Carmichael's lambda-function

prove the existence of primitive elements in Zp

complexity involved in Primality testing

Miller-Rabin probabilistic primality test

Pollard's p-1 factorisation method

References

[1] C. Paar, J. Pelzl, *Understanding Cryptography*. 2010 ed. Springer., Chapter 6.1

[2] C. Paar, J. Pelzl, *Understanding Cryptography*. 2010 ed. Springer., Chapter 6.3.4

[3] Wikipedia, ”Proofs of Fermat’s little theorem”, `https://en.wikipedia.org/wiki/Proofs_of_Fermat%27s_little_theorem` 18-04-2019