B08705021_part3 readme

盧德原 | B08705021 | 資管三

Environment

Language: C++ / C

OS: Ubuntu 18.04 (Parallel on macOS Monterey 12.1)

Complie

By typing `make` in terminal to run the makefile, an executable names **server** and **client** will be compiled and ready to be executed.

```
make
```

Then, type `./server <server IP address> <server port>` to run the server, for example,

```
./server 127.0.0.1 8888
```

After the server is started and ready, type `./client <server IP address> <server port>` to run the client and connect to the server, for example,

```
./client 127.0.0.1 8888
```

Then the server will wait for clients to connect, some commands and status changes will show on the terminal.

makefile code

```
output:
	g++ server.cpp -pthread -o server
	g++ client.cpp -pthread -o client
```

Encryption Approach

1: Creating Keys

Select two large prime numbers x and y, and compute `n = x * y`

Then, n is the modulus of private and the public key

Calculate totient function, `ø(n) = (x - 1)(y - 1)`

Choose an integer e such that e is coprime to ø(n) and 1 < e < ø(n). And e is the public key exponent used for encryption

Finally choose d, so that `d * e mod ø(n) = 1`

2: Encrypting Message

Messages are encrypted using the Public key generated and is known to all.

The public key is the function of both e and n.

If M is the message(plain text), then ciphertext

`C = M ^ n( mod n )`

3: Decrypting Message

The private key is the function of both d and n.

If C is the encrypted ciphertext, then the plain decrypted text M is

`M = C ^ d ( mod n )`

Reference

- 1
- 2
- 3