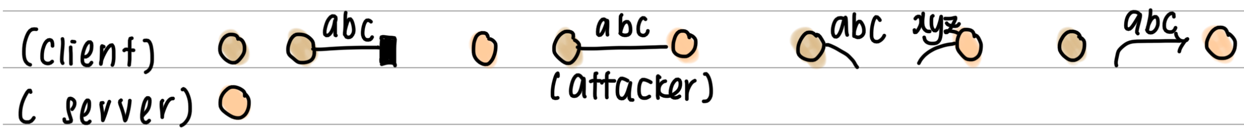
May

Security Taxonomy

taxonomy : Interraption, Interception, modification, Fabrication

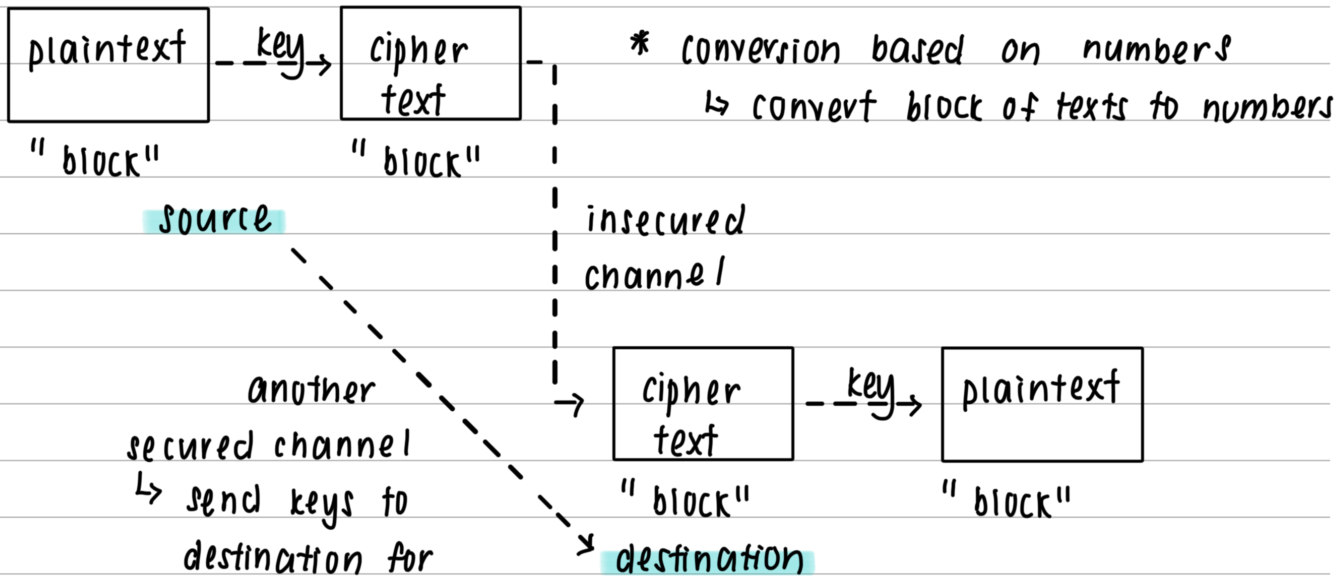
security properties : availability , privacy , integrity , authecity



Cryptography

- symmetric encryption
- public key cryptograph
- hashing algorithmn

Symmetric cryptography
↳ AES (eg. used in browsing server)



* conversion based on numbers
↳ convert block of texts to numbers

* key same key to be able to convert cipher text to plaintext

BUT!

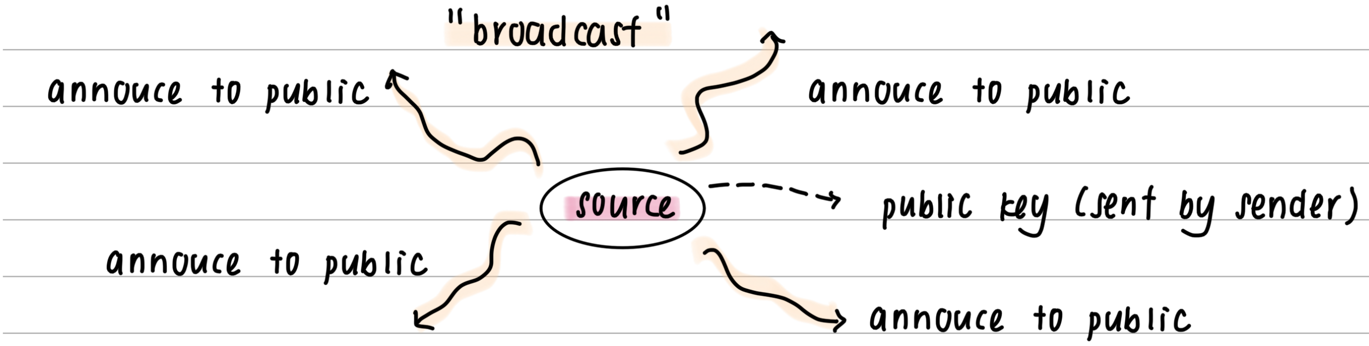
This method of sending keys from source to destination is unsecured
↳ if hacker found out there's another channel between, can hack the channel and get the key to convert cipher text to plaintext

(symmetric cryptography)
main problem : key exchange, confidentiality / privacy only,
scalibility ↳ X integrity, autencity
↳ $^{n}C_{r}$ ↳ X non-repiadility (?)
↳ $^{10}C_2$ vs $^{100}C_2$ vs $^{1000}C_2$
(45 keys) (4950 keys) (499520 keys)

public key cryptography
↳ asymmetric
↳ RSA

source · private key (encrypt)
· public key (decrypt)
↳ if private key is used to encrypt, public key is used to decrypt;
if public key is used to encrypt, private key is used to decrypt
↳ have a pair of keys
↳ can't use private / public key to encrypt and decrypt

* public key is annouced,
private key is kept



* if receiver wanna send a message to sender need use public key to encrypt message and send to original sender

"authencity" sender distribute certificate signed by sender's private key
↳ receiver with the certificate can access the server

Diffie - Hellman key Exchange (DHKE)
↳ exchange symmetric cryptography key through public key cryptography

Design RSA with $p=5, q=11$

① $p=5, q=11$ * normally millions = big numbers

② $N = p \times q = 55$

③ $\phi N = (p-1)(q-1)$

$= 4 \times 10$

$= 40$

④ public key, $e \longrightarrow 1 < e < 40$

$\gcd(e, 40) = 1$

eg. $e \neq 20$ cus $40/20 = 2$, $20/20 = 1$

so $\gcd(20, 40) = 20$

so... $e = 7$; $\gcd(7, 40) = 1$

e can also be $\Rightarrow$ 3, 11, 13 ... etc etc

⑤ private key, d

$$d = \frac{\phi(N)(k-1)+1}{e} = \text{integer}$$

$$d = \frac{40(k-1)+1}{7} \longrightarrow \frac{40(5-1)+1}{7} = 161/7 = 23$$

⋮

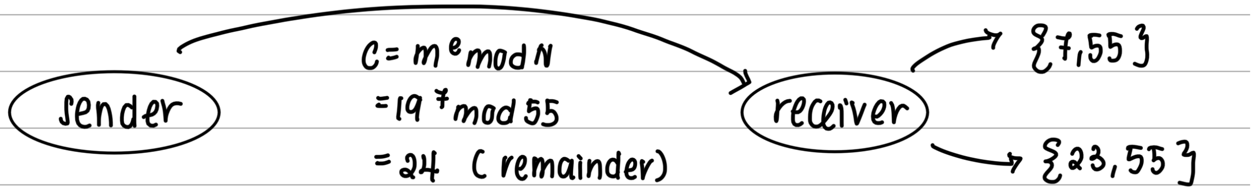
increase / decrease k value until d = integer (whole number)

⑥ $k_u = \{e, N\}$; $k_e = \{d, N\}$

$= \{7, 55\}$ $= \{23, 55\}$

Encrypt 19 using public key & decrypt using private key

* RSA $\Rightarrow$ $N > m$, larger N = encrypt larger message



$C = m^e \bmod N$

$= 19^7 \bmod 55$

$= 24$ (remainder)

* m = message = 19

* $e = 7$

* $N = 55$

$m = c^d \bmod N$

$= 24^{23} \bmod 55$

$= 24^5 \bmod 55 \times 24^5 \bmod 55 \times$

$24^5 \bmod 55 \times 24^5 \bmod 55 \times$

$24^3 \bmod 55$

54^4 ~~mod 55~~

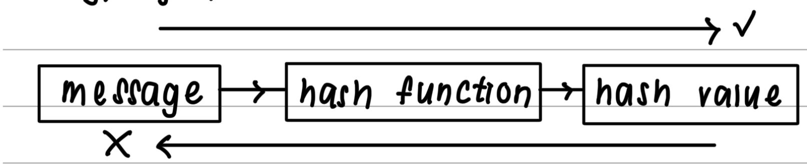
$24^3 \bmod 55$

$54^4 \times 19 \bmod 55 \leftarrow 54^4 \times 24^3 \bmod 55$

the message will do until 19

mod55 until lesser than 55

Cryptographic Hash Function



(fixed length)
↳ 256 bits

- * hash algorithmn
  - ↳ MUST BE FAST & EFFICIENT
  - ↳ GENERATE IMMEDIATELY
- * give hash value, original message can't be found
- * almost similar function have huge difference in hash value
- * same message = same hash value
- * collision resistant
  - ↳ 2 different message will not result in same hash value

Authencity (store password)	Integryty (store document)