Lec6_Authentication Protocols

problem with symmetric key:

- pre-share key with intended recipient, not practical
- make sure key is not intercepted, difficult
- 1 symmetric key by pair of sender, recipient, not scalable

Public key cryptography(asymmetric encryption)

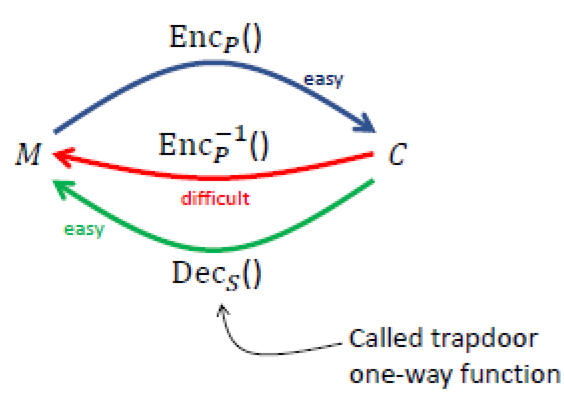
encryption/ decryption: sender encrypts a msg with recipient's public key

Digital signature: sender signs a msg with private key

Key exchange: 2 sides cooperate to exchange session key

public key P & secret key S

$$\mathrm{Dec}_S\left(\mathrm{Enc}_P(M)\right) = M$$



key pair generation

pk = KeyGen(sk)

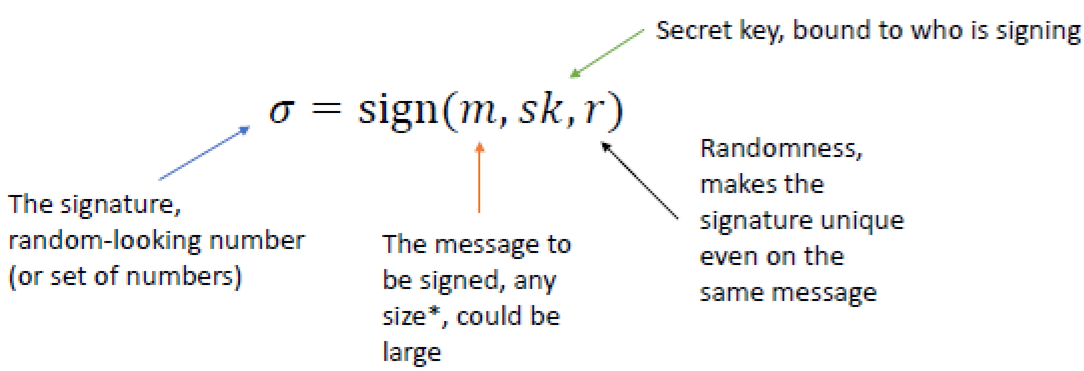
Process

1. Bob distribute his public key PB
2. Alice encrypt her msg using PB， C = Enc(PB, msg)
3. Send ciphertext C to Bob
4. Bob use his private key SB to decrypt Alice’s msg, msg = Dec(SB, C)

Man-in-the-middle

1. Fake intercept Bob’s public key(PC)
2. Fake send his own public key to Alice
3. Alice encrypt her msg for bob with fake public key C = Enc(PC, msg)
4. Alice end encrypted msg to Bob
5. Fake decrypt Alice ciphertext and learn msg, Dec(SC, C) = msg
6. Fake encrypt Alice msg using real Bob public key, C = Enc(PB, msg)
7. Fake send Alice newly encrypt msg to Bob
8. Bob decrypt msg as if nothing bad happened

Digital Signature



- signature is deterministic

Sign verify

Signature to verify, includes r somehow

Public key from the person who signed

$$\{T \mid F\} = \text{Verify}(m, \sigma, pk)$$

Message that was signed

- sign does not hide anything, require original msg to verify sign
- msg is public, msg can be ciphertext

Different public cryptosystem

RSA - encryption, digital signature, key exchange

Diffie-Hellman- no encryption/decryption

Diffie-Hellman using Elliptic Curve: no encryption/decryption

Function of signing

Authentication

- authentication of digital msg

Non-repudiation

- signer cannot claim they did not sign msg, private key remain secret

Integrity

- any change in msg after signature invalidate signature

Diffie-Hellman key exchange

- hard to compute
- no authentication

p - large prime integer

g - primitive root mod p

K(AB) = g^(ab) mod p

TLS 1.3

Security properties ensured by TLS

- confidentiality
- message authenticity
- authentication - server & client
- TLS do not enforce
  - non-repudiation & availability

Plaintext vs. ciphertext

HTTP → HTTPS

Overview

1. server request cert from CA
2. CA verify ownership of domain
3. CA issue cert
4. client initiate connection
5. server respond with server's certificate
6. client validate cert
7. client & server perform authenticated key exchange
8. client and server talk encrypted using key

Certificates

set of public keys and identification info

- provide integrity & authenticity guarantee
- source - trusted 3rd party
- root cert are self-signed

Certificate Authorities

- issue public-key X509 digital certificates that associate pub-key with an identified owner
- verify id before issue certificate
- cert on web are bind a domain name
- TLS secure communication

TLS Handshake