3. Cryptography - Certificate Authority

Wednesday, February 17, 2021 8:25 AM

Whenever someone sends a public key, the identity must be certified. Should be digitally signed.

- Certificate signed by certificate authority

Certificate verification:

1. Bob sends Alice his certificate
2. Contains name, organization, public key, date of expiry/digital signature appended by certificate authority (digest signed by private key of certificate authority)
3. Alice checks if certificate is tampered
4. Using the four fields, Alice computes a digest
5. Then decrypts the certificate digest using public key of CA (available everywhere)
6. If matches, no tampering

Exchange of data:

1. Alice sends public key to certificate authority along with credentials and gets a certificate
2. Bob sends public key to certificate authority along with credentials and gets a certificate
3. Both users exchange certificate
4. Alice verifies certificate and encrypts data with public key

Authentication using Digital Certificate:

1. Web service is subscribers-only: requires authentication
2. Can the client use certificate for authentication? Yes.
3. Certificate specifies unique identity of the person
4. Vulnerable to playback attack
   - Intruder may store certificate and replay it after a while
5. Web service sends encrypted (random) text with public key of client (received from certificate)
6. Client must decrypt using private key and send it to web service

Why random text/string?

- Random text to avoid replay by intruder

Certificate revocation: (why & how)

- Date of expiry
- Suppose there is an intruder between Alice and Bob
- They exchange certificates and encrypt and transfer data
- Data is modified (intruder may have stolen private key of A)

- Alice notes that her private key is compromised. Declare certificate as revoked and update information in server (for revoked certificates)
- Any user before verifying certificate must check the server
- Alice generates new private and public key
- Get certified by certificate authority