Public Key Infrastructure (PKI)

What do we need to establish trust?

- We need a way to **bind a public key to an identity**!

This requires:

1. A set of mechanisms, format and infrastructure to "manage digital identities"

   => Public Key Infrastructure (PKI) to issue and manage Digital Certificates (Cert)

2. A crypto tool to authenticate certificate?

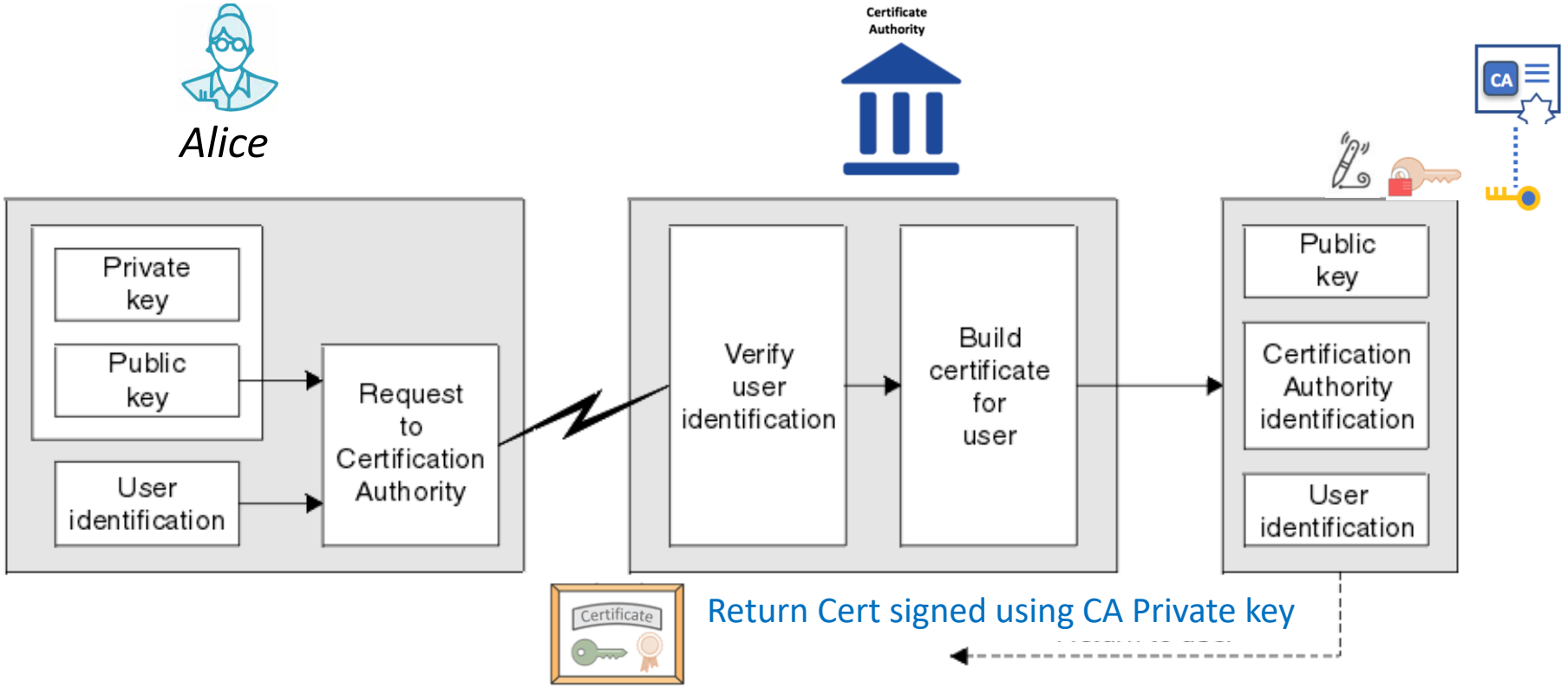
    => Digital Signature of Cert

Ensures that the public key is protected in transit

PKI

- Public Key Infrastructure (PKI) is a set of components, policies, and protocols needed to:
  - Issue, validate and revoke **digital certificates**
  - Support exchange suecure public keys to **establish trust**
  - Define standard format for certificates (X.509 format)
- The biggest PKI usage is secured websites using HTTPS

Obtaining a Digital Certificate



- To obtain a certificate you need to create a Certificate Signing Request (CSR) and sent it to a Certificate Authority (CA)
  - CSR is a digital document that contains your public key and other identifying information
- Once the requester identity has been verified (e.g., ownership of the domain mentioned in CSR), the CA creates a certificate and signs it with the CA private key
  - Anyone can now validate the certificate by checking its digital signature with the CA's public key

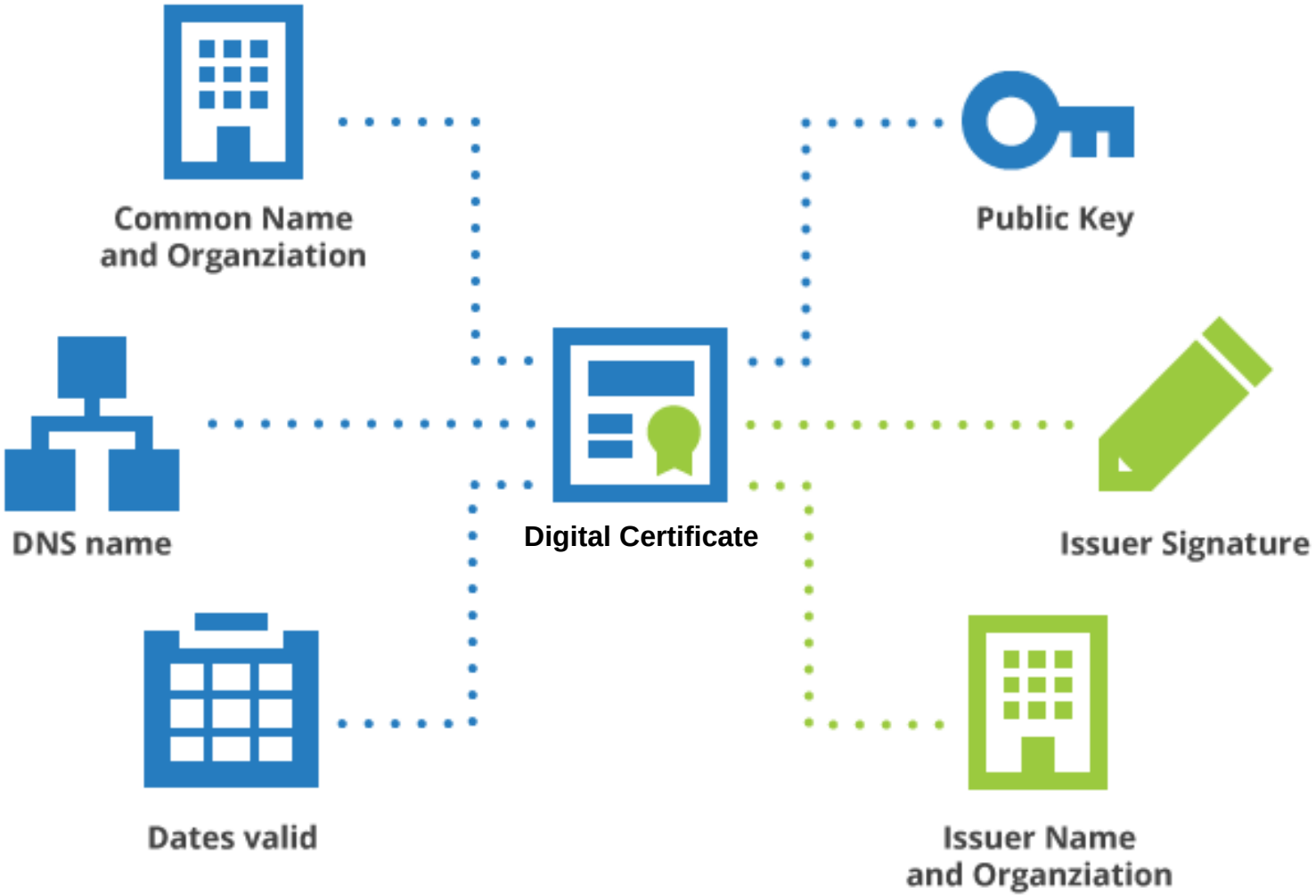
Digital Certificate

- A digital certificate is a document that binds a **public key** to the identity of the legitimate owner

→ Cert = {Owner(ID), Owner(Pub_{key}), CA digital signature of Cert**}**

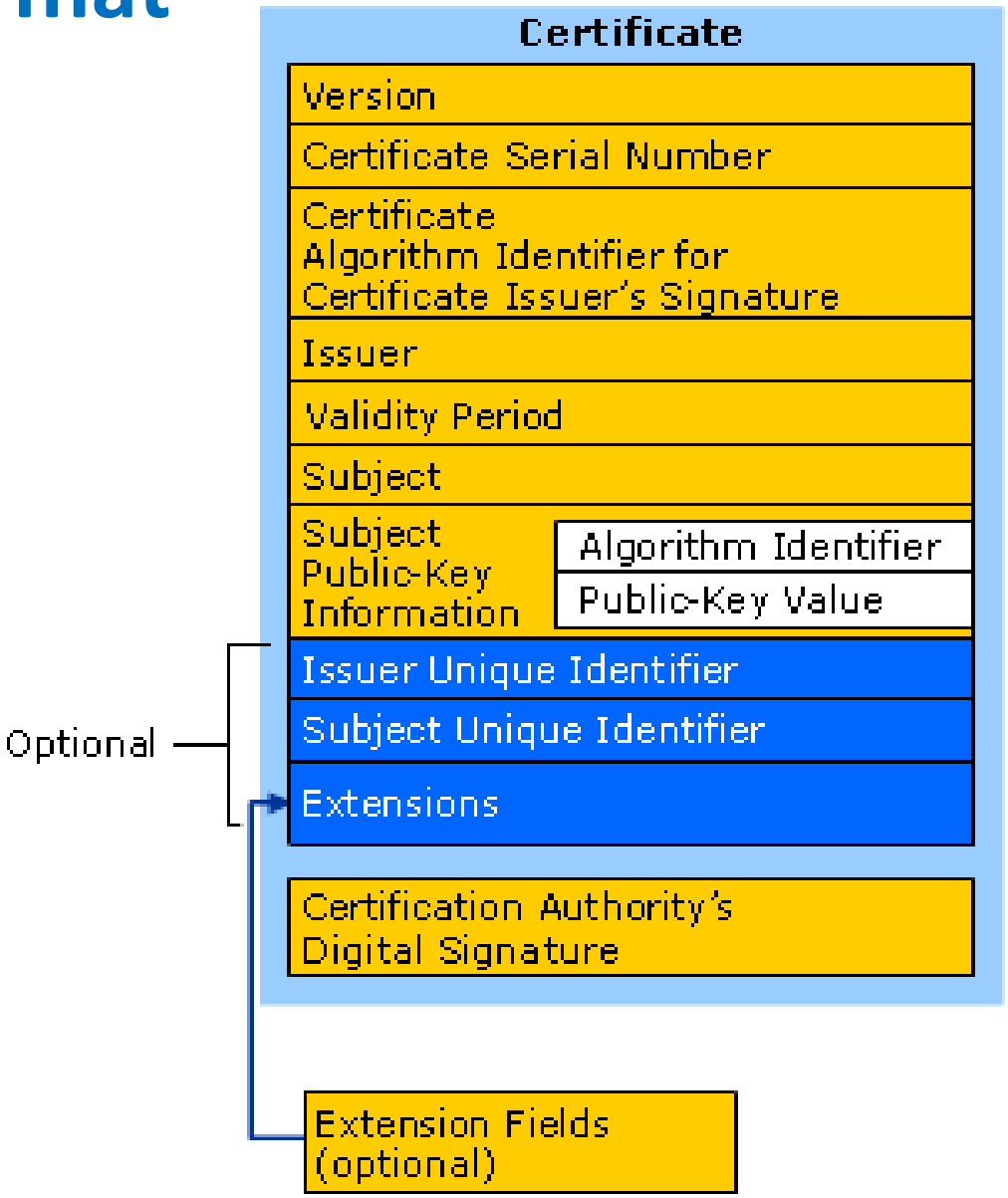
- The binding between {ID, Pub_{key}} is granted by a trusted Certification Authority (CA) that signs Cert
- Provided that we have the CA's public key, we can verify the CA signature and therefore verify the Cert authenticity
  - We trust CA and we have CA's public key (e.g., In Windows use **CertMgr** tool to see the installed Root Certificates)
  - Verify CA signature on the Cert. If OK! then the Cert is authentic
  - Also check the certificate validity period to ensure it is NOT expired

Anatomy of a Certificate



X.509 Certificate Format

- X509 is a popular Cert standard that define the attributes of a certificate
- CA signature is computed over all other attrributes in the certifcate (after hashing them)

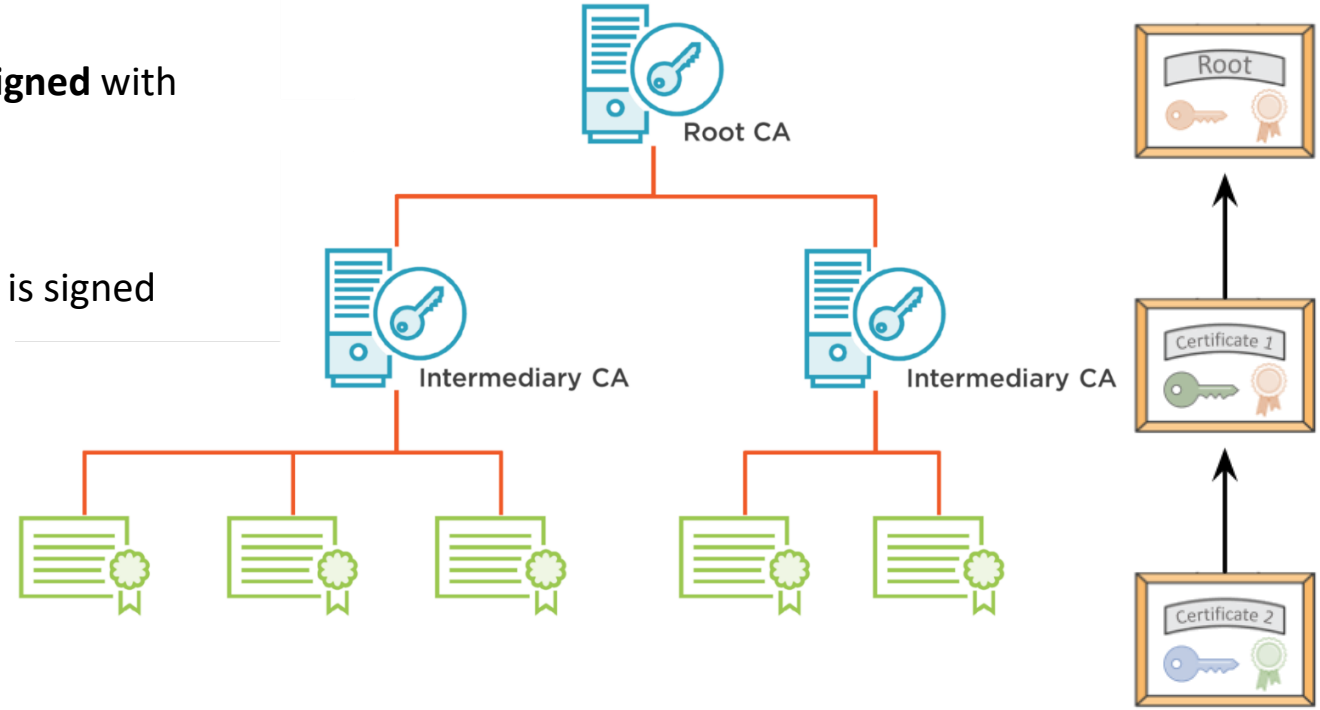


Trust Chain

- Certificate of Root CA is **self-signed** with the Root CA private key

- Certificate of Intermediate CA is signed with the Root CA private key

- Issued certificates are signed with the private key of Issuing Intermediate CA



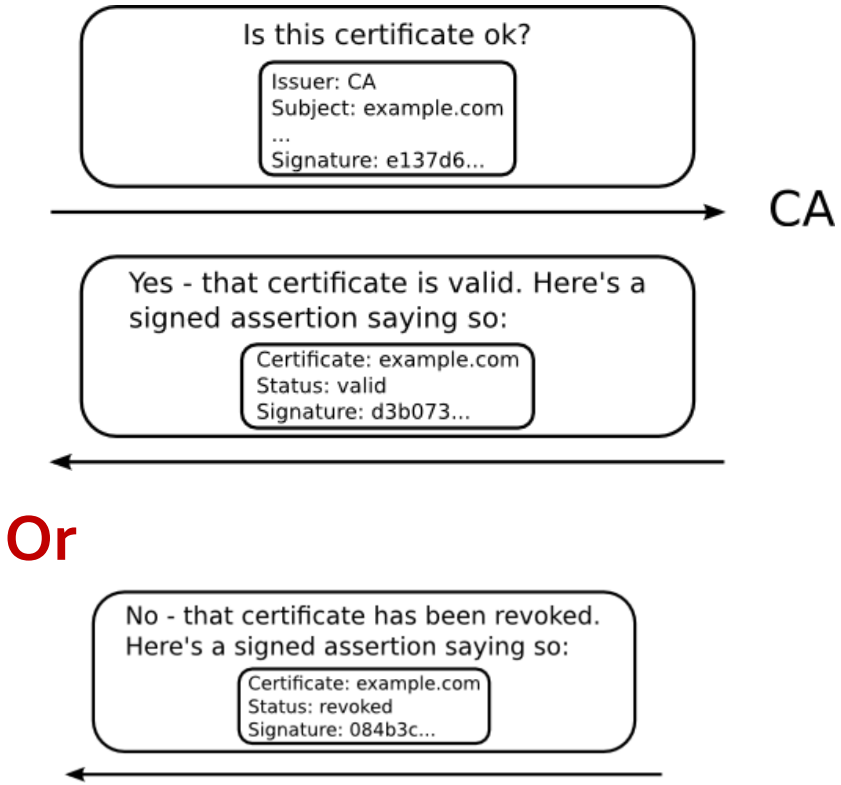
- The **Trust Chain** allows anyone to check the authenticity of any certificate by examining it all the way to a well-known, trusted root certificate
- Trust Chain allows a distributed trust model by enabling the ability to have multiple intermediary Cas with one master root CA
- This allow more scalable solution and limits the risk if one CS is compromised

Certificate Revocation

- A PKI MUST include mechanisms for revoking certificates! In case:
  - Public keys whose private keys have been compronised or lost
  - The domain was transferred to a new owner
  - Analogy: a credit card cancellation when lost or stolen
- Explicit revocation is used. CA maintains and publishes a **Certificate Revocation List** (**CRL**) to inform client about certificates that have been revoked (i.e., no longer valid)
- Client can download and periodically sync the CRL from the CA to verify whether a particular Cert is revoked.

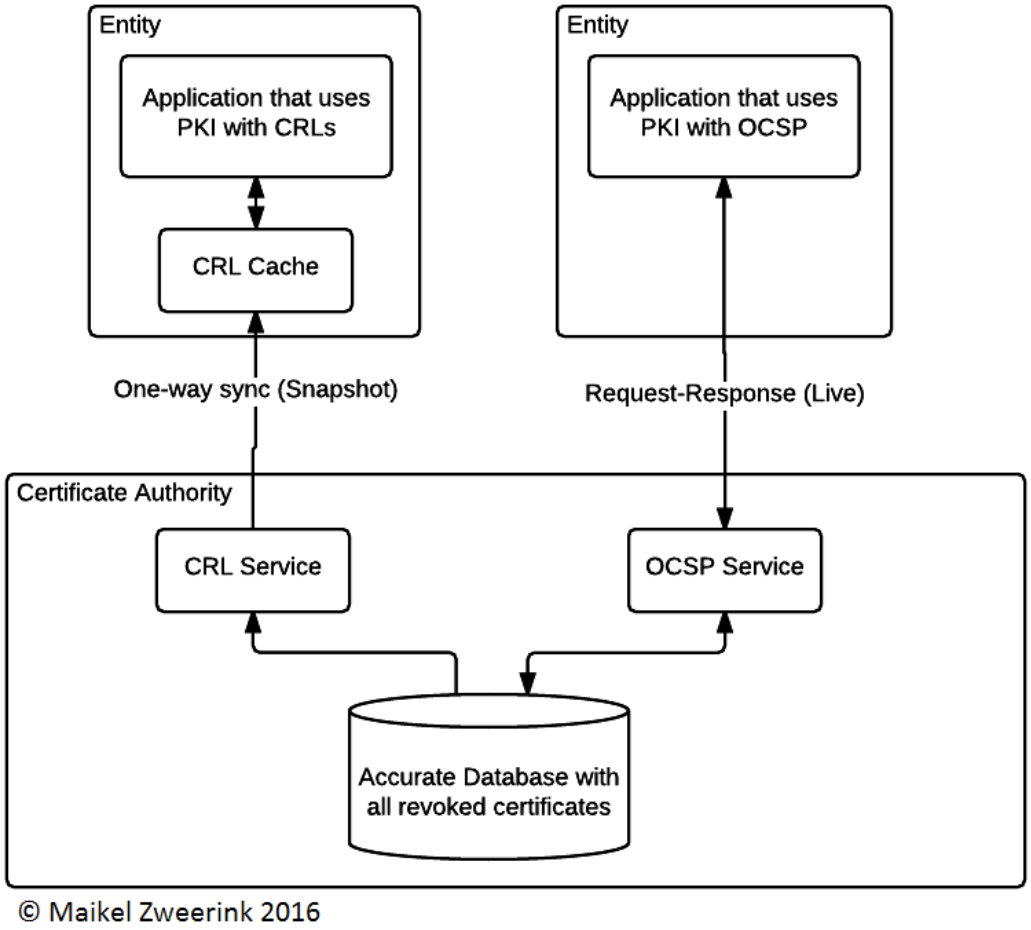
Online Certificate Status Protocol (OCSP)

- OCSP enables real-time verification of the certificate status (alternative to CRLs)
  - If the certificate is fine, the CA can respond with a signed assertion that the certificate is still valid. Otherwise it will state that it is revoked.



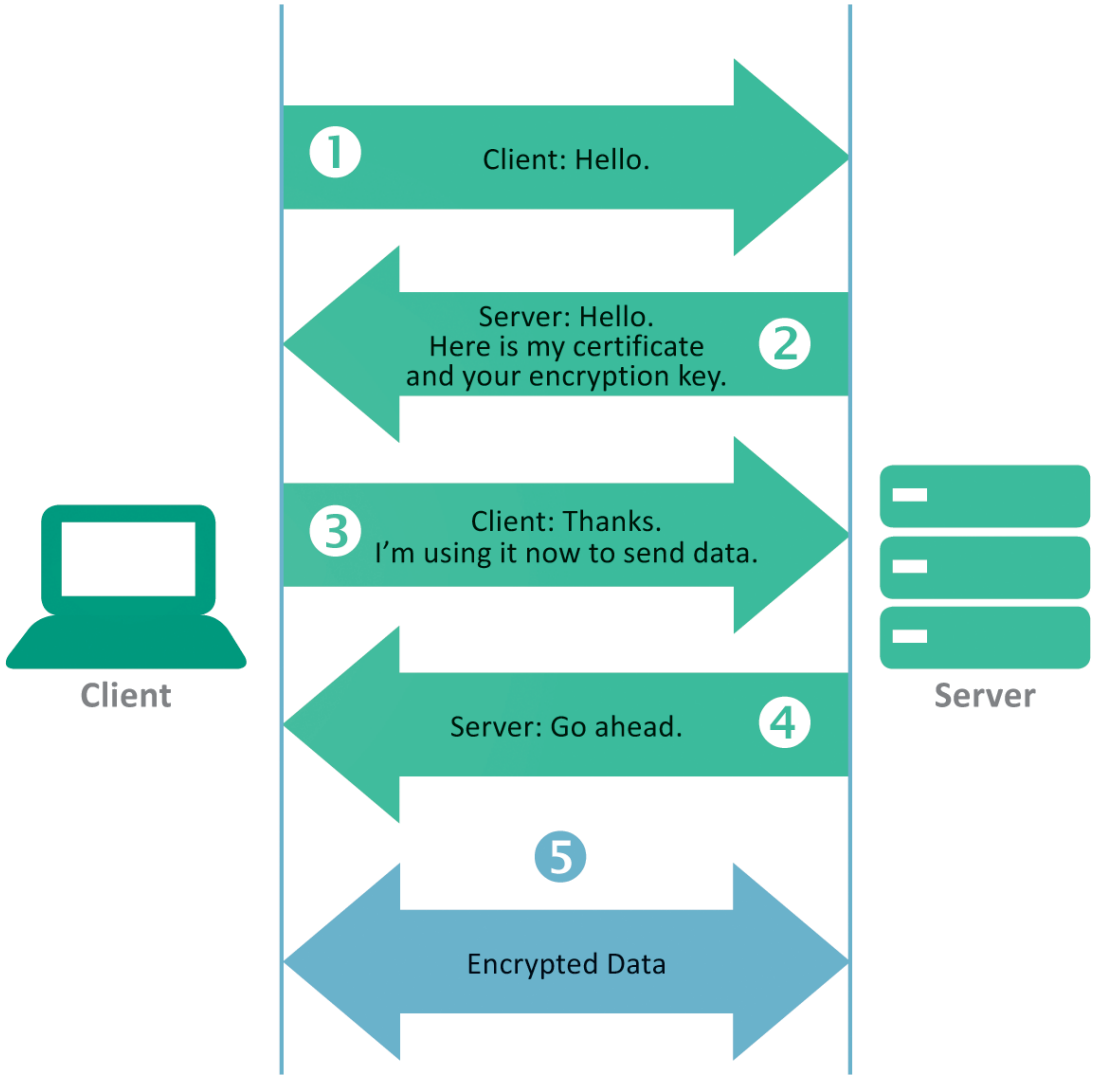
Difference between OSCP and CRL

- OSCP allows direct and on demand check if a certificate is revoked.
- CRLs are downloaded periodically, then locally check if a certificate is revoked.
- OSCP no longer required to download CRLs and the information is more accurate. This comes at the cost of being more prone to availability attacks (such as DDoS) as the service is completely used on-demand.



Source: https://www.maikel.pro/blog/current-state-certificate-revocation-crls-ocsp/

Digital Certificate in TSL Handshake (HTTPS)



Resources

- Digital Certificate

https://en.wikipedia.org/wiki/Public_key_certificate

- Free Certificate (an alternative to Cert from CA)

https://letsencrypt.org/