Public Key Infrastructure: Part 2

Gaurav S. Kasbekar
Dept. of Electrical Engineering
IIT Bombay

NPTEL

References

- J. Kurose, K. Ross, "*Computer Networking: A Top Down Approach*", Sixth Edition, Pearson Education, 2013
- A. Tanenbaum, D. Wetherall, "*Computer Networks*", Fifth Edition, Pearson Education, 2012.
- C. Kaufman, R. Perlman, M. Speciner, "*Network Security: Private Communication in a Public World*", Pearson Education, 2nd edition, 2002

NPTEL

Terminology

- If Alice signs a certificate vouching for Bob's name and public key, then:
  - Alice is the "**issuer**"
  - Bob is the "**subject**"
- If Carol is checking whether a certificate issued to Bob is valid, then Carol is the "**verifier**"
- A **trust anchor** is a public key that the verifier has decided, through some means, is trusted to sign certificates
  - e.g., suppose Carol's browser is pre-loaded with a certificate of the CA Symantec, then Symantec is a trust anchor for Carol
  - e.g., suppose Carol met Alice in person and the latter provided her public key to the former, then Alice is a trust anchor for Carol

Different PKI Models and Pros and Cons of each Model

NPTEL

Monopoly Model

- The world chooses one organization, universally trusted by all organizations, countries, people, etc., to be the single CA for the world
- Public key of that organization embedded in all software and hardware as the PKI trust anchor
- Everyone needs to obtain certificates from that CA

NPTEL

Disadvantages of Monopoly Model

- There is no one universally trusted organization
- Since all hardware and software would come preconfigured with the monopoly organization's public key:
  - it would be infeasible to change the key in case it were compromised
- It would be expensive to have a remote organization certify a person's public key, e.g.:
  - they need some way of reliably checking that person's credentials
  - public key of subject must be securely communicated to the monopoly organization
- Once enough hardware and software was deployed embedded with the monopoly organization's public key, it would be difficult for the world to switch to another CA; hence:
  - the monopoly organization could charge very high rates to issue certificates
- Security of the entire world would be compromised if the monopoly organization had an incompetent or corrupt employee since:
  - he/ she may be tricked or bribed into issuing bogus certificates or disclosing the CA's private key

Monopoly and Registration Authorities (RA) Model

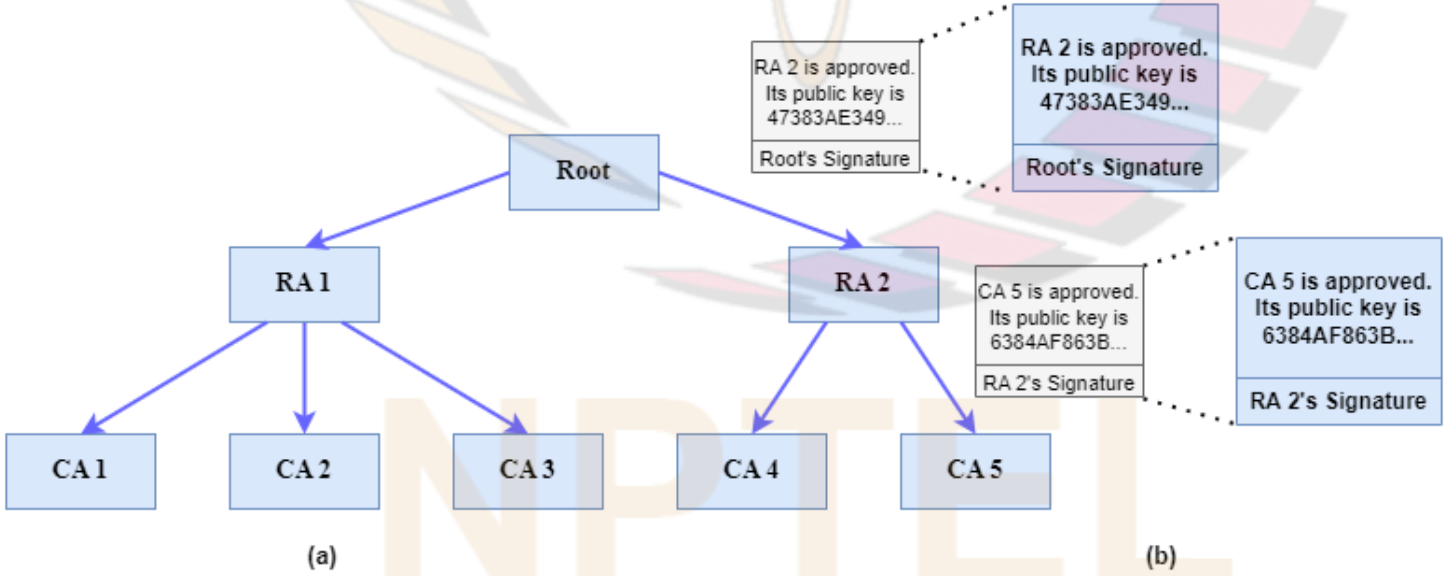
- Similar to monopoly model, except that:
  - the single CA chooses other organizations (called RAs) to securely check identities and vouch for public keys of a subject
  - each RA securely communicates with the CA
  - the CA issues a certificate to a subject vouched for by RA, since the CA trusts the RA
- Advantages over monopoly model:
  - more convenient and secure to obtain certificates, since there are more places to go to for getting a certificate
- However, the other disadvantages of monopoly model also apply to this model

Delegated CAs

- One CA issues certificates to other CAs:
  - vouching for their public keys and
  - vouching for their trustworthiness as CAs
- Users then obtain certificates from one of the delegated CAs instead of having to go to the delegating CA
- There can be multiple levels of delegation
- Difference between delegated CAs and RAs:
  - Delegated CAs issue certificates; RAs don't
  - Suppose Alice is verifying Bob's certificate
    - In latter case, Alice checks only one certificate; in former case, Alice checks a *chain of certificates*

Example

- Top level CA (root) certifies second-level CAs (Regional Authorities (RAs)), which in turn certify CAs that certify organizations and individuals
  - *Note*: Do not confuse “Regional Authority” with “Registration Authority”
- *E.g.*:
  - Suppose Alice wants to communicate with Bob and receives a certificate containing his public key signed by CA5
  - She verifies that Bob is legitimate using above certificate, that CA5 is legitimate using certificate issued to CA5 by RA2 and that RA2 is legitimate using certificate issued to RA2 by Root
  - Root’s public key is well-known (*e.g.*, preloaded in web browsers)



Oligarchy Model

- The model commonly used in web browsers (e.g., Firefox, Chrome)
- The web browser comes pre-configured with certificates (containing public keys) of multiple CAs (roots)
  - ❑ certificate signed by any of these roots, or chain of certificates rooted at any one of these CAs, is accepted
- Possible for a user to add/ delete CAs (roots) to/ from the pre-configured list
- Advantages over monopoly models:
  - ❑ There need not be an organization (root CA) that is trusted by all organizations and individuals
  - ❑ Easier for an individual or organization to obtain a certificate
  - ❑ Organizations chosen as roots will be in competition with each other; so there will not be monopoly pricing
- Disadvantages of oligarchy model:
  - ❑ The set of CAs (roots) whose public keys are included in a browser is selected by the vendor, not by user; vendor may sometimes use inappropriate criteria for the selection (e.g., including all CAs (roots) who pay for being included)
  - ❑ Browsers today come pre-configured with public keys of more than 80 roots; impractical for even a knowledgeable user to examine this list and see if it has been tampered with/ needs modification, etc.

Anarchy Model

- Each user is responsible for configuring some trust anchors
  - ❑ e.g., public keys of people he/ she has met, who have handed him/ her a business card with hash value of public key and sent him/ her email containing a public key with that hash value
- Anyone can potentially sign certificates for anyone else
- Some organizations (e.g., MIT) volunteer to keep a certificate database into which anyone can deposit certificates
- To obtain the public key of someone (say, Alice), you can search through the public database to see if you can find a path (chain of certificates) from one of your trust anchors to Alice
- Shortcomings of anarchy model:
  - ❑ Scaling problems: Database would get extremely large if deployed on Internet scale
    - If every user donated ten certificates on average, database would consist of billions of certificates
    - may be impractical to search through the database and construct paths
  - ❑ Some malicious individuals may add bogus certificates making it difficult to know whether to trust a path

Certificate Revocation

NPTEL

Certificate Revocation

- If someone realizes that their private key has been stolen or an employee who knew an organization's private key left the organization:
  - a way is needed to revoke their certificate
- Certificates have expiration dates on them
  - but validity period of a certificate is usually several months
  - insecure to wait until certificate expires
- Mechanism for a CA to revoke certificates that it earlier issued:
  - CA periodically issues a "*Certificate Revocation List*" (CRL): signed timestamped list of all certificates that have been revoked
  - If Alice wants to verify Bob's certificate, she checks whether Bob's certificate is included in latest CRL
- Assuming that CRLs are periodically issued, why do certificates have expiration dates at all?
  - To keep the sizes of CRLs small (since expired certificates need not be included in CRLs)

Delta CRLs

- Intended for making CRL distribution more efficient
- E.g.:
  - Suppose we want revocation to take effect within one hour
  - Then a CA would have to post a CRL every hour
  - Every verifier would need to download the latest CRL every hour
  - Suppose the CRL became very large, since a company laid off 10,000 people (each of whom had a certificate)
  - Then every hour, every verifier would have to download a huge CRL, even though very few certificates had been revoked after the layoff
- Delta CRL
  - In above e.g., full CRLs are posted much less frequently than once every hour
  - Delta CRL posted once every hour; lists changes from the last full CRL
  - E.g. of a delta CRL: "The following is a list of all the certificates that have been revoked since Feb. 7, 10 am, which is the most recent full CRL: …"
  - Delta CRLs would be very short, often containing no certificates
  - CA does not need to post full CRL (which is often large) frequently
  - Instead, CA can post a full CRL in place of a delta CRL when the delta CRL gets sufficiently large
  - Each verifier has to download:
    - latest full CRL and latest delta CRL