Distributed Systems Security

Authentication Practice - 2

Prof. Steve Wilbur
s.wilbur@cs.ucl.ac.uk

Lecture Objectives

- Examine X.509 as a practical example of Public Key services
X.500 Directory Service

- X.500 is a family of standards for directory services providing information about users
- Developed in late-1980s by ITU
- X.509 defines a certificate structure and protocols which are widely used, eg.:
  - S/MIME
  - IP Security
  - SSL/TLS
  - SET
- X.509:

X.509 Certificate Structure

Example of early definition of Certificate structure in ASN.1

LHS = identifier; RHS = type
certificate ::= SIGNED SEQUENCE {
  signature AlgorithmIdentifier,
  issuer Name,
  validity Validity,
  subject Name,
  subjectpkinfo SubjectPublicKeyInfo}

Validity ::= SEQUENCE {
  notbefore UTCTime,
  notAfter UTCTime}

SubjectPublicKeyInfo ::= SEQUENCE{
  algorithm AlgorithmIdentifier,
  subjPK BITSTRING}

AlgorithmIdentifier ::= SEQUENCE {
  algorithm OBJECT IDENTIFIER,
  parameters ANY DEFINED BY algorithm OPTIONAL}
X.509 Certificates

- Issued by trusted Certification Authority
- Directory Service only stores and distributes them

X.509 Certificate Structure

Version
Certificate Serial Number
Signature Alg. ID
Issuer Name
Validity Period
Subject Name
Subject PK Info.
Issuer Unique ID
Subject Unique ID
Extensions
Signature
Encrypted Hash

Version
Versions 1, 2 and 3
Versions 2 and 3
Version 3 only

Versions 1, 2 and 3
Versions 2 and 3
Version 3 only
## X.509 Certificates - 2

- **Version:** Indicates format of certificate (1, 2 or 3)
- **Serial Number:** Integer associated with this C, unique in issuing CA
- **Sig. Alg. ID:** Algorithm used to sign the C and any parameters (repeated in the Signature field)
- **Issuer Name:** Name of CA that created C
- **Validity Period:** First and last dates on which C is valid
- **Subject Name:** Name of user to whom C applies
- **Subject PK Info.:** Public key, algorithm and any relevant parameters of subject
- **Issuer Unique ID:** Optional bit string to identify uniquely CA in case name is not unique
- **Subject Unique ID:** Optional bit string to identify uniquely CA in case name is not unique
- **Signature:** Covers all other fields of C

## X.509 Certificates - 3

- Inter-domain certification by CA’s producing certificates for each other’s Public Keys
  - **Forward Certificate:** Certificate of X generated by other Cas
  - **Reverse Certificates:** Certificates generated by X for another CA
- Generally arranged hierarchically so that easy for users to find certification chain and request relevant certificates
- Otherwise, similar to “theory”
X.509 Certificate Revocation

- Certificates usually issued for appropriate length of time, e.g., students: 1 academic year
- May need to nullify C earlier if:
  - User’s secret key compromised
  - User no longer within jurisdiction of CA, e.g., left job
  - CA’s certificate has been compromised
- Each CA keeps a list of all revoked but not expired certificates
- Periodically published to directory via Certificate Revocation List (CRL)
- If end user caches C’s they must also cache relevant CRL’s

X.509 Certificate Revocation List

| Signature | Algorithm
| Alg. ID | Parameters |
| Issuer Name | Certificate Serial Number |
| This update date | Revocation Date |
| Next update date |
| Revoked Certificate |
| ... |
| Revoked Certificate |
| Signature | Algorithms
| Parameters | Encrypted Hash |
X.509 Certificates V.3

Limitations of V.2

- Subject field inadequate to fully convey identity of owner (which j smith in that domain esp. if domain is broad eg. ISP)
- May be several different identities for a given user, eg. mail address, URL, etc. - need to specify and relate them
- Need to indicate security policy information, so protocols can relate specific Certificate for this
- Need to limit damage from faulty or malicious CA
- Important to keep separate keys used by same owner at different times - key life cycle management

Flexible structure to deal with these and other needs: extensions

- Each extension consists of:
  - extension identifier
  - criticality indicator
  - extension value
- Criticality indicator indicates whether this extension can safely be ignored
  - if indicator is TRUE and application/protocol cannot deal with this extension type, then the certificate must be treated as invalid
X.509 Certificates V.3
Key and Policy Information -1

- Authority Key Identifier: Identifies which of CAs keys to use to validate C. Allows CAs key pairs to be updated.

- Subject Key Identifier: Similar to above.

- Key Usage: Policy restrictions on key, eg. digital signature, data encryption, key encryption etc.

X.509 Certificates V.3
Key and Policy Information -2

- Private-key Usage Period: Private key may be valid for a much shorter period than the public key, eg. signing (private) key validity less than verifying (public) key.


- Policy Mappings: For Certificates for CAs issued by other CAs. Indicates policies in issuer domain which are equivalent in the subject CAs domain.
**X.509 Certificates V.3**

**Subject & Issuer Attributes**

- Provide alternative names in alternative formats
- Increase user’s confidence that C relates to particular person or entity
- Examples:
  - postal address
  - position within organisation
  - picture
- **Subject Alternative Name**: One or more alternative names. Some apps use their own name forms

---

**Subject & Issuer Attributes -2**

- Subject Alternative Name: One or more alternative names. Some apps use their own name forms
- **Issuer Alternative Name**: Similar to above, but for issuer
- **Subject Directory Attributes**: X.500 directory attributes for the subject
**X.509 Certificates V.3**

**Certification Path Constraints**

- May provide constraints on which cross-certificates may appear in certification chains.
- May constrain the types of certificates that the subject CA can issue.
- **Basic Constraints**: Indicates if subject may act as CA. May include a max. certification path length.
- **Name Constraints**: Limits name space for all subject names in subsequent Cs in a path.
- **Policy Constraints**: May enforce explicit policy specification in the rest of the certification path.

---

**Certificate Creation**

- Algorithmically, this is easy.
- BUT, need to think carefully about processes within organisation or domain.
- Following diagrams show schematics of Certificate issue under authority of a commercial or government issuer.
- Assumes that subject’s company has established its credential beforehand with issuing authority.
Certificate Generation

M is of form:
LA, (RQCert, LA, User, Creds, PK)SKLA

CA - Certification Authority
LA - Local Authority

Local Certificate Generation

CA - Certification Authority
LA - Local Authority
Certificate Issuing Issues

- How do you prove to someone who you are?
  - Driving licence?
  - P45?
  - Letter of reference?
  - Birth Certificate?

- How much certainty is required?

- If you buy/program key-pair generation software how do you know it is sound? What tests would you apply to it?

Certificate Issuing Issues - 2

- Are copies of Private Keys kept - escrowed? All of them, non, some?

- How are Private Keys stored? How secure is this?

- If PKs are encrypted while not in use (key chain) how secure is the encryption? Based on password or phrase?

- What procedures does CA expect LA to carry out? What auditing needs to be done? Does this introduce potential weaknesses?
Further Reading

  - X.509 Authentication and Certificates: pp 341-349

  - Certificates: pp 135-140


Further Reading - 2