Computer security
Key management

Olivier Markowitch

Cryptographic keys management

- exchange of secret keys for symmetric cryptographic algorithms

- management of public keys associated to a private key

Session keys = short-life symmetric secret keys

Secret keys exchange

Key establishment is a protocol whereby a shared secret becomes available to two or more parties

key transport is a key establishment protocol whereby one party creates or obtains a secret value, and securely transfers it to the other parties

key agreement is a key establishment protocol in which a shared secret is derived from information ideally provided by each party and such that no party can predetermine the resulting value

Secret keys exchange

- Key transport protocols based on asymmetric encryption:
  - Needham-Schroeder
  - modified Needham-Schroeder
- Key agreement protocols based on asymmetric techniques:
  - Diffie-Hellman
  - Station-to-station protocol

Secret keys exchange

Needham-Schroeder

$A \rightarrow B : E_{K_B}(k_1, A)$

$B \rightarrow A : E_{K_A}(k_1, k_2)$

$A \rightarrow B : E_{K_B}(k_2)$

Secret keys exchange

Needham-Schroeder: problem

$$A \rightarrow B : E_{K_B}(k_1, A)$$

$$B \rightarrow C : E_{K_C}(k_1, A)$$

$$C \rightarrow B : E_{K_A}(k_1, k_2)$$

$$B \rightarrow A : E_{K_A}(k_1, k_2)$$

$$A \rightarrow B : E_{K_B}(k_2)$$

$$B \rightarrow C : E_{K_C}(k_2)$$

Secret keys exchange

Needham-Schroeder fixed

$A \rightarrow B : E_{K_B}(k_1, A)$

$B \rightarrow A : E_{K_A}(B, k_1, k_2)$

$A \rightarrow B : E_{K_B}(k_2)$

Secret keys exchange

Needham-Schroeder fixed

$A \rightarrow B : E_{K_B}(k_1, A)$

$B \rightarrow C : E_{K_C}(k_1, A)$

$C \rightarrow B : E_{K_A}(C, k_1, k_2)$

$B \rightarrow A : E_{K_A}(C, k_1, k_2)$

$A \rightarrow B :$ STOP

Secret keys exchange

Modified Needham-Schroeder

$A \rightarrow B : E_{K_B}(k_1, r_1, A)$

$B \rightarrow A : E_{K_A}(k_2, r_1, r_2)$

$A \rightarrow B : r_2$

Secret keys exchange

Diffie-Hellman

Let a large prime p and α a generator of $\mathbb{Z}_p^*$

$A \rightarrow B : \alpha^x \bmod p$
(where x is a secret random value chosen by Alice)

$B \rightarrow A : \alpha^y \bmod p$
(where y is a secret random value chosen by Bob)

$k = (\alpha^x)^y = (\alpha^y)^x = \alpha^{xy} \bmod p$

Secret keys exchange

Diffie-Hellman: man-in-the-middle attack

$A \to O : \alpha^x \bmod p$

$O \to B : \alpha^{x'} \bmod p$

$B \to O : \alpha^y \bmod p$

$O \to A : \alpha^{y'} \bmod p$

Alice computes $k_1 = (\alpha^{y'})^x \bmod p$

Bob computes $k_2 = (\alpha^{x'})^y \bmod p$

Oscar computes: $k_1 = (\alpha^x)^{y'} \bmod p$ et
$k_2 = (\alpha^y)^{x'} \bmod p$

Secret keys exchange

Station-to-station protocol

Let a large prime p and α a generator of $\mathbb{Z}_p^*$

$A \rightarrow B : \alpha^x \bmod p$

$B \rightarrow A : \alpha^y \bmod p, E_k(\text{Sig}_B(\alpha^x, \alpha^y))$

$A \rightarrow B : E_k(\text{Sig}_A(\alpha^x, \alpha^y))$

Where $k = (\alpha^x)^y = (\alpha^y)^x = \alpha^{xy} \bmod p$

Lifetime of a cryptographic key

The **cryptoperiod** of a (symmetric ou asymmetric) cryptographic key is the period during which the key is valid

A cryptographic key can be *short-term* or *long-term*

Public key certificates

A **public key certificate** allows to bind a public key with the identity of its owner

A public key certificate contains at least the public key, the information allowing to identify its owner and a digital signature on the key and the information

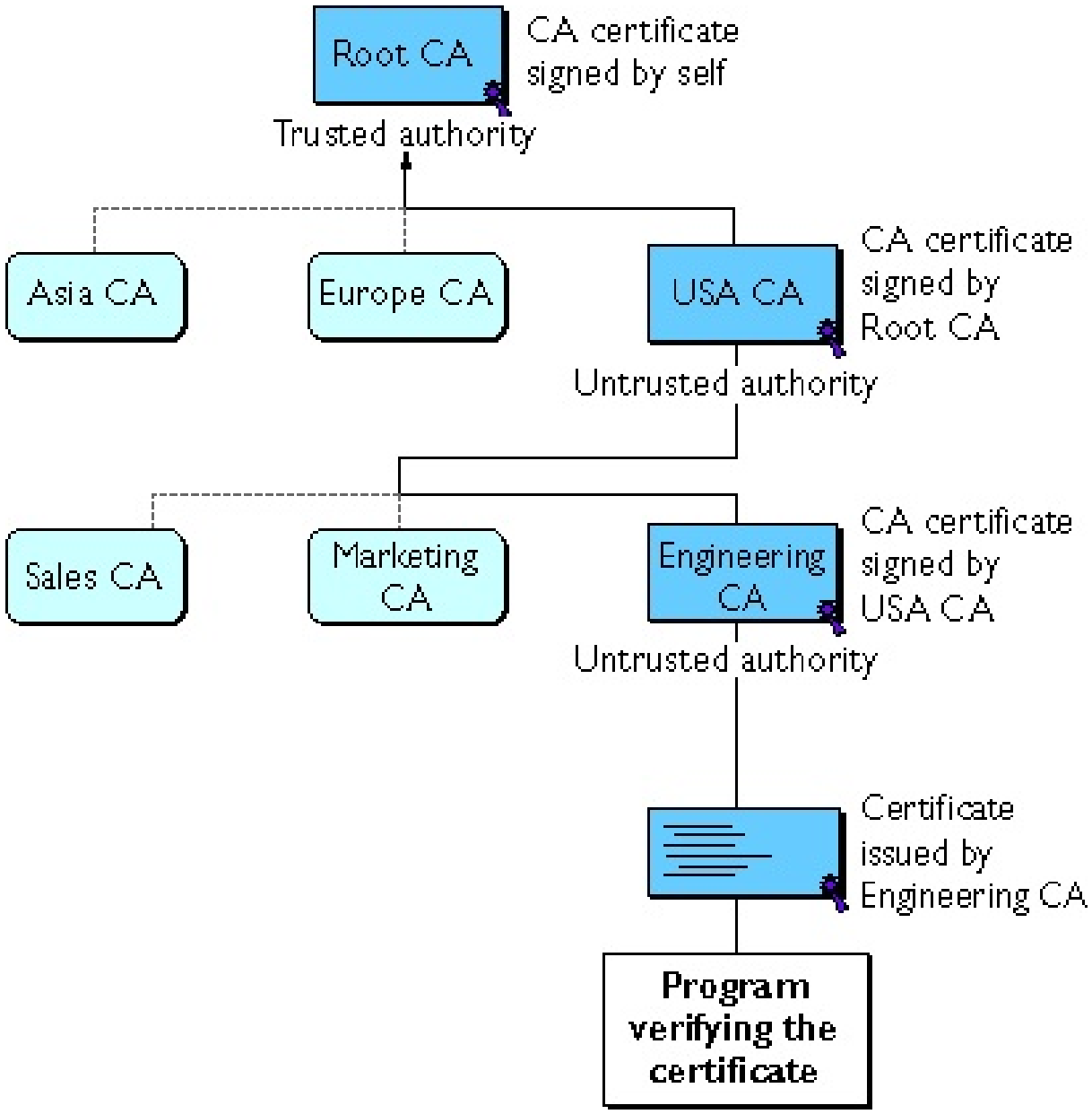
The digital signature is produced by a certification authority

Public key certificates

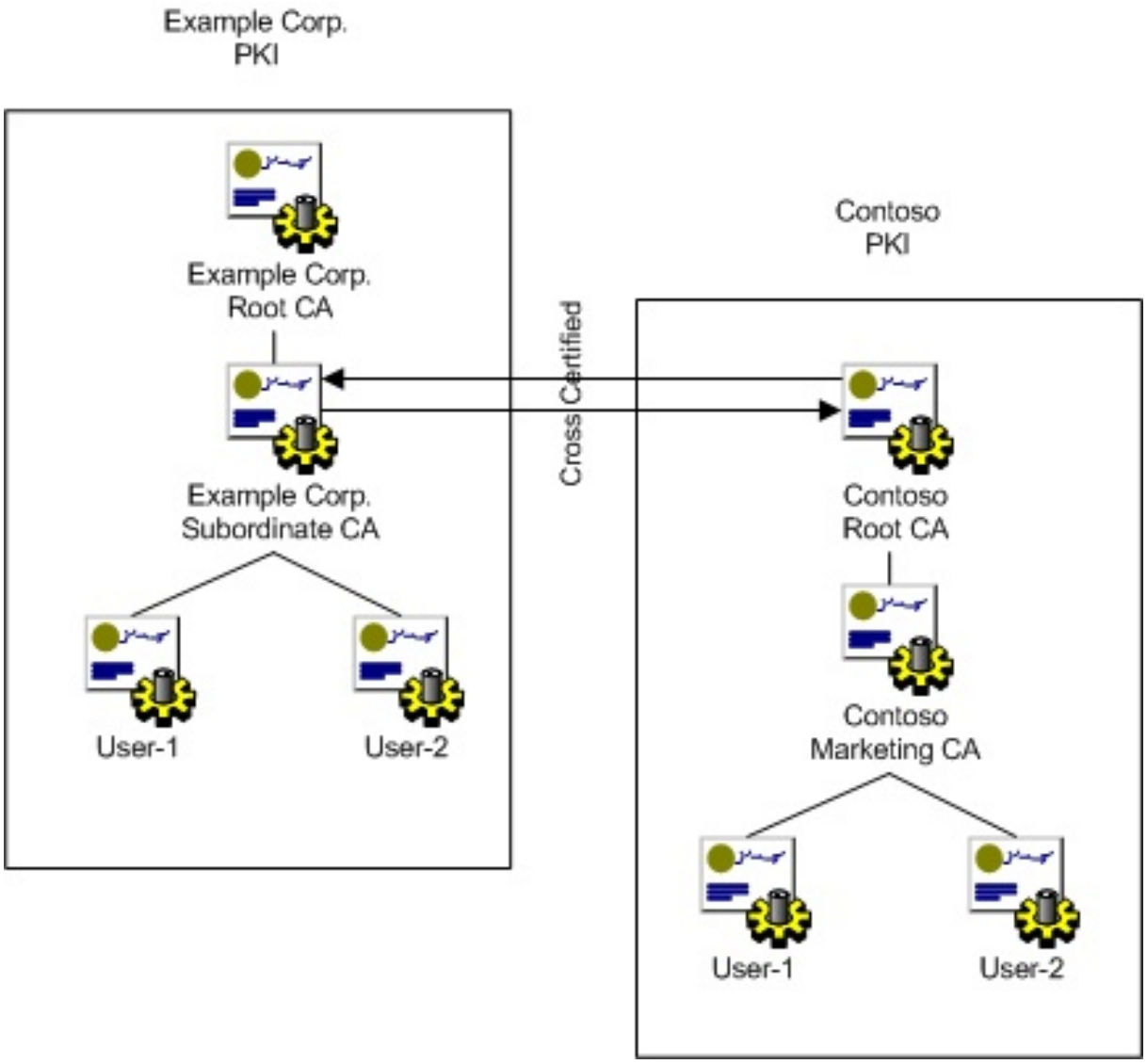
- Certificate
- Version
- Signature algorithm
- Issuer
- Validity (not before, not after)
- Subject
- Subject Public Key Info (public key algorithm, public key)
- Certificate signature algorithm
- Certificate signature

```
Certificate:
 Data:
    Version: 3 (0x2)
    Serial Number: 3 (0x3)
    Signature Algorithm: PKCS #1 MD5 With RSA Encryption
    Issuer: OU=Ace Certificate Authority, O=Ace Industry, C=US
    Validity
       Not Before: Aug  1 00:00:00 2006 GMT
       Not After : Dec 31 23:59:59 2020 GMT
    Subject: CN=Jane Doe, OU=Finance,
                    O=Ace Industry, C=US
    Subject Public Key Info:
       Algorithm: PKCS #1 RSA Encryption
       Public Key:
          Modulus:
               00:d3:a4:50:6e:c8:ff:56:6b:e6:cf:5d:b6:ea:0c:
               68:75:47:a2:aa:c2:da:84:25:fc:a8:f4:47:51:da:
               85:b5:20:74:94:86:1e:0f:75:c9:e9:08:61:f5:06:
               6d:30:6e:15:19:02:e9:52:c0:62:db:4d:99:9e:e2:
               6a:0c:44:38:cd:fe:be:e3:64:09:70:c5:fe:b1:6b:
               29:b6:2f:49:c8:3b:d4:27:04:25:10:97:2f:e7:90:
               6d:c0:28:42:99:d7:4c:43:de:c3:f5:21:6d:54:9f:
               5d:c3:58:e1:c0:e4:d9:5b:b0:b8:dc:b4:7b:df:36:
               3a:c2:b5:66:22:12:d6:87:0d
          Public Exponent: 65537 (0x10001)
    Signature:
        Algorithm: PKCS #1 MD5 With RSA Encryption
        Signature:
          07:fa:4c:69:5c:fb:95:cc:46:ee:85:83:4d:21:30:8e:ca:d9:
          a8:6f:49:1a:e6:da:51:e3:60:70:6c:84:61:11:a1:1a:c8:48:
          3e:59:43:7d:4f:95:3d:a1:8b:b7:0b:62:98:7a:75:8a:dd:88:
          4e:4e:9e:40:db:a8:cc:32:74:b9:6f:0d:c6:e3:b3:44:0b:d9:
          8a:6f:9a:29:9b:99:18:28:3b:d1:e3:40:28:9a:5a:3c:d5:b5:
          e7:20:1b:8b:ca:a4:ab:8d:e9:51:d9:e2:4c:2c:59:a9:da:b9:
          b2:75:1b:f6:42:f2:ef:c7:f2:18:f9:89:bc:a3:ff:8a:23:2e:
          70:47
```

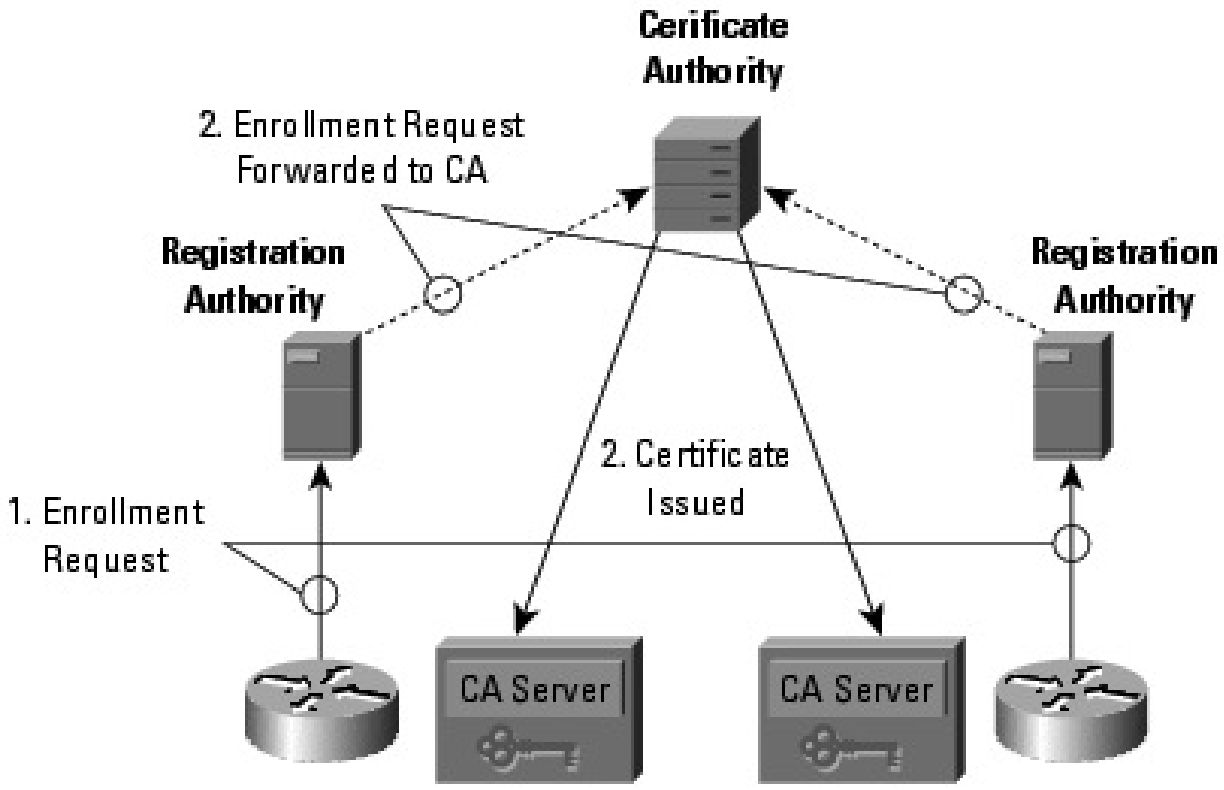
Public key certificates



Public key certificates



Public key infrastructure



Revocation

When an opponent has some information about a cryptographic key, the key is said to be *compromised* and has to be *revoked*

The digital certificates of revoked keys has to be added in a *certificate revocation list*

CRL has to be checked when using a public key