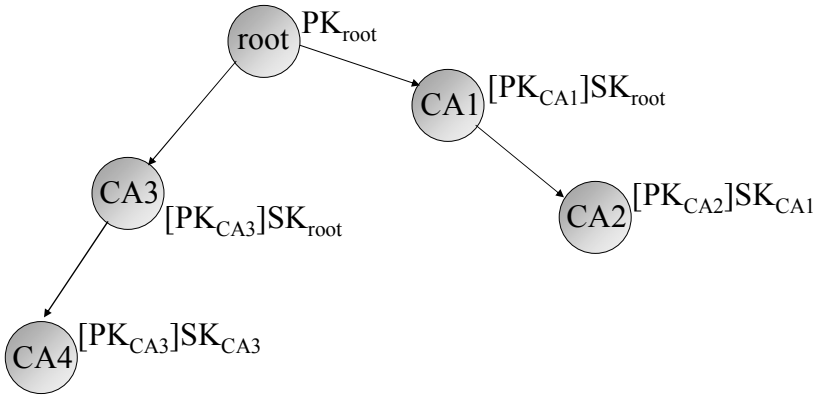
Lecture 18

Public Key Certification and Revocation

1

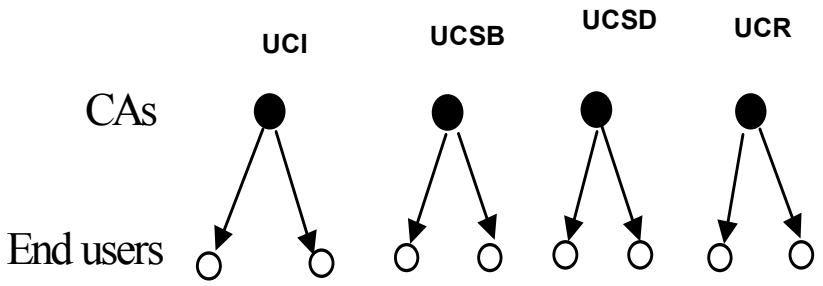
Certification Tree / Hierarchy

Logical tree of CA-s



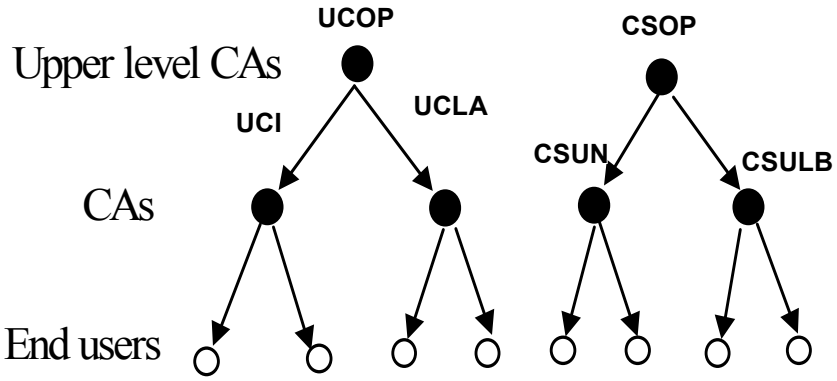
2

Hierarchical PKI Example



3

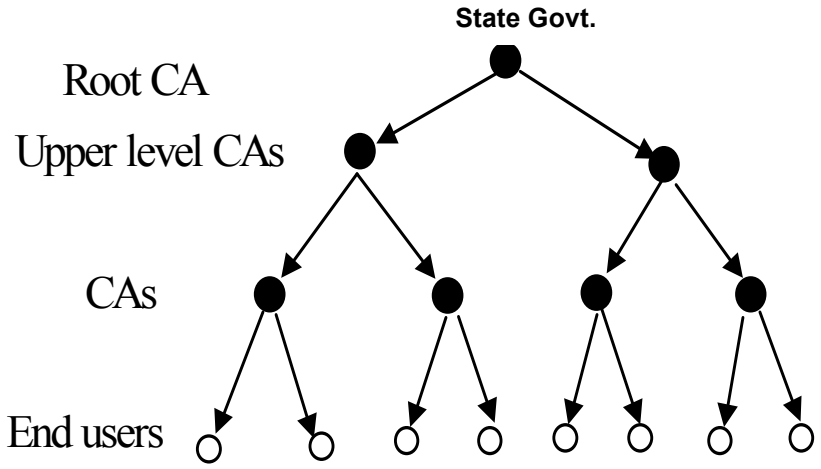
Hierarchical PKI Example



gtsudik@uci.edu

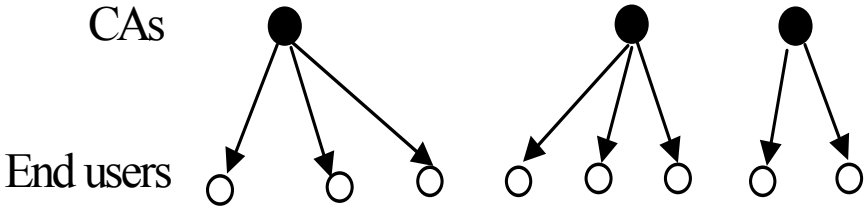
4

Hierarchical PKI Example



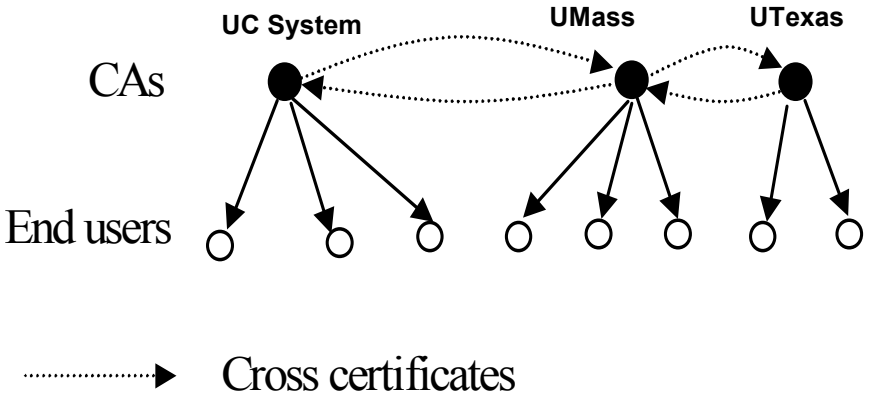
5

Cross Certificate Based PKI Example



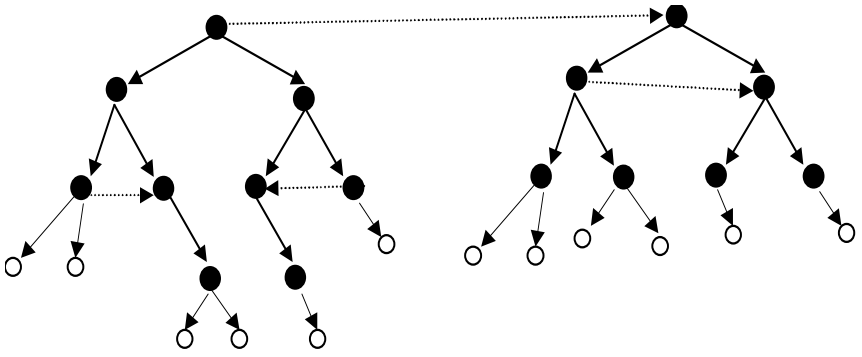
6

Cross Certificate Based PKI Example



7

Hybrid PKI example

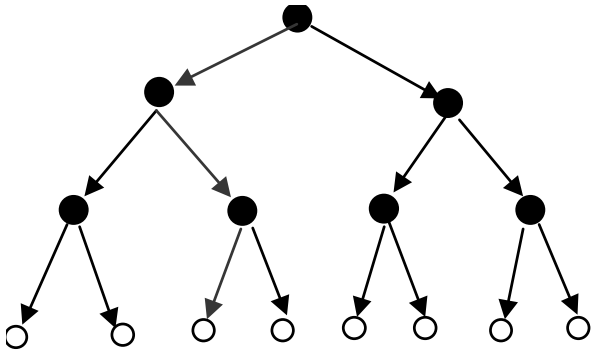


Note that no cross arrows down or up!

8

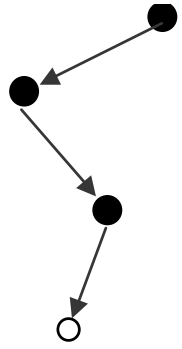
Certificate Paths

Derived from PKI



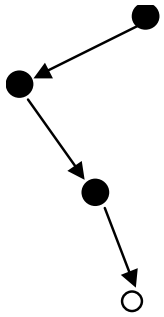
9

Certificate Paths



10

Certificate Paths



- Verifier must know public key of the first CA
- Other public keys are 'discovered' one by one
- All CAs on the path must be (implicitly) trusted by the verifier

11

X.509 Standard

- X.509v3 is the current version
- ITU standard
- ISO 9495-2 is the equivalent ISO standard
- Defines certificate format, not PKI
- Identity and attribute certificates
- Supports both hierarchical model and cross certificates
- **End users cannot be CAs**

12

X.509 Service

- Distributed set of servers that maintains a database about users
- Used in S/MIME, IPSec, SSL/TLS, SET.
- RSA, DSA, MD5, SHA are most commonly used algorithms

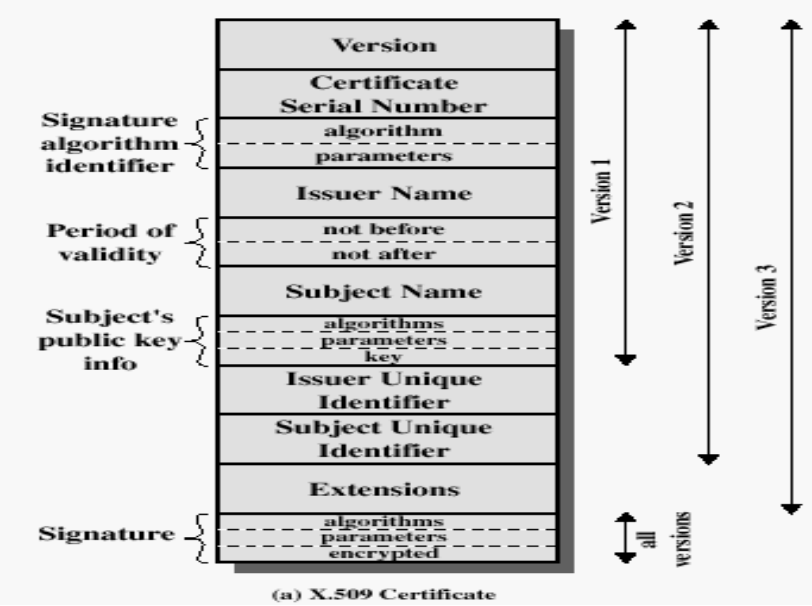
13

Format:

- version
- serial number
- signature algorithm ID
- issuer name(X.500 Distinguished Name)
- validity period
- subject(user) name (X500 Distinguished Name)
- subject public key information
- issuer unique identifier (version 2 and 3 only)
- subject unique identifier (version 2 and 3 only)
- extensions (version 3 only)
- signature on the above fields

14

X.509 Certificate Format



15

A sample certificate

```
Certificate:
   Data:
      Version: 3 (0x2)
      Serial Number: 28 (0x1c)
      Signature Algorithm: md5WithRSAEncryption
      Issuer: C=US, O=Globus, CN=Globus Certification Authority
      Validity
         Not Before: Apr 22 19:21:50 1998 GMT
         Not After : Apr 22 19:21:50 1999 GMT
      Subject: C=US, O=Globus, O=University of Southern California, \
               ou=ISI, CN=bonair.isi.edu
      Subject Public Key Info:
         Public Key Algorithm: rsaEncryption
         RSA Public Key: (1024 bit)
            Modulus (1024 bit):
               00:bf:4c:9b:ae:51:e5:ad:ac:54:4f:12:52:3a:69:
               <snip>
               b4:e1:54:e7:87:57:b7:d0:61
            Exponent: 65537 (0x10001)
Signature Algorithm: md5WithRSAEncryption
      59:86:6e:df:dd:94:5d:26:f5:23:c1:89:83:8e:3c:97:fc:d8:
      <snip>
```

16

Certificates in Practice

- X.509 certificate format is defined in Abstract Syntax Notation 1 (ASN.1)
- ASN.1 structure is encoded using the Distinguished Encoding Rules (DER)
- A DER-encoded binary sting is typically base-64 encoded to get an ASCII representation

17

Certificates in Practice

```
-----BEGIN CERTIFICATE-----
MIIDTzCCAvmgAwIBAgIBATANBgkqhkiG9w0BAQQFADBcMSEwHwYDVQQKExhFdXJv
cGVhbiBJQ0UtVEVMIHByb2plY3QxIzAhBgNVBAsTGlYzLUNlcnRpZmljYXRpb24g
QXV0aG9yaXR5MRIwEAYDVQQHEwlEYXJtc3RhZHQwHhcNOTcwNDAyMTczNTU5WhcN
OTgwNDAyMTczNTU5WjBrMSEwHwYDVQQKExhFdXJvcGVhbiBJQ0UtVEVMIHByb2pl
Y3QxIzAhBgNVBAsTGlYzLUNlcnRpZmljYXRpb24gQXV0aG9yaXR5MRIwEAYDVQQH
EwlEYXJtc3RhZHQxDTALBgNVBAMTBFVTRVIwWTAKBgRVCAEBAgICAANLADBIAkEA
qKhTY0kbk8PDC2yIEVXefmri+VKg3GklxMi/VeExqM7kqSmFmYoVmt72L+G0UF9e
BHWm9HbcPA453Dq+PqRhiwIDAQABo4IBmDCCAZQwHwYDVR0jBBgwFoAUfnLy+DqG
nEKINDRmdcPU/NGiETMwHQYDVR0OBBYEFJfc4B8gjSoRmLUx4Sq/ucIYiMrPMA4G
A1UdDwEB/wQEAwIB8DAcBgNVHSABAf8EEjAQMAYGBCoDBAUwBgYECQgHBjBDBgNV
HREEPDA6gRV1c2VyQGRhcm1zdGFkdC5nbWQuZGWGIWh0dHA6Ly93d3cuZGFybXN0
YWR0LmdtZC5kZS9+dXNlcjCBsQYDVR0SBIGpMIGmgQxnbWRjYUBnbWQuZGWGEWh0
dHA6Ly93d3cuZ21kLmRlghdzYXR1cm4uZGFybXN0YWR0LmdtZC5kZaRcMSEwHwYD
VQQKExhFdXJvcGVhbiBJQ0UtVEVMIHByb2plY3QxIzAhBgNVBAsTGlYzLUNlcnRp
ZmljYXRpb24gQXV0aG9yaXR5MRIwEAYDVQQHEwlEYXJtc3RhZHSHDDE0MS4xMi42
Mi4yNjAMBgNVHRMBAf8EAjAAMB0GA1UdHwQWMBQwEqAQoA6BDGdtZGNhQGdtZC5k
ZTANBgkqhkiG9w0BAQQFAANBAGkM4ben8tj76GnAE803rSEGIk3oxtvxBAu34LPW
DIEDzsNqPsfnJCSkkmTCg4MGQlMObwkehJr3b2OblJmD1qQ=
-----END CERTIFICATE-----
```

18

Certificate Revocation Scenario

What if:

- Someone steals Bob's private key?
- Bob looses his private key?
- Bob willingly discloses his public key?
  - Eve can decrypt/sign while Bob's certificate is still valid…
  - Bob reports key loss to CA (or CA finds out somehow)
  - CA issues a Certificate Revocation List (CRL)
    - Distributed in public announcements
    - Published in public databases
  - When verifying Bob's signature or encrypting a message for Bob, Alice first checks if Bob's certificate is still valid!
  - IMPORTANT: what about signatures "Bob" generated before he realized his key is lost?

19

More generally: Certificate is a capability!

- Certificate revocation needs to occur when:
  - certificate holder key compromise/loss
  - CA key compromise
  - end of contract (e.g. certificates for employees)
  - Certificate Revocation Lists (CRLs) hold the list of certificates that are not yet naturally expired but revoked
  - Reissued periodically (even if no activity!)
  - More on revocation later…

20

Requirements for revocation

- Timeliness
  - Must check most recent revocation status
- Efficiency
  - Computation
  - Bandwidth and storage
  - Availability
- Security

21

Types of Revocation

- Implicit
  - Each certificate is periodically (ore-issued
  - Alice has a fresh certificate ➔ Alice not revoked
  - No need to distribute/publish revocation info
- Explicit
  - Only revoked certificates are periodically announced
  - Alice's certificate not listed among the revoked ➔ Alice not revoked
  - Need to distribute/publish revocation info

22

Revocation methods

- CRL - Certificate Revocation List
  - CRL-DP, indirect CRL, dynamic CRL-DP,
  - delta-CRL, windowed CRL, etc.
  - CRT and other Authenticated Data Structures
- OCSP – On-line Certificate Status Protocol
- CRS - Certificate Revocation System

23

CRL

- Off-line mechanism
- CRL = list of revoked certificates (e.g., SNs) signed by a revocation authority (RA)
- RA not always CA that issued the revoked PKC
- Periodically issued: daily, weekly, monthly, etc.

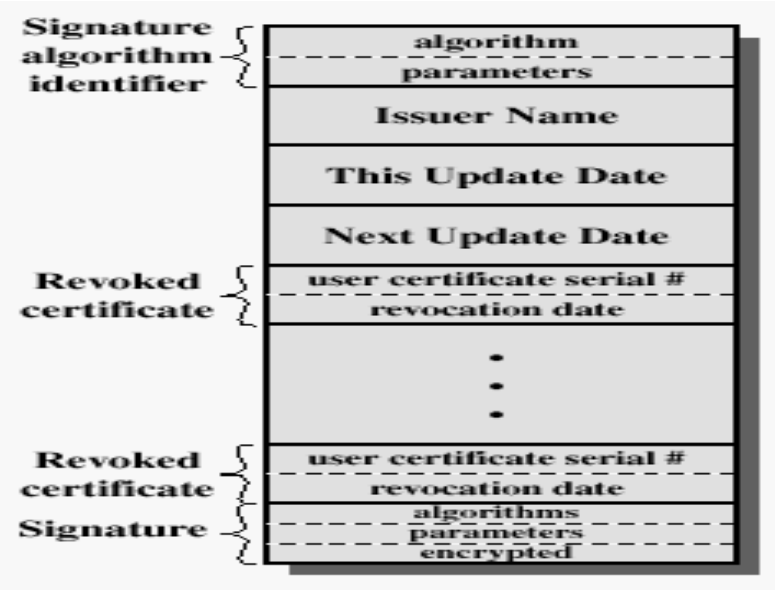
24

Pros & Cons of CRLs

- Pros
  - Simple
  - Don't need secure channels for CRL distribution
- Cons
  - Timeliness: "window of vulnerability"
  - CRLs can be huge
  - How to distribute CRLs reliably?

25

X.509 CRL Format



26

PKI and Revocation

- On January 29 and 30, 2001, VeriSign, Inc. issued two certificates for Authenticode Signing to an individual fraudulently claiming to be an employee of Microsoft Corporation.
- Any code signed by these certificates appears to be legitimately signed by Microsoft.
- Users who try to run code signed with these certificates will generally be presented with a warning dialog, but who wouldn't trust a valid certificate issued by VeriSign, and claimed to be for Microsoft?
- Certificates were very soon placed in a CRL, but:
  - code that checks signatures for ActiveX controls, Office Macros, and so on, didn't do any CRL processing.
- According to Microsoft:
  - since the certificates don't include a CRL Distribution Point (DP), it's impossible to find and use the CRL!

27

Certificate Revocation Tree (CRT)

- proposed by Kocher (1998)
- based on hash trees
  - hash trees first proposed by Merkle in another context in 1979 (one-time signatures)
  - improvement to Lamport-Diffie OTS scheme
  - based on the following idea:
    - A wants to sign a bit of information. A gives B the image (y) produced as y=F(x)
    - Eventually A reveals the pre-image: x
    - B checks that: y=F(x)

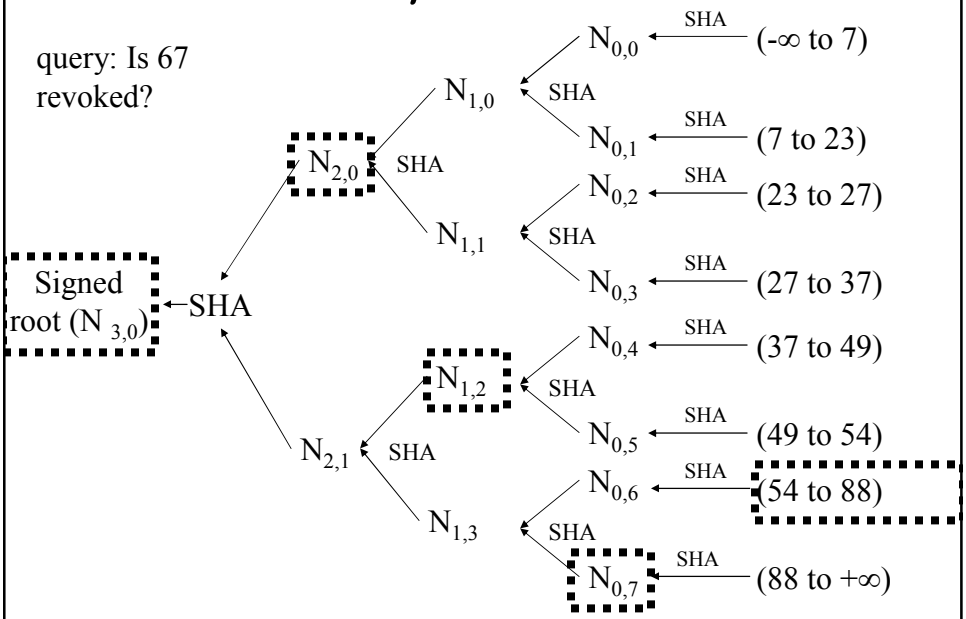
28

CRT contd.

- express ranges of SN of PKC's as tree leaf labels:
  - E.g., (5 -- 12) means: 5 and 12 are revoked, the others larger than 5 and smaller than 12 are okay
  - Place the hash of the range in the leaf
- response includes the corresponding tree leaf, the necessary hash values along the path to the root, the signed root
- the CA periodically updates the structure and distributes to un-trusted servers called Confirmation Issuers

29

Example of CRT



30

Characteristics of CRT

- each response represents a proof
- length of proof is: O(log n)
  - Much shorter than CRL which is O(n)
  - Where n is # of revoked certificates
- only one "real" signature for tree root (can be done off-line)

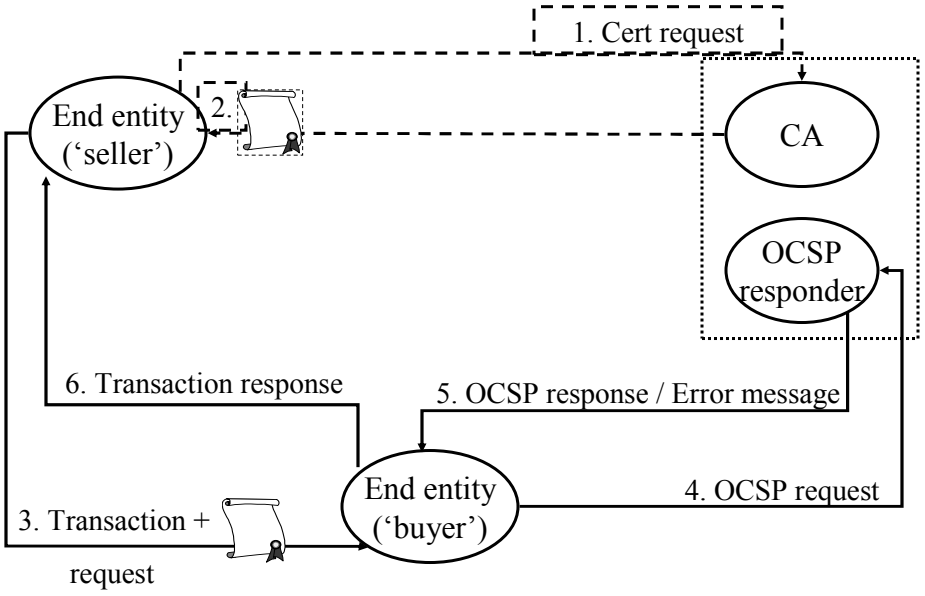
31

Explicit Revocation: OCSP

- OCSP = On-line Certificate Status Protocol (RFC 2560) - June 1999
- In place of or, as a supplement to, checking CRLs
- Obtain instantaneous status of a PKC
- OCSP may be used in sensitive, volatile settings, e.g., stock trades, electronic funds transfer, military

32

OCSP players



33

OCSP definitive response

- all definitive responses have to be signed:
  - either by issuing CA
  - or by a Trusted Responder (OCSP client trusts the TR's PKC)
  - or by a CA Authorized Responder which has a special PKC (issued by the CA) saying that it can issue OCSP responses on CA's behalf

34

Responses for each certificate

- Response format:
  - target PKC SN
  - PKC status:
    - good - positive answer
    - revoked - permanently/temporarily (on-hold)
    - unknown - responder doesn't know about the certificate being requested
  - response validity interval
  - optional extensions

35

Special Timing Fields

- A response contain three timestamps:
  - thisUpdate - time at which the status being indicated is known to be correct
  - nextUpdate - time at or before which newer information will be available
  - producedAt - time at which the OCSP responder signed this response. Useful for response pre-production

36

Security Considerations

- on-line method
- DoS vulnerability
  - flood of queries + generating signatures!
  - unsigned responses → false responses
  - pre-computing responses offers some protection against DoS, but…
- pre-computing responses allows replay attacks (since no nonce included)
  - but OCSP signing key can be kept off-line

37

Open questions

- Consistency between CRL and OCSP responses
  - possible to have a certificate with two different statuses.
- If OCSP is more timely and provides the same information as CRLs, do we still need CRLs?
- Which method should come first - OCSP or to CRL?

38

Implicit Revocation: Certificate Revocation System (CRS)

- proposed by Micali (1996)
- aimes to improve CRL communication costs / size
- basic idea: signing a message for every certificate stating its status
- use of off-line/on-line signature scheme to reduce update cost

39

CRS: creation of a certificate

- Two new parameters in PKC: Y_{MAX} and N

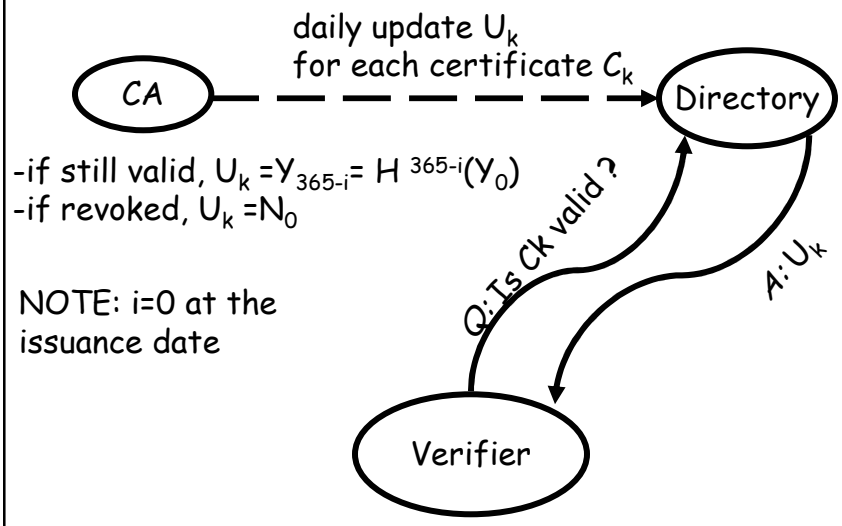
$$Y_{MAX} = H^{MAX}(Y_0)$$

$$N = H(N_0)$$

- [Y_0, N_0] -- per-PKC secrets stored by *CA*
- H() -- public one-way function

40

How CRS works



41