

Merkle's Puzzles (1974)

$0 < i < 2^n = N$

X_i, Y_i -- random secret keys

$index_i = \text{random (secret) value}$

Puzzle $P_i = \{index_i, X_i, S\}^{Y_i}$

S -- fixed string, e.g., "Alice to Bob"



Look up $index_j$

Obtain X_j

$\{P_i \mid 0 < i < 2^n\}$



Pick random j , $0 < j < 2^n$

Select P_j

Break Y_j by brute force

Obtain $\{index_j, X_j, S\}$

$index_j$

Encrypted communication with X_j



Is security computational or unconditional?

1

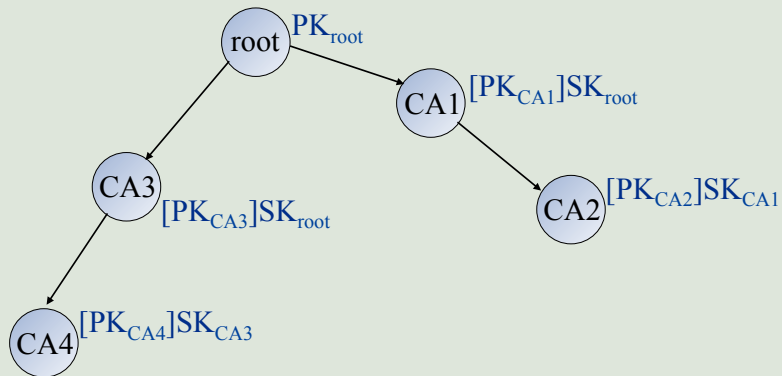
Lecture 18

Public Key Certification and Revocation

2

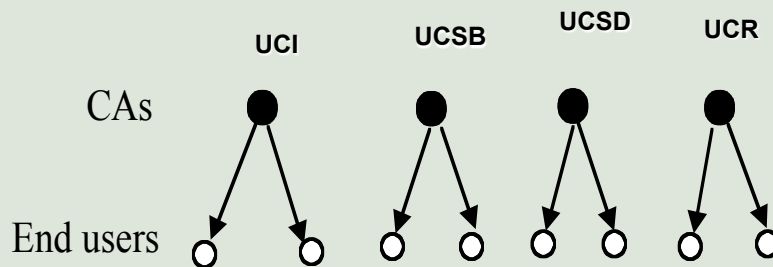
Certification Tree / Hierarchy

Logical tree of CA-s



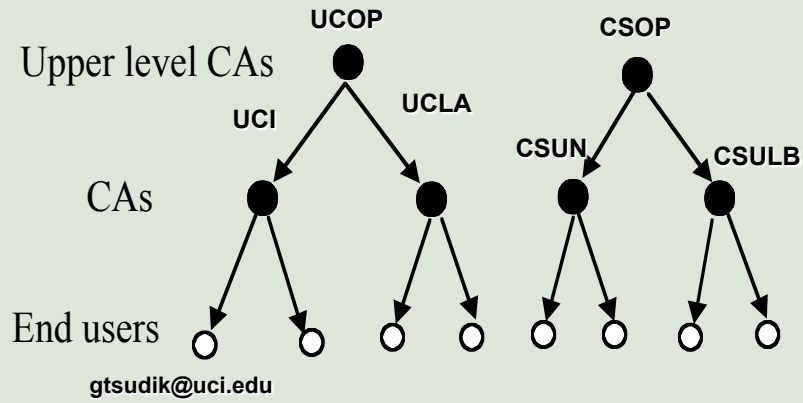
3

Hierarchical PKI Example



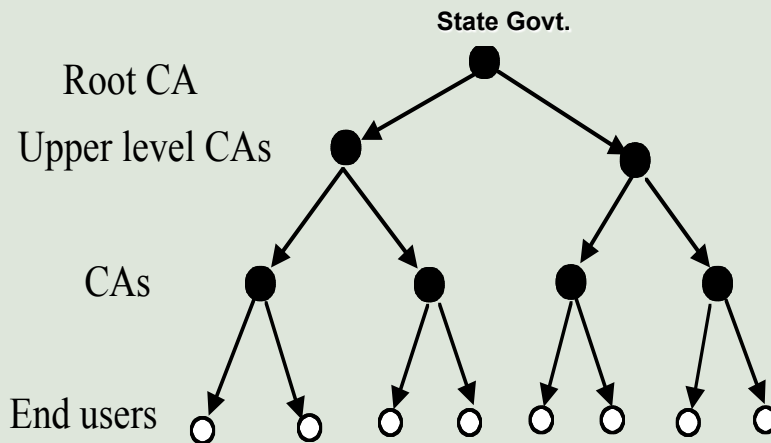
4

Hierarchical PKI Example



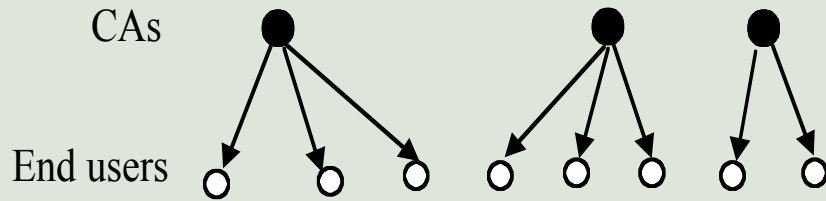
5

Hierarchical PKI Example



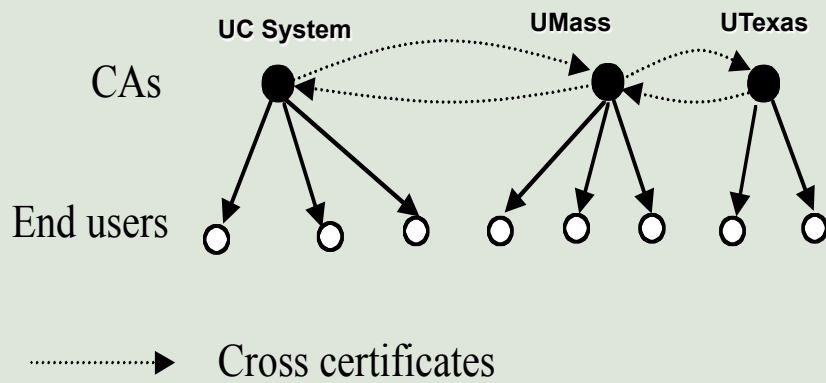
6

Cross Certificate Based PKI Example



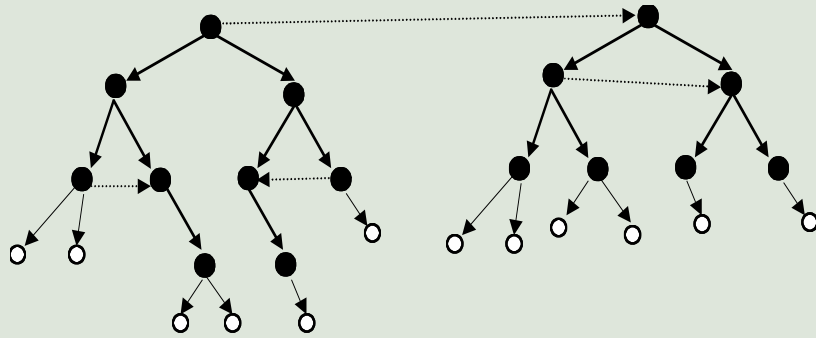
7

Cross Certificate Based PKI Example



8

Hybrid PKI example

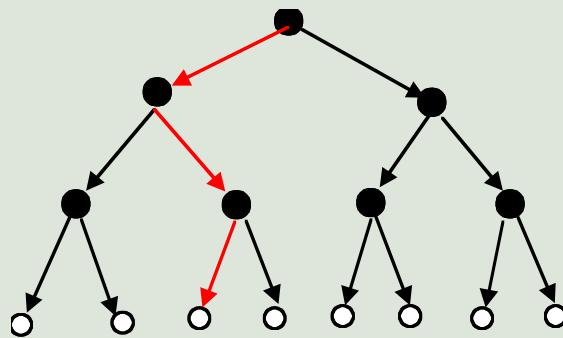


Note that no cross arrows down or up!

9

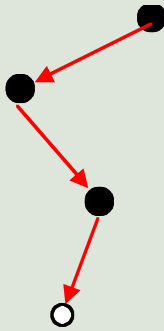
Certificate Paths

Derived from PKI



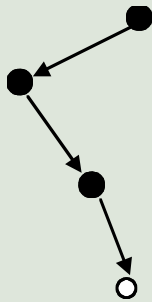
10

Certificate Paths



11

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

12

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**

13

X.509 Service

- ❖ Assumes a distributed set of servers maintaining a database about certificates
- ❖ Used in S/MIME, IPsec, SSL/TLS, SET.
- ❖ RSA, DSA, MD5*, SHA* are most commonly used algorithms

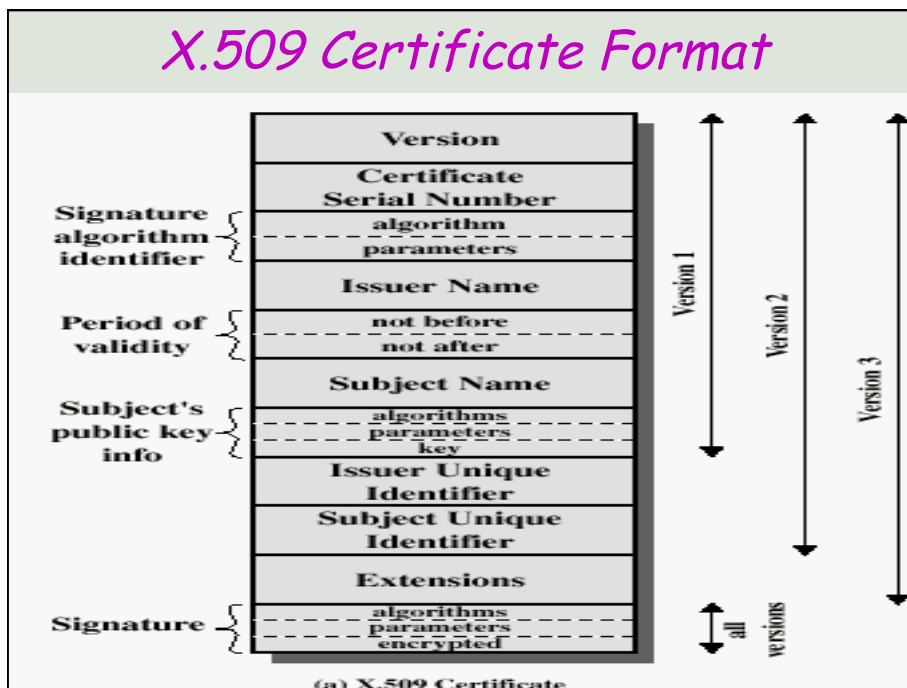
14

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

15

X.509 Certificate Format



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>

17

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 string is typically base-64 encoded to get an ASCII representation

18

Certificates in Practice

```
-----BEGIN CERTIFICATE-----
MIIDTzCCAvmgAwIBAgIBATANBgkqhkiG9w0BAQQFADBcMSEwHwYDVQQKEhhFdxJvcGVhbiBJQ0U0tVEVMIHByb2p1Y3QxIzAhBgNVBAsTG1YzLUN1cnRpZmljYXRpb24gQXV0aG9yaXR5MR1wEAYDVQHEw1EYXJtc3RhZHQwHhcNOTcwNDYMTczNTU5WhcN
OTgwNDYMTczNTU5WjBrMSEwHwYDVQQKEhhFdxJvcGVhbiBJQ0U0tVEVMIHByb2p1
Y3QxIzAhBgNVBAsTG1YzLUN1cnRpZmljYXRpb24gQXV0aG9yaXR5MR1wEAYDVQHE
Ew1EYXJtc3RhZHQwDTALBgNVBAMTBVTRVWVWTAkBgRVCAEBAgICAANLADBIkEA
qKhTY0kbk8PDC2yIEVXeFmri+VKg3GklxMi/VeExqM7kqSmFmYoVmt72L+G0UF9e
BHwM9HbcPA453Dg+PqRhiwIDAQABo4IBmDCCAZQwHwYDVROjBBgwFoAUfnLy+DqG
nEKINDRmdcPU/NGiETMwHQYDVR0OBByEFJfc4B8gjSoRmLUx4Sg/ucIYiMrPMA4G
A1UdDwEB/wQEAWIB8DAcBgNVHSABAF8EEjAQMAYGBCoDBAUwBgYECQgHBjBDBGNV
HREEPDA6gRV1c2VyQGRhcm1zdGFkdC5nbWQuZGwGIWh0dHA6Ly93d3cuZGFybnXN0
YWR0LmdtZC5kZS9+dXN1c2VYDVR0SBIgPmIGmgQxnbWRjYUBnbWQuZGwGEWh0
dHA6Ly93d3cuZ21kLmRlghdzYXR1cm4uZGFybnXN0YWR0LmdtZC5kZaRcMSEwHwYD
VQKKEhhFdxJvcGVhbiBJQ0U0tVEVMIHByb2p1Y3QxIzAhBgNVBAsTG1YzLUN1cnRp
ZmljYXRpb24gQXV0aG9yaXR5MR1wEAYDVQHEw1EYXJtc3RhZHSHDDE0MS4xMi42
Mi4yNjAMBGNVHRMBAf8EAJAAMB0GA1UdHwQWMBQwEQAQoA6BDGdtZGNhQGdtZC5k
ZTANBgkqhkiG9w0BAQQFAANBAGkM4ben8tj76GnAE803rSEGIk3oxtvxBAu34LPW
DIEDzsNqPsfNJCSkkmTCg4MGQlMObwkehJr3b2Ob1JmD1qQ=
-----END CERTIFICATE-----
```

19

Certificate Revocation Scenario

What if:

- ❖ Bob's CA goes berserk?
- ❖ Bob forgets his private key?
- ❖ Someone steals Bob's private key?
- ❖ Bob loses his private key?
- ❖ Bob willingly discloses his private 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?

20

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 List (CRL) lists certificates that are not yet naturally expired but revoked
- ❖ CRL reissued periodically, even if no activity!
- ❖ More on revocation later...

21

Requirements for revocation

- ❖ Timeliness
 - ◆ Before using a certificate, must check most recent revocation status
- ❖ Efficiency
 - Computation
 - Bandwidth and storage
 - Availability
- ❖ Security

22

Types of Revocation

❖ Implicit

- ◆ Each certificate is periodically (re-)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

23

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

24

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.

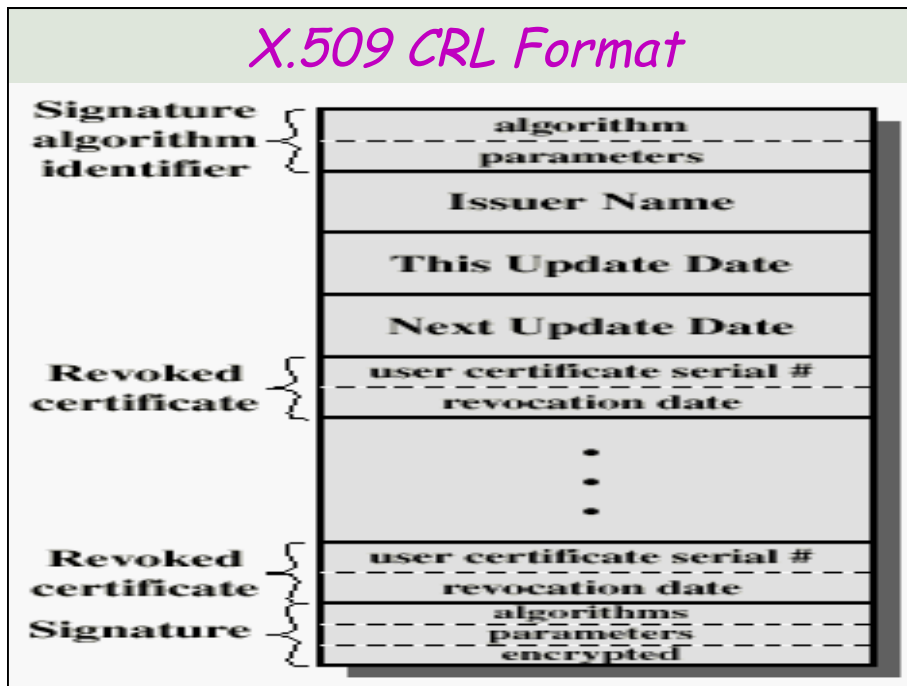
25

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?

26

X.509 CRL Format



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!

Certificate Revocation Tree (CRT)

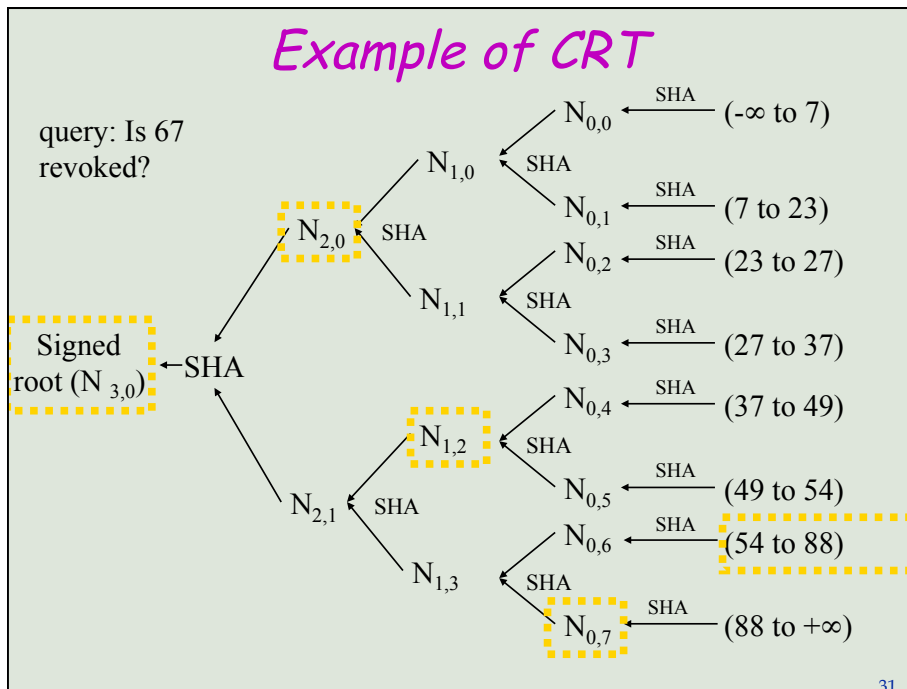
- ❖ proposed by P. Kocher (1998)
- ❖ based on hash trees
 - hash trees first proposed by R. Merkle in another context in 1979 (one-time signatures)
 - improvement to Lamport-Diffie OTS scheme
 - based on the following idea:
 - ◆ A wants to sign a bit of information. A gives B the image (y) produced as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

29

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

30

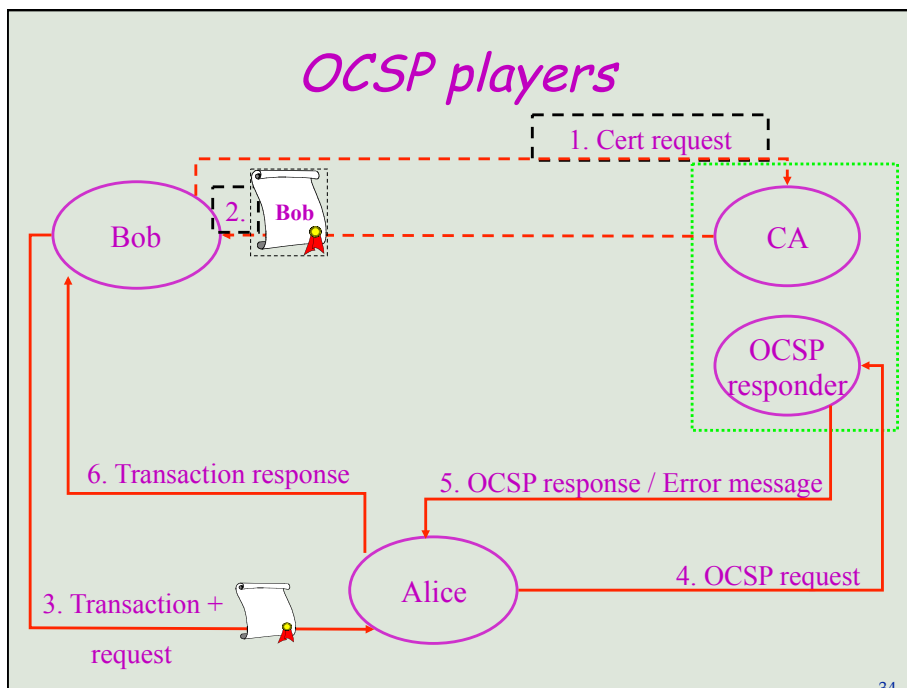


- ### Characteristics of CRT
- ❖ each response represents a proof
 - ❖ length of proof is: $O(\log n)$
 - Much shorter than CRL which is $O(n)$
 - Where n is # of revoked certificates
 - ❖ only one “real” signature for tree root (can be done off-line)
- 32

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

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

35

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

36

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**

37

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

38

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?

39

Implicit Revocation: Certificate Revocation System (CRS)

- ❖ proposed by Micali (1996)
- ❖ aims 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

40

CRS: creation of a certificate

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

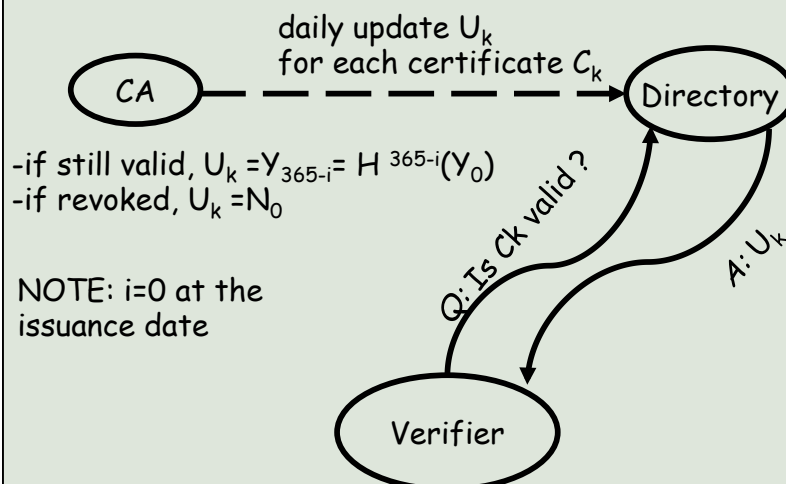
$$Y_{MAX} = H^{MAX}(Y_0)$$

$$N = H(N_0)$$

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

41

CRS example: certificate issued for 1 year



42