

PKI

Part 3: Key and Certificate Management

Kjell Jørgen Hole
NoWires Research Group
Department of Informatics
University of Bergen

last changed September 29, 2008

Outline

- Key / certificate life-cycle management
 - a) initialization phase
 - b) issued phase
 - c) cancellation phase
- Building and validating certification paths

Life-cycle management

- **Key/certificate life-cycle management** covers the functions associated with **creation**, **issuance**, and **cancellation** of public-private key pairs and their associated certificates
- note that this definition concerns the **keying material** associated with an entity—not the identity of the entity

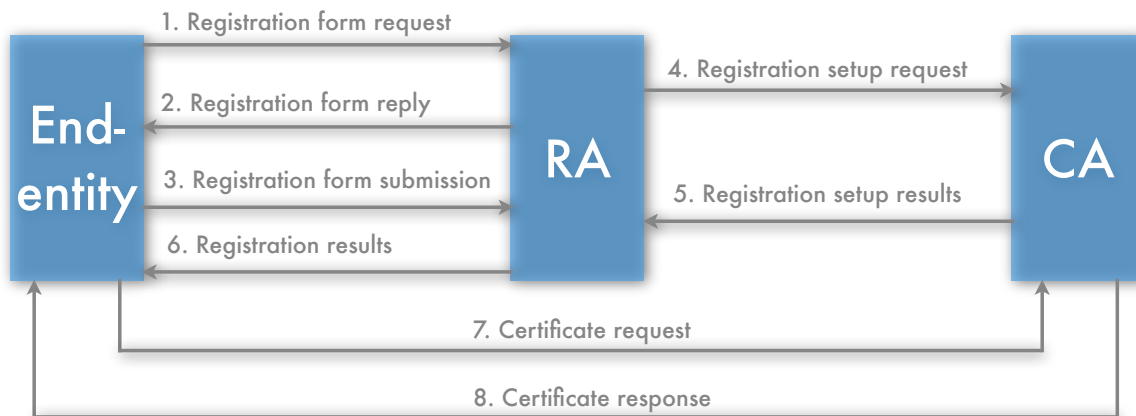
3

Initialization phase

- **Initialization:**
 1. end-entity registration
 2. key pair generation
 3. certificate creation
 4. certificate dissemination
 5. key backup

4

1. End-entity registration



5

Registration ...

- An on-line registration can be used to obtain a certificate for an e-mail application
 - the registration must be authenticated and protected
- A registration process to authorize multimillion dollar transactions requires
 - physical presence at the RA
 - identification documents containing a photo
 - requisite authorization forms

Shared secret

- The registration process typically includes assigning a **shared secret** to the end entity
- The shared secret is used by the end entity to request a certificate (see Step 7 in fig.)
- Sometimes a preexisting shared secret can be used to simplify the registration

7

2. Key pair generation

- A key pair may be generated within the
 - end entity's client system
 - RA
 - CA
- A trusted third party facility may also be used for key generation

8

Key pair generation ...

It is recommended that keys are generated within the client system, especially when the keys are to be used for *non-repudiation* services

- An alternative view states that the keys should be generated within the CA since this is the most trusted entity within the PKI
- performance limitations may also dictate that key generation is done within the CA

9

Two-key pair model

- Key-pairs:
 - one key pair for **confidentiality** services
 - one key pair for **non-repudiation** services
- Different key-pairs are needed because they must satisfy different management requirements (see next page)

10

Why use two key-pairs?

	private key for digital signature	private key for decryption
local storage	yes	yes
third party backup	no	yes
destroy after active lifetime	yes	no
long-term archive	no	yes

11

More than two key-pairs?

- It may be necessary to have different pairs of keys for digital signing, e.g.
 - one private key for signing of large purchase orders,
 - another key to sign rental forms at the video store,
 - and yet another key to sign e-mails

12

Private-key protection

- Several methods exist to protect private keys:
 1. storage in a tamper-resistant smart card or PCMCIA card (recommended method)
 2. storage in an encrypted file on a hard drive or other storage medium
 3. storage on a server
- Access to a key needs to be protected via a password or PIN in all cases

13

3. Certificate creation

- An authorized CA is always responsible for the creation of certificates
- The CA generates a certificate based on the public key received from the client and the information obtained from the RA
- Completed certificates may be distributed directly to the end entities or to a repository

14

Certificate creation ...

- A secure protocol is needed to request a new certificate and to receive a certificate from a CA
- Two RFCs of interest are:
 - RFC2510, *The Internet X.509 Public Key Infrastructure Certificate Management Protocols (CMP)*
 - RFC2511, *The Internet X.509 Certificate Request Message Format (CRMF)*

15

4. Certificate dissemination

- The certificates must be conveyed to other end entities using
 - out-of-band distribution, e.g. physical delivery
 - repository or database to facilitate on-demand and on-line retrieval
 - in-band protocol distribution

16

5. Key backup

- A business needs to recover encrypted data inaccessible to owners due to loss of keys or malfunctioning smart cards
- The private key and certificate may be stored at a trusted third party as a protection during the time the certificate is valid
- archiving is used for long-term storage

17

Key backup ...

- In addition, end-entities may back up their own keys and certificates

Private keys used to support non-repudiation services should *never* be backed up by a third party

18

Issued phase

Issued:

1. certificate retrieval
2. certificate validation
3. private-key recovery
4. key-pair update

19

1. Certificate retrieval

- Certificate retrieval is the process of obtaining an end-entity certificate when and as needed
- An end entity retrieves a certificate to
 1. encrypt data destined for another end entity
 2. verify a digital signature received from another end entity

20

Certificate retrieval ...

- When the sender and receiver have obtained each other's certificates, the parties can agree on a secret symmetric key for data encryption
- When signing data, the certificate of the signer is often sent with the data to avoid having to obtain the certificate from a remote repository

21

2. Certificate validation

1. Verify that the certificate has been issued by a recognized trust point / anchor
2. Digital signature is valid
3. The certificate is within its established validity period
4. The certificate has not been revoked
5. The certificate is used according to the policy

22

3. Private-key recovery

- Some end users will lose access to the private-keying material that is used for decryption purposes
- Necessary to store copies of private keys at a remote backup facility such as a trusted
 - key recovery center, or
 - CA

23

4. Key-pair update

- When a certificate is close to its expiration date, it is necessary to issue a new public-private key pair and the associated certificate
- It is suggested that key updates should occur once 70% to 80% of the current key lifetime has been exhausted

24

Cancellation phase

Cancellation:

1. certificate expiration
2. certificate revocation
3. key history
4. key archive

25

1. Certificate expiration

- When a certificate expires, one of the following events occur:
 - **no action** because the end entity is no longer enrolled in the PKI
 - **certificate renewal** with the same public key placed into a new certificate
 - **certificate update** with a new public-private key pair and certificate

26

Renewal versus update

- **Certificate renewal** preserves the original public-private key pair, whereas a new key pair is generated during **certificate update**
- Certificate renewal may be used when
 1. circumstances/policies associated with certificate issuance have not changed
 2. the cryptographic strength of the key pair is still thought to be sound

27

2. Certificate revocation

- A certificate must be revoked before it expires when there is a
 - private-key compromise
 - change in job status
 - termination of employment
- An end user or an authorized administrator may request a certificate revocation

28

3. Key history

- A private key for decryption expires when the corresponding public-key certificate expires
- It is often necessary to reliably and securely store a private key after it has expired to recover old encrypted data
- Typically, this **key history** storage is local to the key owner

29

4. Key archive

- **Key archive** is the long-term storage of certificates at a CA or another trusted party
- This service is coupled with
 - time-stamping services
 - audit trails
 - restoration of an end entity's key history (when local history is lost)

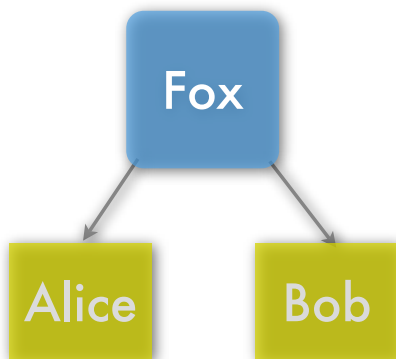
30

Certification paths revisited

- Certification paths were introduced in Part 1, and PKI architectures were first described in Part 2
- We now briefly discuss the **building** and **validation** of certification paths for different PKI architectures

31

Single CA

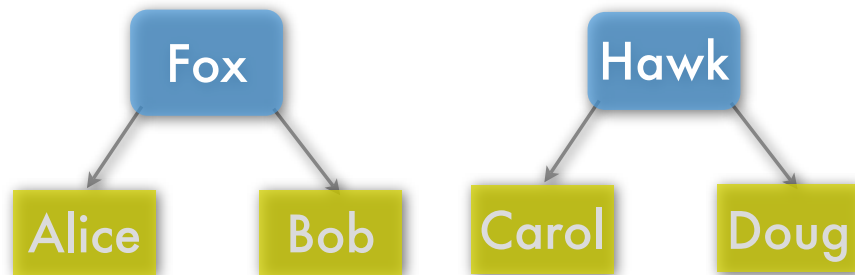


- A certification path consists of a single certificate
- Certification paths:
 - Alice: Fox → Alice
 - Bob: Fox → Bob

32

CA trust list

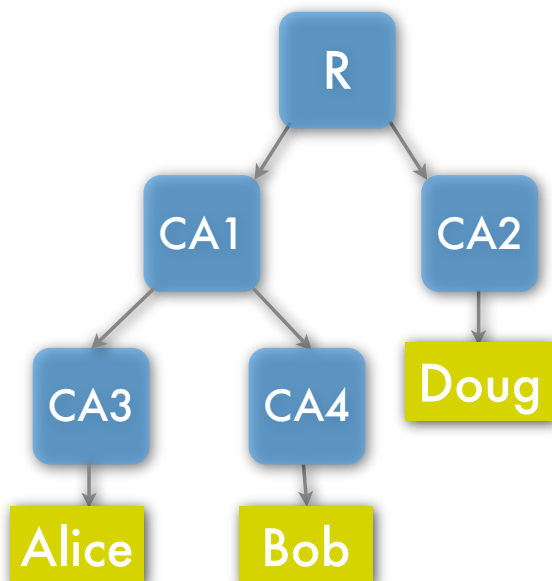
Trust list
Fox
Hawk
...



- All certification paths consist of single certificates
- Carol: Hawk → Carol

33

Hierarchical PKI



- Alice:
Root → CA1 → CA3 → Alice
- Bob:
Root → CA1 → CA4 → Bob
- Dough:
Root → CA2 → Dough

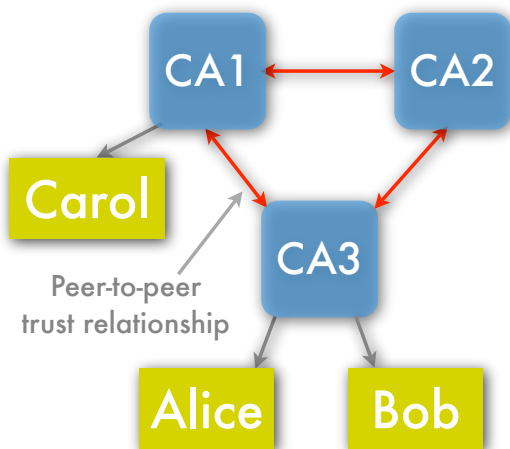
34

Hierarchical PKI ...

- A certification path originates at the root and terminates at the end entity
- Note, however, that the construction of the path starts at the end entity

35

Mesh PKI



- The CA that issued an end user's certificate is the trust point
- Paths constructed by Alice:
 - Bob: CA3 → Bob
 - Carol: CA3 → CA1 → Carol

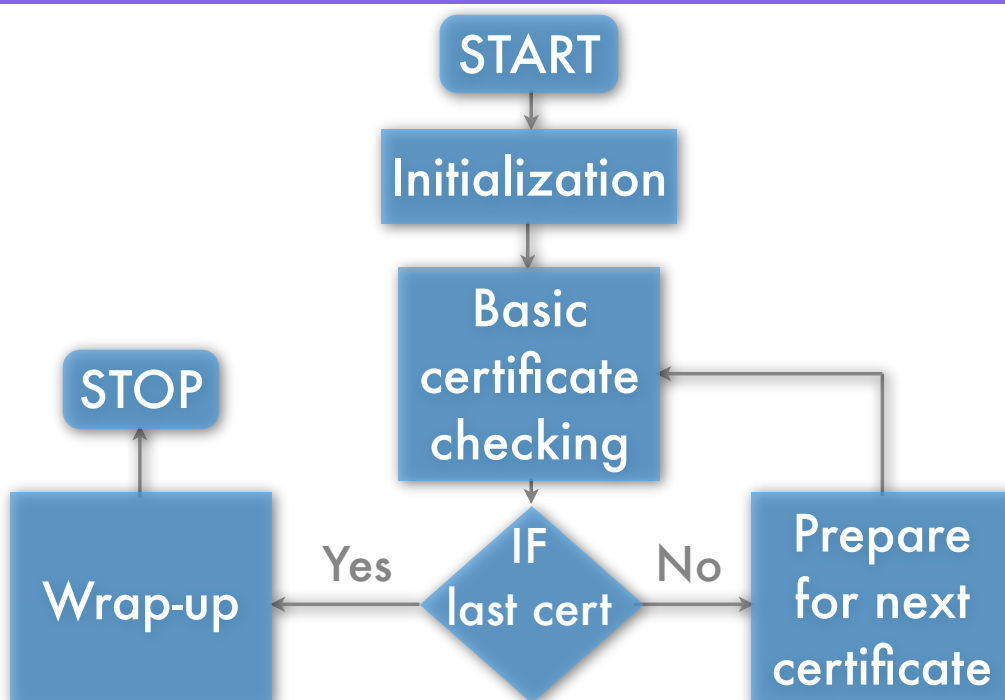
36

Mesh PKI ...

- Note that there may be more than one path to a given end entity
 - Paths constructed by Alice:
 - Carol: CA3 → CA1 → Carol
 - Carol: CA3 → CA2 → CA1 → Carol
 - Paths can contain loops

37

Certification path validation



Initialization

- Possible inputs to **initialization**:
 - potential certification path
 - set of acceptable certification policy identifiers (often set to any-policy)
 - trust point information (self-signed certificate)

39

Basic certificate checking

- **Certificate validity**: cert. inside validity period
- **Certificate revocation**: cert. not revoked
- **Certificate signature**: validate digital signature
- **name chaining**: issuer and subject names
- **Certificate policies**:
- **Name constraints**: satisfies X.500 DNs

40

Summary

- Key management includes many operations to create, store and remove public-private key pairs
- It is necessary to build and validate certification paths in more “advanced” PKI architectures

41

Sources

- C. Adams and S. Lloyd, [Understanding PKI](#), 2nd Edition, Addison Wesley, 2003
- W. Ford and M. S. Baum, [Secure Electronic Commerce](#), 2nd Edition, Prentice Hall, 2001
- R. Housley and T. Polk, [Planning for PKI](#), Wiley, 2001

42

