Management of Asymmetric keys

Problems to solve

Ensure proper and correct use of asymmetric key pairs

Privacy of private keys

- To prevent the repudiation of digital signatures

Correct distribution of public keys

- To ensure confidentiality
- To ensure the correct validation of digital signatures

Problems to solve

Temporal evolution of entity <-> key pair mappings

To tackle catastrophic occurrences

- e.g. loss of private keys

To tackle normal exploitation requirements

- e.g. refresh of key pairs for reducing impersonation risks

Ensure a proper generation of key pairs

- Random generation of secret values
- Increase efficiency without reducing security

Problems to solve

Ensure a proper generation of key pairs

Random generation of secret values

- So that they cannot be easily predicted

Increase efficiency without reducing security

- Make security mechanisms more useful
- Increase performance

Goals

1. **Key pair generation**
   - When and how should they be generated
2. **Handling of private keys**
   - How do I maintain them private
3. **Distribution of public keys**
   - How are they correctly distributed worldwide
4. **Lifetime of key pairs**
   - Until when should they be used
   - How can I check the obsolesce of a key pair

Generation of key pairs: Design principles

Good random generators for producing secrets

Result is indistinguishable from noise

- All values have equal probability
- No patterns resulting from the iteration number or previous values

Example: Bernoulli ½ generator

- Memoryless generator
- P(b=1) = P(b=0) = ½
- Coin toss

Generation of key pairs: Design principles

Facilitate without compromising security

Efficient public keys

- Few bits, typically 2^k+1 values (3, 17, 65537)
- Accelerates operations with public keys
- No security issues

Generation of key pairs: Design principles

Self-generation of private keys

Maximizes privacy as no other party will be able to use a given private key

Principle can be relaxed when not involving signature generation

Handling of private keys

Correctness

The private key represents a subject

- Its compromise must be minimized
- Physically secure backup copies can exist in some cases

The access path to the private key must be controlled

- Access protection with password or PIN
- Correctness of applications that use it

Handling of private keys

Confinement

Protection of the private key inside a (reduced) security domain (ex. cryptographic token)

- The token generates key pairs
- The token exports the public key but never the private key
- The token internally encrypts/decrypts with the private key

Example: SmartCards

- We ask the SmartCard to cipher/decipher something
- The private key never leaves the SmartCard

Distribution of public keys

Distribution to all senders of confidential data

- Manual
- Using a shared secret
- Ad-hoc using digital certificates

Distribution to all receivers of digital signatures

- Manual
- Ad-hoc using digital certificates

Distribution of public keys

Trustworthy dissemination of public keys

- Trust paths / graphs

If A trusts K_X^+, and B trusts A, then B trusts K_X^+

- Certification hierarchies / graphs
  - With the trust relations expressed between entities
  - Certification is unidirectional!

Public key (digital) certificates

Digital Document issued by a Certification Authority (CA)

Binds a public key to an entity

- Person, server or service

Are public documents

- Do not contain private information, only public one
- Can have additional binding information (URL, Name, email, etc..)

Are cryptographically secure

- Digitally signed by the issuer, cannot be changed

Public key (digital) certificates

Can be used to distribute public keys in a trustworthy way

A certificate receiver can validate it

- With the CA's public key

If the signer (CA) public key is trusted, and the signature is correct, then the receiver can trust the (certified) public key

- As the CA trust the public key, if the receiver trusts on the CA public key, the receiver can trust on the public key

Public key (digital) certificates

X.509v3 standard

- Mandatory fields
  - Version
  - Subject
  - Public key
  - Dates (issuing, deadline)
  - Issuer
  - Signature
  - etc.
- Extensions
  - Critical or non-critical

PKCS #6

- Extended-Certificate Syntax Standard

Binary formats

- ASN.1 (Abstract Syntax Notation)
  - DER, CER, BER, etc.
- PKCS #7
  - Cryptographic Message Syntax Standard
- PKCS #12
  - Personal Information Exchange Syntax Standard

Other formats

- PEM (Privacy Enhanced Mail)
- base64 encoding of X.509

Key pair usage

The public certificate binds the key pair to a usage profile

- Public keys are seldom multi-purpose

Typical usage profiles

- Authentication / key distribution
  - Digital signature, Key encipherment, Data encipherment, Key agreement
- Document signing
  - Digital signature, Non-repudiation
- Certificate issuing
  - Certificate signing, CRL signing

Public key certificates have an extension for this

- Key usage (critical)

Certification Authorities (CA)

Organizations that manage public key certificates

Define policies and mechanisms for:

- Issuing certificates
- Revoking certificates
- Distributing certificates
- Issuing and distributing the corresponding private keys

Manage certificate revocation lists

- Lists of revoked certificates

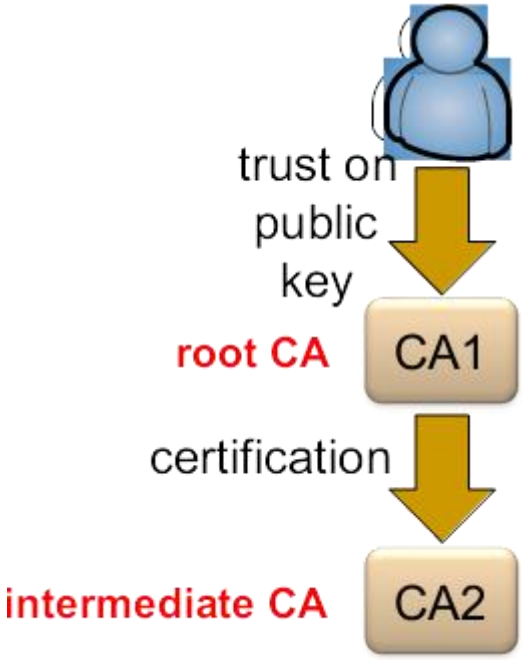
Trusted Certification Authorities

Intermediate CAs: CAs certified by other trusted CAs

- Using a certificate
- Enable the creation of certification hierarchies

Trusted anchor (or certification root) : One has a trusted public key

- Usually implemented by self-certified certificates
  - Issuer = Subject
- Manual distribution
  - e.g. within browsers code (Firefox, Chrome, etc.), OS, distribution...



***.ua.pt**

Issued by: TERENA SSL CA 3

Expires: Thursday 19 July 2018 at 13 h 00 min 00 s Western European Summer Time

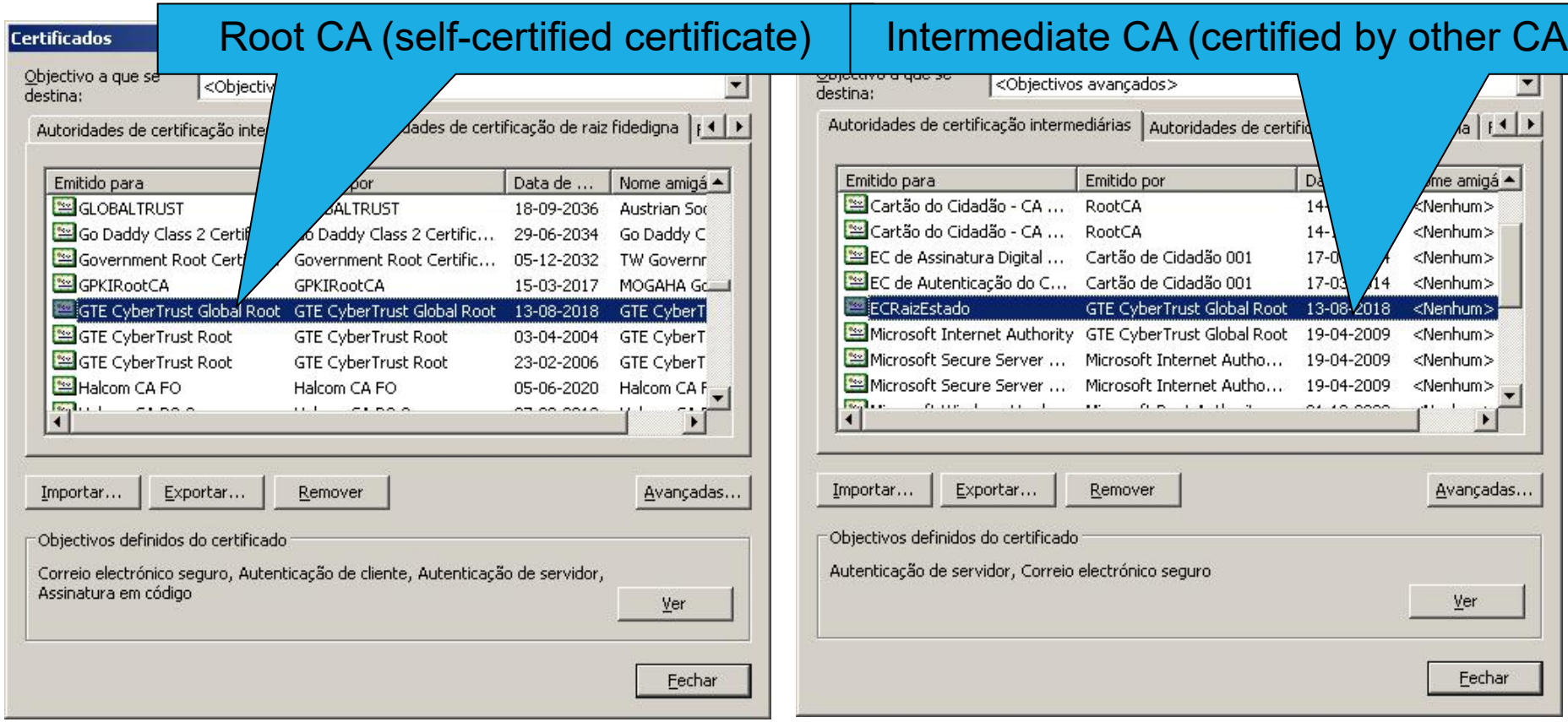
This certificate is valid

Details

Subject Name	
Country	PT
Locality	Aveiro
Organization	Universidade de Aveiro
Organizational Unit	sTIC
Common Name	*.ua.pt
Issuer Name	
Country	NL
State/Province	Noord-Holland
Locality	Amsterdam
Organization	TERENA
Common Name	TERENA SSL CA 3
Serial Number	0E DF B0 1F D8 2A DE 67 C9 E9 7C D1 68 2E C0 33
Version	3
Signature Algorithm	SHA-256 with RSA Encryption (1.2.840.113549.1.1.11)
Parameters	none
Not Valid Before	Wednesday 15 July 2015 at 01 h 00 min 00 s Western European Summer Time
Not Valid After	Thursday 19 July 2018 at 13 h 00 min 00 s Western European Summer Time
Public Key Info	
Algorithm	RSA Encryption (1.2.840.113549.1.1.1)
Parameters	none
Public Key	256 bytes : E5 9F C2 04 0C D7 15 C9 ...
Exponent	65537
Key Size	2048 bits
Key Usage	Encrypt, Verify, Wrap, Derive
Signature	256 bytes : 54 4C 6E 63 AF 8D B8 A7 ...

Extension	Key Usage (2.5.29.15)
Critical	YES
Usage	Digital Signature, Key Encipherment
Extension	Basic Constraints (2.5.29.19)
Critical	YES
Certificate Authority	NO
Extension	Extended Key Usage (2.5.29.37)
Critical	NO
Purpose #1	Server Authentication (1.3.6.1.5.5.7.3.1)
Purpose #2	Client Authentication (1.3.6.1.5.5.7.3.2)
Extension	Subject Key Identifier (2.5.29.14)
Critical	NO
Key ID	47 C4 9C 28 CA 35 93 B3 DB 3A AC 69 40 ED 1B 8B 45 74 DB 36
Extension	Authority Key Identifier (2.5.29.35)
Critical	NO
Key ID	67 FD 88 20 14 27 98 C7 09 D2 25 19 BB E9 51 11 63 75 50 62
Extension	Subject Alternative Name (2.5.29.17)
Critical	NO
DNS Name	*.ua.pt
DNS Name	ua.pt
Extension	Certificate Policies (2.5.29.32)
Critical	NO
Policy ID #1	(2.16.840.1.114412.1.1)
Qualifier ID #1	Certification Practice Statement (1.3.6.1.5.5.7.2.1)
CPS URI	https://www.digicert.com/CPS
Extension	CRL Distribution Points (2.5.29.31)
Critical	NO
URI	http://crl3.digicert.com/TERENASSLCA3.crl
URI	http://crl4.digicert.com/TERENASSLCA3.crl

Manual distribution of trusted public keys (as root certificates):



Certification hierarchies (or path): Portuguese Citizen Card

Certification hierarchies: PEM (Privacy Enhanced Mail) model

Distribution of certificates for Privacy-enhanced Electronic Mail

IETF Proposed Standard in 1993 (RFC1421-1423)

Worldwide hierarchy (monopoly model)

- Single root (IPRA)
- Several PCA (Policy Creation Authorities) bellow the root
- Several CA below each PCA
  - Possibly belonging to organizations or companies
- Certification paths

Certification hierarchies: PEM (Privacy Enhanced Mail) model

Model was never actively deployed

- Except for a small number of implementations (90s)

Forest of hierarchies below CAs without a root PCA

- Independent hierarchies with an independent root CA
- Oligarchy

Each root CA negotiates the distribution of its public key along with some applications or operating systems

- ex. Browsers, Operating Systems

Certification hierarchies: PGP (Pretty Good Privacy) Model

Web of trust model

No central trustworthy authorities

- Each person is a potential certifier
- Anyone can certify a public key (issue a certificate) and publish the signature for others

People uses two kinds of trust

- Trust in the keys they know
  - Validated using any means (FAX, telephone, direct meeting, etc.)
- Trust in the correct behavior of certifiers
  - Assuming they know what they are doing when issuing a certificate

Certification hierarchies: PGP (Pretty Good Privacy) Model

Transitive trust

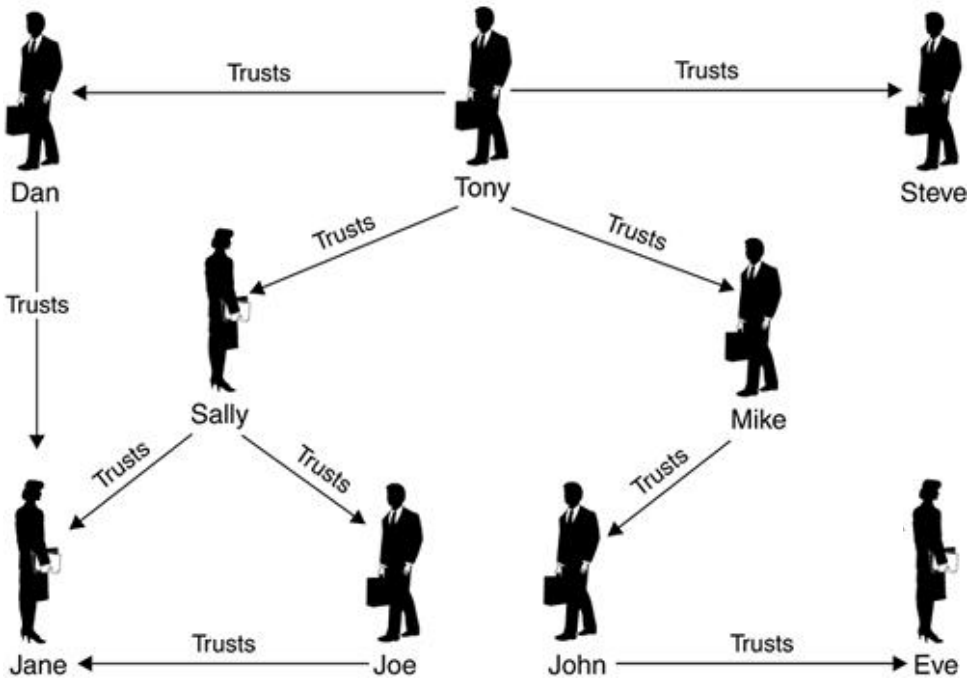
If

Mike trusts John is a correct certifier;
and

John certified the public key of Eve,

Then

Mike trusts Eve's public key



PGP public key certificates: Validity vs. trust

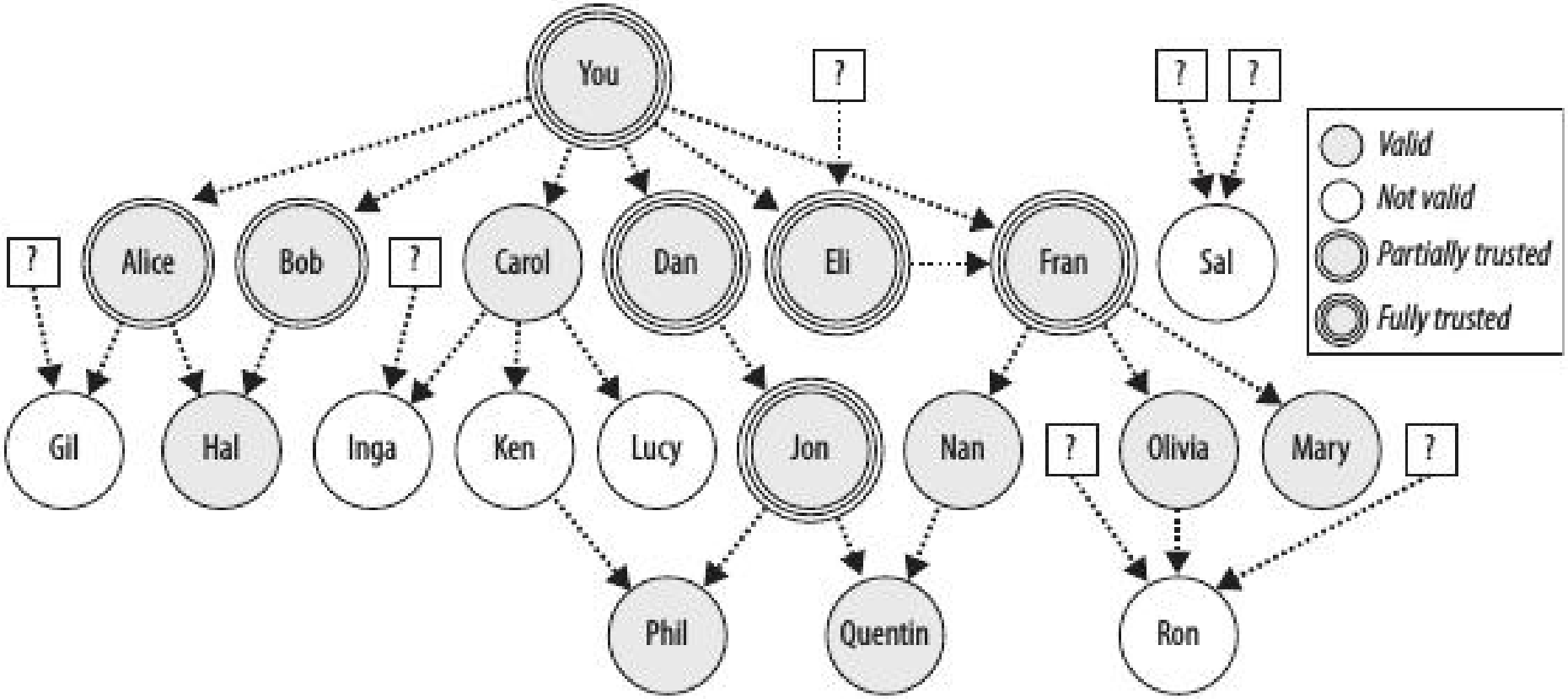
Trust: How much one trusts the other person

- Trust is unidirectional, personal and subjective
- Levels:
  - Ultimate (our own keys, we have the private key)
  - Complete trust
  - Marginal trust
  - Notrust (or Untrusted)

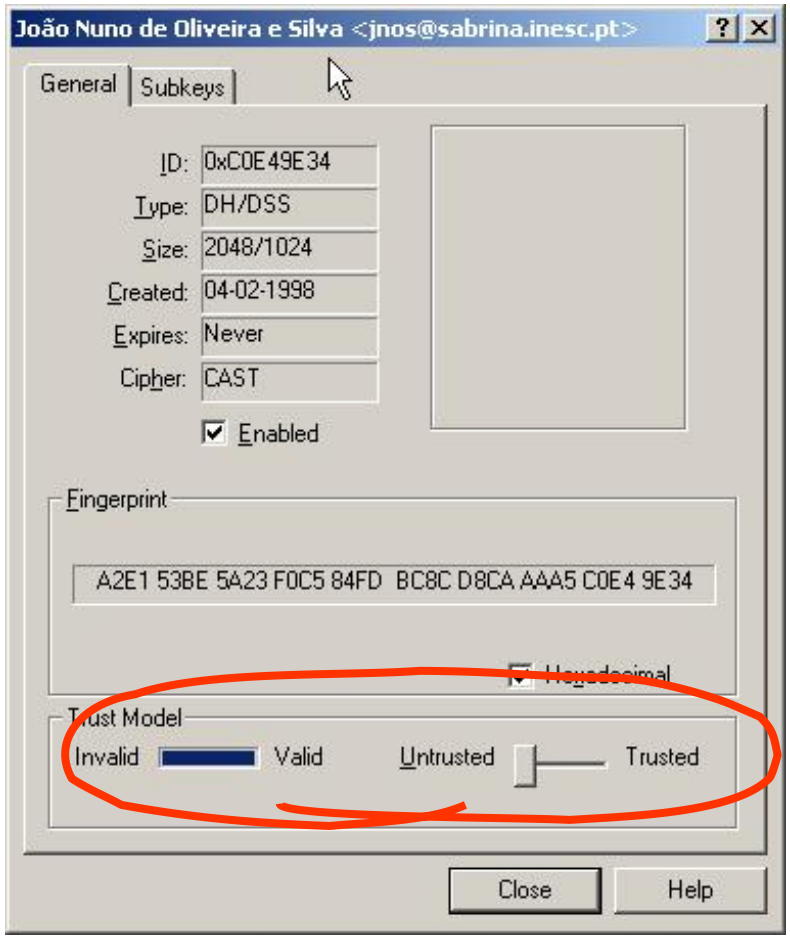
Validity: How much verification this key has (eg, A regarding user E)

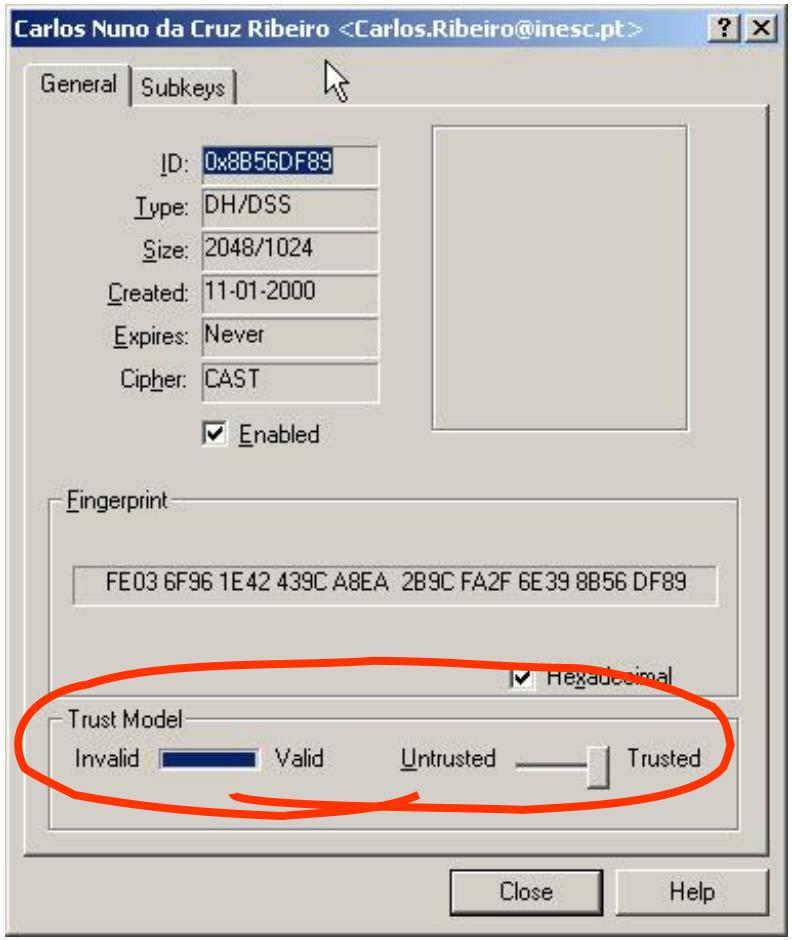
- **Valid:** A completely trusts B, or marginally trusts C and D; D or B and C signed E key
- **Marginally valid**: A marginally trusts B and B signed E key
- **Invalid**: No path

PGP public key certificates:
Validity vs. trust



PGP public key certificates:
Validity vs. trust





Refreshing of asymmetric key pairs

Key pairs should have a limited lifetime

- Because private keys can be lost or discovered
- To implement a regular update policy

Problem

- Certificates can be freely copied and distributed
- The universe of holders of certificates is unknown
  - Therefore we cannot contact them to eliminate specific certificates

Solutions

- Certificates with a validity period (not before, not after)
- Certificate revocation lists
  - To revoke certificates before expiring their validity

Certificate revocation lists (CRL)

Base or delta

- Complete / differences

Signed lists of certificate (identifiers) prematurely invalidated

- Must be regularly consulted by certificate holders
- OCSP protocol for single certificate validation
  - RFC 2560
- Can tell the revocation reason

Publication and distribution of CRLs

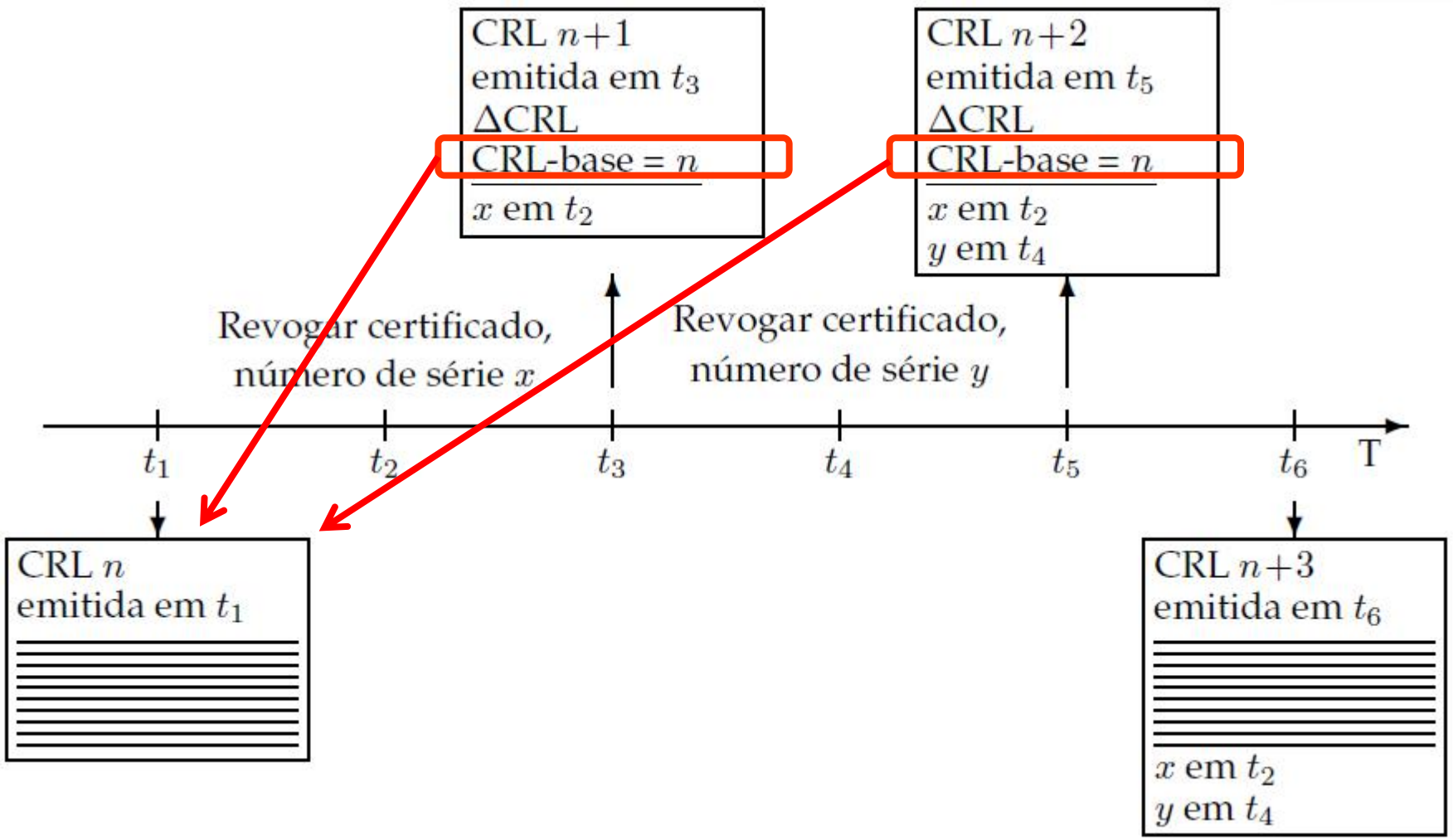
- Each CA keeps its CRL and allows public access to it
- CAs exchange CRLs to facilitate their widespreading

RFC 3280

unspecified (0)
keyCompromise (1)
CACompromise (2)
affiliationChanged (3)
superseded (4)
cessationOfOperation (5)
certificateHold (6)

removeFromCRL (8)
privilegeWithdrawn (9)
AACompromise (10)

CRL and Delta CRL



Online Certificate Status Protocol

HTTP based protocol to assert certificate status

- Request includes the certificate serial number
- Response states if the certificate is revoked
  - Response is signed by the CA and has a validity
- One check per certificate

Request lower bandwidth to clients

- One check per certificate instead of a bulk download of the CRL

Involves higher bandwidth to CAs

- One check per certificate
- Privacy issues as the CA will know that a system is accessing a service

OCSP Stapling

- Including a recently signed timestamp in the server response to assert validity
- Reduces verification delay and load on CA

Distribution of public key certificates

Transparent (integrated with systems or applications)

- Directory systems
  - Large scale (ex. X.500 through LDAP)
  - Organizational (ex. Windows 2000 Active Directory (AD), Manually (UA IDP))
- On-line: within protocols using certificates for peer authentication
  - eg. secure communication protocols (TLS, IPSec, etc.)
  - eg. digital signatures within MIME mail messages or within documents

Explicit (voluntarily triggered by users)

- User request to a service for getting a required certificate
  - eg. request sent by e-mail
  - eg. access to a personal HTTP page

PKI (Public Key Infrastructure) (1/2)

Infrastructure for enabling a proper use of asymmetric keys and public key certificates

Creation of asymmetric key pairs for each enrolled entity

- Enrolment policies
- Key pair generation policies

Creation and distribution of public key certificates

- Enrolment policies
- Definition of certificate attributes

PKI (Public Key Infrastructure) (2/2)

Definition and use of certification chains (or paths)

- Insertion in a certification hierarchy
- Certification of other CAs

Update, publication and consultation of CRLs

- Policies for revoking certificates
- Online CRL distribution services
- Online OCSP services

Use of data structures and protocols enabling inter-operation among components / services / people

PKI Example: Citizen Card

Enrollment

- In loco, personal enrolment

Multiple key pairs per person

- One for authentication
- One for signing data
- Both generated inside smartcard, not exportable
- Both require a PIN to be used in each operation

Certificate usage (authorized)

- Authentication
  - SSL Client Certificate, Email (Netscape cert. type)
  - Signing, Key Agreement (key usage)
- Signature
  - Email (Netscape cert. type)
  - Non-repudiation (key usage)

Certification path

- Uses a well-known, widely distributed root certificate
  - GTE Cyber Trust Global Root
- PT root CA below GTE
- CC root CA below PT root CA
- CC Authentication CA and CC signature CA below CC root CA

CRLs

- Signature certificate revoked by default
  - Revocation is removed if the CC owner explicitly requires the usage of CC digital signatures
- All certificates are revoked upon a owner request
  - Requires a revocation PIN
- CRL distribution points explicitly mentioned in each certificate

PKI Trust relationships

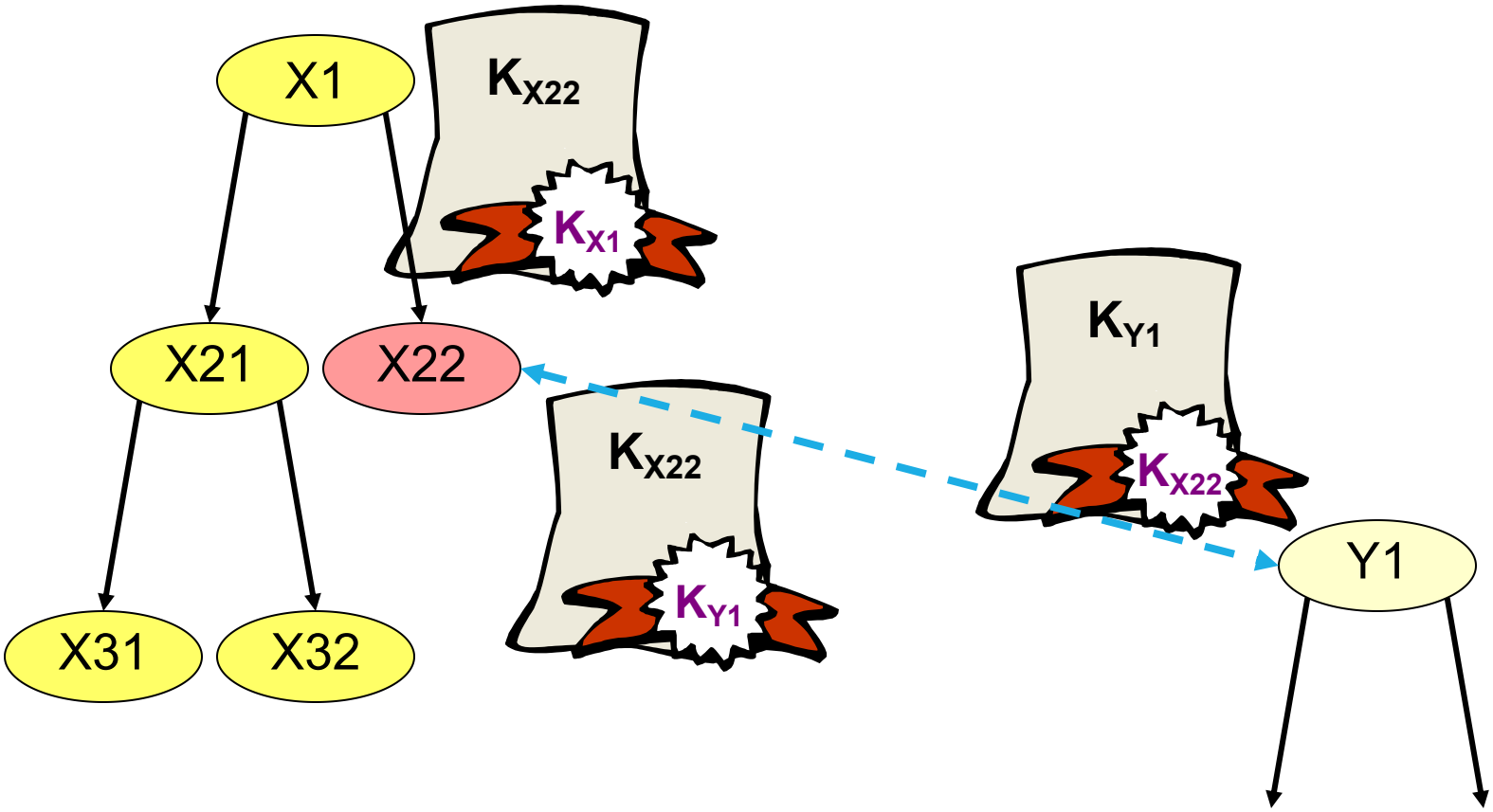
A PKI defines trust relationships in two different ways

- By issuing certificates for the public key of other CAs
  - Hierarchically below; or
  - Not hierarchically related
- By requiring the certification of its root public key by another CA
  - Above in the hierarchy; or
  - Not hierarchically related

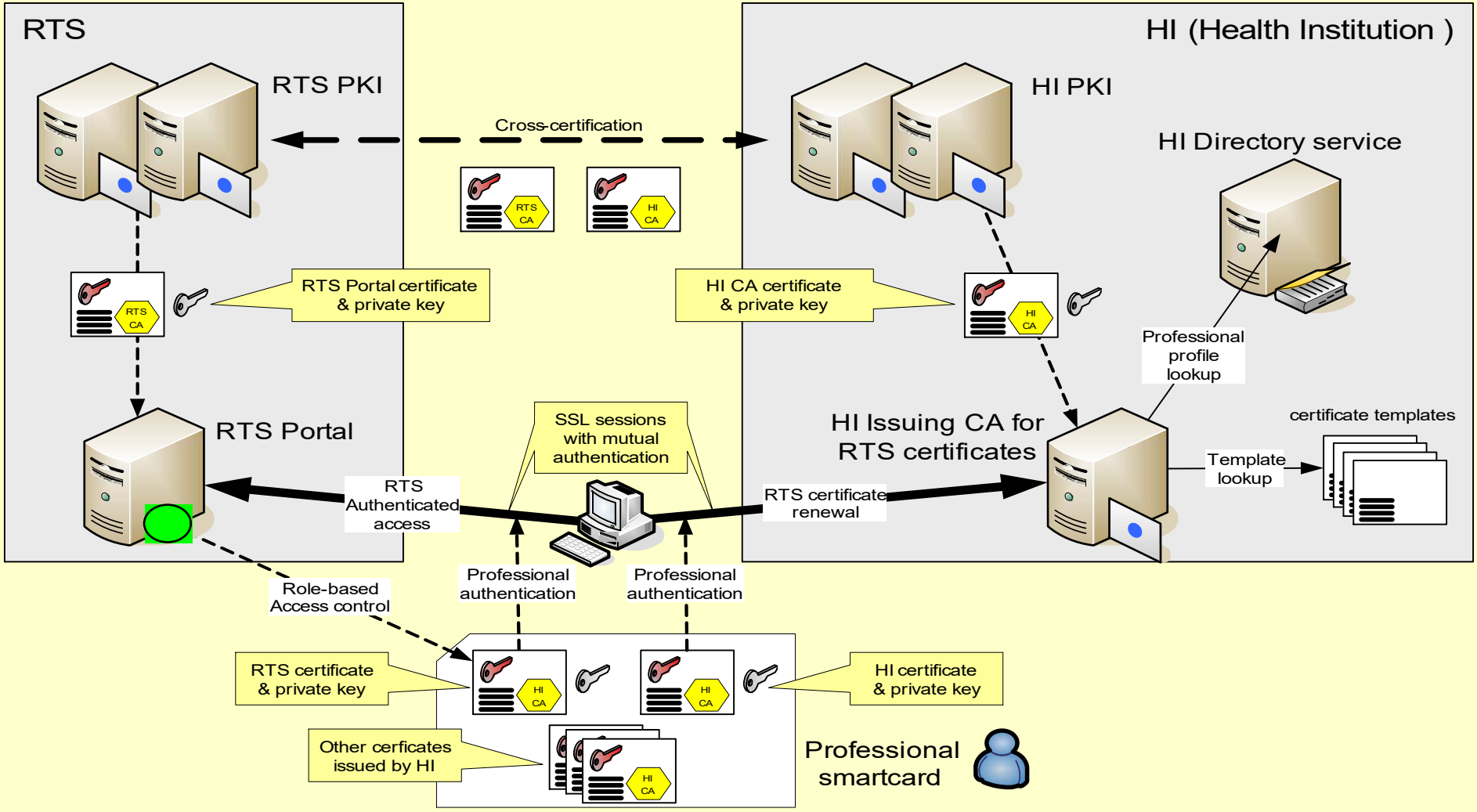
Usual trust relationships

- Hierarchical
- Crossed (A certifies B and vice-versa)
- Ad-hoc (mesh)
  - More or less complex certification graphs

PKI: Hierarchical and crossed certifications



Cross-certification of PKIs: A practical example



Certificate Pinning

If attacker has access to trusted Root, it can impersonate every entity

- Manipulate a trusted CA into issuing certificate (unlikely)
- Inject custom CA certificates in the victim's database (likely)

Certificate Pinning: add the fingerprint of the PubK to the code

- Fingerprint is a SHA256 hash
- Also store the URL to access

Validation process:

- Certificate must be valid according to local rules
- Certificate must have a public they with the given fingerprint

Additional Documentation

[RFC 3280] Internet X.509 Public Key Infrastructure: Certificate and CRL Profile

Other RFCs

- [RFC 2510] Internet X.509 PKI Certificate Management Protocols.
- [RFC 2511] Internet X.509 Certificate Request Message Format.
- [RFC 2559] Internet X.509 PKI Operational Protocols - LDAPv2.
- [RFC 2560] X.509 Internet PKI Online Certificate Status Protocol - OCSP.
- [RFC 2585] Internet X.509 PKI Operational Protocols: FTP and HTTP.
- [RFC 2587] Internet X.509 PKI LDAPv2 Schema.
- [RFC 3029] Internet X.509 PKI Data Validation and Certification Server Protocols.
- [RFC 3161] Internet X.509 PKI Time-Stamp Protocol (TSP).
- [RFC 3279] Algorithms and Identifiers for the Internet X.509 PKI Certificate and Certificate Revocation List (CRL) Profile.
- [RFC 3281] An Internet Attribute Certificate Profile for Authorization.
- [RFC 3647] Internet X.509 PKI Certificate Policy and Certification Practices Framework.
- [RFC 3709] Internet X.509 PKI: Logotypes in X.509 Certificates.
- [RFC 3739] Internet X.509 PKI: Qualified Certificates Profile.
- [RFC 3779] X.509 Extensions for IP Addresses and AS Identifiers.
- [RFC 3820] Internet X.509 PKI Proxy Certificate Profile.