Management of Asymmetric keys

Goals

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

Design principles for new key pairs: self-generation of private keys

- Maximizes privacy as no other party will be able to use a given private key
  - Only the owner has the key
  - Even better: the owner cannot observe the key, but may use the key
- This principle can be relaxed when not involving signature generation
  - Where there are not issues related with non-repudiation

Secure handling of private keys

- The private key represents a subject
  - e.g., a citizen, a service
  - 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
    - To prevent voluntary or involuntary private key leaks

Secure handling of private keys

- Protection of the private key inside an isolated 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

- Examples
  - Smartcards
    - e.g. Cartão de Cidadão
  - FIDO2 tokens

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

Public key (digital) certificate

- Document issued by a Certification Authority (CA)
- Binds a public key to an entity
  - Person, server or service
- Public document
  - Do not contain private information, only public one
  - Can have additional binding information (URL, Name, email, etc.)
- Cryptographically secure
  - Digitally signed by the issuer, cannot be changed

Public key certification

- Certificates allow a trustworthy distribution of public keys

- A certificate receiver can validate it in many ways
  - With the CA’s public key
  - Can also validate the identification
  - Validate the validity
  - Validate is the key is being properly used

- A certificate receiver trusts the behavior of the CA
  - Therefore, will trust the documents they sign
  - When a CA associates a certificate to X
    - If the receiver trusts the CA
    - Then it will trust that the association of a public key to X is correct

Public key (digital) certificates

- X.509v3 standard
  - Mandatory fields
    - Version
    - Subject
    - Public key
    - Dates (issuing, deadline)
    - Issuer
    - Signature
    - etc.
  - Extensions
    - Critical or non-critical

- Binary format
  - ASN.1 (Abstract Syntax Notation)
    - DER, CER, BER, etc.

- Other formats
  - PEM (Privacy Enhanced Mail)
    - base64 encoding of X.509
  - PKCS #12
    - Personal Information Exchange Syntax Standard
    - Used to pack a private key and a public key certificate
    - The private key can be PWD-protected

Extensions: key pair usage

- The public certificate binds the key pair to a usage profile
  - Private keys are seldom multi-purpose
- There is extension for this
  - Key usage (critical)

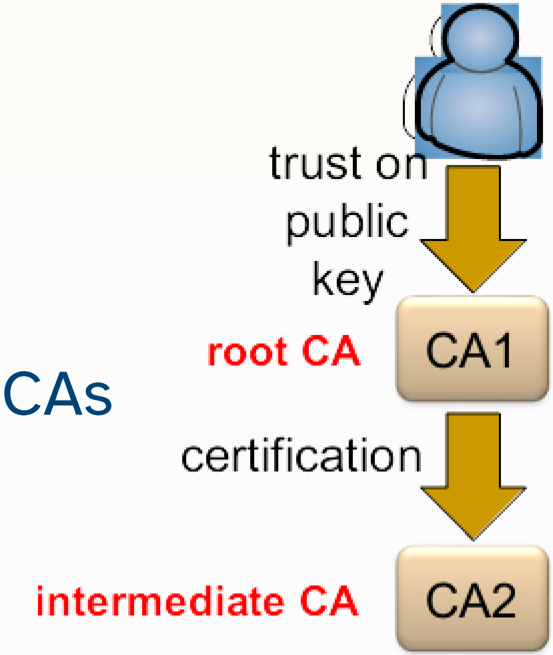
- Typical usage profiles
  - Authentication / key distribution
    - Digital signature, Key encipherment, Data encipherment, Key agreement
  - Document signing
    - Digital signature, Non-repudiation
  - Certificate issuing (exclusively for CAs)
    - Certificate signing, CRL signing
  - Timestamping (exclusively for TSAs)

Certification Authorities (CA)

- Organizations that manage public key certificates
  - Companies, not-for-profit organizations or governmental
  - Have the task of validating key-identity relationships
- 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
  - Programmatic interfaces to verify the current state of a certificate

Certification hierarchies

- Formed by Intermediate and Root Cas
- Intermediate CAs: CAs certified by other CAs
  - Using a certificate
- Trusted anchor (or certification root)
  - One that 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…



trust on public key

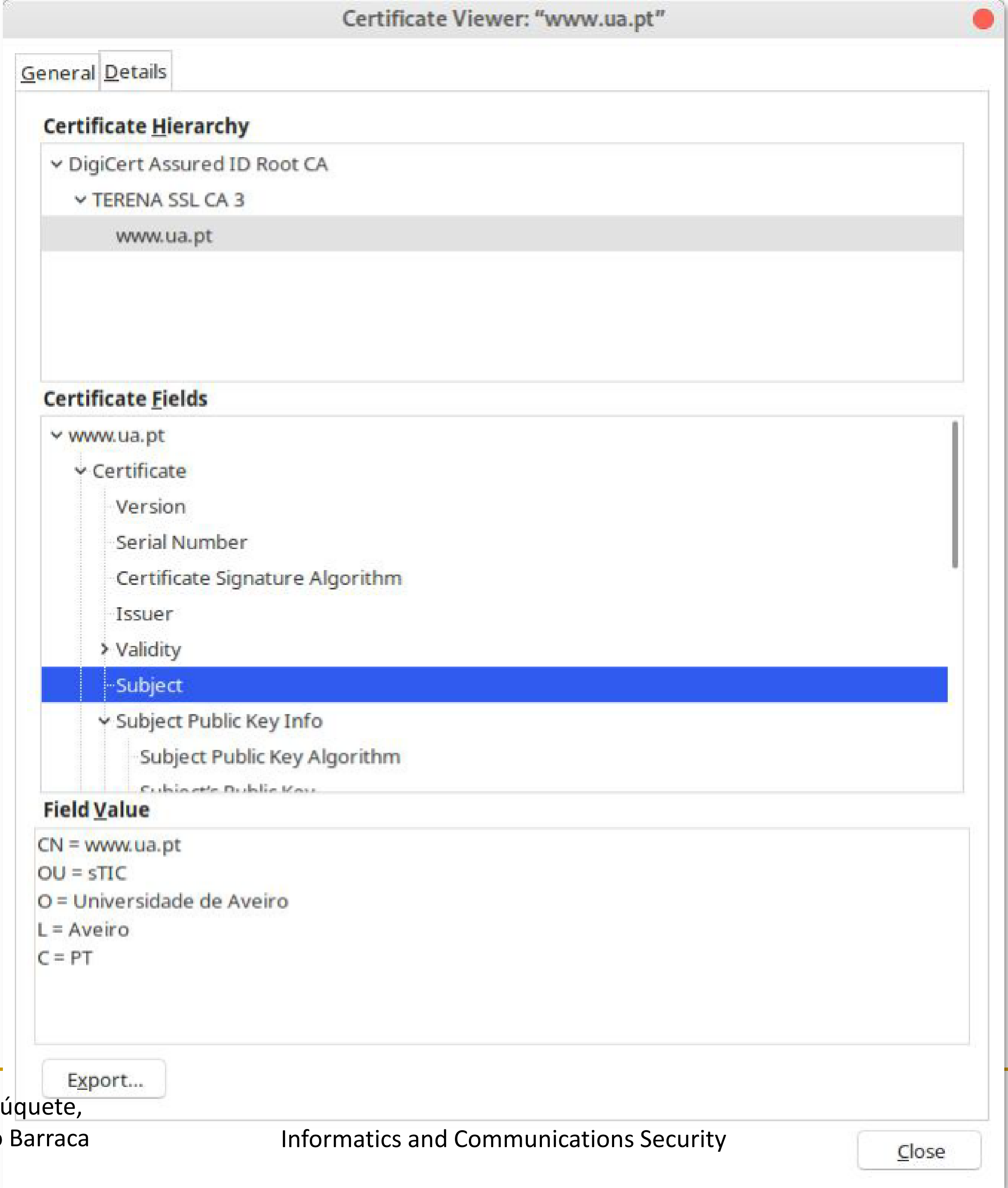
root CA

CA1

certification

intermediate CA

CA2



Certificate Hierarchy

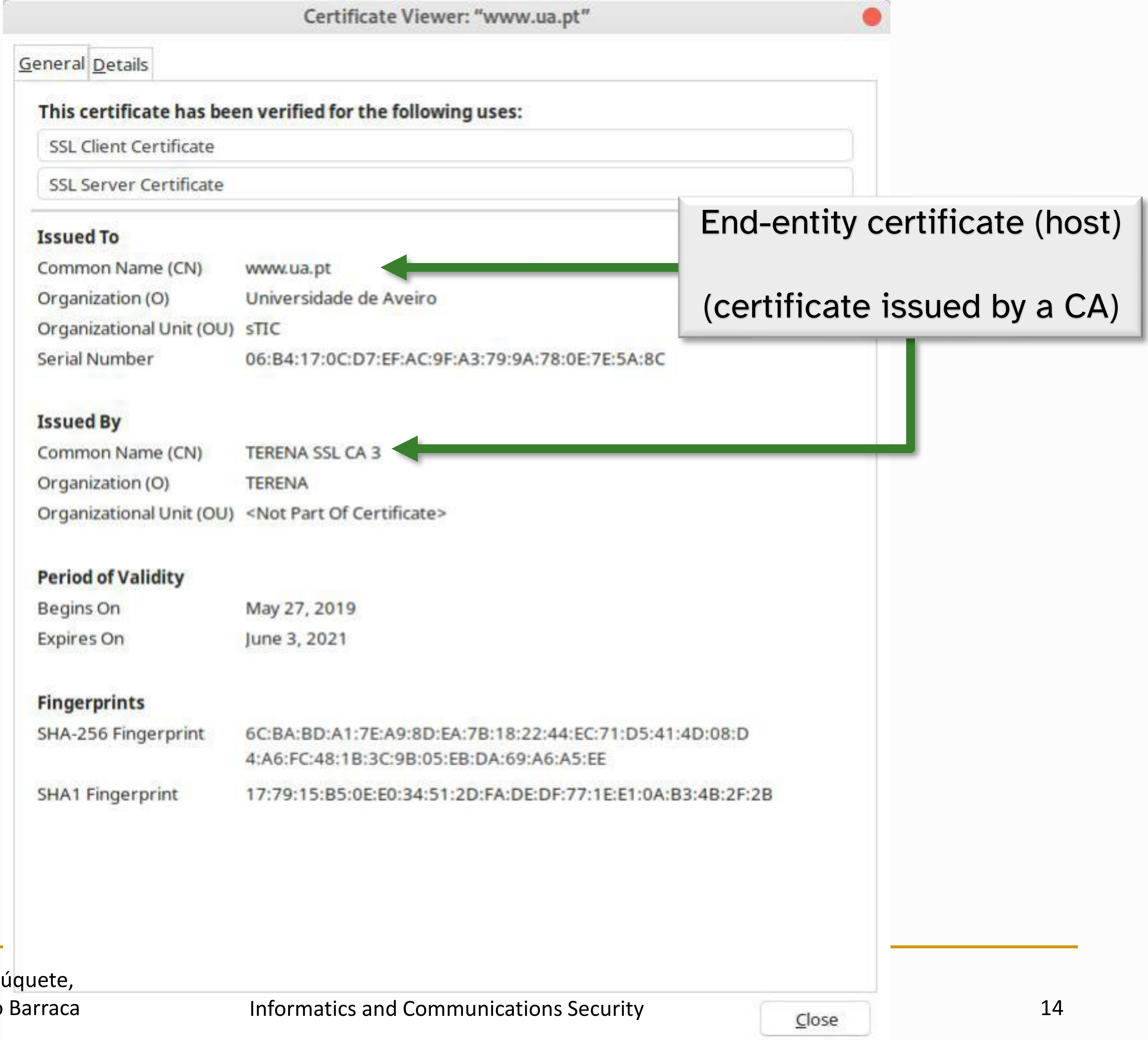
- DigiCert Assured ID Root CA
  - TERENA SSL CA 3
    - www.ua.pt

Certificate Fields

- www.ua.pt
  - Certificate
    - Version
    - Serial Number
    - Certificate Signature Algorithm
    - Issuer
    - Validity
    - Subject
    - Subject Public Key Info
      - Subject Public Key Algorithm
      - Subject's Public Key

Field Value

CN = www.ua.pt
OU = sTIC
O = Universidade de Aveiro
L = Aveiro
C = PT



This certificate has been verified for the following uses:

SSL Client Certificate

SSL Server Certificate

Issued To

Common Name (CN)	www.ua.pt
Organization (O)	Universidade de Aveiro
Organizational Unit (OU)	sTIC
Serial Number	06:B4:17:0C:D7:EF:AC:9F:A3:79:9A:78:0E:7E:5A:8C

Issued By

Common Name (CN)	TERENA SSL CA 3
Organization (O)	TERENA
Organizational Unit (OU)	<Not Part Of Certificate>

Period of Validity

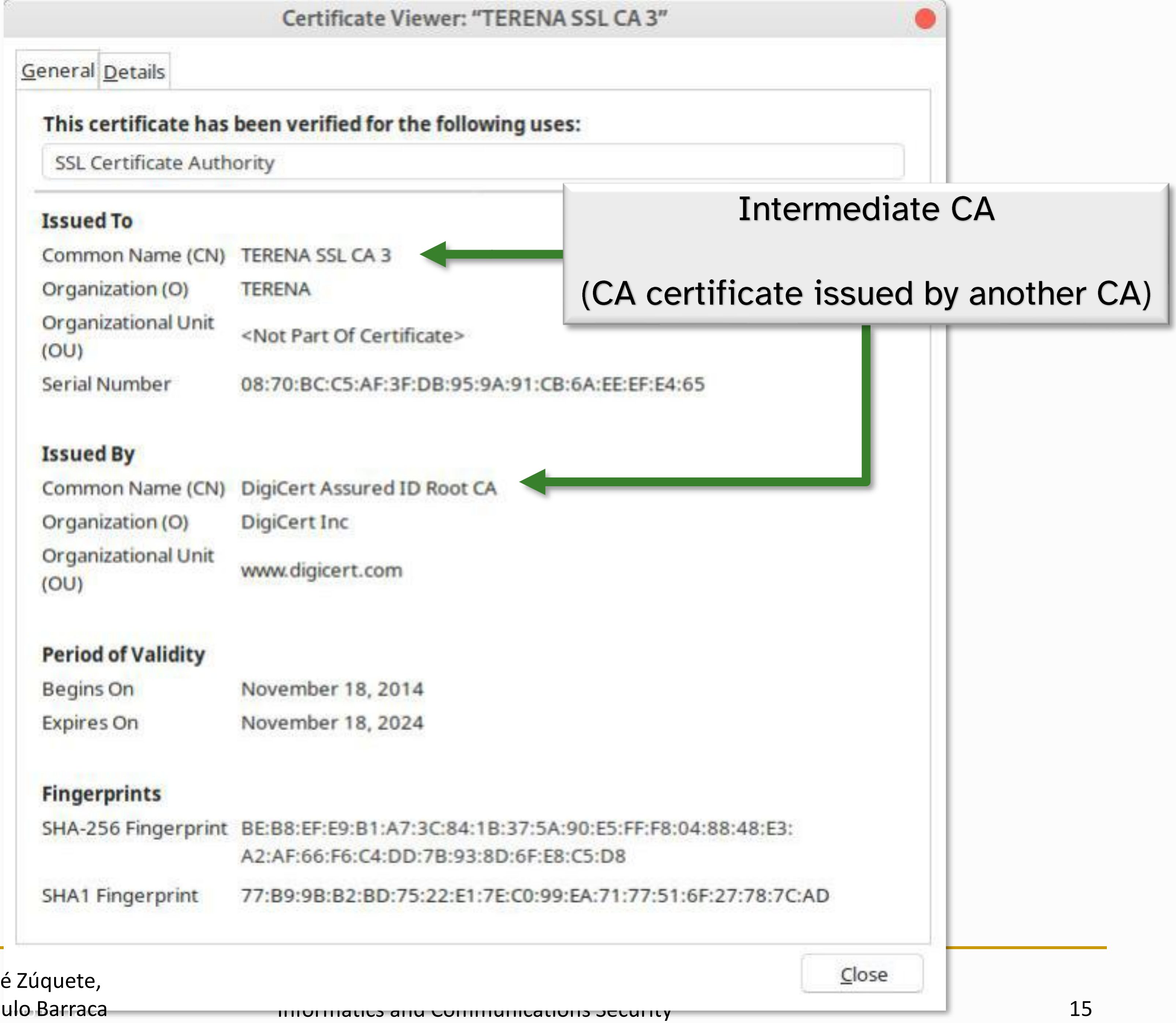
Begins On	May 27, 2019
Expires On	June 3, 2021

Fingerprints

SHA-256 Fingerprint	6C:BA:BD:A1:7E:A9:8D:EA:7B:18:22:44:EC:71:D5:41:4D:08:D 4:A6:FC:48:1B:3C:9B:05:EB:DA:69:A6:A5:EE
SHA1 Fingerprint	17:79:15:B5:0E:E0:34:51:2D:FA:DE:DF:77:1E:E1:0A:B3:4B:2F:2B

End-entity certificate (host)

(certificate issued by a CA)



Certificate Viewer: "TERENA SSL CA 3"

This certificate has been verified for the following uses:

SSL Certificate Authority

Issued To

Common Name (CN)	TERENA SSL CA 3
Organization (O)	TERENA
Organizational Unit (OU)	<Not Part Of Certificate>
Serial Number	08:70:BC:C5:AF:3F:DB:95:9A:91:CB:6A:EE:EF:E4:65

Issued By

Common Name (CN)	DigiCert Assured ID Root CA
Organization (O)	DigiCert Inc
Organizational Unit (OU)	www.digicert.com

Period of Validity

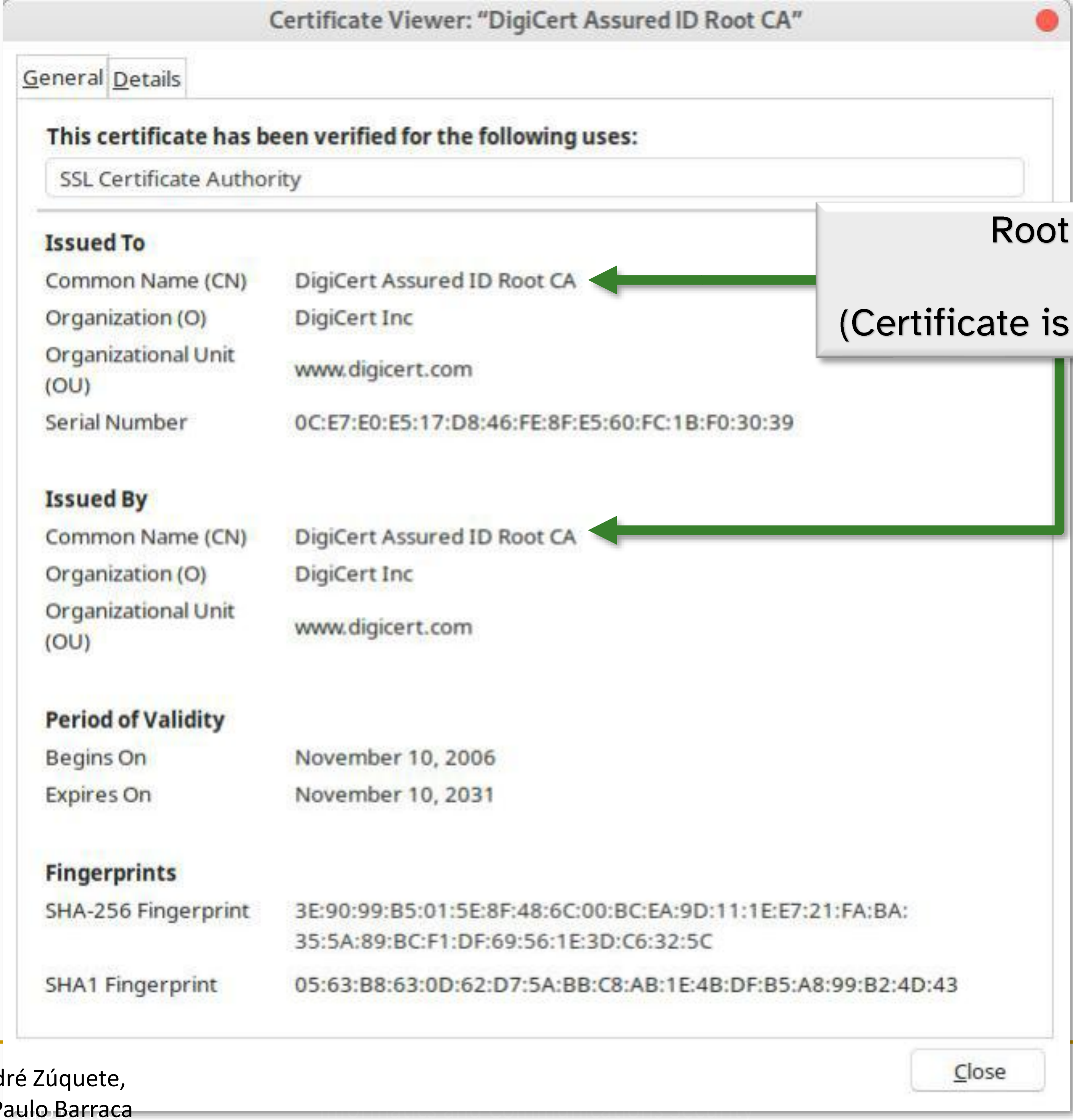
Begins On	November 18, 2014
Expires On	November 18, 2024

Fingerprints

SHA-256 Fingerprint	BE:B8:EF:E9:B1:A7:3C:84:1B:37:5A:90:E5:FF:F8:04:88:48:E3:A2:AF:66:F6:C4:DD:7B:93:8D:6F:E8:C5:D8
SHA1 Fingerprint	77:B9:9B:B2:BD:75:22:E1:7E:C0:99:EA:71:77:51:6F:27:78:7C:AD

Intermediate CA

(CA certificate issued by another CA)



Certificate Viewer: “DigiCert Assured ID Root CA”

This certificate has been verified for the following uses:

SSL Certificate Authority

Issued To

Common Name (CN)	DigiCert Assured ID Root CA
Organization (O)	DigiCert Inc
Organizational Unit (OU)	www.digicert.com
Serial Number	0C:E7:E0:E5:17:D8:46:FE:8F:E5:60:FC:1B:F0:30:39

Issued By

Common Name (CN)	DigiCert Assured ID Root CA
Organization (O)	DigiCert Inc
Organizational Unit (OU)	www.digicert.com

Period of Validity

Begins On	November 10, 2006
Expires On	November 10, 2031

Fingerprints

SHA-256 Fingerprint	3E:90:99:B5:01:5E:8F:48:6C:00:BC:EA:9D:11:1E:E7:21:FA:BA: 35:5A:89:BC:F1:DF:69:56:1E:3D:C6:32:5C
SHA1 Fingerprint	05:63:B8:63:0D:62:D7:5A:BB:C8:AB:1E:4B:DF:B5:A8:99:B2:4D:43

Root CA

(Certificate is self-signed)

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)

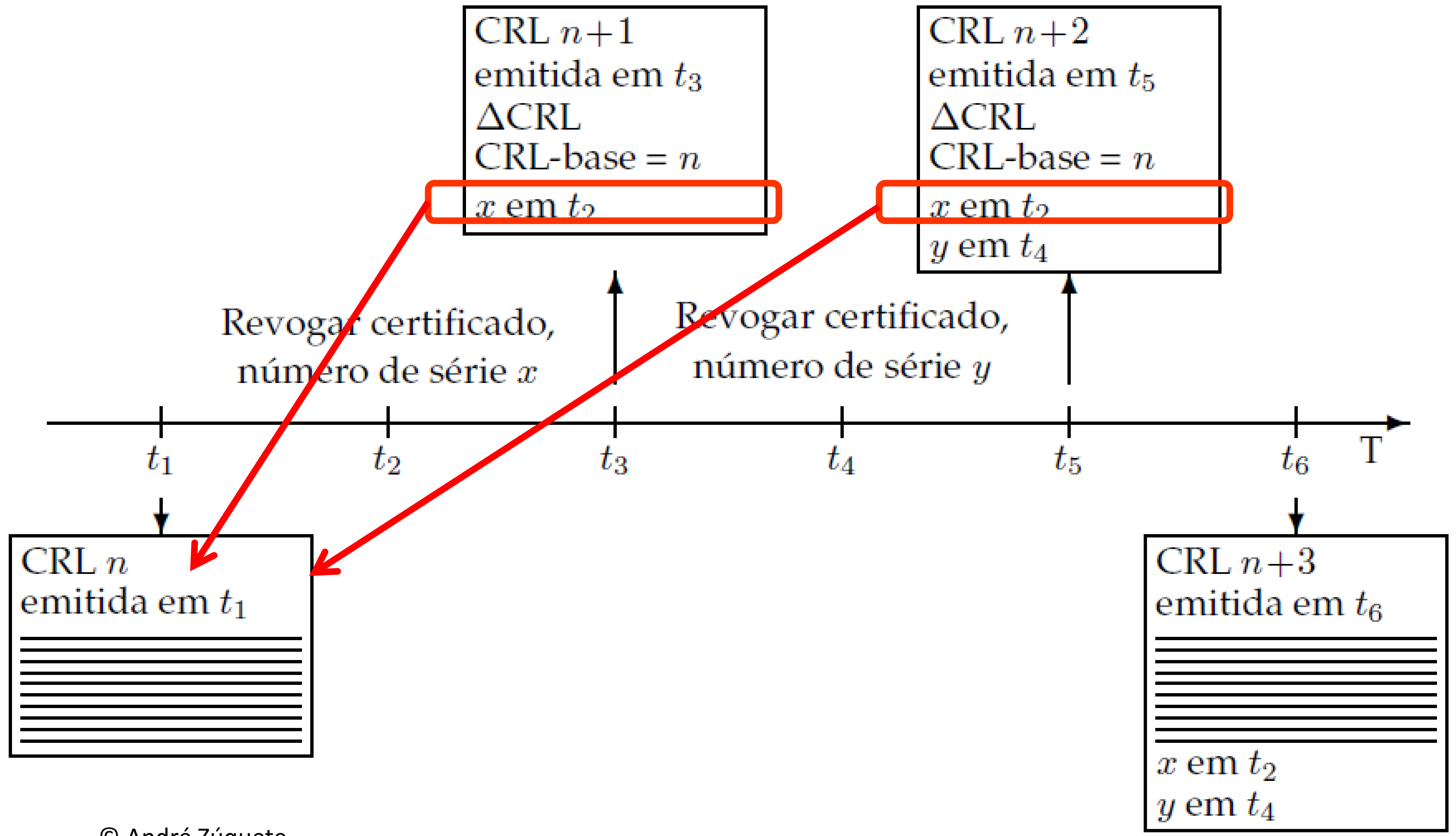
- Base or delta
  - Complete / differences
- Signed lists of certificates (identifiers) prematurely invalidated
  - Can tell the revocation reason
  - Must be regularly consulted by certificate holders
    - Certificates usually contain a reference to its CRL
- OCSP protocol for single certificate validation
  - RFC 2560
- Publication and distribution of CRLs
  - Each CA keeps its CRL updated
  - And provides public access to its CRL

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

- Requires 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 certificate is being used

- OCSP stapling (RFC 6961)
  - Include a recently signed timestamp in a TLS server response to assert validity
  - Reduces verification delay and load on the CA
  - Avoids privacy issues

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, manually (UA IDP))
  - On-line: within protocols using certificates for peer authentication
    - e.g. secure communication protocols (TLS, IPSec, etc.)
    - e.g. digital signatures within MIME mail messages or within documents
- Explicit (voluntarily triggered by users)
  - User request to a service for getting a required certificate
    - e.g. request sent by e-mail
    - e.g. access to a personal HTTP page

Public Key Infrastructure (PKI, 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

Public Key Infrastructure (PKI, 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
  - CRL distribution services
  - OCSP services
- Use of data structures and protocols enabling inter-operation among components / services / people

Certificate Pinning

- If an attacker compromises a trusted Root, it can impersonate every entity below in the hierarchy
  - Inject custom CA certificates in a victim's database (likely)
- Certificate Pinning: add the Kpub fingerprint to the source code
  - Fingerprint is a hash (e.g. SHA256)
- Validation process:
  - Certificate must be valid according to local rules
  - Certificate must have a public key with the given fingerprint

Certification Transparency (RFC 6962)

- Problems
  - CAs can be compromised (e.g., DigiNotar)
    - By attackers
    - By governments, etc.
  - Compromise is difficult to detect
    - Result in the change of assumptions associated to the behavior of the CA
    - Owner will seldom know

- Definition: a global system records all public certificates created
  - Ensure that only a single certificate has the correct roots
  - Stores the entire certification chain of each certificate
  - Presents this information for auditing
    - Organizations or ad-hoc by the end-users