INTERNATIONAL STANDARD

ISO 15782-1

Second edition
2009-10-15

Certificate management for financial services —

Part 1: Public key certificates

Gestion de certificats pour les services financiers —

Partie 1: Certificats de clé publique

Reference number
ISO 15782-1:2009(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO 15782-1 was prepared by Technical Committee ISO/TC 68, *Financial services*, Subcommittee SC 2, *Security management and general banking operations*.

This second edition cancels and replaces the first edition (ISO 15782-1:2003), which has been technically revised.

ISO 15782 consists of the following parts, under the general title *Certificate management for financial services*:

— *Part 1: Public key certificates*

— *Part 2: Certificate extensions*

Introduction

This part of ISO 15782 adopts ISO/IEC 9594-8 for the financial services industry and defines certificate management procedures and data elements.

Detailed requirements for the financial industry for the individual extensions are given in ISO 15782-2.

While the techniques specified in this part of ISO 15782 are designed to maintain the integrity of financial messages and support the service of non-repudiation, this part of ISO 15782 does not guarantee that a particular implementation is secure. It is the responsibility of the financial institution to put an overall process in place with the necessary controls to ensure that the process is securely implemented, with these controls including the application of appropriate audit tests in order to validate compliance.

The binding association between the identity of the owner of a public key and that key is documented in order to prove the ownership of the corresponding private key. This binding is called a public key certificate. Public key certificates are generated by a trusted entity known as a Certification Authority (CA).

The proper implementation of this part of ISO 15782 is intended to provide assurances of the binding of the identity of an entity to the key used by that entity to sign documents, including wire transfers and contracts.

This part of ISO 15782 defines a certificate management framework for authentication, including the authentication of keys for encryption. The techniques specified by this part of ISO 15782 can be used when initiating a business relationship between legal entities (entities).

Certificate management for financial services —

Part 1:
Public key certificates

1 Scope

This part of ISO 15782 defines a certificate management system for financial industry use for legal and natural persons that includes

— credentials and certificate contents,

— Certification Authority systems, including certificates for digital signatures and for encryption key management,

— certificate generation, distribution, validation and renewal,

— authentication structure and certification paths, and

— revocation and recovery procedures.

This part of ISO 15782 also recommends some useful operational procedures (e.g. distribution mechanisms, acceptance criteria for submitted credentials).

Implementation of this part of ISO 15782 will also be based on business risks and legal requirements.

This part of ISO 15782 does not include

— the protocol messages used between the participants in the certificate management process,

— requirements for notary and time stamping,

— Certificate Policy and Certification Practices requirements, or

— Attribute Certificates.

While this part of ISO 15782 provides for the generation of certificates that could include a public key used for encryption key management, it does not address the generation or transport of keys used for encryption.

Implementers wishing to comply with ISO/IEC 9594-8 can utilize the certificate structures defined by that International Standard. Those wishing to implement compatible certificate and certificate revocation structures but without the overhead associated with the X.500 series can utilize the ASN.1 structures defined in ISO 15782-2. ISO 15782-2 can also be referred to for a financial services profile of certificate and CRL extensions.

ISO 21188 provides additional information for implementers on Certificate Policies, Certification Practice Statements, and PKI controls. ISO 21188 sets out a framework of requirements to manage a PKI through Certificate Policies and Certification Practice Statements and to enable the use of public key certificates in the financial services industry. It also defines control objectives and supporting procedures to manage risks.

NOTE The use of a bold sans serif font, such as **CertReqData** or **CRLEntry,** denotes the use of abstract syntax notation (ASN.1), as defined in ISO/IEC 8824-1 to ISO/IEC 8824-4 and ISO/IEC 8825-1 and ISO/IEC 8825-2. Where it makes sense to do so, the ASN.1 term is used in place of normal text. Refer to ISO 15782-2 for related ASN.1 modules.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 8824-1, *Information technology — Abstract Syntax Notation One (ASN.1): Specification of basic notation — Part 1*

ISO/IEC 8824-2, *Information technology — Abstract Syntax Notation One (ASN.1): Information object specification — Part 2*

ISO/IEC 8824-3, *Information technology — Abstract Syntax Notation One (ASN.1): Constraint specification — Part 3*

ISO/IEC 8824-4, *Information technology — Abstract Syntax Notation One (ASN.1): Parameterization of ASN.1 specifications — Part 4*

ISO/IEC 8825-1, *Information technology — ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER) — Part 1*

ISO/IEC 8825-2, *Information technology — ASN.1 encoding rules: Specification of Packed Encoding Rules (PER) — Part 2*

ISO/IEC 9594-8, *Information Technology — Open Systems Interconnection — The Directory: Public-key and attribute certificate frameworks — Part 8*

ISO/IEC 15408 (all parts), *Information technology — Security techniques — Evaluation criteria for IT security*

ISO 15782-2:2001, *Banking — Certificate Management — Part 2: Certificate extensions*

ISO 16609, *Banking — Requirements for message authentication using symmetric techniques*

ISO 21188:2006, *Public key infrastructure for financial services — Practices and policy framework*

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

3.1
ASN.1 module
identifiable collection of ASN.1 types and values

3.2
attribute
characteristic of an entity

3.3
audit journal
chronological record of system activities which is sufficient to enable the reconstruction, review and examination of the sequence of environments and activities surrounding or leading to each event in the path of a transaction from its inception to the output of the final results

3.4
authorization
granting of rights

3.5
CA certificate
certificate whose subject is a CA, and whose associated private key is used to sign certificates

3.6
certificate hold
certificate suspension
temporary interruption of the validity of a certificate by the CA

3.7
certificate information
information in a certificate which is signed

3.8
certificate policy
named set of rules that indicates the applicability of a certificate to a particular community and/or class of application with common security requirements

EXAMPLE A particular Certificate Policy might indicate the applicability of a type of certificate to the authentication of electronic data interchange transactions for the trading of goods within a given price range.

NOTE 1 The Certificate Policy should be used by the user of the certificate to decide whether or not to accept the binding between the subject (of the certificate) and the public key. Some of the components in the Certificate Policy framework are given concrete values and represented by a registered object identifier in the X.509, Version 3 certificate. The object owner also registers a textual description of the policy and makes it available to the Relying Parties.

NOTE 2 The Certificate Policy object identifier can be included in the following extensions in the X.509, Version 3 certificates: Certificate Policies, policy mappings, and policy constraints. The object identifier(s) may appear in none, some, or all of these fields. These object identifiers may be the same (referring to the same Certificate Policy) or may be different (referring to different Certificate Policies).

3.9
certificate policy framework
comprehensive set of security- and liability-related components that can be used to define a Certificate Policy

NOTE A subset of the components in the Certificate Policy framework are given concrete values to define a Certificate Policy.

3.10
certificate re-key
process whereby an entity with an existing key pair and certificate receives a new certificate for a new public-key, following the generation of a new key pair

3.11
certificate renewal
process whereby an entity is issued for a new instance of an existing certificate with a new validity period

3.12
certificate request data
credentials
signed information in a certificate request, including the entity's public key, entity identity and other information included in the certificate

3.13
certificate revocation list
CRL
list of revoked certificates

3.14
certification
process of creating a public key certificate for an entity

3.15
certification authority
CA
entity trusted by one or more entities to create, assign, and revoke or hold public key certificates

3.16
certification authority system
set of entities, including a CA, that manages certificates throughout the life of the certificate

NOTE The entities are responsible for

— generation,

— submission,

— registration,

— certification,

— distribution,

— use,

— renewal,

— re-key,

— revocation or hold, and

— expiry.

3.17
certification path
ordered sequence of certificates of entities which, together with the public key of the initial entity in the path, can be processed to obtain the public key of the final entity in the path

3.18
certification practice statement
CPS
statement of the practices which a Certification Authority employs in issuing and managing certificates through their life cycle

3.19
compromise
violation of the security of a system such that an unauthorized disclosure or modification of sensitive information may have occurred

3.20
confidentiality
property that information is not made available or disclosed to unauthorized individuals, entities, or processes

3.21
CRL distribution point
directory entry or other distribution source for CRLs

NOTE A CRL distributed through a CRL distribution point may contain revocation entries for only a subset of the full set of certificates issued by one CA or may contain revocation entries for multiple CAs.

3.22
cross-certification
process by which two CAs mutually certify each other's public keys

cf. **policy mapping** (3.44)

3.23
(cryptographic) key
parameter that determines the operation of a cryptographic function

NOTE Cryptographic functions include the following:

— the transformation from plain text to cipher text and vice versa;

— synchronized generation of keying material;

— digital signature generation or validation.

3.24
cryptographical hash
mathematical function which maps values from a large (possibly very large) domain into a smaller (fixed) range

NOTE It satisfies the following properties:

— it is computationally unfeasible to find any input which maps to a pre-specified output (i.e. pre-image resistant);

— it is computationally unfeasible to find any two distinct inputs which map to the same output (i.e. collision-resistant).

3.25
cryptographic module
device wherein cryptographic functions (e.g. encryption, authentication, key generation) are performed

3.26
cryptography
discipline which embodies principles, means and methods for the transformation of data in order to hide its information content, prevent its undetected modification, prevent its unauthorized use or a combination thereof

3.27
cryptoperiod
time span during which a specific key is authorized for use or in which the keys for a given system may remain in effect

3.28
data integrity
property whereby data has not been altered or destroyed

3.29
delta-CRL
subset of a CRL indicating changes since the prior CRL update

3.30
(digital) signature
cryptographic transformation of data which, when associated with a data unit, provides the services of origin authentication and data integrity, and may support signer non-repudiation

3.31
directory
repository
method for distributing or making available certificates or CRLs

EXAMPLE A database or an X.500 Directory.

3.32
distinguished name
globally unique name for an entity

NOTE 1 Methods for determining global uniqueness are outside the scope of this part of ISO 15782.

NOTE 2 An entity may be issued for more than one certificate with the same distinguished name.

3.33
dual control
process of utilizing two or more separate entities (usually persons), who are operating in concert, to protect sensitive functions or information

NOTE 1 Both entities are equally responsible for the physical protection of materials involved in vulnerable transactions. No single person is able to access or to utilize the materials (e.g. cryptographic key).

NOTE 2 For manual key and certificate generation, conveyance, loading, storage, and retrieval, dual control requires split knowledge of key among the entities. Also see **split knowledge** (3.52).

3.34
end entity
certificate subject, other than a CA, which uses its private key for purposes other than signing certificates

3.35
entity
legal (e.g. a corporation, labour union, state or nation) or natural person

EXAMPLE CA, RA or end entity.

3.36
financial message
communication containing information which has financial implications

3.37
key agreement
method for negotiating a key value without transferring the key, even in an encrypted form

EXAMPLE The Diffie-Hellman technique.

3.38
key fragment
part of a private key which has been divided into pieces (also called shares) that are distributed amongst entities such that the pooled fragments of specific subsets of entities can reconstitute the key

3.39
key management
handling of keying material throughout its life cycle in accordance with a security policy

3.40
key pair
⟨public key cryptography⟩ public key and its corresponding private key

3.41
keying material
data, such as keys, certificates and initialization vectors, necessary to perform cryptographic operations

3.42
non-repudiation
service which provides proof of the integrity and origin of data which can be verified by a third party

NOTE The non-repudiation service protects against the signing entity falsely denying the action and can provide rebuttable presumption. It requires that appropriate processes and procedures (registration, audit journals, contractual arrangements, personnel, etc.) be in place.

3.43
out-of-band notification
notification using a communication means independent of the primary communications means

3.44
policy mapping
recognition that, when a CA in one domain certifies a CA in another domain, a particular Certificate Policy in the second domain may be considered by the authority of the first domain to be equivalent (but not necessarily identical in all respects) to a particular Certificate Policy in the first domain

cf. **cross-certification** (3.22)

3.45
policy qualifier
policy-dependent information that accompanies a Certificate Policy identifier in an X.509 certificate

3.46
private key
⟨asymmetric (public) key cryptosystem⟩ key of an entity's key pair which is known only by that entity

3.47
public key
⟨asymmetric (public) key cryptosystem⟩ key of an entity's key pair which is publicly known

3.48
public key certificate
public key and identity of an entity together with some other information, rendered unforgeable by signing the certificate information with the private key of the certifying authority that issued that public key certificate

3.49
public key validation
PKV
process that does arithmetic tests on a candidate public key to provide assurance that it conforms to the specifications of the standard

NOTE 1 Attacks may be possible on the owner and/or user if a non-conforming public key is used.

NOTE 2 Public key validation can include arithmetic property tests (range, order, primality, etc.), canonical generation tests and consistency tests between components of a public key. Methods for public key validation are typically found in financial industry signature standards.

3.50
registration authority
RA
entity that is responsible for identification and authentication of subjects of certificates, but is not a CA and hence does not sign or issue certificates

NOTE An RA may assist in the certificate application process, revocation process or both.

3.51
relying party
user
recipient of a certificate who acts in reliance on that certificate

3.52
split knowledge
condition under which two or more entities separately have key fragments which, individually, convey no knowledge of the resultant cryptographic key

3.53
subject
entity whose public key is certified in a public key certificate

3.54
subject CA
CA that is certified by the issuing CA

3.55
subscriber
entity subscribing with a Certification Authority on behalf of one or more subjects

3.56
trusted CA public key
public key used to validate the first certificate in a chain of certificates as a part of certification path processing

EXAMPLE The root key in a centralized trust model or a local CA key in a decentralized trust model (see Annex B).

NOTE If an end entity validates a chain of certificates from a trusted CA public key to the end certificate, then the end certificate is considered valid.

3.57
zeroize
active destruction of electronically stored data, such as by degaussing, erasing or overwriting

4 Symbols and abbreviations

Symbols	Meaning
X{information}	Signing of “information” by X
X_p	X's public key. (e.g. X_{1p} is X_1's public key)
X_s	X's private key
$X_1«X_2»$	X_2's certificate issued by the CA, X_1
$X_1«X_2»X_2«X_3»X_{n-1}«X_n»$	Certificate path. Each item Xi<Xi+1> in the path is the certificate for the CA which produced the next item. This path is of arbitrary length and is functionally equivalent to $X_1«X_n»$. Possession of X_{1p} allows a user to extract the authenticated public key of X_n.
$X_{1p}{\cdot}X_1«X_2»$	Unwrapping of a certificate or path. The public key of the leftmost CA (X_1) is used to extract the authenticated public key of the rightmost certificate ($X_1«X_2»$) by working through the path of intervening certificates. This example extracts X_{2p}.

NOTE The notation used in this part of ISO 15782 is a variant of the X.509 notation for certificates, certification paths and related information.

Abbreviations	Meaning
ASN.1	Abstract syntax notation
BER	Basic encoding rules
CA	Certification Authority
CPS	Certification Practice Statement
CRL	Certificate Revocation List
DER	Distinguished encoding rules
DSA	Digital signature algorithm
ECDSA	Elliptic curve digital signature algorithm
PKI	Public key infrastructure
RA	Registration Authority
RSA	Rivest Shamir Adleman algorithm
SHA-1	Secure hash algorithm-1
URI	Uniform resource identifier

5 Public key infrastructure

5.1 Overview

Public key infrastructure (PKI) is a term used to describe the technical, legal and commercial infrastructure that enables the wide deployment of public key technology.

Public key technology is used for creating digital signatures and for managing symmetric keys. With public key cryptography, two keys are used: one is kept private with the user, the other is made publicly available. That which is signed or processed with one key (public or private) may be validated with its complement (public or private). Revealing the public key does not in any way compromise the private key.

The authentication of public keys is an essential requirement and, for this reason, public keys are housed in public key certificates. A certificate contains the public key and its identifying data and is digitally signed by a

Certification Authority (CA). This part of ISO 15782 is based on the ISO/IEC 9594-8 format for public key certificates.

As described in ISO 21188, the term “Certification Authority” reflects the aggregate of the following PKI roles including

— Certificate Issuer,

— Certificate Manufacturer,

— Registration Authority,

— Repository,

— Certificate Validation Service Provider, and

— Subject Cryptographic Mechanism Provider.

In this part of ISO 15782, we focus on the CA as a whole and on the RA function.

5.2 Public key management infrastructure process flow

The responsibilities, services and procedures required by a public key management infrastructure are as follows:

— key generation;

— registration;

— certification;

— distribution;

— usage;

— revocation hold;

— expiry;

— renewal;

— re-key.

The main steps involved in certification are shown in Figure 1.

5.3 Certification Authority (CA)

A CA has a public/private key pair and uses a digital signature algorithm to produce certificates.

The binding of the entity's public key to its identity is accomplished by having the CA generate the certificate, thereby attesting to the relationship of the information therein and providing assurances of its integrity.

The binding of an entity's public key and identity is validated by using the public key of one or more CAs as described in 6.3.1. The certificate(s) and proof of validation shall be maintained by the validator in an audit journal. A CA may issue certificates to any entities, including CAs.

Entities (including CAs) can use these certificates to authenticate themselves to Relying Parties. Hence, authentication may involve a chain of certificates. The verification of a chain of certificates begins with the

trusted CA public key and ends with the certificate being validated. The trusted CA public key shall be obtained and authenticated by some means other than by the use of certificates. This is to ensure that the process begins securely. See 6.3.1, Annex C and ISO/IEC 9594-8.

Once a certificate has been generated, the integrity of its contents is protected. This part of ISO 15782 does not require that certificates be given confidentiality protection. A valid copy of the CA's public key is required by the Relying Party in order to validate a certificate. Given that the CA is a trusted entity, this permits the validation of the binding between an entity's public key, its identity, and other information needed.

Two general architectures may be configured for certification paths: hierarchical and non-hierarchical. In a hierarchical architecture, authorities are arranged under a “root” CA that issues certificates to subordinate CAs. These subordinate CAs may issue certificates to CAs subordinate to them or to end entities. In a hierarchical architecture the public key of the root CA functions as the trusted CA public key and is known to every entity. Any entity's certificate may be validated by validating the certification path of signature certificates that leads from the certificate being validated back to the trusted CA public key of the root CA. In this architecture, the root CA is a mutual point of trust for all entities.

To communicate outside the root CA's domain, the root CA shall cross-certify with the desired remote domain. Certification path validation then involves building a chain of certificates from the remote entity to the root CA by way of the cross-certified remote CA.

In a non-hierarchical architecture, independent CAs may cross-certify each other by issuing public key certificates to each other. This results in a general network of trust relationships between CAs and allows each group (such as a retail credit authorization network, a clearing house, a financial institution or a subgroup thereof) to have its own CA. An entity uses the public key of a selected CA for its trusted CA public key. The certification path consists of those certificates that chain back from the certificate being validated to the trusted CA of the Relying Party.

In a bridge architecture, which is a common non-hierarchical architecture, a central CA cross-certifies to the other CAs. The public key of the central bridge CA is known to all other trusted CAs and the public key of each trusted CA is known to the bridge CA. The certification path consists of two certificate chains, Subject to Subject's trusted CA and Relying Party to Relying Party's trusted CA, interconnected by the bridge CA-certificates.

A compromise of the private key of a CA compromises all users of certificates of the CA, because the holder of that private key can generate fraudulent certificates and then masquerade as one or more end entities. Failure to provide compensating controls to deal with the possibility of compromise of transactions in a financial network can have catastrophic effects on financial institutions and their customers.

Figure 1 summarizes the life cycle of an end-entity certificate.

5.4 Registration Authority (RA)

An RA is an entity that is responsible for identification and authentication of the subjects of certificates, but is not a CA, and hence does not sign or issue certificates. An RA may assist in the certificate application process, revocation process, or both. The RA does not need to be a separate body, but can be part of the CA.

5.5 End entities

An end entity is a certificate subject which uses its private key for purposes other than signing certificates.

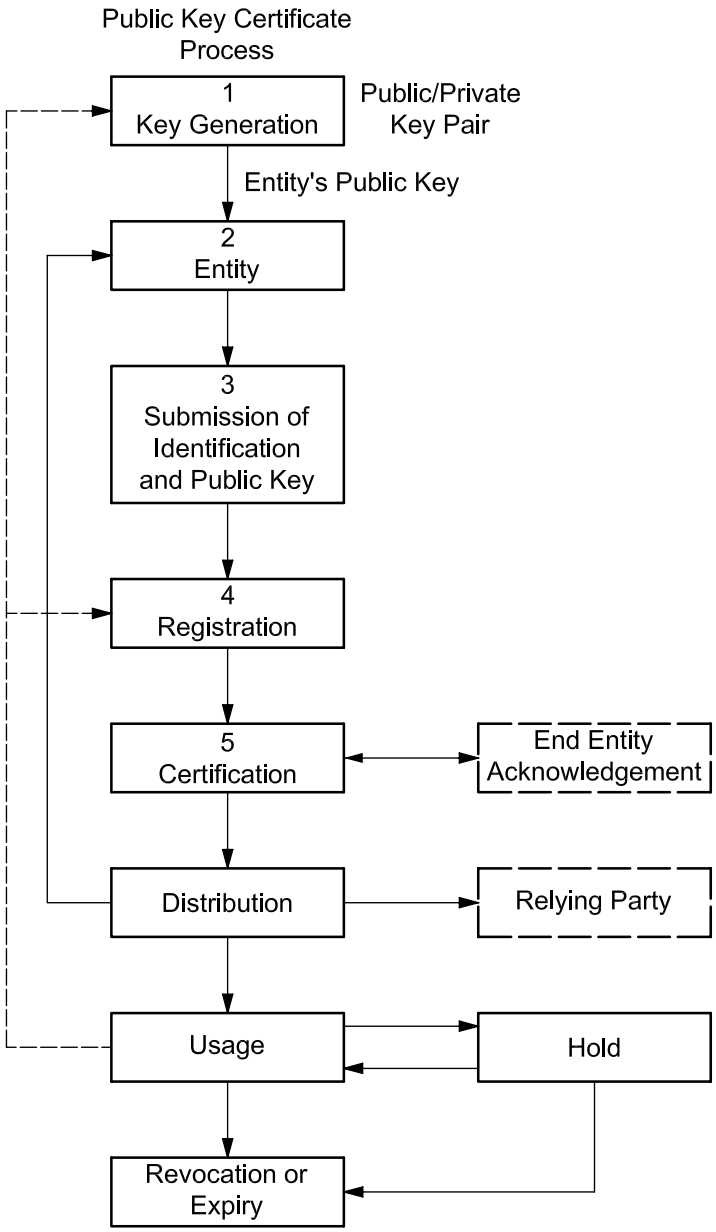
6 Certification Authority systems

6.1 General

A CA and one or more RAs may be configured as a CA system, and the functional allocation between CAs and RAs will vary, based on implementation.

RAs may be appointed to look after the secure registration of entities or this function may be part of the services provided by the CA. Each RA is responsible for a specific RA domain.

When an RA is used, communications between the RA and the CA shall be authenticated. A digital signature may be used for this process.



NOTE The numbers correspond to the process steps detailed in Figures 2 and 3.

Figure 1 — Typical public key management infrastructure process flow

6.2 Responsibilities in CA systems

6.2.1 Sole responsibilities and best practices of a CA

6.2.1.1 Sole responsibilities of a CA

The CA shall be solely responsible for

a) securing the private key associated with the public key contained in the CA's certificate,

b) ensuring that public keys certified by the CA are unique within the CA's domain,

c) ensuring that there is no duplication of the requester's distinguished name with that of any other entity certified by the CA,

d) generating certificates by applying a signature to the certificate information,

e) maintaining a revocation-checking mechanism appropriate for Relying Parties,

f) creating, maintaining and distributing Certificate Revocation Lists (see Annex F),

g) generating key pairs in accordance with a detailed CA key generation ceremony script and CPS requirements (see ISO 21188:2006, 7.3.1 “CA key generation” for more details),

h) producing the CPS and making it available to all appropriate parties,

i) ensuring it owns and controls its private keys, and

j) changing the CA's public/private key pair at intervals appropriate to the business risk.

A CA system shall ensure that the responsibilities of the CA and RA are assigned.

The operation of a CA shall be in accordance with prudent business practices, the CA's Certification Practice Statement (CPS), and applicable Certificate Policies (CP).The CA shall be responsible for certification and shall be able to print the information contained in certificates in human-readable form.

Certification path constraints are defined in ISO 15782-2. Examples of their use are given in Annex A of ISO 15782-2:2001.

Distribution requirements for a CA's public key are defined in 6.3.5.

6.2.1.2 Best practices of a CA system

It is recommended that the CA system possess the following attributes.

a) It should have sufficient resources to maintain its operations in conformity with its duties.

b) It should be reasonably able to bear its risk of liability to end entities and persons relying on certificates issued by the CA, as dictated by its policy.

c) It should employ personnel practices which provide reasonable assurance of trustworthiness.

d) It should use standardized (ISO or national) cryptographic techniques and cryptographic modules designed to meet the requirements for financial institution use based on one of the four progressive assurance levels of module security defined as follows.

1) Level 1 cryptographic module

Security Level 1 provides the lowest level of security. It specifies basic security requirements for a cryptographic module, but it differs from the higher levels in several respects. No physical security mechanisms are required in the module beyond the requirement for production-grade equipment.

Level 1 allows software cryptographic functions to be performed in a general purpose personal computer (PC). Such implementations are often appropriate in low-level security applications. The implementation of PC cryptographic software may be more cost-effective than hardware-based mechanisms. This will enable organizations to avoid the situation that exists today whereby the decision is often made not to cryptographically protect data because hardware is considered too expensive.

2) Level 2 cryptographic module (tamper evident)

Level 2 provides physical security by including requirements for tampe-evident coatings or seals, or for pick-resistant locks. Tampe-evident coatings or seals would be placed on a cryptographic module so that the coating or seal would have to be broken in order to attain physical access to the plain-text cryptographic keys and other critical security parameters within the module. Pick-resistant locks would be placed on covers or doors to protect against unauthorized physical access. These requirements provide a low cost means for physical security and avoid the cost of the higher level of protection involving hard opaque coatings or significantly more expensive tamper detection and zeroization circuitry.

3) Level 3 cryptographic module (tamper protected)

Level 3 attempts to prevent the intruder from gaining access to critical security parameters held within the module. For example, a multiple-chip embedded module shall be contained in a strong enclosure, and if a cover is removed or a door is opened, the critical security parameters are zeroized. As another example, a module shall be enclosed in a hard, opaque potting material to deter access to the contents.

4) Level 4 cryptographic module (tamper enveloped)

Level 4 physical security provides an envelope of protection around the cryptographic module. Whereas the tamper-detection circuits of lower level modules may be bypassed, the intent of Level 4 protection is to detect a penetration of the device from any direction. For example, if one attempts to cut through the enclosure of the cryptographic module, the attempt should be detected and all critical security parameters should be zeroized. Level 4 devices are particularly useful for operation in a physically unprotected environment where an intruder could possibly tamper with the device.

NOTE These levels are not to be confused with ISO/IEC 15408's common criteria evaluation assurance levels (EALS) or security functional levels (FCNs), or with levels of trust that may be defined in a CA's Certificate Policy/CPS.

e) It should make its own certificates, public key and Certificate Revocation Lists (CRL) readily and reliably available to Relying Parties.

f) It should utilize trustworthy systems in performing its services.

g) It should define and document internal operation policies for the CA system domain.

h) It should ensure that it complies with all appropriate regulations.

6.2.2 Responsibilities to be allocated to either a CA or an RA

The following requirements are applicable to the CA or RA.

a) It shall validate the identity of the entity requesting the certificate as being that of the subject of the certificate.

b) It shall validate the identity of the entity requesting a certificate if the requesting entity will not be the subject of the certificate.

c) It shall, if appropriate, advise the party identified in the certificate that a certificate has been issued.

d) A validated means shall be used to convey this advice, and that means shall be independent of any method used to convey the certificate to the entity. Examples of validated means include normal mail for low-risk (retail) systems and registered mail for private banking systems. The implementation of this functionality is determined by business risk.

e) It shall keep records supporting the certificate issuance process for the length of time determined by records-retention requirements.

f) It shall register authenticated entities securely.

g) It shall provide guidance to its end entities on the secure management of the end entity's private key.

h) It shall inform the end entity that the integrity of the end entity's operation will be considered compromised if the private key of the end entity is ever revealed to, or used by, any unauthorized entity.

i) It shall use any appropriate means to ascertain that the end entity understands the responsibilities of 6.2.3. and is able to comply with them.

j) It shall inform Relying Parties in the domain when the CA private key has been compromised.

k) It shall manage certificate renewal, re-key, revocation, suspension and re-instatement requests from entities.

Annex C contains suggested requirements for the acceptance of certificate request data. See 6.4 for security quality assurance and audit requirements.

6.2.3 Responsibilities of an end entity

The following requirements are applicable to the end entity.

a) It shall ensure that the end entity's private key is

   — kept within the secure confines of a cryptographic module, and

   — used only by the authorized person with proper access controls (e.g. user password or PIN), in accordance with the relevant CPS.

b) It shall understand the requirements for business continuity as specified in the CPS.

c) It shall ensure that the private key, when stored in a backup environment, is kept within the secure confines of the cryptographic module, where the cryptographic modules are kept in secure storage.

   Security requirements for the backup of private keys should be based on 6.3.1.2, except that Level 2 cryptographic modules may be used. In low-risk applications, a Level 1 cryptographic module may be used. The private key shall be exported in a secure manner.

   NOTE Both Levels 1 and 2 of key management allow a single individual access to an un-split encryption key. This practice might be acceptable in certain perceived “low-risk” situations (such as home banking), but might not be acceptable within commercial financial institutions, which have traditionally required dual control of keying material.

d) It shall undertake that the private key is kept under its exclusive control, with due care being exercised to prevent unauthorized use throughout the life of the key.

e) It shall undertake that the CA System is notified as soon as possible if the end entity knows or suspects that its private key has been lost, disclosed, revealed or otherwise compromised.

f) It shall ensure that, following the compromise or withdrawal of an end entity's private key, the use of that private key is immediately and permanently discontinued.

g) It shall ensure that any end-entity private key, that has been previously lost or compromised and later recovered, will be destroyed.

h) It shall ensure that, if the end entity ceases to exist, there exists a designated responsible party which will ensure that the end entity's key is destroyed and which will notify the CA System.

i) It shall provide the level of security required by the CPS and in accordance with an end-entity agreement (e.g. Subscriber Agreement).

j) It shall provide accurate and complete data for the certificate request.

k) It shall report to the RA/CA any inaccuracy or changes to the certificate content (e.g. change of names after marriage, etc.).

6.3 Certificate life cycle requirements

6.3.1 Generation

6.3.1.1 General

This subclause addresses the requirements for the generation of public/private key pairs.

6.3.1.2 Security requirements for a CA's certificate-signing private key

Since the certificate generated by a CA shall be used to provide proof of the identity and integrity of the entity's public key, the part of the CA which performs private key cryptographic operations shall be implemented in a cryptographic module not controlled or accessed by any subscribing entity. Since the cryptographic module employs the private key of the CA that issues the entity certificate, this key shall be given a high level of protection, as its possession would enable an intruder to masquerade as the CA and generate forged certificates.

The security requirements for a CA's signing private key are the following.

a) The private key shall be generated internally to a cryptographic module that, as a minimum, meets the requirements of a Level 3 cryptographic module.

b) The public/private key generation process shall ensure that the keying material is arithmetically consistent with the requirements of a valid public key for the algorithm specified.

c) The public/private key pair shall be at least as strong as any key that it will be used to certify.

d) Neither the private key, nor any part thereof, shall exist in plain text outside of a cryptographic module that, as a minimum, meets the requirements of a Level 3 cryptographic module.

e) A CA shall have exclusive control over its own private key.

f) If copies of the CA's certificate-signing private key are required, the private key shall be securely exported using one of the following methods:

— under at least dual control with split knowledge; or

— encrypted using a key of at least equivalent strength of the public/private key pair which itself is afforded the same high level of protection as a CA signing private key.

g) After key expiry, all copies of the CA private key (and fragments if they exist) must remain securely protected or be securely destroyed.

6.3.1.3 End-entity key-pair generation requirements

The generation of an end entity's public/private key-pair shall occur in at least a Level 1 cryptographic module. Key-pair generation for high-risk applications should occur in at least a Level 2 cryptographic module. Security requirements for the backup of private keys should provide the same level of trust as for the operational private key.

Generation of the end entity's key may take place in the end-entity cryptographic module or in a central key management infrastructure. It is strongly recommended that signature keys be generated in the end entity's cryptographic module. The following additional requirements apply if the key pair is centrally generated.

a) The central key generation authority shall generate the key inside at least a Level 3 cryptographic module.

b) The central key generation authority shall ensure that the end-entity digital signature private key is not disclosed to any entity other than the owner of the key.

c) The central key generation authority shall not maintain a copy of any digital signature private key, once that key is delivered to the end entity.

d) The use of centralized key generation requires a secure channel to deliver the key pair to the end entity. The delivery of the end-entity's private key requires confidentiality and integrity protection.

EXAMPLE 1 Delivery in a Level 1 cryptographic module under dual control with split knowledge as defined in 6.3.5 or in another appropriate International Standard on banking.

EXAMPLE 2 Delivery under dual control with split knowledge.

EXAMPLE 3 Delivery in a PIN Mailer for low-risk systems.

The end entity may need to interact with the certificate management system to prove possession of the private key corresponding to the public key in the request using the process known as key validation.

6.3.2 Submission of certificate request data

An end entity shall compile its certificate request data and submit it to an RA. The certificate request data provides the RA with sufficient information to allow the RA to verify the identity of the end entity as required by the CPS. This includes the distinguished name of the end entity.

The certificate request process shall use the entity's private key to sign the certificate request data (Refer to Annex C for suitable forms of identity verification), and the RA or CA shall validate the signature on the certificate request data to ensure

a) the integrity of the entity's distinguished name, public key, and other information during the application process using the private key,

b) that the public key in the certificate request data corresponds to the entity's private key, and

c) that there has been no failure in the key generation and transmission process.

Subsequent to key generation and prior to key usage, the end entity shall

— compile the certificate request data, including the end entity's distinguished name and newly generated public key,

— digitally sign the certificate request data using the private key that relates to the public key contained in the certificate request data, and

— submit the signed certificate request data and other information required by the CPS to an RA or a CA.

6.3.3 Registration

When the RA or a CA performing RA functions applies for a certificate on behalf of the end entity, the CA or RA shall

a) validate the identity of the requesting end entity, according to the CPS of the CA (suggested requirements for accepting certificate request data are contained in Annex C),

b) validate the end entity's possession of the private key corresponding to the public key for which a certificate is requested,

c) accept certificate request data from the end entity whose identity has been validated (if the certificate request data was generated by the end entity),

d) check the certificate request data for errors or omissions,

e) validate the end entity's digital signature on the certificate request data,

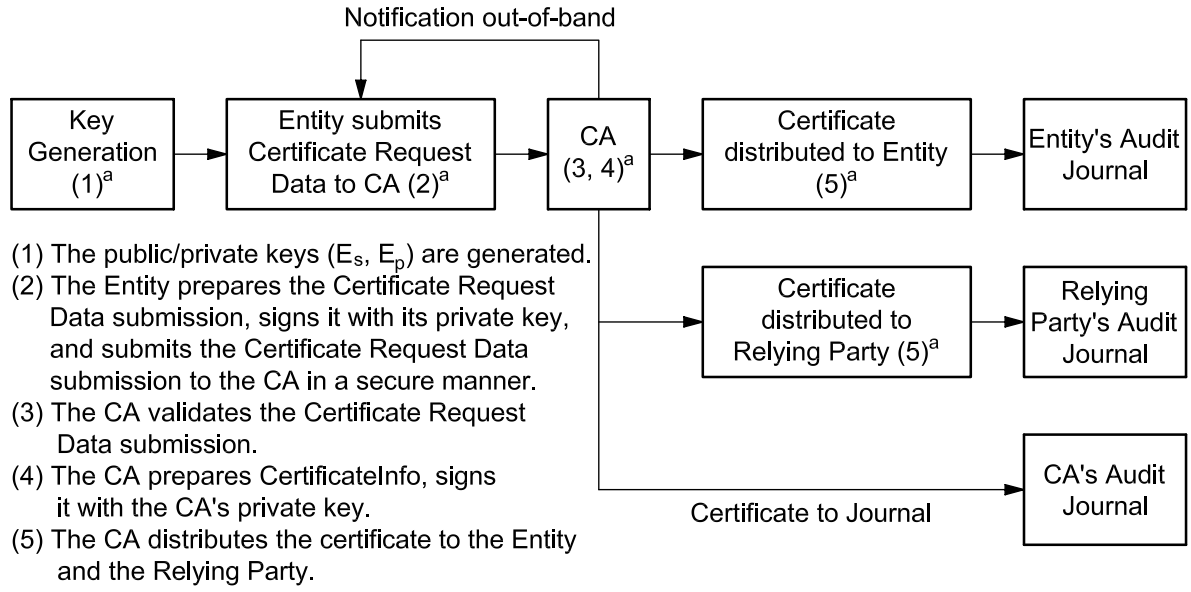
f) deliver a copy of the CA's public key in accordance with the requirements of 6.3.5, and a copy of the end entity's certificate to the end entity,

g) provide a notification to the end entity confirming successful registration and issuance of the certificate using an out-of-band method, and

h) depending on business, record its actions in an audit journal.

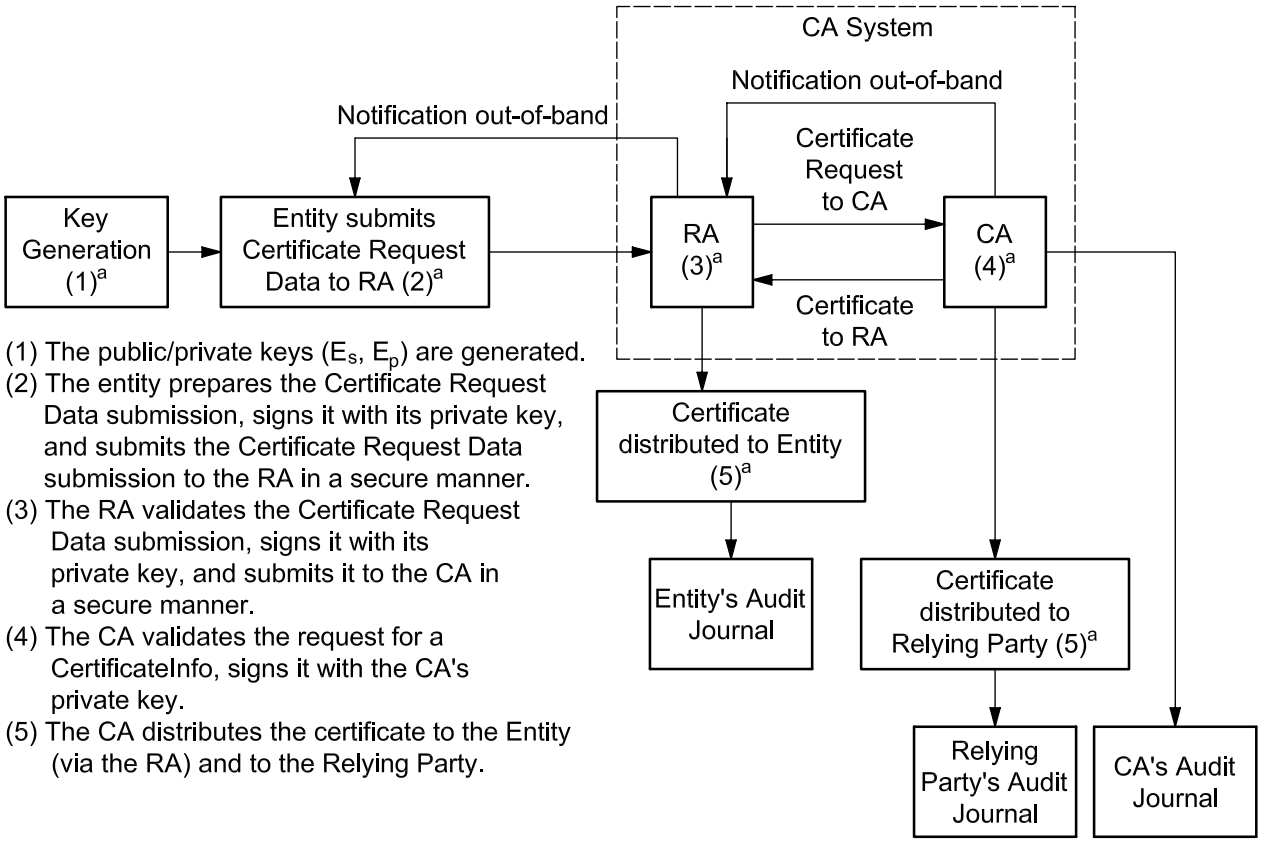
See 6.4.1 and Annex A.

Figure 2 summarizes a process for issuing a certificate using a CA alone. Figure 3 summarizes one approach for issuing a certificate by a CA using an RA.



a The numbers in the boxes refer to process steps in Figure 1.

Figure 2 — Issuance of certificate by a CA alone



a The numbers in the boxes refer to process steps in Figure 1.

Figure 3 — Issuance of certificate by a CA using an RA

6.3.4 Certification

For requests for new public key certificates (X.509, v3), the CA shall

a) verify the authenticity of the submission by the RA,

b) validate the signature on the certificate request data submission,

c) ensure that there is no duplication of the requester's distinguished name with that of any other entity certified by the CA,

d) ensure the uniqueness of the public key submitted for certification within the CA's domain, and

e) when duplicate public keys are detected, the CA shall

   1) revoke all certificates that contain the duplicated public key, and

   2) reject the request for a certificate.

CA Systems shall perform public key validation based on business risk. If any of these checks fail, the CA shall reject the certificate request.

For certificate requests, the CA shall

— ensure that the elements of the certificate information provided in the certificate request data are in compliance with the CA's CPS and, if appropriate, complete or modify those elements to achieve compliance,

— generate or obtain additional data elements to complete the certificate information (**CertificateInfo**), depending on the type of certificate requested,

— use one or more CA private keys to sign the **CertificateInfo,** thereby creating one or more certificates depending upon the business model in operation,

— verify its own signature on the certificate prior to issuing the certificate,

— deliver a copy of the CA's public key or keys, in accordance with the requirements of 6.3.5, and a copy of the end-entity certificate(s) to the RA [1)] or directly to the end entity, depending on the business model,

— securely notify the RA, using an out-of-band means if necessary, that one or more certificates have been generated, and

— record its actions in an audit journal.

6.3.5 Distribution of CA public keys

The CA system shall provide its entities and Relying Parties with one or more trusted CA public keys and associated parameters, which these entities can use to validate the first certificate in a path.

The CA may distribute multiple public keys to provide for the replacement of a public key upon the expiration of the cryptoperiod of a given public/private key pair and for backup and recovery purposes.

1) The RA does not need to receive the CA public key on every request and thus can provide the CA public key to the subject at the time of submission.

The integrity of a CA's public key and any associated parameters is essential. When a single CA or a certification path is employed, the integrity of the certification path depends on the integrity of the public key and the confidentiality of the private key of every CA in that path.

The CA shall distribute its public key, associated parameters including the validity period and the CA's distinguished name, and ensure the integrity and authenticity of that key during distribution.

The method of distribution will depend upon the business-risk model employed, and the reason for distribution, but shall be one of the following.

a) Trusted CA public keys may be initially distributed using

   — machine readable media (e.g. IC card),

   — embedding in an entity's cryptographic module,

   — a self-signed certificate, which contains the trusted public key, signed with the corresponding private key (note that the signature does not, in itself, prove the identity of the signer, this being done via out-of-band means), or

   — non-automated means.

b) if a Subscriber or Relying Party already has an authenticated copy of a trusted CA public key, a new trusted CA public key may be distributed by

   — direct electronic transmission from the CA,

   — placing in a remote cache or directory,

   — loading into a cryptographic module, or

   — any of the methods for initial distribution.

Regardless of the method of transmission, the integrity and authenticity of the trusted CA public key shall be ensured. This may be achieved by using one of the following, or an equivalent method.

— Distribute media under dual control with split knowledge and obtain receipts. This method is not applicable to direct electronic transmission.

— Sign a new trusted CA public key using an existing trusted CA private key. If this method is used, the recipient shall validate the signature on receipt. This option is not available if the CA private key is (suspected) compromised.

— Compute a message authentication code (MAC) in accordance with ISO 16609 over the CA public key. If this method is used, the recipient shall validate the MAC on receipt.

A CA's key may be contained in a certificate signed by another CA and be a part of a certification path, in which case, the key may be validated using the public key from the previous certificate in the path.

To validate a path that includes multiple CA certificates, the initial certificate in a path is validated by a trusted CA public key that shall be distributed by one of the methods described in this part of ISO 15782.

The validator shall ascertain that this trusted CA public key is currently valid. Annex D describes a method for extracting a CA's public key from a cross-domain certification path (a path that includes one or more CAs using two or more digital signature algorithms, e.g. DSA and RSA).

6.3.6 Distribution of certificates

Certificates may be distributed to Relying Parties using one or both of the following methods:

— certificates are explicitly included with the electronic message;

— certificates are implicitly in the electronic message and are contained in a remote cache or repository.

The RA may also distribute the certificate to Relying Parties. Note that the means for distribution are not specified in this part of ISO 15782.

6.3.7 Usage of certificates

The usage of certificates shall meet the requirements in 6.3.7.1 and 6.3.7.2.

6.3.7.1 In order to validate a certificate and obtain a validated public key for immediate use, the Relying Party shall check

a) the validity period of the certificates (note that synchronization and secure maintenance of the clocks of the sender, recipient and all CAs in the path is an issue and that solutions should be based on business risk).

b) the revocation status of all certificates in the certification path, using CRLs or other appropriate status mechanisms (e.g. Online Certificate Status Protocol (OCSP) or another standard certificate status protocol), and

c) the validity of the signatures on all certificates in the certification path:

   — if the system uses a trusted time mechanism and the certificate was revoked *before* the trusted time parameter contained in the signed message then the certificate shall be treated as invalid, and any business decision that contradicts this requirement will be outside the scope of the certification system and will implicitly accept the possibility that the message may be repudiated, or

   — if the system does not use a trusted time mechanism, then it may be possible to validate signed messages using a certificate that has been revoked but that appears to have originated before the revocation took place.

   A business decision shall be made by the Relying Party as to whether to accept or reject the message. This will depend upon the business-risk profile of the system and will take into account the possibility that the message may be repudiated.

6.3.7.2 Whenever a validation failure occurs, the following action shall be taken:

a) for high-risk applications, Relying Parties shall record validation failures in an audit journal;

b) Relying Parties shall retain records associated with validation failures for the period of time required by law, regulation, or prudent business practice, and these records shall include

   — the message that failed validation,

   — the certificate chain associated with the message, and

   — the Certificate Revocation Lists (CRL) or the use of other standard certificate status-checking protocol.

A certificate shall be used only for the purposes outlined in the Certificate Policy.

See ISO/IEC 9594-8 and ISO 15782-2 for detailed certificate path processing requirements.

The integrity of any public key and associated parameters retained for future use shall be ensured. One method is to store the entire certificate and validate the certificate as required. Another method is to extract the public key for operational use, while protecting the key from accidental or deliberate modification. Regardless of the method, the Relying Party shall allow re-validation of the public key. This includes the validation of the time span during which the key is authorized for use and any possible certificate revocations or holds.

In accordance with 6.3.3 and 6.3.4, it is required that an entity be provided with either

— the public key of the CA which signs the entity's certificate, or

— the public key of any CA in a possible certification path between the originator and receiver of signed information.

If the public key of the Relying Party's CA is used to unwrap a certification path, the path extends from the Relying Party's CA to the originator's CA to the originator.

If the public key of another CA in this path is trusted, the path may be shortened because fewer certificates would need to be validated. For example, the trusted public key of the topmost CA in a hierarchy could be used, allowing the path to extend from that topmost CA to the originator, with no need for a path from the Relying Party to the topmost CA. Some examples of these alternative trust models are presented in Annex B.

6.3.8 Revocation and hold or expiry

6.3.8.1 General

A certificate has a lifetime that is indicated by a validity period stated in the certificate but may be further constrained by the CA as defined in the Certificate Policy.

A CA may place a temporary hold on a certificate without permanently revoking it. Reasons for such an action include

a) a desire to reduce liability for erroneous revocation when a revocation request is not authenticated and there is inadequate information to determine whether the revocation request is valid, and

b) other business needs, such as temporarily disabling the certificate of an entity pending an audit or investigation.

A certificate revocation or hold may be requested on an entity's certificate by that entity or by an RA associated with that entity. RA association is established at the time of a certificate request according to CA and RA policy. Where the prevention of denial of service is required, the revocation or hold request shall be authenticated. In this case, a digital signature may be used.

Certificates may be revoked or held only by the CA that issued the certificate. This may occur for a number of reasons:

— end-entity private key compromise;

— CA private key compromise;

— change of affiliation (e.g. to a different CA);

— a public key being superseded with another;

— cessation of operations;

— certificate hold; or

— unspecified reasons.

However, the reason code CRL entry extension should be absent instead of using the **unspecified** reason code value.

A CA (or RA) shall validate the identity of an entity requesting the revocation or holding of a certificate, according to commercially reasonable procedures corresponding to the relative risk of an unauthorized revocation or hold.

A procedure and means of rapid communication shall be in place to facilitate the secure and authenticated revocation or holding of

— one or more certificates of one or more entities,

— the set of all certificates issued by a CA based on a single public/private key pair used by a CA to generate certificates, and

— all certificates issued by a CA, regardless of the public/private key pair used.

Whether certificates expire, are revoked or are held, copies of old certificates and CRLs shall be retained by the issuing CA for the period of time required by law, prudent business practice and regulations. Note that this is a records-retention requirement. Requirements for keeping CRL entries for expired certificates are stated in 6.3.8.2.

In the case of revocation of a certificate, all certificates containing the same public key shall be immediately revoked.

6.3.8.2 Certificate revocation process

Certificates that have been revoked or held may be distributed in time-stamped Certificate Revocation Lists (CRLs). Time-stamping is critical for indicating the time at which the binding between an entity's public key and identity has been terminated (or held). Upon revocation, the entity whose certificate is revoked cannot generate signatures with the private key corresponding to the public key in the revoked certificate, which can be verified to be valid.

When Certificate Revocation Lists are used, these lists shall be

— created and signed by the CA so that Relying Parties can validate the integrity of the CRL and the date of issuance,

— issued by the CA at regular intervals, even if no changes have occurred since the last issuance, and

— accessible to all entities and Relying Parties of the system.

The frequency and timing of CRL issuance is defined in the CPS. However, the online distribution of CRLs need not be made to all entities. CRLs may also be made available by a Directory Service. As a minimum, CRL entries identifying revoked certificates shall remain on the CRL until the end of the validity period of that certificate.

See ISO/IEC 9594-8 and ISO 15782-2 for a discussion of ways to partition CRLs (e.g. delta CRLs and distribution points).

6.3.8.3 Certificate hold and release

A certificate may be placed on hold by issuing a CRL entry with a reason code of **certificatehold**. An optional hold instruction code may also be included to convey additional information to Relying Parties.

The duration of a certificate hold will vary according to the purpose of the hold. In the case of an unauthenticated revocation request where the CA is unsure of the identity of the requester, the hold might last for a few days or even only a few hours, as needed to investigate and validate the authenticity of the request.

If the certificate of a substantial enterprise is placed on hold, the CA will wish to complete its investigation as quickly as possible. Holds issued for other business purposes might apply for a longer time.

Once a hold has been issued, the hold shall be handled in one of three ways:

a) remain on the CRL with no further action, causing validation of the certificate to fail during the hold period;

b) be replaced by a revocation, for the same certificate, in which case, the reason shall be one of the standard reasons for revocation and the optional instruction code extension field shall not appear;

c) be released and the entry removed from the CRL.

The certificate may be used after the release of the hold. The hold is released by not putting the certificate on the next CRL.

6.3.8.4 Actions taken when a certificate is revoked or held

6.3.8.4.1 General

The actions that shall be taken when a certificate is revoked or held are defined in Table 1. Additional actions to be taken for each reason code are defined in Tables 2 to 6.

6.3.8.4.2 Actions taken during certificate revocation

During the certificate revocation process, the CA shall

a) verify the identity and authority of the entity requesting revocation of a certificate,

b) validate the revocation request,

c) prepare the revocation notice, sign it with its private key and (optionally) send it to the requesting entity,

d) prepare the **CRLInfo** and sign it with its private key,

e) ensure availability to all entities of information on certificate status, and

f) record its actions in an audit journal.

The CA should provide an authenticated acknowledgement of the revocation to the entity, and may also distribute the revocation notice to other Subscribers and Relying Parties.

When the RA assists an entity in the certificate revocation process, the RA shall

— verify the identity and authority of the entity requesting revocation of a certificate,

— submit certificate revocation requests to the CA in an authenticated manner as required by the CPS,

— receive and verify the confirmation that the CA has received the revocation request,

— secure that part of the revocation process for which the RA assumes responsibility,

— provide an authenticated acknowledgement of the revocation to the requesting entity,

— ensure availability to all Subscribers and Relying Parties of information on certificate status, and

— record its actions in an audit journal.

It should also provide an authenticated acknowledgement of the revocation to the entity as required by the CPS.

When the RA assists a requesting entity in the certificate hold process, the RA shall

— verify the identity and authority of the entity requesting the hold of a certificate, and

— submit the certificate hold request to the CA in an authenticated manner,

— provide a written acknowledgement of the hold.

Also see Table 1.

6.3.8.4.3 Additional actions to be taken

See Tables 2 to 6.

Further additional actions that may be taken by an entity or RA requesting a hold are the following.

— An entity or RA may request that the CA that issued a certificate to be held place a hold on the certificate giving the **CertificateSerialNumber** to identify the certificate, and an optional **CRLEntry reasonCode** value of **certificateHold**.

— If the certificate to be held is that of a CA, the Certificate Revocation List of the CA performing the revocation shall contain entries for all revoked and held certificates of the subject CA.

Table 1 — Actions to be taken whenever a certificate is revoked or held for any reason

Entity	Actions to be taken
Certified entity or RA	The certified entity or RA may: 1. request that the CA revoke, hold, or release the hold on a certificate, giving the **CertificateSerialNumber** to identify the certificate, and the optional **CRLEntry ReasonCode requested**, 2. send a message to notify other entities and Relying Parties and identify the certificate for the CRLEntry, 3. update the audit journal to reflect the actions taken and the reasons for the actions. Revoked or held certificates shall be journalized.
CA	The CA shall ascertain the validity of the revocation or hold request, according to the CA's CPS and perform the following actions. 1. Update the CRL. In the case of a certificate revocation, the certificate shall remain on the revocation list until the first CRL issued following the expiration date of the certificate, as a minimum. In the case of a certificate hold, the certificate shall remain on the CRL until either the explicit release of the hold, the expiration of the hold, or the expiration of the underlying certificate (whichever comes first). 2. Send (optionally) a signed message (out-of-band) containing the CRLEntry to all entities and Relying Parties. A revocation notice may be used for this purpose. 3. Update the audit journal to reflect the actions taken and the reasons for the actions. Revoked or held certificates shall be journalized.
Users of the certificate	The user shall: 1. reject any message signed after the revocation date requiring the use of the revoked certificate, 2. update the audit journal to reflect the actions taken and the reasons for the actions. Revoked or held certificates shall be journalized. Optionally, other entities may be notified. Table 6 defines additional requirements when certificates for public keys that are used to protect symmetric algorithm key exchanges are revoked or held.

Table 2 — Additional actions taken on the compromise or suspected compromise of an entity's private key

Entity	Additional actions to be taken
Certified entity or RA	The certified entity or RA may request that the entity's CA revoke the certificate, giving the **CertificateSerialNumber** to identify the certificate and with an optional **CRLEntry reasonCode** value of **keyCompromise** or **caCompromise**. If the entity is a CA, all the suspected certificates shall be revoked, and the **CertificateRevocationList** of the CA itself may contain entries for all certificates of the suspect CA, with an optional **reasonCode** value of **caCompromise**.
CA	The issuing CA shall update the Certificate Revocation List (**CertificateRevocationList**). The **CRLEntry** in the **CertificateRevocationList** may optionally contain a reasonCode value of **keyCompromise** or **caCompromise**, as appropriate.
Users of the certificate	Discontinue the use of all keying material ever sent and protected by that certificate.

Table 3 — Additional actions taken because of cessation of operations

Entity	Additional actions to be taken
Certified entity or RA	The certified entity or RA may request that the entity's CA revoke the certificate, giving the **CertificateSerialNumber** to identify the certificate and a **CRLEntry** and optional **reasonCode** value of **cessationOfOperation**. If the entity is a CA, it shall revoke all certificates that the CA has issued. The request may be submitted by the entity or its legal representative.
CA	The issuing CA shall update the Certificate Revocation List (**CertificateRevocationList**) giving the optional **reasonCode** value of **cessationOfOperation**. The business issues regarding the management and ownership of the entity or CA that has ceased operations shall be addressed.

Table 4 — Additional actions taken because of change of affiliation of entity

Entity	Additional actions to be taken
Certified entity or RA	The certified entity or RA may request that the entity's CA revoke the certificate giving the **CertificateSerialNumber** to identify the certificate with a optional requested **CRLEntry reasonCode** value of **affiliationChanged**. If the entity is a CA, the **CertificateRevocationList** of the CA itself may contain entries for all revoked certificates.
CA	The CA shall update the Certificate Revocation List (**CertificateRevocationList**), giving the optional **reasonCode** value of **affiliationChanged**.

Table 5 — Additional actions taken when certificates are revoked for reasons other than for key compromise, cessation of operations or change of affiliation

Entity	Additional actions to be taken
Certified entity or RA	The entity or RA may request that the entity's CA revoke the certificate giving the **CertificateSerialNumber** to identify the certificate and an optional Certificate Revocation List entry (**CRLEntry**) **reasonCode** value of **superseded** or **unspecified**. If the entity is a CA, the **CertificateRevocationList** of the CA itself shall contain entries for all revoked certificates, with a **reasonCod**e value of **superseded** or **unspecified**.
CA	The CA shall update the Certificate Revocation List (**CertificateRevocationList**) giving the optional **reasonCode** value of **superseded** or **unspecified**, as appropriate.

Table 6 — Additional actions taken when certificates for public keys used to protect symmetric algorithm key exchanges are revoked or held

Reason for revocation or hold	Additional actions to be taken by users of the certificate
Compromise or suspected compromise of an entity's private key	The use of all keying material ever sent and protected by that certificate (without regard to type) shall be discontinued. If the entity whose certificate is revoked or held is a CA, a different CA shall be used in the process of replacing keying material.
Certificate expires or is revoked for reasons other than actual or suspected compromise	Replace all keying material sent and protected by that certificate (without regard to type) as soon as it is operationally convenient.
Certificate is held for reasons other than actual or suspected compromise	When a certificate is held for reasons other than actual or suspected compromise, optionally suspend the use of, or replace, all keying material sent and protected by that certificate (without regard to type) as soon as it is operationally convenient.

6.3.9 Renewal of certificates

6.3.9.1 A certificate has a lifetime indicated by the validity period stated in the certificate.

Before this validity period has expired, an end entity may request the renewal of a certificate by requesting an extension of its validity period (i.e. requesting a **notAfter** date later than the existing certificate's **notAfter** date). However, the start date (**notBefore** date) must be the same in the renewed certificate as it is in the original certificate. The renewed certificate will have the same key pair as the original certificate.

Certificates may be renewed only by the CA that issued the certificate.

An end entity requesting the renewal of a currently valid certificate shall compile its certificate renewal data and submit it to the RA, or the CA performing RA functions, that had previously generated this certificate. The certificate renewal data provides the RA with sufficient information to allow the RA to verify the identity of the end entity and to identify the certificate to renew. This includes

a) the distinguished name of the end entity,

b) the serial number of the certificate, and

c) the requested validity period.

The end entity shall digitally sign the certificate renewal data.

6.3.9.2 For renewing public key certificates, the CA shall

a) validate the signature on the certificate renewal data submission,

b) verify the existence and validity of the certificate to be renewed,

c) verify that the request of the validity period includes the same **notBefore** date as the **notBefore** date in the original certificate, and that the **notAfter** date is later than the **notAfter** date in the original certificate, and

d) verify that the request, including the extension of the validity period, meets the requirements defined in the Certificate Policy.

If none of these checks fails, the CA shall generate and sign a new instance of the certificate, differing from the previous certificate only by the validity period, certificate serial number and the CA signature. If a check fails, then the CA shall reject the certificate renewal request.

The CA shall make the new certificate available to the end entity in accordance with the CP. A certificate renewal results in two certificates with the same public key. This implies that

— renewal of a certificate does not require revocation of the previous instance of the certificate, and

— when the validity periods overlap, any of the different instances of a certificate may be used.

6.3.10 Certificate Re-key

6.3.10.1 A public key has a lifetime, known as cryptoperiod, which may differ from the validity period stated in the certificate. Before the end of the validity period, or whenever there is a risk that the key could be compromised, an end entity may request a re-key, that is to say to produce a new certificate for a new public key following the generation of a new key pair. This new certificate will have a validity period set in accordance with the CP. If a re-key was requested due to compromise, the old certificate should be revoked and placed on the CRL.

An end entity requesting the re-key of a currently valid certificate shall compile its certificate re-key data and submit it to the RA, or the CA performing RA functions, that had previously generated this certificate. The certificate re-key data provides the RA with sufficient information to allow the RA to verify the identity of the end entity and to identify the certificate to re-key. This includes

a) the distinguished name of the end entity,

b) the serial number of the certificate, and

c) the requested validity period.

6.3.10.2 The certificate re-key request process shall use the entity's private key to sign the certificate re-key data, and the RA or CA shall validate the signature on the certificate re-key data to ensure

a) the integrity of the entity's distinguished name, the new public key, and other information during the application process using the new private key,

b) that the public key in the certificate re-key data corresponds to the entity's new private key, and

c) that there has been no failure in the key generation and transmission process.

6.3.10.3 Subsequent to key generation and prior to key usage, the end entity shall

— compile the certificate re-key data, including the end entity's distinguished name and newly generated public key,

— digitally sign the certificate re-key data using the new private key that relates to the new public key contained in the certificate re-key data, and

— submit the signed certificate re-key data and other information as required by the CPS to an RA or a CA.

6.3.10.4 For re-keying public key certificates, the CA shall

a) validate the signature on the certificate re-key data submission,

b) verify the existence and validity of the certificate to be re-keyed, and

c) verify that the request, including the requested validity period, meets the requirements defined in the Certificate Policy.

If none of these checks fails, the CA shall generate and sign a new certificate (with a new certificate serial number). If a check fails, then the CA shall reject the certificate re-key request.

The CA shall process a re-key request only if the previous certificate is valid (neither expired nor revoked).

Certificates may be re-keyed only by the CA that issued the certificate.

6.4 Security quality assurance and audit requirements

6.4.1 General

CAs and RAs shall maintain sound management and control practices confirmed via security quality-assurance processes and procedures, and compliance audits.

6.4.2 Audit journal requirements

CA systems are required to keep audit journals that provide sufficient detail to reconstruct events, provide "due care" requirements and meet legal requirements. All entries in audit journals shall be date and time stamped. While audit journals will normally be created and maintained by the CA management system, some audit journals may out-of-necessity be manual. The audit requirements shall be specified in the CA's Certification Practice Statement.

CA audit journal entries shall include all certificate and key management operations, such as key generation, backup, recovery and destruction, together with the identity of the person authorizing the operation and persons handling any key material (such as key fragments or keys stored in portable devices or media). Changes in the custody of private keys and associated parameters, and of devices or media holding keys shall be recorded in the audit journals. Audit journals shall not record the plain text values of any private keys but may hold hash values as a means of identifying keys and validating their correctness, as well as that of public keys derived from private keys by means of a one-way function.

A list of audit journal contents is provided in Annex A.

Audit journals shall be maintained in a form that prevents unauthorized modification or destruction. Automated audit journals shall be protected from modification or substitution. The use of a hash and a digital signature may be used. The private key pair used for signing the audit journal shall not be used for any other purpose. In addition, the audit journal shall only be retrieved by authorized individuals for valid business or security reasons.

6.4.3 Security quality assurance

Documented security quality-assurance processes and procedures are required as part of the system of internal security control over certificate management. The audit journal shall be reviewed regularly (e.g. daily) by a security quality-assurance function. In some organizations, this function may be fulfilled by the audit department. The review shall include the validation of the audit journal's integrity, and the identification and follow-up of exceptional, unauthorized or suspicious activity (e.g. digital signature failures, access at unusual times or from unusual sources, unexpected increases in volume or saturation of system resources).

The extent and frequency of review and management escalation requirements should be determined by a threat/risk evaluation. In high-risk applications or for legal purposes or both, CA systems may require end entities and Relying Parties to maintain an audit journal.

6.4.4 CA and RA audit

Compliance reviews shall be performed by an independent audit function (internal, external or both) at a frequency (e.g. annually) to be determined by the audit function. The audit should include compliance with the Certificate Policy and Certification Practice Statement, procedures manuals and configuration specifications. Audit results shall be formally documented and provided to the auditee for follow-up.

6.4.5 End-entity audit

In high-risk applications, end entities and Relying Parties may also be audited.

6.5 Business continuity planning

The requirements for business continuity planning depend on the availability and continuity needs of the business applications supported by the CA. In general, high-risk, high-availability applications require more robust business continuity processes and techniques than low-risk, low-availability applications.

As a minimum, business continuity planning shall include disaster recovery processes for all critical components of a CA system, including the hardware, software and keys, in the event of a failure of one or more of these components.

Disaster recovery options may include the re-installation of the CA system and the re-issuance of all certificates, the use of a fully redundant system, or a “hotsite”.

Disaster recovery processes are also required if a critical security component is compromised. In particular, the compromise or suspected compromise of a CA's private key shall be considered a disaster. All certificates signed with that key after the compromise date (where available) shall be considered suspect and therefore be revoked. The security measures implemented at the CA in order to protect the private key of the CA shall ensure that the probability of a compromise of that key is negligible.

In the event that a CA has to replace or recover its private key, procedures shall be in place for the secure and authenticated revocation of all certificates issued by the CA using this private key.

Examples of disaster recovery procedures are described in Annex E.

An entity may have one or more valid certificates in anticipation of a need for transition or recovery. These certificates provide continuity of service when a certificate expires, a cryptographic module fails or a private key is compromised.

7 Data elements and relationships

The authentication framework defined in this part of ISO 15782 provides for either a hierarchical or non-hierarchical structure for point-to-point connections and is based on ISO/IEC 9594-8. The authentication framework is based on providing assurances that the public key of an entity is contained in the certificate request data of that entity. In turn, this assumes that there are assurances that the entity presenting certificate request data to be signed is the entity holding the private key corresponding to the public key contained in the certificate request data. Refer to ISO 15782-2 for associated ASN.1 structures.

8 Public key certificate and Certificate Revocation List extensions

ISO/IEC 9594-8 provides the syntax and semantics for a public key certificate. Version 3 (V3) public key certificates provide a mechanism for CAs to append additional information about the entity's public key, issuer's public key and issuer's CRLs. Standard certificate extensions are defined. Since ISO/IEC 9594-8 is intended to be applicable to a widely diverse community of users, it offers numerous optional features.

Refer to ISO 15782-2 for a description of a profile of the certificate and CRL extensions that appear in ISO/IEC 9594-8 with requirements for financial applications.

Annex A
(normative)

Certification Authority audit journal contents and use

A.1 CA and RA audit journal contents and protection

A.1.1 General

Audit journal entries shall be precisely identified as to source, date and entry number. Only information authorized and required for the audit journal may be entered into the audit journal.

Most items journalized are in electronic form; however, some significant security-related events shall be manually recorded.

The provisions of 6.4.2, last paragraph, shall be duly noted.

A.1.2 Elements to be included in all journal entries

The following elements shall be included in all journal entries:

a) date and time of the entry;

b) serial or sequence number of entry;

c) type of entry;

d) source (terminal, port, location, customer, etc.);

e) identity of the entity making the journal entry.

Manual journal entries shall also include the following, if appropriate:

— identity of the entity authorizing the operation and entities handling any keying material (such as key fragments or keys stored in portable devices or media);

— custody of keys and of devices or media holding keys.

A.1.3 Certificate application information to be journalized by an RA or CA

The following information shall be included in the audit journal:

a) type of identification document(s) presented by the applicant;

b) record of unique identification data, numbers, or a combination thereof of identification documents (e.g. applicant's driver's licence number), if applicable;

c) storage location of copies of applications and identification documents;

d) identity of entity accepting the application;

e) method used to validate identification documents, if any;

f) name of receiving CA or submitting RA, if applicable;

g) the subject's acceptance of the Subscriber Agreement;

h) where required under privacy legislation, the Subscriber's consent to allow the CA to keep records containing personal data and pass this information to specified third parties; and publication of certificates.

A.1.4 Events to be journalized

Events to be journalized shall include the following information concerning keying material:

a) generation;

b) installation of manual cryptographic keys and its outcome (with the identity of the operator);

c) backup;

d) storage;

e) recovery;

f) withdrawal of keying material from service;

g) escrow;

h) archival;

i) destruction;

j) generation of certificates;

k) receipt of requests for certificate(s) including initial certificate requests, renewal requests, and re-key requests;

l) certificate revocation and suspension requests;

m) certificate revocation, suspension, and re-activation;

n) receipt, generation and sending of Certificate Revocation Lists;

o) distribution of the CA's public key;

p) submissions of public keys for certification;

q) actions taken on compromise of a private key.

A.1.5 Security-sensitive events to be journalized

Audit journal records shall include information needed to analyse patterns. Security-sensitive events and associated information to be journalized include the following:

a) security-sensitive files or records read or written;

b) actions taken against security-sensitive data;

c) security profile changes;

d) use of identification and authentication mechanisms, both successful and unsuccessful (including multiple failed authentication attempts);

e) security-sensitive non-financial transactions (e.g. account or name/address changes, etc.);

f) system crashes, hardware failures and other anomalies;

g) actions taken by individuals in Trusted Roles: computer operators, system administrators, and system security officers;

h) change of affiliation of an entity;

i) audit journal access;

j) decisions to bypass encryption/authentication processes or procedures;

k) access to the CA system or any component thereof.

A.1.6 Messages and data to be journalized

The following messages and data shall be included in the audit journal:

a) all messages/data (in electronic form) in or out of the CA (including Certificate Revocation Lists);

b) all certificates generated;

c) identity of cryptographic keys used and their hash (where applicable)[2)];

d) the identity of the persons receiving the key fragments.

A.2 Audit journal backup

The CA and RA audit journal data shall be backed up off-site at appropriate intervals.

A.3 Audit journal use

Manual or automated procedures shall be available to facilitate the use, review and maintenance of the audit journal. Procedures should identify when the violation(s) should be reported to management.

Examples of conditions requiring analysis and possible action include

a) unusual saturation of system resources,

b) sudden, unexpected increases in volume, and

c) access at unusual times or from unusual places.

2) Check values derived from keys using an approved hash should be journalized, as a means of identifying keys and verifying their correctness.

Annex B
(informative)

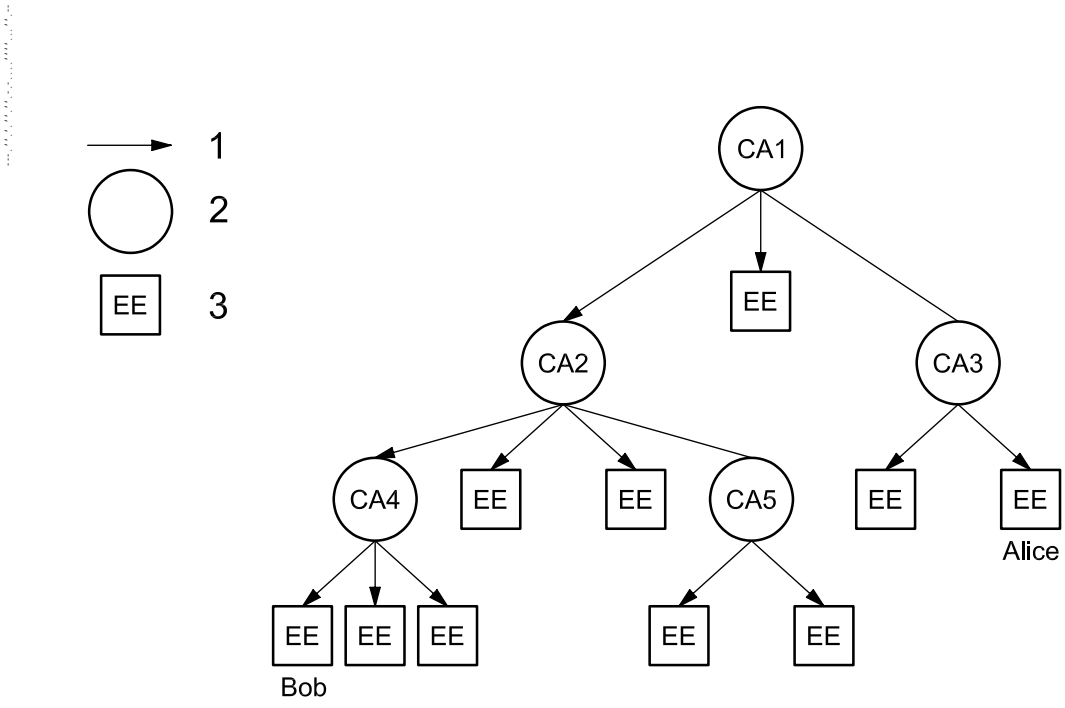
Alternative trust models

B.1 General

Trust models define how trust is transferred between CAs. To validate a certificate, a Relying Party finds a trusted path of certificates from the certificate being validated to a CA certificate that the entity trusts. An entity of one CA might not hold (or trust) the public key of the CA that signed another entity 's certificate.

Various trust models define mechanisms to construct a certification path through a CA hierarchy. These trust models share the common characteristic that a Relying Party need only trust one CA public key in order to obtain and validate the entity's certificate. The trusted key is typically that of the root CA (a hierarchical trust model) or of the CA that issued the entity's certificate (a non-hierarchical trust model).

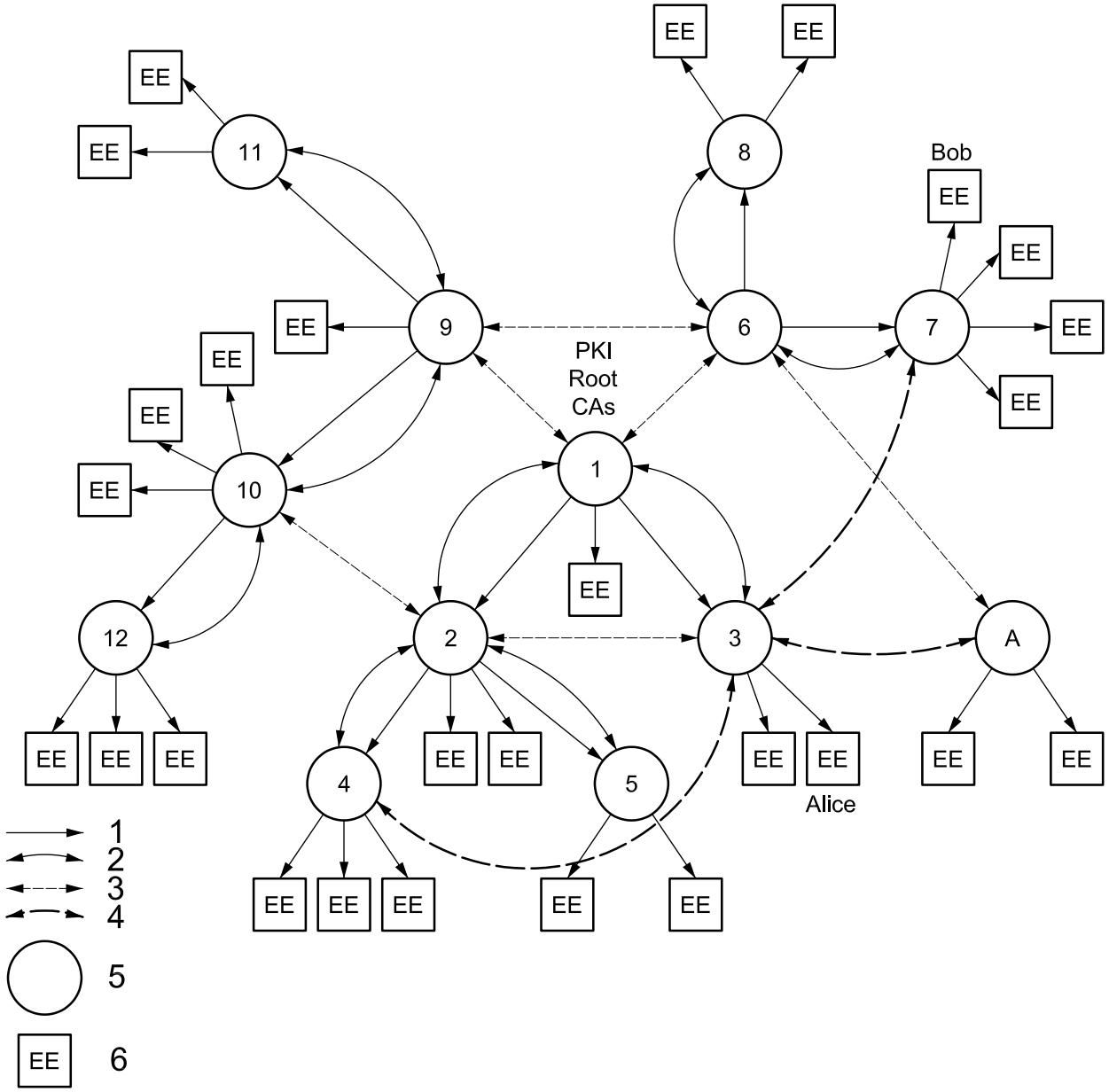
Trust models are identified by the method used by CAs to issue certificates to other CAs. CAs can issue certificates to each other in a systematic and ordered way or in a more flexible and less ordered way. The systematic, ordered topology of certification paths that is normally employed is a hierarchy, as illustrated in Figure B.1. The more general topology is a network of cross-certified CAs as illustrated in Figure B.2. Hybrid topologies are discussed in B.4 and are illustrated in Figure B.3. All three models are discussed in B.2 to B.4 and the advantages and disadvantages of each model are summarized in Table B.1.



Key

1 certificate (point to entity certified)
2 Certification Authority
3 end entity

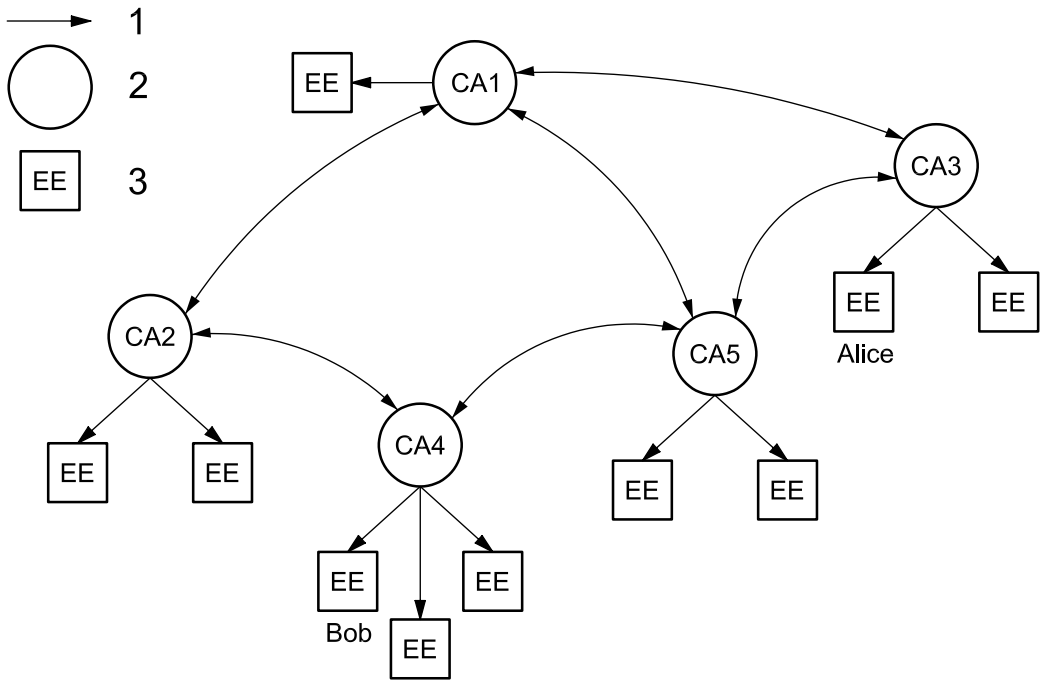
Figure B.1 — Hierarchical trust model



Key

1 certificate (point to entity certified)
2 required hierarchical cross-certificate
3 general cross-certificate
4 special cross-certificate
5 Certification Authority
6 end entity

Figure B.2 — Non-hierarchical trust model



Key

1 certificate (point to entity certified)

2 Certification Authority

3 end entity

Figure B.3 — Hybrid trust model

B.2 Hierarchical trust model

In the hierarchical trust model, the level of trust in any validated certificate is only as good as the level of trust in the root's public key. In this model, the root's public key is not certified, since there is implicitly no issuer to certify that public key, and the public key is distributed by an out-of-band means (e.g. by delivery using a properly insured, bonded courier). This requires the use of a hierarchical certification model, and has the property that an entity need not have a trusted copy of its own CA's public key.

In the hierarchical trust model, CAs are arranged hierarchically under a “root” CA that issues certificates to subordinate CAs. Figure B.1 illustrates a hierarchical trust model. These CAs may issue certificates to CAs below them in the hierarchy, or to entities.

In a hierarchical PKI:

a) the public key of the root CA is known to every entity;

b) any entity's certificate may be verified by verifying the certification path of certificates that leads back to the root CA.

Alice

— verifies Bob's certificate, issued by CA4, then

— CA4's certificate, issued by CA2, and then

— CA2's certificate issued by CA1, the root, whose public key she knows.

A reverse certification path (from the Relying Party to the root) is never required; only the path from the root to the entity being authenticated (i.e. the originator) is needed. Depending on the environment, this may minimize the average length of a certification path (e.g. if most communication is between widely separated entities).

Table B.1 — Trust model advantages and disadvantages

Trust Model	Advantages	Disadvantages
Hierarchical	— The organizational management structure of many organizations is largely hierarchical. Trust relationships are frequently aligned with organizational structure, so it is natural to align the certification path with the organizational structure. — The hierarchy may be aligned with hierarchical directory names. — The certification path search strategy is straightforward. — Some existing systems are designed hierarchically. — Each entity has a certification path back to the root. The entity can provide its path to any other entity, and any entity can verify the path, since all entities know the root's public key.	— It is improbable that there will be a single root CA for the world PKI. — Commercial and business trust relationships are not necessarily hierarchical. — Compromise of the root private key is catastrophic, and recovery requires the secure distribution of the new public key to every entity. However, the compromise of the issuing CA's private key invalidates only the certificates issued by that CA.
Non-hierarchical	— It is flexible, facilitates *ad hoc* associations and trusted relationships, and reflects the bilateral trust relationships of business. — An entity trusts at least the CA that issued its own certificate in any PKI, and it is reasonable to make this the foundation of all trust relationships. — CAs that are organizationally remote, but whose end entities work together with a high degree of trust, can be directly cross-certified under a high-trust policy that is not extended to other CAs and is higher than would be practical through a long, hierarchical chain of certificates. — It allows direct cross-certification of CAs whose entities communicate frequently, reducing the certification path processing load. — Recovery from the compromise of any CA's private key requires only that the new public key (and certificates signed with the corresponding new private key) be securely distributed to the holders of certificates from that CA.	— Certification path search strategies can be more complex. — An entity cannot provide a single certification path that is guaranteed to enable verification of his signatures by all other entities of the PKI.
Hybrid	— Implements a hierarchical network of trust that provides a certification path back to the root for each user. — Allows direct cross-certification of CAs whose users communicate frequently, reducing the certification path processing load. — Reflects the bilateral trust relationships of businesses. — Aligns the certification path with the organizational structure.	— Certification path search strategies can be much more complex. — Cross-certification among subordinate CAs presents a risk.

B.3 Non-hierarchical trust model

In the non-hierarchical model, each CA certifies its parent and subordinate CAs, and an entity holds the public key of its issuing CA. The entity trusts this key, regardless of whether the key is certified or not. This model allows the easy addition of entities, CAs, and levels of hierarchy. The impact of a key compromise or change to a CA is minimized since the shortest certification path will be via the least common ancestor CA of the communicating entities. In a large environment where most communication is between entities closely related to one another, this model may result in the shortest average certification path.

In a non-hierarchical trust model, independent CAs cross certify each other (that is, issue certificates to each other), resulting in a general network of trust relationships between CAs. Figure B.2 illustrates a non-hierarchical trust model. An entity trusts the public key of a local CA, generally the one that issued its certificate, and verifies the certificates of other entities by verifying a certification path of certificates that leads back to this trusted CA.

EXAMPLE

— Alice knows the public key of CA3, and

— Bob knows the public key of CA4.

There are several certification paths that lead from Bob to Alice, but the shortest requires Alice to verify

— Bob's certificate, issued by CA4, then

— CA4's certificate issued by CA5, and finally

— CA5's certificate, issued by CA3.

CA3 is Alice's CA, and she trusts CA3 and knows its public key.

B.4 Hybrid trust model

The hybrid trust model incorporates characteristics of both the hierarchical and non-hierarchical trust models. The fundamental building block is a hierarchy which includes root CAs, subordinate CAs, and end entities. Root CAs certify subordinate CAs and perform cross-certification with other root CAs. Every CA that is not a root CA will have a certification path to a root CA. This establishes a hierarchical path of certificates extending from a root CA to its subordinate CAs, and from each of these CAs to their respective subordinates.

Cross-certificate pairs are in parallel to the certificates hierarchically linking CAs to the root. These parallel cross-certificate pairs, shown in Figure B.3 as double-headed arrows, allow client applications that perform certification path validation from the verifier's parent CA, using the cross-certificate pair, to operate from any CA. This allows cross-certification with other domains to occur not only at the top (root CA) level, but also among subordinate CAs.

Within the hybrid trust model, three types of cross-certificates are defined: *hierarchical, general,* and *special.*

Hierarchical cross-certificates parallel the hierarchical certification path to the root CA. They ensure that Relying Parties that trust their local CA (as opposed to the root CA), can always find a certification path to any other entity in the PKI.

General cross-certificates supplement the certification hierarchy and allow for shorter certification paths. The rules regarding the use of general cross-certificates allow for a propagation of trust that is at least as restrictive as the propagation that would result from the use of the least restrictive certification path from the Relying Party's root CA to the certificate being validated.

Special cross-certificates provide certification paths that need not conform to the restrictions imposed hierarchically along the paths from the root CAs. Special cross-certificates may be created between “leaf” CAs. Leaf CAs are CAs that have a hierarchical certification path to the root and hold a certificate with a path length constraint of “0”. This permits further propagation of trust to another CA along the hierarchical certification

path. Special cross-certificates are appropriate when each of the two CAs operate under policies that allow a higher trust level or less restrictions than would otherwise be permitted.

In Figure B.3, Alice is shown in a different hierarchy from Bob within the PKI. If she wishes to verify Bob's signature, Alice has a variety of certification paths to choose from which to establish trust to her immediate CA, CA3, or her root CA, CA1. Alice's client certification-path validation process finds at least one certification path that meets the policy or other criteria that Alice requires, to validate Bob's certificate.

Annex C
(informative)

Suggested requirements for the acceptance of certificate request data

C.1 General

This annex provides guidance for the acceptance of certificate request data from an individual or legal entity in various situations. These are only suggestions, and are meant to illustrate that the level of effort required for acceptance should be proportional to the risk associated with accepting an invalid certification request (e.g. where the individual requesting a certificate is not the subject of the certificate, having presented falsified identification documents). The actual requirements will be determined by the CA's (and application's) security policy.

This procedure does not prohibit a policy of the CA that uses RAs that validate the identity of the applicant based upon the personal presence and presentation of identification credentials of the applicant.

The requirements presented in this annex are only one factor in determining how much trust to place in certificates issued by a CA. Other factors include CA operating policy, procedures and security controls, end-entity policy and procedures for handling private keys, etc. The liability assumed by the Certificate Issuers and end entities also play a role in the degree of trust. The certificate may contain Certificate Policies; these identifiers allow the user of the certificate to decide how much trust to place in the binding of the entity's identity and its public key. A Certificate Policy is thus "*a named set of rules that indicates the applicability of a certificate to a particular community and/or class of application with common security requirements.*" Specific security policies and the mechanisms that implement them are generally defined in a detailed document, a Certification Practice Statement (CPS).

For a comprehensive framework that identified the elements (items) to be considered in formulating a Certificate Policy, see ISO 21188.

C.2 Acceptance of certificate request data of an individual

C.2.1 Low-risk applications

EXAMPLE Retail banking credit cards and point of sale terminals (POS).

The acceptance of certificate request data should be based on identifying the individual applying for a certificate. For low-risk applications, certificate request data does not have to be presented in person. The means for identifying the individual should conform to reasonable commercial practices.

C.2.2 Medium-risk applications

EXAMPLE Low volume/low-value commercial EDI applications or personal asset management systems.

The acceptance of certificate request data should be based on one or more mechanisms to identify the individual applying for a certificate. These techniques may vary in strength somewhere between not being presented in person and being presented in person with reasonable commercial practices to identify the person. Mechanisms which could be used in combination include

— use and confirmation of a valid user ID and password,

— call-back procedures,

— message authentication codes (MACs), and

— IC cards or tokens.

C.2.3 High-risk applications

EXAMPLE Private banking system.

The acceptance of certificate request data should be based on the appearance in person (or by an authorized agent of that person) of the individual applying for the certificate, and the use of reasonable commercial practices to identify the person (or an agent of that person if required). This may involve the validation of the identity of that person by a trusted entity such as a bank account officer or a law firm approved by the financial institution.

C.3 Acceptance of certificate request data of a legal entity

C.3.1 A financial institution in a peer-to-peer relationship

EXAMPLE Clearing houses (e.g. CHIPS, CHAPS) and automated settlement systems of central banks.

The acceptance of certificate request data should be based on the hand delivery of the certificate request data by a representative of the entity, and

— a corporate resolution, signed and sealed (where applicable) on a letterhead of a senior official of the entity authorized to apply for the certificate, and

— a means for identifying the representative delivering the certificate request data that conforms to reasonable commercial practices.

C.3.2 A business customer of a financial institution

The acceptance of Certificate Request Data should be based on the hand delivery of the certificate request data by two or more representatives of the entity, and

— the signature and seal (where applicable) on a letterhead authorizing the application for a certificate,

— the use of reasonable commercial practices to identify the signature and seal (where applicable) of the entity, and

— the means for identifying the representatives delivering the certificate request data that conforms to reasonable commercial practices.

C.4 Acceptance of the certificate request data of a hardware device

The acceptance of certificate request data should be based on the unique identity of a hardware device and the owner of the hardware device. The owner of the hardware device may be an individual or an entity. Therefore, in addition to a unique device identity, the certificate request data also includes the appropriate information that would be required for acceptance of the certificate request data of an individual or a legal entity.

Annex D
(informative)

Multiple algorithm certificate validation example

D.1 Multiple algorithm certification paths

A multiple algorithm certification path (path) is one that includes CAs using different signature algorithms. Multiple algorithm paths may be encountered when the paths cross (e.g. financial system or national boundaries).

The Relying Party of a multiple algorithm path validates the signatures for all of the signature algorithms used in creating the certificates that constitute the path.

Note that the Relying Party does not need the capability to generate signatures.

D.2 Unwrapping DSA/RSA multiple algorithm certification paths

In the example that follows, the subscripts of the CA issuing each certificate denote the algorithm used.

Subscript	Algorithm
rsa	RSA
dsa	DSA
both	DSA and RSA

Consider the following case of a certificate chain with three CAs (CA^1, CA^2, and CA^3):

a) Entity E^1 subscribes to CA^1, and obtains the certificate:

$CA^1_{rsa}«E^1»$

b) which is computed using the RSA algorithm.

c) CA^2 has issued the following DSA certificate:

$CA^2_{dsa}«CA^1»$

d) CA^3 has issued the following DSA certificate:

$CA^3_{dsa}«CA^2»$

e) Entity E^2 has an authenticated copy of CA^3's public key.

f) Entity E_2 has the capability to validate RSA and DSA signatures and receives a signed transaction from E_1 and the certificates

$CA^1_{rsa}«E^1»$,

$CA^2_{dsa}«CA^1»$, and

$CA^3_{dsa}«CA^2»$

g) Note that $E^1{}_p$ may be either a DSA or an RSA public key.

h) E^2 validates the certificates to obtain an authenticated copy of E^1's public key by unwrapping the certification path as follows:

$CA^3{}_p \cdot CA^3{}_{dsa} \langle\langle CA^2 \rangle\rangle$, $CA^2{}_p \cdot CA^2{}_{dsa} \langle\langle CA^1 \rangle\rangle$, $CA^1{}_p \cdot CA^1{}_{rsa} \langle\langle E^1 \rangle\rangle$

i) To unwrap the last certificate, $CA^1{}_s \langle\langle E^1 \rangle\rangle$, E^2 validates an RSA signature.

Refer to ISO 15782-2 for additional information regarding algorithms and other parameters used to unwrap each certificate.

Annex E
(informative)

Certification Authority techniques for disaster recovery

E.1 General

The loss or compromise of a CA's private key is considered a disaster. All of the certificates signed with that private key are suspect. This includes all certificates issued for past, present and future use. Proper disaster recovery includes the revocation and re-issuance of all certificates that were signed with the CA's private key. Some techniques for reissuing certificates are

— notification with a CA's secondary key pair (CRL is signed with the secondary key),

— re-issuance of certificates with the CA's secondary key pair,

— re-issuance of certificates with the CA's new primary key pair, and

— notification with multiply signed certificates.

For each technique, the CA has two public key pairs: its primary key pair (P^1,S^1) and its secondary key pair (P^2,S^2). The following notation is used:

— **$P=>E_P$**: For a given certificate CA<<E>>, the public key E_P is validated using the CA's public key P. Hence, for CA<<E>>, P yields E_P.

— **$P^1=>E_P$**: E's public key is validated using the CA's primary public key.

— **$P^2=>E_P$**: E's public key is validated using the CA's secondary public key.

— **S^1<<E>>**: E's primary certificate is signed using the CA's primary private key.

— **S^2<<E>>**: E's secondary certificate is signed using the CA's secondary private key.

E.2 Notification with CA's secondary key pair

When a new end entity (E) submits its public key (E_P) to the CA, the CA generates a primary certificate CA^1<<E>> signed by the CA's primary private key, and a secondary certificate CA^2<<E>> signed by the CA's secondary private key. Before disaster recovery, Relying Parties have both certificates and both CA public keys, but only use the CA's primary key to validate the primary certificate.

Before disaster recovery: CA^1<<E>>, $P^1=>E_P$ and CA^2<<E>>, P^2

After disaster recovery: CA^1<<E>>, P^1 and CA^2<<E>>, $P^2=>E_P$

In the unlikely event that the CA's primary private key is compromised, disaster recovery consists of revoking all certificates signed with the primary private key and notifying the end entities to switch to the secondary certificate. After disaster recovery, each Subscriber validates the secondary end-entity certificates using the CA's secondary public key.

Note that this recovery technique is feasible only once.

A variation of this technique is used when certificates are distributed indirectly via a centralized service. In this case, the CA deletes all of the invalid primary certificates and notifies the end entities to use the existing secondary certificates which are still valid.

E.3 Re-issuance with CA's secondary key pair

When a new end entity (E) submits its public key (E_P) to the CA, the CA generates a certificate, CA^1<<E>> signed by the CA's primary private key. Before disaster recovery, Relying Parties have only the primary certificates and both CA public keys, but only use the CA's primary key to validate the primary certificate.

Before disaster recovery: CA^1<<E>>, $P^1 => E_P$ and P^2

After disaster recovery: CA^1<<E>>, P^1 and CA^2<<E>>, $P^2 => E_P$

In the unlikely event that the CA's primary private key is compromised, disaster recovery consists of revoking all primary certificates and reissuing secondary certificates. After disaster recovery, each validator validates the secondary end-entity certificates using the CA's secondary public key.

Note that this recovery technique is feasible only once.

A variation on this technique is used for central distribution services. In this case, the CA replaces all of the invalid primary certificates with the new valid secondary certificates and notifies the end entities to use the new certificates.

E.4 Re-issuance with CA's new primary key pair

When a new end entity (E) submits its public key (E_P) to the CA, the CA generates a certificate, CA^1<<E>> signed by the CA's primary private key. Before disaster recovery, Relying Parties have only the primary certificates and both CA public keys, but only use the CA's primary key to validate the primary certificate.

Before disaster recovery: CA^1<<E>>, $P^1 => E_P$ and P^2

During disaster recovery: CA^1<<E>>, P^1 and CA^2<<P^3>>, $P^2 => P^3$

After disaster recovery: CA^3<<E>>, $P^3 => E_P$ and CA^2<<P^3>>, $P^2 => P^3$

In the unlikely event that the CA's primary private key is compromised, disaster recovery initially consists of revoking all certificates signed with the primary private key. During disaster recovery, the CA generates a new primary public key pair (P^3,S^3) and issues a new certificate CA^2<<P_3>> of its new primary public key (P^3), signed by the CA's secondary private key (S^2). Each Subscriber validates the new CA certificate using the CA's secondary public key. Disaster recovery is completed by the CA reissuing certificates signed by the CA's new primary private key. After disaster recovery, each end entity validates the new end-entity certificates using the CA's new primary public key.

Note that this recovery technique is repeatable.

A variation on this technique is used when the certificates are distributed indirectly via a centralized service. In this case, the CA adds its own new certificate, replaces all of the invalid end-entity certificates with the new valid certificates, and notifies the end entities to validate the new CA certificate and begin using the new end-entity certificates.

E.5 Re-issuance with CA's next key pair

When a new end entity (E) submits its public key (E_P) to the CA, the CA generates a end-entity certificate, CA^1<<E>> signed by the CA's primary private key. The CA does not distribute its secondary public key, but a hash of it is contained in the self-signed certificate CA^1<<P^1, Hash(P^2)>> of its primary public key. Before disaster recovery, Relying Parties have only the end-entity primary certificates and the CA primary certificate, and use the CA's primary key to validate the end-entity primary certificate.

Before disaster recovery: CA^1<<E>>, P^1=>E_P and CA^1<<P1, Hash(P2)>>

During disaster recovery: CA^1<<E>>, P^1 and (P^1, Hash(P^2)) => P^2

After disaster recovery: CA^2<<E>>, P^2=>E_p and CA^2<<P^2, Hash(P^3)>>

In the unlikely event that the CA's primary private key is compromised, disaster recovery consists of revoking all primary certificates, and by distributing the secondary CA public key. The CA's secondary public key is distributed in a self-signed certificate CA^2<<P^2, Hash(P^3)>> which also contains the hash of their next public key — the tertiary CA public key. The Subscriber validates the new CA certificate by verifying the self-signed signature and by checking that the hash of the public key P2 is indeed equal to the hash contained in the CA primary certificate CA^1<<P^1, Hash(P^2)>>.

Disaster recovery is completed by the CA reissuing secondary end-entity certificates. After disaster recovery, each end entity validates its new end-entity certificates using the CA's new public key.

Note that this recovery technique is repeatable and that end-entity certificates generated by “future” CA keys could, if necessary, be distributed before the future CA key becomes the “live” CA key.

Annex F
(informative)

Distribution of certificates and Certificate Revocation Lists

F.1 General

This annex discusses mechanisms for distributing certificates to entities other than the certificate subject, and for distributing Certificate Revocation Lists (CRLs). Note that the CRL distribution may be via electronic means or may use other media, such as CD-ROMs or printed lists.

F.2 Certificate distribution

If an entity has multiple certificates, it is desirable to indicate in the signature structure (or data being signed) which certificate should be used to validate the signature. For a signed message, the certificate serial number and issuer may be identified; in the case of a certificate that is signed by a CA with more than one certificate, the Relying Party uses the certificate whose **subjectUniqueID** field matches the **issuerUniqueID** field of the certificate being validated.

For transactions being signed, there are a variety of alternatives, including the following:

— to convey the certificate (or certification path) with the transaction;

— to convey an indication of the certificate required to validate the signature along with the transaction, for example,

  — the signer's name (and optionally the signer's unique ID), or

  — the issuer and serial number.

Certificates not conveyed with the transaction could be retrieved from an X.500 directory or similar server.

F.3 CRL distribution

CRLs may be distributed to other entities and Relying Parties (e.g. the entities certified by the CA, other CAs, or one or more centralized “CRL servers”). Distribution mechanisms include the following.

— A cryptographic service message containing the CRL.

— Inclusion in a message header, as in the case of certificates.

— Query by an entity for a CA's CRL, or a single entry. In this case, the response is signed by the revoking CA to prevent denial of service by spoofing a revocation of a valid certificate. Queries would be directed to an X.500 directory or similar well-known server. Processing the query at the CA violates the assumption that the CA is not a real-time database.

— Transmission of a single CRL entry (signed by the revoking CA). This mechanism would be used to provide “real-time” notice of revocation, while the entire CRL would be sent on a scheduled basis per the next update time. The standards do not, of course, preclude sending an entire, new (unscheduled) CRL when a certificate is revoked.

The reason code could be used to determine whether or not to distribute a revocation notice before the **nextUpdate** (see 6.3.8). Alternatively, multiple CRLs with varying update frequencies could be kept, based on the **reasonCode**. The definition of multiple CRLs is outside the scope of this part of ISO 15782.

Bibliography

[1] ISO/IEC 9594-1, *Information technology — Open Systems Interconnection — The Directory: Overview of concepts, models and services — Part 1*

[2] ISO/IEC 9594-2, *Information technology — Open Systems Interconnection — The Directory: Models — Part 2*

[3] ISO/IEC 9594-6, *Information technology — Open Systems Interconnection — The Directory: Selected attribute types — Part 6*

[4] ISO/IEC 9834-1, *Information technology — Open Systems Interconnection — Procedures for the operation of OSI Registration Authorities: General procedures and top arcs of the International Object Identifier tree — Part 1*

[5] ISO/IEC 10118-3, *Information technology — Security techniques — Hash-functions — Part 3: Dedicated hash-functions*

[6] ISO 13491-1, *Banking — Secure cryptographic devices (retail) — Part 1: Concepts, requirements and evaluation methods*

[7] ISO/IEC TR 14516, *Information technology — Security techniques — Guidelines for the use and management of Trusted Third Party services*

[8] ISO/IEC 15945, *Information technology — Security techniques — Specification of TTP services to support the application of digital signatures*

[9] ISO/IEC 18033-2, *Information technology — Security techniques — Encryption algorithms — Part 2: Asymmetric ciphers*

[10] ANS X9.30-1, *Public Key Cryptography Using Irreversible Algorithms for the Financial Services Industry, Part 1: The Digital Signature Algorithm (DSA)*

[11] ANS X9.31-1, *Public Key Cryptography Using Reversible Algorithms for the Financial Services Industry, Part 1: The RSA Signature Algorithm*

[12] ANS X9.55, *Public Key Cryptography For the Financial Services Industry: Extensions to Public Key Certificates and Certificate Revocation Lists*

[13] ANS X9.57, *Public Key Cryptography For the Financial Services Industry: Certificate Management*

[14] ANS X9.62, *Public Key Cryptography For the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)*

[15] ANS X9.79, *Public Key Infrastructure — Practices and Policy Framework*

[16] Federal Information Processing Standard 140-2, *Security Requirements for Cryptographic Modules*

[17] FRANKEL, Y. and DESMEDT, Y. *Parallel reliable threshold multisignature*, TR-92-04-02, U. of Wisconsin, Milwaukee, 1992

[18] FRANKEL, Y., GEMMEL, P., MACKENZIE, P. and YUNG, M. *Proactive RSA*, manuscript, 1996

[19] GENNARO, R., JARECKI, S., KRAWCZYK, H. and RABIN, T. *Robust Threshold DSS Signatures*, Proceedings of Eurocrypt '96, 1996

[20] GENNARO, R., JARECKI, S., KRAWCZYK, H. and RABIN, T. *Robust and Efficient Sharing of RSA Functions*, Proceedings of Crypto '96, 1996

[21] HERZBERG, A., JAKOBSSON, M., JARECKI, S., KRAWCZYK, H. and YUNG, M. *Proactive Public Key and Signature Systems*, 3rd ACM Conference on Computer and Communications Security, 1996

[22] MENEZES, A., VAN OORSCHOT, P. and VANSTONE, S. *Handbook of Applied Cryptography*, CRC Press, New York, 1996

[23] SHAMIR, A. *How to Share a Secret*, Communications of the ACM, November 1979

[24] TC 68 Web site: http://www.iso.org/tc68.

ICS 35.240.40

Price based on 49 pages