

CA-CA Interoperability
Interface Specification for experiment
Version 1.0

Changes

Version	Date	Comments	Detail
0.9	20030203	Ask for Comments	Draft Edition
1.0	-	Publication	First Edition
1.0.1	20030623	Change in CRLDP	
1.0.2	20030701	Modify backslash characters	

- Table of Contents -

1. Introduction.....	1
2. References	2
3. Symbols and Abbreviations	3
4. CA-CA Model.....	4
4.1 Overview.....	4
4.1.1 Cross Certification (CC).....	4
4.1.2 Cross Recognition (CR)	5
4.2 IWG Architecture.....	5
4.3 Policy Structure Consideration	7
5. Interface of PKI Components.....	8
5.1 PKI Components	8
6. Profile	10
6.1 Policy of Designing Certificate/CRL Profiles.....	10
6.2 CA Certificate Profile	10
6.2.1 ROOT CA Certificate Profile.....	11
6.2.2 CC Certificate	13
6.2.3 SubCA Certificate	15
6.3 EE Certificate Profile	17
6.3.1 Common EE Profile	17
6.3.2 Identification Certificate (digital signature)	18
6.3.3 Secure E-Mail Certificate (data Encipherment and digital signature).....	18
6.4 ARL/CRL Profile	19
6.4.1 Common ARL/CRL Profile	19
6.5 Interoperability consideration (Certificate & CRL)	20
6.5.1 Encoding rules of DirectoryName.....	20
6.5.2 The escape method to describe the LDAPURI in case that "comma character" is included in RDN value (e.g. value of cRLDP.distname.fullname etc.)	20
6.5.3 Value of cRLDistributionPoints and issuingDistributionPoints	21
APPENDIX OCSP responder	24
6.6 Repository Profile	25
6.6.1 DIT.....	25
6.6.2 Schema (objectclass, attribute).....	25
6.6.3 Interoperability Considerations.....	29

- List of Figures –

Fig. 1 Cross Certification (Mutual CC, Unilateral CC) 4

Fig. 2 Cross Certification (IWG Architecture) 6

Fig. 3 Cross Recognition (IWG Architecture) 6

Fig. 4 PKI Components 8

Fig. 5 sample DIT Tree (3 parties) in one directory 25

- List of Tables -

Table. 1 CA-CA interface 9

Table. 2 CA-EE interface 9

Table. 3 End Entity-Repository interface and VA-Repository interface 9

Table. 4 End Entity-VA interface 9

Table. 5 End Entity-End Entity interface..... 9

- Trademarks, Registered Trademarks -

Microsoft® is a registered trademark of Microsoft Corp. in the U.S. and other countries.

Windows® is a registered trademark of Microsoft Corp. in the U.S. and other countries.

All other trademarks and product names are property of their respective owners.

1. Introduction

This specification describes a minimum set of technical interfaces to the JKST (Japan / Korea / Singapore / Chinese Taipei) PKI Model needed by the Certification Authority (CA), application developers, and the end-entity users to interconnect one another.

The specification includes the certificate and CRL profile, and directory profile for multiple PKI domains' interoperability, with greater harmony with the Internet Engineering Task Force (IETF) Public Key Infrastructure, ITU-T Recommendation, and other standard documents. The specification establishes a profile that is a largely subset of the PKI profile in IETF in order to help maintain the interoperability in multiple PKI domain environments. All of the other technical details are also referred from the documents published by standardization organizations.

The specification is designed to facilitate the interoperability of multiple PKI domains when the domains interconnect one another. This specification is still generic in a sense that potential PKI designers still can customize this specification for their needs. However, in order to make interoperable environments in multiple PKI domains, this specification suggests the interfaces of PKI components and recommended profile.

It is important that the JKST PKI model be compatible as much as possible with PKI efforts established in other activities conducted and hope that the JKST PKI model could be used as a reference model for PKI initiatives in Asia regions.

A final report in 2001 experiment can be found at the PKI-J web page:

<http://www.japanpkiforum.jp/E/index.htm>

2. References

[x500]	ITU-T Recommendation X.500 – Information technology – Open Systems Interconnection – The Directory: Overview of concepts, models and services, 2001
[x501]	ITU-T Recommendation X.501 – Information technology – Open Systems Interconnection – The Directory: Models, 2001
[x509]	ITU-T Recommendation X.509 – Information technology –Open Systems Interconnection – The Directory: Authentication Framework, 1997
[x520]	ITU-T Recommendation X.520 – Information technology – Open Systems Interconnection – The Directory: Selected attribute types, 2001
[x521]	ITU-T Recommendation X.521 – Information technology – Open Systems Interconnection – The Directory: Selected object classes, 2001
[x690]	ITU-T Recommendation X.690 – Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER), 1998
[2251]	Lightweight Directory Access Protocol (v3) Internet Request For Comments 2251 December 1997
[2252]	Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions. Internet Request For Comments 2252 December 1997
[2253]	Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names Internet Request For Comments 2253 December 1997
[2254]	The String Representation of LDAP Internet Request For Comments 2254 December 1997
[2255]	The LDAP URL Format Internet Request For Comments 2255 December 1997
[2256]	A Summary of the X.500 (96) User Schema for use with LDAPv3 Internet Request For Comments 2256 December 1997
[2279]	UTF-8, a transformation format of ISO 10646 Internet Request For Comments 2279 January 1998
[2396]	Uniform Resource Identifiers (URI): Generic Syntax Internet Request For Comments 2396 August 1998
[2459]	Internet X.509 Public Key Infrastructure Certificate and CRL Profile Internet Request For Comments 2459 January 1999
[2559]	Internet X.509 Public Key Infrastructure Operational Protocols - LDAPv2 Internet Request For Comments 2559 April 1999
[2560]	X.509 Internet Public Key Infrastructure Online Certificate Status Protocol – OCSP Schema Internet Request For Comments 2560 June 1999.
[2587]	Internet X.509 Public Key Infrastructure LDAPv2 Schema Internet Request For Comments 2587 June 1999
[3280]	Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile Internet Request For Comments 3280 April 2002
[p10]	PKCS 10: Certification Request Syntax Version 1.0, 1993
[p12]	PKCS 12 v1.0: Personal Information Exchange Syntax, 1999

3. Symbols and Abbreviations

ARL	Authority Revocation List
ASN.1	Abstract Syntax Notation One
B2B	Business to Business
BER	Basic Encoding Rules
CA	Certification Authority
CRL	Certificate Revocation List
CC	Cross Certification
DAP	Directory Access Protocol
DER	Distinguished Encoding Rules
DIT	Directory Information Tree
DN	Distinguished Name
EE	End entity
LDAP	Lightweight Directory Access Protocol
OCSP	Online Certificate Status Protocol
OID	Object Identifier
PKCS	Public Key Cryptograph Standard
RDN	Relative Distinguished Name
RA	Registration Authority
SCA	Subordinate CA
VA	Validation Authority

4. CA-CA Model

The PKI technology develops several CA-CA models in which the relying party (RP) can trust the information and digital certificates signed by other parties in multiple domains PKI. It is unlikely that end-entity transactions can be accomplished with the PKI applications without considering PKI CA-CA model. In Asian countries, the Cross Certification model and its variants are often used in the government-initiated PKI. The Cross Recognition model and its variants are used in the commercial PKI, including a Trusted-List solution in the Web browsers. Some countries do not employ the Cross Certification model at all.

After evaluating several possibilities, the JKST identifies two fundamental models that interconnect one another: Cross Certification and Cross Recognition. The JSKT does not preclude other PKI models such as Bridge model. The Cross Certification and Cross Recognition are at least considered as common denominators used for the multiple domains PKI.

The JKST also utilizes a hybrid model between the Cross Certification and Cross Recognition – trust point management in RP’s perspective, in which CA allows its EE to switch its trust point for the certificate path processing based on the relationship with the other domain. This model envisions that the Cross Certification and Cross Recognition could coexist.

4.1 Overview

This section describes brief summary of Cross Certification and Cross Recognition.

4.1.1 Cross Certification (CC)

The concept of Cross Certification is that a CA publishes a certificate to another CA. There are two kinds of Cross Certification. One is “Mutual Cross Certification”. The other is “Unilateral Cross Certification”. These are described below.

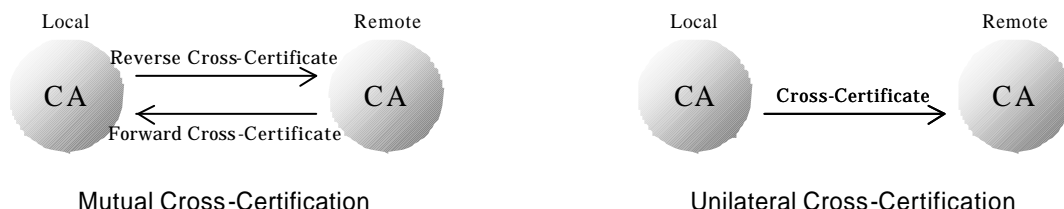


Fig. 1 Cross Certification (Mutual CC, Unilateral CC)

Mutual Cross Certification is the case where one CA publishes a certificate to the

other, and vice versa. The relationship of “Cross Certification” is shown at the left of the Fig. 1. Unilateral Cross Certification is the case where one CA publishes a certificate only to a remote CA. The model “Unilateral Cross Certification” is used when adopting a hybrid model and when a CA publishes a certificate to a subordinate CA.

In multiple PKI domains environment, especially in international context, it is more suitable for each party to use the Mutual CC model when the Cross Certification model is employed.

4.1.2 Cross Recognition (CR)

Cross Recognition is a concept considered by APEC TEL WG, and is defined as follows:

*An interoperability arrangement in which a relying party in one PKI domain can use authority information in another PKI domain to authenticate a subject in the other PKI domain, and vice-versa.*¹

An example of application that utilizes the concept of Cross Recognition is “Web browser model”. Web browser has a lot of certificates as a trusted list. An example of the method to establish Cross Recognition is that a relying party stores the trust anchor certificates in the application, decides whether to accept the sender’s certificate or not, and validates the certificate. The Cross Recognition covers an important concept of the acceptance framework on how the relying party can decide to accept the trust anchor certificate of the other parties. However, this is out of scope in this document.

4.2 IWG Architecture

IWG Architecture is constituted on the basis of the following basic concepts:

- For Cross Certification, Root CAs publish cross certificates one another. No subordinate CA’s cross certification is considered yet.
- For Cross Recognition, an RP domain can use the Root CA’s information on the other domain based on the RP Perspective to authenticate a subject (Presumably some information is given in advance to evaluate the trust relationship.).
- The Root CAs have zero or more subordinate CAs.

The IWG Architecture becomes the following models.

¹ ACHIEVING PKI INTEROPERABILITY (<http://www.apectelwg.org/apecdata/telwg/eaTG/eatf06.doc>)

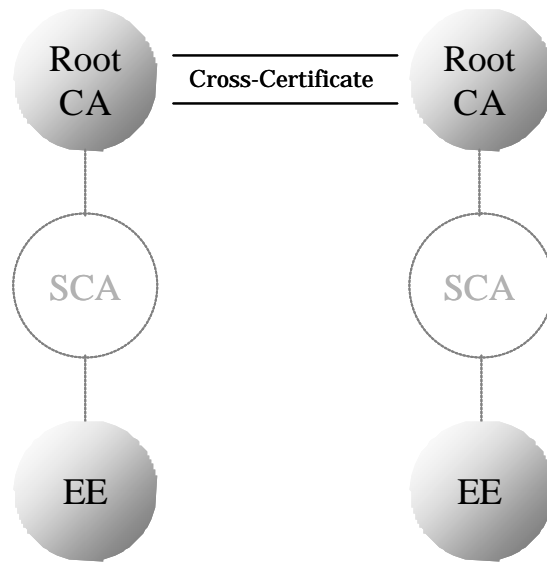


Fig. 2 Cross Certification (IWG Architecture)

In Fig.2, both Root CAs issue the cross certificate and exchange them in order for the RP to process the certificate path across the domains.

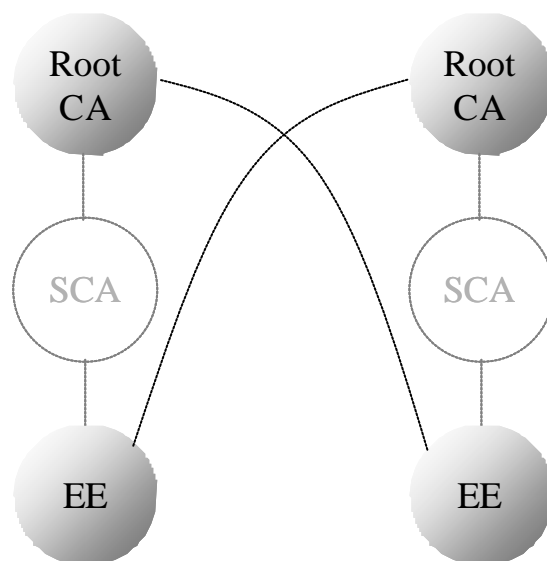


Fig. 3 Cross Recognition (IWG Architecture)

In Fig. 3 an EE uses the Root CA's information of the other domain to authenticate a subject. Note this architecture assumes that some acceptance framework has been established between the two domains.

Fig. 4 shows a hybrid model between Cross Certification and Cross Recognition.

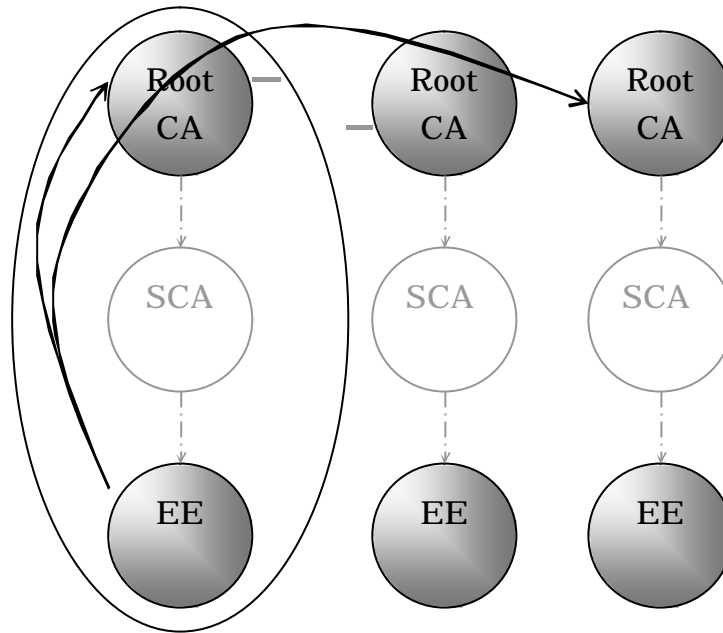


Fig 4. Hybrid Model (IWG Architecture)

One domain manages trust points with the parties involved, and EE switches it's the trust point based on the CA-CA relationship with others domains. Fig 4 indicates that this model accepts both of the cross certification and cross recognition, and the EE can change the trust point in order to validate a certificate in question. Note that this architecture should be carefully articulated when several additional parameters (such as constraints) are set in the cross certificates for an instance.

4.3 Policy Structure Consideration

In the multiple PKI domains environment, it is desirable to establish, evaluate, agree upon, and check the certificate policy in consistent manners. IWG architecture assumes that the multiple domains establish certificate policies and agree upon between the CAs, and/or evaluated by some trusted audited organization. It is highly expected that a valid policy path should be constructed in the certificate path and validated in the relying party applications. Even the CR model should provide some mechanisms to notify the information of certificate policies to help the EE decision.

5. Interface of PKI Components

5.1 PKI Components

The following figure shows the PKI components in the IWG architecture. There is a minimum set of the PKI components interfaces to be agreed upon between involved parties. The internal CA-RA-EE interfaces are not important for the multiple domains PKI environment. Rather, the CA-CA interface and the EE-Repository interface are important and have to be agreed. The solid line is the interface to other domains, and the broken line is out of scope.

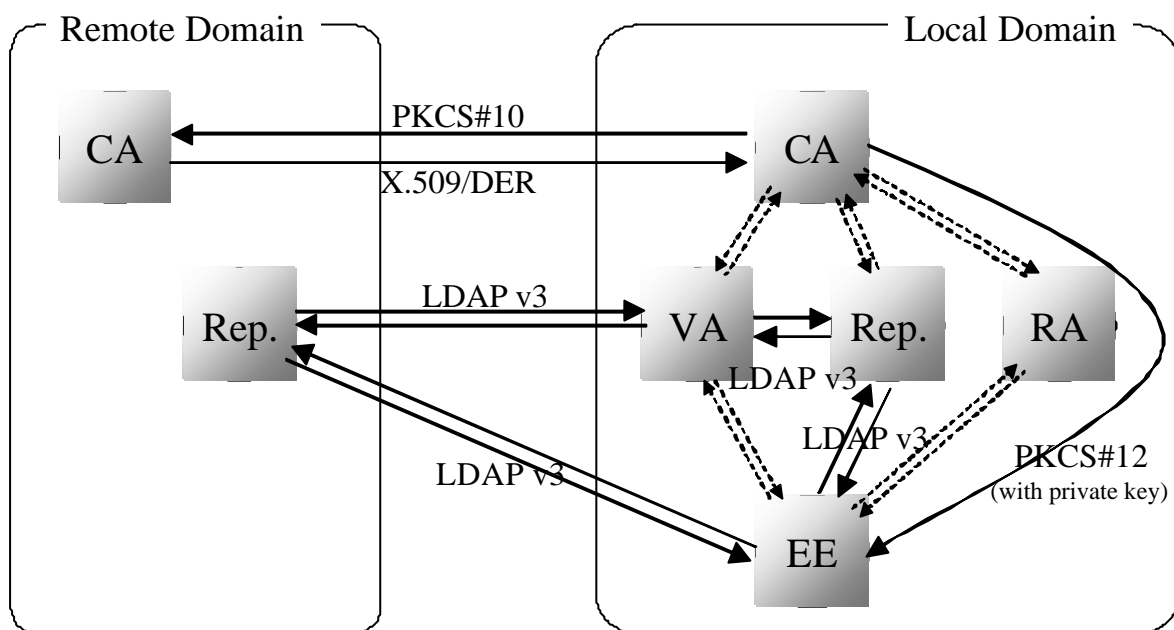


Fig. 3 PKI Components

Here is the summary of the PKI components interfaces that be agreed. For the certificate profile, the detail will be described later.

Content	Interface
Certificate profile	X.509(97) v3[x509], RFC3280[3280]
Certificate encoding format	DER[x690]
CRL profile	X.509(97) v3, RFC3280
CRL encoding format	DER
Cross-Cert request format	PKCS#10[p10] PKCS#10 version1.0 (No extension fields must be included in the cross cert request file.)
Cross-Cert response format.	X.509 in DER
The method to sends the fingerprint.	E-Mail
POP (proof of possession)	Verification of digital signature on certificate request format

Table. 1 CA-CA interface

Content	Interface
EE Certificate response format	PKCS#12[p12] (Private-key included)

Table. 2 CA-EE interface

Content	Interface
Repository access protocol (e.g., LDAPv2, LDAPv3, DAP)	LDAPv3[2251]

Table. 3 End Entity-Repository interface and VA-Repository interface

Content	Interface
EE-VA access protocol	OPTIONAL
Role of VA	Certificate Validation Server (Path Construction, Path Validation)

Table. 4 End Entity-VA interface

Content	Interface
Certificate path validation method	RFC3280
Certificate validation entity	VA, EE

Table. 5 End Entity-End Entity interface

6. Profile

The certificate and CRL profile is based on the X.509 and RFC 3280 standards. The RFC 3280 provides the information on the details of the data fields and format and the guidance on the choices of the fields, and the values in each field. The JKST employs a profile that is a great harmony with the standards and that is more specific to the choice of the data values and fields to maintain the interoperability in multiple PKI domains. The profile contains the basic and extension fields. The basic fields are needed to set the value in mandatory fashion. An extension can be non-critical or critical. If an extension is critical and an application does not recognize or cannot process that extension, the application must reject any transaction. The handling of the criticality follows the RFC 3280.

6.1 Policy of Designing Certificate and CRL Profile

- Certificate/CRL profile is based on RFC 3280 and X.509 (97).
- The profile is primarily designed for the digital signature usage for document exchange applications and for the secure email usage. However this is not designed for a specific business transaction. Rather the profile should be considered common minimum sets of technical agreements to achieve the interoperability in the IWG architecture.
- The profile is also designed to be applicable to both the web browser application and standalone system. The flexibility and expendability should be maintained such as SSL and Code Signing using this profile.
- Local encryption algorithm and private extensions of each country are not used.
- The character set in Certificate/CRL must be within the range of PrintableString. (Multi-byte code is out of scope in this experiment.)
- xxxConstraint extensions MAY be used in the test environments. However in the real usage, complex xxxConstraint extensions are not recommended.
- Some parts are based on the present implementation and the limitations of the application such as Microsoft® Windows® operating systems and etc.

6.2 CA Certificate Profile

There are four types of the CA certificates; Root CA certificate, Self-issued, Subordinate CA certificate, and Cross certificate. The CA product is at least expected to

produce the Root CA certificate and Subordinate CA certificate in the IWG architecture. For Cross Certification, the CA product must produce the Cross Certificate. Currently the Self-issued certificates for key rollover are not defined.

6.2.1 ROOT CA Certificate Profile

The ROOT CA's self-signed certificate is used for signing other CA certificates, Self-issued certificate, Cross certificate, and Subordinate CA certificate. The ROOT CA certificate will be used to provide the public key of the trust anchor and the initial information of the certificate path processing.

Certificate Basic field

FIELD	NOTE
version (Mandatory)	Since several extension fields appear in this profile, the value MUST be set to 2 (v3).
serialNumber (Mandatory)	Unique integer. Up to 20 octets.
Signature (Mandatory)	1.2.840.113549.1.1.5 (sha1WithRSAEncryption)
issuer (Mandatory)	X.500 DN. DN is encoded by UTF8STRING, according to description of the X.520 (2001) except for the Country attribute encoded by PrintableString. This profile strongly recommends using the UTF8STRING as default encoding. However the PrintableString is acceptable and still valid to maintain the backward compatibility with legacy and web browser systems.
Validity (Mandatory)	UTC TIME
subject (Mandatory)	X.500 DN. And see issuer.
subjectPublicKeyInfo (Mandatory)	1.2.840.113549.1.1.1 (rsaEncryption) CA: 2,048bit
issuerUniqueID (not used)	
subjectUniqueID (not used)	

Certificate Extension field

FIELD	NOTE
authorityKeyIdentifier (optional, non-critical)	KeyId(Mandatory): The hash value of Issuer's public key (SHA1 160bit). The 1 st calculation method in RFC3280 ch.4.2.1.2. authorityCertIssuer(optional): DN authCertSerialNum(optional): INTEGER When AuthCertIssuer is used, AuthCertSerialNum must be set as well. Vice versa.
subjectKeyIdentifier	The hash value of Issuer's pubic key (SHA1 160bit).

(Mandatory, non-critical)	The 1st calculation method in RFC3280 ch.4.2.1.2
keyUsage (optional, critical)	When used, keyCertSign and cRLSign should be included at least.
extKeyUsage (not used)	
privateKeyUsagePeriod (not used)	
certificatePolicies (optional, critical)	When used, policyID MUST be present.
policyMappings (not used)	
subjectAltName (optional, non-critical)	If the PKI domain wants to include email address or etc in the certificate, this field will be used.
issureAltName (optional, non-critical)	If the PKI domain wants to include email address or etc in the certificate, this field will be used.
subjectDirectryAttributes (not used)	
basicConstraints (Mandatory, critical)	cA=TRUE pathLen=optional (INTEGER)
nameConstraints (not used)	
policyConstraints (not used)	
cRLDistributionPoints (optional, non-critical)	When used, distributionPoint.fullName must be used. The Generalname must be either directoryName or URI. For LDAP URI, the format MUST be ldap://hostname[:portnumber]/dn?attribute[;binary] The portnumber part is optional; the attribute part should be mandatory and the binary option is optional.
authorityInfoAccess (optional, non-critical)	If the PKI domain uses OCSP, this field will be used.
inhibitAnyPolicy (not used)	
freshestCRL (not used)	
subjectInfoAccessSyntax (not used)	

6.2.2 CC Certificate

The CC certificate is a certificate, issued by the issuer domain to the subject domain. The CC certificate represents the subject domain policy is equivalent to the issuer domain policy. It is allowed to use constraint-related extensions such as basic constraints, policy constraints, and name constraints in this certificate. However, extreme cautions must be required in order to design such extensions in multiple domains PKI environment.

Since the cross certificate request file does not contain any extension fields in the data, the cross certificate-issuing CA must produce the extensions specified in the profile. The OID information in the policyMappings extension have to be exchanged beforehand via out of band.

Certificate Basic field

the same as ROOT CA Certificate

Certificate Extension field

Regarding the certificatePolicies, the critical-flag can be set as “non-critical”, considering the implementation of the present application (e.g. Microsoft® Windows® 2000 operating system or earlier etc). However, it is necessary to check the policy in the path processing.

FIELD	NOTE
authorityKeyIdentifier (Mandatory, non-critical)	keyId(Mandatory): The hash value of Issuer's public key (SHA1 160bit). The 1st calculation method in RFC3280 ch.4.2.1.2 authorityCertIssuer(optional): DN authCertSerialNum(optional): INTEGER When AuthCertIssuer is used, AuthCertSerialNum must be set as well. Vice versa.
subjectKeyIdentifier (Mandatory, non-critical)	The hash value of Issuer's public key (SHA1 160bit). The 1st calculation method in RFC3280 ch.4.2.1.2
keyUsage (Mandatory, critical)	keyCertSign, cRLSign
extKeyUsage (not used)	
privateKeyUsagePeriod (not used)	
certificatePolicies (Mandatory, either critical or non-critical ²)	policyID MUST be present.
policyMappings (Mandatory, non-critical)	issuerDomain and subjectDomain OIDs MUST be present. OID must be exchanged beforehand via out

² It must be verified of a policy by the case of non-critical as well as the case of critical.

	of band
subjectAltName (optional, non-critical)	If the PKI domain wants to include email address or etc in the certificate, this field will be used.
issureAltName (optional, non-critical)	If the PKI domain wants to include email address or etc in the certificate, this field will be used.
subjectDirectryAttributes (not used)	
basicConstraints (Mandatory, critical)	cA=TRUE pathLen=optional (INTEGER)
nameConstraints (optional, critical)	
policyConstraints (optional, critical)	If the PKI domain wants to strictly validate of certificate policies, this field will be set as requireExplicitPolicy=0.
cRLDistributionPoints (Mandatory, non-critical)	<p>“distributionPoint.fullname” must be used and contain either directoryName or URI.</p> <p>For LDAP URI, the format MUST be ldap://hostname[:portnumber]/dn?attribute[;binary]</p> <p>The portnumber part is optional; the attribute part should be mandatory and the binary option is optional.</p>
authorityInfoAccess (not used)	If the PKI domain uses OCSP, this field will be used.
inhibitAnyPolicy (not used)	
freshestCRL (not used)	
subjectInfoAccessSyntax (not used)	

6.2.3 Subordinate CA Certificate

The Subordinate CA certificate is a certificate, issued by the upper CA, typically the ROOT CA. This CA is the entity that issues certificates to the End Entity. There is no restriction on the number of the subordinate CA in the hierarchy. However proper managed CA hierarchy is expected.

Certificate Basic field

The same as ROOT CA Certificate

Certificate Extension field

Regarding the certificatePolicies, the critical-flag can be set as “non-critical”, considering the implementation of the present application (e.g. Microsoft® Windows® 2000 operating systems or earlier etc). However, it is necessary to check the policy in the path process.

FIELD	NOTE
authorityKeyIdentifier (Mandatory, non-critical)	keyId(Mandatory): The hash value of Issuer's public key (SHA1 160bit). The 1st calculation method in RFC3280 ch.4.2.1.2 authorityCertIssuer(optional): DN authCertSerialNum(optional): INTEGER When AuthCertIssuer is used, AuthCertSerialNum must be set as well. Vice versa.
subjectKeyIdentifier (Mandatory, non-critical)	The hash value of Issuer's public key (SHA1 160bit). The 1st calculation method in RFC3280 ch.4.2.1.2
keyUsage (Mandatory, critical)	keyCertSign, cRLSign
extKeyUsage (not used)	
privateKeyUsagePeriod (not used)	
certificatePolicies (Mandatory, either critical or non-critical ³)	policyID MUST be present.
policyMappings (not used)	
subjectAltName (optional, non-critical)	If the PKI domain wants to include email address or etc in the certificate, this field will be used.
issuerAltName (optional, non-critical)	If the PKI domain wants to include email address or etc in the certificate, this field will be used.
subjectDirectoryAttributes (not used)	
basicConstraints (Mandatory, critical)	cA=TRUE pathLen=optional (INTEGER)
nameConstraints	

³ It must be verified of a policy by the case of non-critical as well as the case of critical.

(optional, critical)	
policyConstraints (not used)	
cRLDistributionPoints (Mandatory, non-critical)	<p>“distributionPoint.fullname” must be used and contain either directoryName or URI.</p> <p>For LDAP URI, the format MUST be <code>ldap://hostname[:portnumber]/dn?attribute[:binary]</code></p> <p>The portnumber part is optional; the attribute part should be mandatory and the binary option is optional.</p>
authorityInfoAccess (not used)	If the PKI domain uses OCSP, this field will be used.
inhibitAnyPolicy (not used)	
freshestCRL (not used)	
subjectInfoAccessSyntax (not used)	

6.3 EE Certificate Profile

The EE Certificate is used by individual or the electric ID to identify the entity for certain transactions. The issuer and subject name in the certificate is the DN for a corresponding entry in the directory.

The common fields of the EE Certificate are specified in “6.3.1”. The following sections, “6.3.2” and “6.3.3” specify the differences from “6.3.1” for individual applications.

6.3.1 Common EE Profile

Certificate Basic field

The same as ROOT CA Certificate. (The subject DN may be subject to change according to the usage of certificate.)

Certificate Extension field

Regarding the certificatePolicies, critical-flag can be set to non-critical in consideration of the present application implementation (e.g. windows2000 or earlier etc). However, it is necessary to validate of a policy also the same as the case of critical.

FIELD	NOTE
authorityKeyIdentifier (Mandatory, non-critical)	keyId(Mandatory): The hash value of Issuer's public key (SHA1 160bit). The 1st calculation method in RFC3280 ch.4.2.1.2 authorityCertIssuer(optional): DN authCertSerialNum(optional): INTEGER When AuthCertIssuer is used, AuthCertSerialNum must be set as well. Vice versa.
subjectKeyIdentifier (Mandatory, non-critical)	The hash value of Issuer's public key (SHA1 160bit). The 1st calculation method in RFC3280 ch.4.2.1.2
keyUsage (Mandatory, critical)	Please see 6.3.2 and 6.3.3 about a value.
extKeyUsage (not defined)	
privateKeyUsagePeriod (not used)	
certificatePolicies (Mandatory, ether critical or non-critical ⁴)	policyID MUST be present.
policyMappings (not used)	
subjectAltName (optional, non-critical)	If the PKI domain wants to include email address or etc in the certificate, this field will be used. And see 6.3.3.
issureAltName (optional, non-critical)	If the PKI domain wants to include email address or etc in the certificate, this field will be used.

⁴ It must be verified of a policy by the case of non-critical as well as the case of critical.

subjectDirectoryAttributes (not used)	
basicConstraints (optional, critical)	It recommends that CAs don't include a this field.
nameConstraints (not used)	
policyConstraints (not used)	
cRLDistributionPoints (Mandatory, non-critical)	<p>"distributionPoint.fullname" must be used and contain either directoryName or URI.</p> <p>For LDAP URI, the format MUST be ldap://hostname[:portnumber]/dn?attribute[;binary]</p> <p>The portnumber part is optional; the attribute part should be mandatory and the binary option is optional.</p>
authorityInfoAccess (not used)	If the PKI domain uses OCSP, this field will be used.
inhibitAnyPolicy (not used)	
freshesCRL (not used)	
subjectInfoAccessSyntax (not used)	

6.3.2 Identification Certificate (digital signature)

Certificate Extension field

keyUsage (Mandatory, critical)	digitalSignature (, nonRepudiation)
--------------------------------	-------------------------------------

6.3.3 Secure E-Mail Certificate (data Encipherment and digital signature)

Certificate Extension field

keyUsage (Mandatory, critical)	keyEncipherment, dataEncipherment
subjectAltName (Mandatory, non-critical)	If the PKI domain wants to include multi byte code or email address or etc in the certificate, this field will be used.

6.4 ARL/CRL Profile

Authority Revocation List (ARL) and Certificate Revocation List (CRL) are used to check whether a certificate in the certification path has not been revoked or not. This profile distinguishes the ARL and CRL. In addition, this profile accepts the partitioned CRL distribution policy based on the revocation reasons and serial number, for instance. This design policy suggests that the CA can customize their revocation policy. The customization is up to the decision of the CA issuing policy. The application is expected to handle the revocation policy of the CA.

6.4.1 Common ARL/CRL Profile

ARL/CRL Basic field

FIELD	NOTE
Version (Mandatory)	Since extension field appears in this profile, the value MUST be set to 1 (v2).
signature (Mandatory)	1.2.840.113549.1.1.5 (sha1WithRSAEncryption)
issue (Mandatory)	X.500 DN. Although DN is generally encoded by UTF8STRING, according to description of the X.520(2001), Country attribute is encoded by PrintableString.
thisUpdate (Mandatory)	UTCTIME
nextUpdate (Mandatory)	UTCTIME
revokedCertificates (Mandatory)	

ARL/CRL EntryExtensions

FIELD	NOTE
ReasonCode (Mandatory, non-critical)	See the RFC 3280
holdInstructionCode (not used)	
invalidityDate (optional, non-critical)	GeneralizedTime
CertificateIssuer (not used)	

ARL/CRL Extensions

FIELD	NOTE
authorityKeyIdentifier (Mandatory, non-critical)	keyId(Mandatory): The hash value of Issuer's public key (SHA1 160bit). The 1st calculation method in RFC3280 ch.4.2.1.2 authorityCertIssuer(optional): DN authCertSerialNum(optional): INTEGER When AuthCertIssuer is used,

	AuthCertSerialNum must be set as well. Vice versa.
issureAltName (not-used)	
cRLNumber (Mandatory, non-critical)	unique integer. up to 20 octets.
deltaCRLIndicator (optional, critical)	If the PKI domain wants to use dCRL, this field will be used.
issuingDistributionPoint	Please see 6.5.3 about this value. When the distributionPoint.fullName is used, the value must be the exact same as the one in the CRLDP. distributionPoint.fullName in certificate to do the comparison.
freshestCRL (optional, non-critical)	If the PKI domain wants to use dCRL, this field will be used.
crlScope (not-used)	

6.5 Interoperability consideration (Certificate & CRL)

6.5.1 Encoding rules of DirectoryName

Although DN is generally encoded by UTF8STRING, according to description of the X.520(2001), Country attribute is encoded by PrintableString. In addition, this profile accepts the PrintableString for whole DN encoding, the DN matching rule of X.500 series described in RFC 3280 is expected.

6.5.2 The escape method to describe the LDAPURI in case that "comma character" is included in RDN value (e.g. value of cRLDP.distname.fullname etc.)

Since "comma character (,)" is used as a delimiter of RDN in DN, the extreme cautions are needed when the comma character is included in RDN value. (Of course, there are other important characters that must take care about as well.)

In order to change the DN into the URI, it is necessary to make the DN "string representation" first using the method described by RFC2253 and then change it to the LDAPURI format.

Since "comma character" is used as a delimiter character at this time, it is necessary to be escaped. Four kinds of methods exist. For example, assume "country name=AA, organization name=ABC Co., Ltd.", it is as follows.

1. o=ABC Co.\2C Ltd.,c=AA
2. o=ABC Co.\2c Ltd.,c=AA
3. o=ABC Co.\, Ltd.,c=AA
4. o="ABC Co., Ltd.",c=AA (discouraged, since not V3 compliant)

And it is as follows when above four are URI.

- 1'. ldap://example.tld/o=ABC%20Co.%5C2C%20Ltd.,c=AA
- 2'. ldap://example.tld/o=ABC%20Co.%5C2c%20Ltd.,c=AA
- 3'. ldap://example.tld/o=ABC%20Co.%5C,%20Ltd.,c=AA
- 4'. ldap://example.tld/o=%22ABC%20Co.,%20Ltd%22,c=AA (discouraged, since not V3 compliant)

The special character including "comma character" can be used by being escaped escaping as mentioned above.

6.5.3 Value of `issuingDistributionPoints`

The value of `issuingDistributionPoints` changes according to the CRL publication policy. This profile allows CA to have the partitioned CRL distribution policy. There are four types of publication policies of CRL that considered⁵.

- 1) CA publishes one full CRL
- 2) CA publishes partitioned CRLs only
- 3) CA publishes one complete CRL and one complete ARL
- 4) CA publishes partitioned CRLs, and one complete ARL or partitioned ARLs

This profile defines the following three terms to avoid the confusion on the CRL distribution terms.

1. full CRL is a CRL that lists all revoked certificate including the all EE and CA certificates.
2. complete CRL(ARL) is a CRL that lists all revoked certificates within two given scopes. One is the set of the certificates covered by the CRL that contains all the EE certificates only. The other is the set of the certificates covered by the CRL that contains all the CA certificates only.
3. partitioned CRL is a partition of a full CRL or complete CRL(ARL), partitioned with some kinds of the criteria such as the range of the certificate serial number or some other ad hoc range. The criteria depend on the CA policy. The CA makes sure that the union of the full set of the partitioned CRL should be equivalent to a full CRL. This profile assumes that the partitioned CRL must be

⁵ The publication policies of delta or indirect CRL are out of scope. If your domain employs such CRL policies, you should make sure that your CRL policy could be interoperable with the other four CRL publication policies. In addition, only `CompleteReasons` fields are not used.

published at the locations of the
cRLDistributionPoint.DistributionPoint.fullName and
issuingDistributionPoint.distributionPoint.fullName fields.

The CRL publication policies of CA are expected to comply the following assumptions:

- 1) A CRL without the issuingDistributionPoint(iDP) extension is expected to cover all the revocation information of all unexpired certificates, including both CA and EE certificates, all issued by the CRL issuer (assuming the certificate-issuing CA). The CRL must be a full CRL.
- 2) A CRL with the iDP extension with the onlyContainsUserCerts field is expected to cover all the revocation information of all unexpired certificates including EE certificate, all issued by the CRL issuer (assuming the certificate-issuing CA). A CRL with the iDP extension with the onlyContainsCACerts field is expected to cover all the revocation information of all unexpired certificates including CA certificate, all issued by the CRL issuer (assuming the certificate-issuing CA). The CRLs must be a complete CRL and complete ARL.
- 3) A CRL with the iDP extension with distributionPoint.fullName field is expected to be a partitioned CRL, (or a full CRL and complete CRL/ARL published at the location of the distributionPoint.fullName). It is important that the RP must make sure that one of the names in the distribution point fields in the CRL distribution point extension (cRLDP) must match one of the names in the distribution point field in iDP to prevent the CRL from the substitution attack. The CA should make one of the names in the distribution point field of the certificate's cRLDP EXACTLY the same as one of the names in the distribution point fields in the CRL's iDP. It is also important that this is the CA's responsibility to issue valid partitioned CRLs with a given scope.

The values of issuingDistributionPoints are specified based on the CRL publication policies above.

CA publishes only one full CRL (no ARL)

iDP -- Optional (critical/non-critical)

distributionPoint -- Optional

fullName – Optional (EXACTLY the same value as one of the names in the
cRLDP in the certificate)
nameRelativeToCRLIssuer -- not defined
onlyContainsUserCerts -- forbidden to use
onlyContainsCACerts -- forbidden to use
onlySomeReasons -- forbidden to use
indirectCRL -- not defined

CA publishes partitioned CRLs (no ARL)

iDP -- Mandatory (critical)
distributionPoint -- Mandatory
fullName – Mandatory (EXACTLY the same value as one of the names in the
cRLDP in the certificate)
nameRelativeToCRLIssuer -- not defined
onlyContainsUserCerts -- forbidden to use
onlyContainsCACerts -- forbidden to use
onlySomeReasons -- forbidden to use
indirectCRL -- not defined

CA publishes one complete CRL and one complete ARL

iDP -- Mandatory (critical)
distributionPoint -- Optional
fullName – Optional (EXACTLY the same value as one of the names in the
cRLDP in the certificate)
nameRelativeToCRLIssuer -- not defined
onlyContainsUserCerts -- Mandatory in CRL
onlyContainsCACerts -- Mandatory in ARL
onlySomeReasons -- forbidden to use
indirectCRL -- not defined

CA publishes partitioned CRLs and ARL(s)

iDP -- Mandatory (critical)
distributionPoint – Mandatory
fullName – Mandatory (EXACTLY the same value as one of the names in the
cRLDP)
nameRelativeToCRLIssuer -- not defined

onlyContainsUserCerts -- Mandatory in CRL

onlyContainsCACerts -- Mandatory in ARL

onlySomeReasons -- forbidden to use

indirectCRL -- not defined

APPENDIX OCSP responder

Certificate Basic field

the same as ROOT CA Certificate

Certificate Extension field

extKeyUsage (Optional, non-critical)	OCSPSigning
To-be-defined [TBD]	TBD

6.6 Repository Profile

To store the certificate and CRL/ARL information in repository, this IWG architecture employs the LDAP directory. This profile uses LDAP v3, primarily to use the referral function to fetch the certificates and CRLs/ARLs in multiple PKI domains' environment. To simplify the directory operations, no replication and integrated directory environments are considered. The profile emphasizes that the referral is a focal function in order to access to the information in other domains.

6.6.1 DIT

DIT structure in each country is not specified. This specification only mandates that the DN in a certificate should be corresponding to the structure of the DN in DIT.

A sample DIT is following.

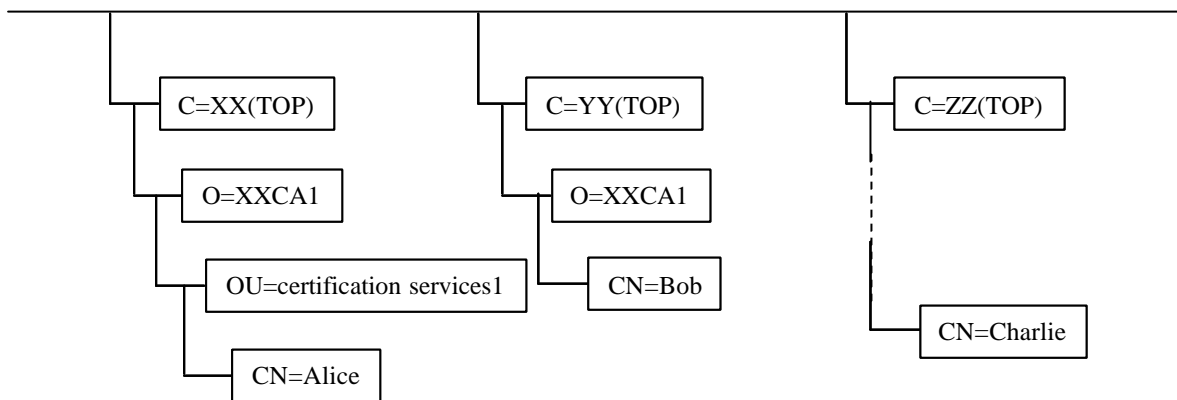


Fig. 4 sample DIT Tree (3 parties) in one directory

In Fig. 5, the “c=XX” entry, appropriate subordinate entry, and the referral should be defined. Note: in real usage, c=XX will not likely be used for the actual referral entry, since there is no such a representative directory server. The O or OU entry is the most likely.

6.6.2 Schema (objectclass, attribute)

No base objectclasses are described currently. Objectclass and attribute of each entry MUST be compliant with X.520, X.521, X.509, RFC2256, RFC2587, RFC2798, and other standard documents.

(1) CA

Objectclass and attribute

For the CA, the following object classes MUST be used.

- pkiCA (2.5.6.21) or certificationAuthority (2.5.6.16)

```
pkiCA OBJECT-CLASS ::= {  
    SUBCLASS OF {top}  
    KIND auxiliary  
    MAY CONTAIN {cACertificate /  
                 certificateRevocationList /  
                 authorityRevocationList /  
                 crossCertificatePair }  
    ID joint-iso-ccitt(2) ds(5) objectClass(6) pkiCA(22)}
```

```
cACertificate ATTRIBUTE ::= {  
    WITH SYNTAX Certificate  
    EQUALITY MATCHING RULE certificateExactMatch  
    ID joint-iso-ccitt(2) ds(5) attributeType(4) cACertificate(37) }
```

```
crossCertificatePairATTRIBUTE::={  
    WITH SYNTAX CertificatePair  
    EQUALITY MATCHING RULE certificatePairExactMatch  
    ID joint-iso-ccitt(2) ds(5) attributeType(4) crossCertificatePair(40)}
```

```
certificateRevocationListATTRIBUTE::={  
    WITH SYNTAX CertificateList  
    EQUALITY MATCHING RULE certificateListExactMatch  
    ID joint-iso-ccitt(2) ds(5) attributeType(4)  
    certificateRevocationList(39)}
```

```
authorityRevocationListATTRIBUTE::={  
    WITH SYNTAX CertificateList  
    EQUALITY MATCHING RULE certificateListExactMatch  
    ID joint-iso-ccitt(2) ds(5) attributeType(4)  
    authorityRevocationList(38)}
```

(2.5.6.16 NAME 'certificationAuthority' SUP top AUXILIARY
 MUST (authorityRevocationList \$ certificateRevocationList \$
 cACertificate) MAY crossCertificatePair)

(2) End Entity

No base objectclass is described currently.

Objectclass and attribute

For the EE, the following object classes MUST be used.

- pkiUser (2.5.6.21) or inetOrgPerson (2.16.840.1.113730.3.2.2)

```
pkiUser OBJECT-CLASS ::= {
    SUBCLASS OF      {top}
    KIND              auxiliary
    MAY CONTAIN       {userCertificate}
    ID                id-oc-pkiUser }
```

```
userCertificate ATTRIBUTE ::= {
    WITH SYNTAX
    Certificate
    EQUALITY MATCHING RULE
    certificateExactMatch
    ID                id-at-userCertificate }
```

(2.16.840.1.113730.3.2.2

NAME 'inetOrgPerson'

SUP organizationalPerson

STRUCTURAL

MAY (

audio \$ businessCategory \$ carLicense \$ departmentNumber \$
 displayName \$ employeeNumber \$ employeeType \$ givenName \$
 homePhone \$ homePostalAddress \$ initials \$ jpegPhoto \$
 labeledURI \$ mail \$ manager \$ mobile \$ o \$ pager \$
 photo \$ roomNumber \$ secretary \$ uid \$ userCertificate \$
 x500uniqueIdentifier \$ preferredLanguage \$
 userSMIMECertificate \$ userPKCS12

)

)

(3) CRLDP

Objectclass and attribute

For the CRLDP, the following object class MUST be used.

- cRLDistributionPoint (2.5.6.19)

```
cRLDistributionPoint          OBJECT-CLASS      ::= {  
    SUBCLASS OF                { top }  
    KIND                        structural  
    MUST CONTAIN                { commonName }  
    MAY CONTAIN                 { certificateRevocationList  
/ authorityRevocationList /    deltaRevocationList }  
    ID  
    id-oc-cRLDistributionPoint }
```

(4) Referral

Objectclass and attribute

For the Referral, the following object class MUST be used.

- Referral (2.16.840.1.113730.3.2.6)

```
( 2.16.840.1.113730.3.2.6  
    NAME 'referral'  
    DESC 'named subordinate reference object'  
    STRUCTURAL  
    MUST ref )
```

```
( 2.16.840.1.113730.3.1.34  
    NAME 'ref'  
    DESC 'named reference - a labeledURI'  
    EQUALITY caseExactMatch  
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15  
    USAGE distributedOperation )
```


6.6.3 Interoperability Consideration

The CRL/ARL are basically stored in the CA entry. However this is not the necessary condition as long as you specify the CRLDP entry and store them in the entry. To publish partitioned CRLs, the CRLDP entry must be specified for each partitioned CRL.

For the subordinate certificates, there are two possibilities to be stored. One is in cACertificate attribute of the subordinate CA entry. The other is in the crossCertificatePair attribute in the issuer's CA entry. To ensure the interoperability in the path construction and possibly increase the efficiency, it is recommended to store them in both of the entries.