IWG Recommended Profile

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 -

- Trademarks, Registered Trademarks -

1 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.

1.1 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.

1.2 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.

1.1.1 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 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. Vise versa.
subjectKeyIdentifier (Mandatory, non-critical)	The hash value of Issuer's pubic key (SHA1 160bit). 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)	

1.1.2 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 has 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 pubic 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. Vise versa.
subjectKeyIdentifier (Mandatory, non-critical)	The hash value of Issuer's pubic 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, ether critical or non-critical[1])	policyID MUST be present.
policyMappings (Mandatory, non-critical)	issuerDomain and subjectDomain OIDs MUST be present. OID must be exchanged beforehand via out 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)	"distributionPoint.fullname" must be used and contain 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 (not used)	If the PKI domain uses OCSP, this field will be used.
inhibitAnyPolicy (not used)	
freshestCRL (not used)	
subjectInfoAccessSyntax (not used)	

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

1.1.3 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 pubic 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. Vise versa.
subjectKeyIdentifier (Mandatory, non-critical)	The hash value of Issuer's pubic 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, ether critical or non-critical [2])	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	

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

(not used)	
basicConstraints (Mandatory, critical)	cA=TRUE pathLen=optional (INTEGER)
nameConstraints (optional, critical)	
policyConstraints (not used)	
cRLDistributionPoints (Mandatory, non-critical)	"distributionPoint.fullname" must be used and contain 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 (not used)	If the PKI domain uses OCSP, this field will be used.
inhibitAnyPolicy (not used)	
freshestCRL (not used)	
subjectInfoAccessSyntax (not used)	

1.3 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 "1.1.4". The following sections, "1.1.5" and "1.1.6" specify the differences from "1.1.4" for individual applications.

1.1.4 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. Vise 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 1.1.5 and 1.1.6 about a value.
extKeyUsage (not defined)	
privateKeyUsagePeriod (not used)	

certificatePolicies (Mandatory, ether 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. And see 1.1.6.
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 (optional, critical)	It recommends that CAs don't include a this field.
nameConstraints (not used)	
policyConstraints (not used)	
cRLDistributionPoints (Mandatory, non-critical)	"distributionPoint.fullname" must be used and contain 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 (not used)	If the PKI domain uses OCSP, this fieldwill be used.
inhibitAnyPolicy (not used)	
freshestCRL (not used)	
subjectInfoAccessSyntax (not used)	

1.1.5 Identification Certificate (digital signature)

? **Certificate Extension field**

keyUsage (Mandatory, critical)	digitalSignature (, nonRepudiation)

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

1.1.6 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.

1.4 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 handle the revocation policy of the CA.

1.1.7 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)
issure (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	**keyId(Mandatory):** The hash value of Issuer's

(Mandatory, non-critical)	pubic 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. Vise 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 1.1.10 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)	

1.5 Interoperability consideration (Certificate & CRL)

1.1.8 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.

1.1.9 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.

1.1.10 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 considered[4].

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.

[4] 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, onlyComeReasons fields are not used.

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 published at the locations of the cRLDistributionPoint.DistriubtionPoint.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

1.6 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.

1.1.11 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.

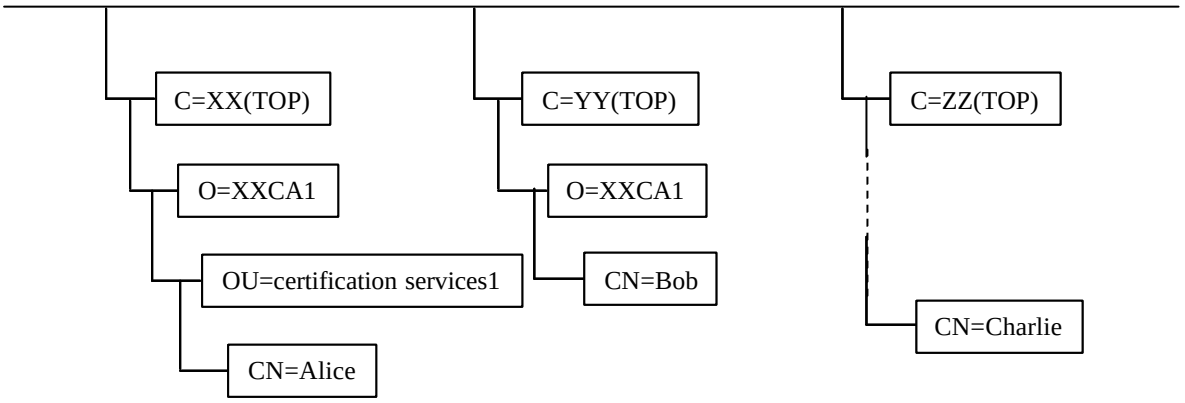

Fig. 1 sample DIT Tree (3 parties) in one directory

In Fig. 1, 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.

1.1.12 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)

1.1.13 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.