An LDAPv3 Schema for X.509 Certificates
draft-klasen-ldap-x509certificate-schema-03

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Comments and suggestions on this document are encouraged. They should be sent to the PKIX working group discussion list <ietf-pkix@imc.org> or directly to the authors.

This Internet-Draft will expire on December 29, 2003.

Copyright Notice

Copyright (C) The Internet Society (2003). All Rights Reserved.
Please see the full copyright statement near the end of this document for more information

Abstract

This document describes an LDAP schema which can be used to implement a certificate store for X.509 certificates. Specifically, two structural object classes for X.509 user and CA certificates are defined. Key fields of a certificate are stored in LDAP attributes...
so that applications can easily retrieve the certificates needed by using basic LDAP search filters. Multiple certificates for a single entity can be stored and retrieved.

Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

The following syntax specifications use the augmented Backus-Naur Form (ABNF) as described in [RFC2234].

Schema definitions are provided using LDAPv3 description formats [RFC2252]. Definitions provided here are formatted (line wrapped) for readability.
1. Introduction ................. 6
2. Comparison with Values Return Filter Control ............... 7
3. Comparison with component matching approach ............... 8
4. The x509certificate object classes and their attribute types ......................................................................................... 8
   4.1 Attributes for mandatory fields of an X.509 certificate . 8
      4.1.1 X.509 version ........................................ 8
      4.1.2 Serial number ......................................... 8
      4.1.3 Signature algorithm .................................... 9
      4.1.4 Issuer .................................................. 9
      4.1.5 Validity .................................................. 9
      4.1.6 Subject .................................................. 10
      4.1.7 Subject public key info algorithm .................... 10
      4.2 Attributes for selected extensions ......................... 11
         4.2.1 Authority key identifier extension ................ 11
         4.2.1.1 Authority key identifier ........................ 12
         4.2.1.2 Authority cert issuer ............................. 12
         4.2.1.3 Authority cert serial number ..................... 12
         4.2.2 Subject key identifier extension ................... 12
         4.2.3 Key usage extension ................................ 13
         4.2.4 Policy information identifier extension .......... 13
         4.2.5 Subject alternative name extension ................ 13
            4.2.5.1 Subject RFC822 name ........................... 14
            4.2.5.2 Subject DNS name ............................... 14
            4.2.5.3 Subject directory name .......................... 14
            4.2.5.4 Subject Uniform Resource Identifier .......... 15
            4.2.5.5 Subject IP address ............................. 15
            4.2.5.6 Subject registered ID ........................... 15
         4.2.6 Issuer alternative name extension ..................... 15
            4.2.6.1 Issuer RFC 822 name ............................. 16
            4.2.6.2 Issuer DNS name ................................ 16
            4.2.6.3 Issuer directory name ........................... 16
            4.2.6.4 Issuer Uniform Resource Identifier .......... 17
            4.2.6.5 Issuer IP address ............................... 17
            4.2.6.6 Issuer registered ID ............................. 17
         4.2.7 Basic constraints extension ........................... 17
         4.2.8 Extended key usage extension ......................... 18
         4.2.9 CRL distribution points extension ..................... 18
      4.3 Additional attributes .................................... 19
         4.3.1 Certificate location .................................. 19
         4.3.2 Certificate holder .................................... 19
         4.3.3 X.509 user certificate ............................... 19
         4.3.4 X.509 CA certificate .................................. 19
         4.4 X.509 PKC object class .................................. 19
         4.5 X.509 user certificate object class ..................... 20
         4.6 X.509 CA certificate object class ......................... 21
4.7 X.509 certificate holder object class  .............. 21
5. DIT structure and naming  . .......................... 21
6. Security Considerations ......................... 23
7. Open issues ...................................... 23
8. Acknowledgments .................................. 24
9. References ........................................ 24
     Normative references  ............................. 24
     Non-normative references ...................... 26
Authors’ Addresses ................................. 27
A. Sample directory entries  ....................... 27
B. Sample searches  .................................. 29
C. Changes from previous Drafts  ................. 30
   C.1 Changes in Draft 01 ......................... 30
   C.2 Changes in Draft 02  ......................... 30
   C.3 Changes in Draft 03 ......................... 31
   Full Copyright Statement ....................... 32
1. Introduction

A key component in the widespread adoption of a PKI infrastructure is the general availability of public keys and their certificates. Today, certificates are often published in an X.500 compliant directory service. These directories are accessed by applications using the LDAP v3 [RFC3377] protocol. An LDAPv3 schema for PKI repository objects is specified in [pkix-ldap-schema], where a set of object classes, attribute types, syntaxes, and extended matching rules are defined. For storing certificates, the "userCertificate" and "cACertificate" attribute types are used. All certificates of an entity are stored as values in these multi-valued attributes. This solution has a serious drawback. In LDAP, the smallest granularity of data access is the attribute. The directory server will therefore always return the full list of certificates of an entry to clients dealing with certificates. If the number of certificates for an entity is large this will result in considerable overhead and burden to the client.

This document proposes to solve this problem by the use of the structural object classes x509userCertificate and x509caCertificate for storing certificates. Each certificate will be stored in a separate entry in the directory.

While it is a simple matter to modify the DIT in such a way that all certificate information is removed from the entries and placed in the container directly beneath the entries according to the definitions of this specification, it is less simple to simultaneously modify all of the applications that depend on certificates being stored in the entry. Thus, it may be desirable to duplicate the certificate information, by having it appear in the entry, as well as in the container beneath the entry for a short period of time, in order to allow for migration of the applications to the new LDAP schema. As in any situation in which information is duplicated, great care must be taken in order to ensure the integrity and consistency of the information.

Fields of certificates which are needed to identify a certificate and those which are often used in searching for an appropriate certificate, are extracted from the certificate and stored as attributes of the entry. Each attribute type uses existing LDAP syntax, so that no new matching rules need to be defined. Applications can thus search for specific certificates with simple LDAP filters. This approach could be named a metadata approach, since data (.attributes) about data (certificate) are stored.

The use of simple attributes also makes a large scale widely distributed certificate repository service possible by using an
indexing service based on The Common Indexing Protocol (CIP) [RFC2651], which defines a protocol between index servers for exchanging indexobjects in order to facilitate query routing. The Tagged Index Object format as specified in [RFC2654] was specified to carry directory server information, by collecting the single attribute types and values.

This document is one of a set following this approach comprising:

1. the LDAP schema for X.509 public key certificates (this document)
2. the LDAP schema for X.509 attribute certificates [ldap-ac-schema]
3. the LDAP schema for X.509 CRLs [ldap-crl-schema]

Two alternative approaches are discussed in the next two sections.

2. Comparison with Values Return Filter Control

In [matchedval] a control has been defined that allows for only a subset of values of a specified attribute to be returned from a matching entry, by defining a filter for the returned values. In this section, this approach is compared with the one proposed in this document.

The major benefit of the Values Return Filter Control is that it does not require any changes to the DIT.

There are several advantages in using the x509certificate object class. No special matching rules are needed to retrieve a specific certificate. Any field in the certificate can be used in the search filter. Even information that doesn’t appear in the certificate can be used in a search filter. It is easier to remove certificates from the DIT, since the entire certificate BER/DER encoding does not have to be supplied in the modify operation. Searches that don’t need extensible matching rules and Values Return Filter Control will perform faster.

Another advantage of the solution proposed here is that it will not be necessary to modify existing server implementations to support this schema. The extended matching rules proposed in [pkix-ldap-schema] would require substantial changes in the servers’ indexing mechanisms. In contrast, servers implementing the x509certificate schema can easily leverage their indexing support for standard LDAPv3 syntaxes.

A CIP-based indexing system for a wide scale distributed certificate repository will rather be possible by using the solution proposed
here due to its dependency on attribute values.

3. Comparison with component matching approach

[componentmatch] defines a new mechanism for matching in complex syntaxes, by defining generic matching rules that can match any user selected component parts in an attribute value of any arbitrarily complex attribute syntax. We believe that this might be the proper way to solve search problems in the longer term, but that it will take a long time until such ASN.1 based mechanisms will be implemented in LDAP servers and clients. Even if this has happened the mechanism proposed here, will still be useful in the frame of CIP. A simple and easy to implement mechanism is needed today and this is what this memo wants to provide.

4. The x509certificate object classes and their attribute types

The description of all attributes with relevance to fields and extensions of an X.509 certificate include a respective reference to [X.509-2000] and to [RFC3280].

4.1 Attributes for mandatory fields of an X.509 certificate

4.1.1 X.509 version

X.509 Version of the encoded certificate (See X.509(2000) 7, RFC3280 4.1.2.1.) or of the CRL.

( 1.3.6.1.4.1.10126.1.5.3.1
   NAME 'x509version'
   DESC 'X.509 Version of the certificate, or of the CRL'
   EQUALITY integerMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
   SINGLE-VALUE )

Values of this attribute may either be 0, 1, 2 or 3 corresponding to X.509 v1, v2, v3, or v4.

4.1.2 Serial number

The serial number is an integer assigned by the CA to each certificate. It is unique for each certificate issued by a given CA (i.e., the issuer name and serial number uniquely identify a certificate). See X.509(2000) 7, RFC3280 4.1.2.2
( 1.3.6.1.4.1.10126.1.5.3.2
  NAME 'x509serialNumber'
  DESC 'Unique integer for each certificate issued by a
  particular CA'
  EQUALITY integerMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 )

4.1.3 Signature algorithm

  OID identifying the algorithm used by the CA in signing the
  certificate (see X.509(2000) 7, RFC3280 4.1.2.3) or the CRL.

( 1.3.6.1.4.1.10126.1.5.3.3
  NAME 'x509signatureAlgorithm'
  DESC 'OID of the algorithm used by the CA in
  signing the CRL or the certificate'
  EQUALITY objectIdentifierMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.38
  SINGLE-VALUE )}

4.1.4 Issuer

  String representation of the issuer’s distinguished name (see
  X.509(2000) 7, RFC3280 4.1.2.4)

( 1.3.6.1.4.1.10126.1.5.3.4
  NAME 'x509issuer'
  DESC 'Distinguished name of the entity who has signed and
  issued the certificate'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
  SINGLE-VALUE )

  Values of this attribute type must be encoded according to the syntax
given in [RFC2253].

4.1.5 Validity

  The "validity" attribute in an X.509 certificate (see X.509(2000) 7,
  RFC3280 4.1.2.5) consists of an ASN.1 sequence of two timestamps
  which define the begin and end of the certificate’s validity period.
  This sequence has been split up into two separate attributes
  "x509validityNotBefore" and "x509validityNotAfter". The times are
  represented in string form as defined in [RFC2252].
( 1.3.6.1.4.1.10126.1.5.3.5
  NAME 'x509validityNotBefore'
  DESC 'Date on which the certificate validity period begins'
  EQUALITY generalizedTimeMatch
  ORDERING generalizedTimeOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
  SINGLE-VALUE )

( 1.3.6.1.4.1.10126.1.5.3.6
  NAME 'x509validityNotAfter'
  DESC 'Date on which the certificate validity period ends'
  EQUALITY generalizedTimeMatch
  ORDERING generalizedTimeOrderingMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
  SINGLE-VALUE )

Note that the field in the certificate may be in UTC or GeneralizedTime format. If in UTC format, the creator of this attribute MUST convert the UTC time into GeneralisedTime format when creating the attribute value.

4.1.6 Subject

String representation of the subject’s distinguished name (see X.509(2000) 7, RFC3280 4.1.2.6).

( 1.3.6.1.4.1.10126.1.5.3.7
  NAME 'x509subject'
  DESC 'Distinguished name of the entity associated with this public-key'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
  SINGLE-VALUE )

Values of this attribute type must be encoded according to the syntax given in [RFC2253].

4.1.7 Subject public key info algorithm

OID identifying the algorithm associated with the certified public key (see X.509(2000) 7, RFC3280 4.1.2.7).
( 1.3.6.1.4.1.10126.1.5.3.8
  NAME 'x509subjectPublicKeyInfoAlgorithm'
  DESC 'OID identifying the algorithm associated with the certified
   public key'
  EQUALITY objectIdentifierMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.38
  SINGLE-VALUE )

4.2 Attributes for selected extensions

As this specification intends to only facilitate applications in
finding certificates, only those extensions have to be defined that
might be searched for. Thus extensions described in [RFC3280] like
the following are not dealt with here:

  o  private key usage period extension
  o  policy mappings extension
  o  subject directory attributes extension
  o  basic constraints extension
  o  name constraints extensions
  o  policy constraints extensions
  o  inhibit any policy extension
  o  freshest CRL extension
  o  authority information access extension
  o  subject information access extension

4.2.1 Authority key identifier extension

This attribute identifies the public key to be used to verify the
signature on this certificate or CRL (see X.509(2000) 8.2.2.1,
RFC3280 4.2.1.1). The key may be identified by an explicit key
identifier in the keyIdentifier component, by identification of a
certificate for the key (giving certificate issuer in the
authorityCertIssuer component and certificate serial number in the
authorityCertSerialNumber component), or by both explicit key
identifier and identification of a certificate for the key.
4.2.1.1 Authority key identifier

( 1.3.6.1.4.1.10126.1.5.3.11
   NAME 'x509authorityKeyIdentifier'
   DESC 'Key Identifier field of the Authority Key Identifier extension'
   EQUALITY octetStringMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
   SINGLE-VALUE )

4.2.1.2 Authority cert issuer

( 1.3.6.1.4.1.10126.1.5.3.12
   NAME 'x509authorityCertIssuer'
   DESC 'Authority Cert Issuer field of the Authority Key Identifier extension'
   EQUALITY distinguishedNameMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
   SINGLE-VALUE )

In this specification, only the "Name" choice, encoded according to [RFC2253], of the "GeneralName" type may be used.

4.2.1.3 Authority cert serial number

( 1.3.6.1.4.1.10126.1.5.3.13
   NAME 'x509authorityCertSerialNumber'
   DESC 'Authority Cert Serial Number field of the Authority Key Identifier extension'
   EQUALITY integerMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
   SINGLE-VALUE )

4.2.2 Subject key identifier extension

This attribute identifies the public key being certified (see X.509(2000) 8.2.2.2, RFC3280 4.2.1.2). It enables distinct keys used by the same subject to be differentiated.

( 1.3.6.1.4.1.10126.1.5.3.14
   NAME 'x509subjectKeyIdentifier'
   DESC 'Key identifier which must be unique with respect to all key identifiers for the subject'
   EQUALITY octetStringMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
   SINGLE-VALUE )
4.2.3 Key usage extension

This attribute defines the purpose (e.g., encipherment, signature, certificate signing) of the key contained in the certificate (see X.509(2000) 8.2.2.3, RFC3280 4.2.1.3).

( 1.3.6.1.4.1.10126.1.5.3.15
  NAME 'x509keyUsage'
  DESC 'Purpose for which the certified public key is used'
  EQUALITY caseIgnoreMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )

Values of this type are encoded according to the following BNF, so that each value corresponds to the respective bit in the ASN.1 "KeyUsage" bitstring:

x509keyUsage-value =
  "digitalSignature" / "nonRepudiation" / "keyEncipherment" / "dataEncipherment" / "keyAgreement" / "keyCertSign" / "cRLSign" / "encipherOnly" / "decipherOnly"

4.2.4 Policy information identifier extension

This attribute contains OIDs which indicate the policy under which the certificate has been issued and the purposes for which the certificate may be used (see X.509(2000) 8.2.2.6, RFC3280 4.2.1.5).

( 1.3.6.1.4.1.10126.1.5.3.16
  NAME 'x509policyInformationIdentifier'
  DESC 'OID which indicates the policy under which the certificate has been issued and the purposes for which the certificate may be used'
  EQUALITY objectIdentifierMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.38
  SINGLE-VALUE )

4.2.5 Subject alternative name extension

The subject alternative name extension allows additional identities to be bound to the subject of the certificate (see X.509(2000) 8.3.2.1, RFC3280 4.2.1.7). Separate attribute types are defined for all choices of the ASN.1 type "GeneralName" except for "otherName", "x400Address" and "ediPartyName".
4.2.5.1 Subject RFC822 name

\[
\begin{aligned}
\text{NAME} & \text{ 'x509subjectRfc822Name' } \\
\text{DESC} & \text{ 'Internet electronic mail address of the entity associated with this public-key' } \\
\text{EQUALITY} & \text{ caseIgnoreIA5Match } \\
\text{SUBSTR} & \text{ caseIgnoreIA5SubstringsMatch } \\
\text{SYNTAX} & \text{ 1.3.6.1.4.1.1466.115.121.1.26 }
\end{aligned}
\]

Values of this attribute must be encoded according to the syntax given in [RFC0822].

4.2.5.2 Subject DNS name

\[
\begin{aligned}
\text{NAME} & \text{ 'x509subjectDnsName' } \\
\text{DESC} & \text{ 'Internet domain name of the entity associated with this public-key' } \\
\text{EQUALITY} & \text{ caseIgnoreIA5Match } \\
\text{SUBSTR} & \text{ caseIgnoreIA5SubstringsMatch } \\
\text{SYNTAX} & \text{ 1.3.6.1.4.1.1466.115.121.1.26 }
\end{aligned}
\]

Values of this attribute must be encoded as Internet domain names in accordance with [RFC1035].

4.2.5.3 Subject directory name

\[
\begin{aligned}
\text{NAME} & \text{ 'x509subjectDirectoryName' } \\
\text{DESC} & \text{ 'Distinguished name of the entity associated with this public-key' } \\
\text{EQUALITY} & \text{ distinguishedNameMatch } \\
\text{SYNTAX} & \text{ 1.3.6.1.4.1.1466.115.121.1.12 }
\end{aligned}
\]

Values of this attribute type must be encoded according to the syntax given in [RFC2253].
4.2.5.4 Subject Uniform Resource Identifier

\begin{verbatim}
( 1.3.6.1.4.1.10126.1.5.3.20
  NAME 'x509subjectUniformResourceIdentifier'
  DESC 'Uniform Resource Identifier for the World-Wide Web of the entity associated with this public-key'
  EQUALITY caseExactIA5Match
  SUBSTR caseExactIA5SubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
\end{verbatim}

Values of this attribute must be encoded according to the syntax given in [RFC2396].

4.2.5.5 Subject IP address

\begin{verbatim}
( 1.3.6.1.4.1.10126.1.5.3.21
  NAME 'x509subjectIpAddress'
  DESC 'Internet Protocol address of the entity associated with this public-key'
  EQUALITY caseIgnoreIA5Match
  SUBSTR caseIgnoreIA5SubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
\end{verbatim}

Values of this attribute type must be stored in the syntax given in Appendix B of [RFC2373].

4.2.5.6 Subject registered ID

\begin{verbatim}
( 1.3.6.1.4.1.10126.1.5.3.22
  NAME 'x509subjectRegisteredID'
  DESC 'OID of any registered object identifying the entity associated with this public-key'
  EQUALITY objectIdentifierMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.38 )
\end{verbatim}

registeredID is an identifier of any registered object assigned in accordance with ITU-T Rec. X.660.

4.2.6 Issuer alternative name extension

The issuer alternative names extension allows additional identities to be bound to the subject of the certificate or CRL (see X.509(2000) 8.3.2.2, RFC3280 4.2.1.8). Separate attribute types are defined for all choices of the ASN.1 type "GeneralName" except for "otherName", "x400Address" and "ediPartyName".
4.2.6.1 Issuer RFC 822 name

( 1.3.6.1.4.1.10126.1.5.3.23
  NAME 'x509issuerRfc822Name'
  DESC 'Internet electronic mail address of the entity who has
       signed and issued the certificate'
  EQUALITY caseIgnoreIA5Match
  SUBSTR caseIgnoreIA5SubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

Values of this attribute must be encoded according to the syntax
given in [RFC0822].

4.2.6.2 Issuer DNS name

( 1.3.6.1.4.1.10126.1.5.3.24
  NAME 'x509issuerDnsName'
  DESC 'Internet domain name of the entity who has
       signed and issued the certificate'
  EQUALITY caseIgnoreIA5Match
  SUBSTR caseIgnoreIA5SubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

Values of this attribute must be encoded as Internet domain names in
accordance with [RFC1035].

4.2.6.3 Issuer directory name

( 1.3.6.1.4.1.10126.1.5.3.25
  NAME 'x509issuerDirectoryName'
  DESC 'Distinguished name of the entity who has
       signed and issued the certificate'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 )

Values of this attribute type must be encoded according to the syntax
given in [RFC2253].
4.2.6.4 Issuer Uniform Resource Identifier

    ( 1.3.6.1.4.1.10126.1.5.3.26
      NAME 'x509issuerUniformResourceIdentifier'
      DESC 'Uniform Resource Identifier for the World-Wide Web
            of the entity who has signed and issued the certificate'
      EQUALITY caseExactIA5Match
      SUBSTR caseExactIA5SubstringsMatch
      SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

Values of this attribute must be encoded according to the syntax
given in [RFC2396].

4.2.6.5 Issuer IP address

    ( 1.3.6.1.4.1.10126.1.5.3.27
      NAME 'x509issuerIpAddress'
      DESC 'Internet Protocol address of the entity who has
            signed and issued the certificate'
      EQUALITY caseIgnoreIA5Match
      SUBSTR caseIgnoreIA5SubstringsMatch
      SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

Values of this attribute type must be stored in the syntax given in
Appendix B of [RFC2373].

4.2.6.6 Issuer registered ID

    ( 1.3.6.1.4.1.10126.1.5.3.28
      NAME 'x509issuerRegisteredID'
      DESC 'OID of any registered object identifying the entity who has
            signed and issued the certificate'
      EQUALITY objectIdentifierMatch
      SYNTAX 1.3.6.1.4.1.1466.115.121.1.38 )

registeredID is an identifier of any registered object assigned in
accordance with ITU-T Rec. X.660.

4.2.7 Basic constraints extension

This attribute indicates whether the subject of the certificate is a
CA (see X.509(2000) 8.4.2.1, RFC3280 4.2.1.10). If the value of this
attribute is "TRUE", the certificate MUST be stored in the
"cacertificate" attribute.
4.2.8 Extended key usage extension

This attribute indicates one or more purposes for which the certified public key may be used, in addition to or in place of the basic purposes indicated in the "x509keyUsage" attribute (see X.509(2000) 8.2.2.4, RFC3280 4.2.1.13). These purposes are identified by their OID.

( 1.3.6.1.4.1.10126.1.5.3.30
  NAME 'x509extKeyUsage'
  DESC 'Purposes for which the certified public key may be used, identified by an OID'
  EQUALITY objectIdentifierMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.38 )

4.2.9 CRL distribution points extension

This attribute identifies how the full CRL information for this certificate can be obtained (see X.509(2000) 8.6.2.1, RFC3280 4.2.1.14).

( 1.3.6.1.4.1.10126.1.5.3.32
  NAME 'x509fullCRLDistributionPointURI'
  DESC 'URI type of DistributionPointName for the full CRL'
  EQUALITY caseExactIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

In this specification, only the "uniformResourceIdentifier" choice of "distributionPoint fullName" field is supported. If this attribute exists in an entry, both the "reasons" and "cRLIssuer" fields MUST be absent from the certificate, i.e. the CRL distributed by the distribution point contains revocations for all revocation reasons and the CRL issuer is identical to the certificate issuer.

Values of this attribute must be encoded according to the URI syntax given in [RFC2396].
4.3 Additional attributes

4.3.1 Certificate location

This attribute contains a pointer to the directory entry of a certificate. Thus it is possible to point to the certificate from an, e.g., white pages entry.

( 1.3.6.1.4.1.10126.1.5.4.74
  NAME 'x509certLocation'
  DESC 'Pointer to a x509certificate Entry'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 )

4.3.2 Certificate holder

This attribute contains a pointer to the directory entry of the end entity to which this certificate was issued. Thus it is possible to link a certificate entry in a certificate repository to, e.g., a white pages entry of the subject.

( 1.3.6.1.4.1.10126.1.5.4.75
  NAME 'x509certHolder'
  DESC 'Pointer to the directory entry of the end entity to which this certificate was issued'
  EQUALITY distinguishedNameMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 )

4.3.3 X.509 user certificate

This attribute is used to store the complete certificate. Since it has to be single valued the multi valued attribute userCertificate [pkix-ldap-schema] cannot be used.

( 1.3.6.1.4.1.10126.1.5.4.76
  NAME 'x509userCert'
  DESC 'Complete x.509 user certificate'
  SUP userCertificate
  SINGLE-VALUE )

4.3.4 X.509 CA certificate

This attribute is used to store the complete CA certificate. Since it has to be single valued the multi valued attribute caCertificate [pkix-ldap-schema] cannot be used.
4.4 X.509 PKC object class

This abstract object class contains the fields of an X.509 user certificate or CA certificate that are used in searches as attributes and in name forms. It is derived from the abstract objectclass x.509base as specified in [ldap-crl-schema] and is base for the two following object classes.

( 1.3.6.1.4.1.10126.1.5.4.2.3
NAME 'x509PKC'
SUP x509base
ABSTRACT
MUST ( x509serialNumber $ x509validityNotBefore $ x509validityNotAfter $ x509subjectPublicKeyInfoAlgorithm )
MAY ( x509authorityKeyIdentifier $ x509authorityCertIssuer $ x509authorityCertSerialNumber $ x509subjectKeyIdentifier $ x509keyUsage $ x509policyInformationIdentifier $ x509subjectRfc822Name $ x509subjectDnsName $ x509subjectDirectoryName $ x509subjectUniformResourceIdentifier $ x509subjectIpAddress $ x509subjectRegisteredID $ x509issuerRfc822Name $ x509issuerDnsName $ x509issuerDirectoryName $ x509issuerUniformResourceIdentifier $ x509issuerIpAddress $ x509issuerRegisteredID $ x509extKeyUsage $ x509FullcRLDistributionPointURI $ x509certHolder $ x509issuerSerial $ x509basicConstraintsCa ) )

The attribute description of x509issuerSerial can be found in [ldap-ac-schema]

4.5 X.509 user certificate object class

This object class is for storing user certificates.
The attribute type x509subject is specified here as a MAY attribute. Nevertheless if this attribute is not used at least one of the following attributes MUST be filled in: x509subjectRfc822Name, x509subjectDnsName, x509subjectDirectoryName, x509subjectUniformResourceIdentifier, x509subjectIpAddress, or x509subjectRegisteredID.

4.6 X.509 CA certificate object class

This object class is for storing CA certificates.

( 1.3.6.1.4.1.10126.1.5.4.2.5
  NAME 'x509caCertificate'
  SUP x509PKC
  STRUCTURAL
  MUST ( x509caCert $ x509subject ) )

4.7 X.509 certificate holder object class

This auxiliary object class has an attribute that contains a pointer to an entry with x509certificate objectclass. Thus it is possible to link, e.g., an entry of a white pages directory to an entry in a certificate store.

( 1.3.6.1.4.1.10126.1.5.4.2.2
  NAME 'x509certificateHolder'
  AUXILIARY
  MAY ( x509certLocation ) )

5. DIT structure and naming

If the schema presented in this document is used to store certificate information in a directory that contains entries for organizations, persons, services, etc., each certificate belonging to an entity SHOULD be stored as a direct subordinate to the entity’s entry. In this case, these entries MUST be named by a multi-valued RDN formed by the certificate issuer and serial number, as this is the only way to enforce unique RDN under the siblings. This is expressed in the following two name forms:

Gietz & Klasen          Expires December 29, 2003              [Page 20]
There are some LDAP implementations that don’t support multi-valued RDNs. These can use following alternative two name forms:

( 1.3.6.1.4.1.10126.1.5.5.8
  NAME "x509userCertificateAltNameForm"
  OC x509userCertificate
  MUST x509issuerSerial )

( 1.3.6.1.4.1.10126.1.5.5.9
  NAME "x509PcaCertificateAltNameForm"
  OC x509caCertificate
  MUST x509issuerSerial )

The attribute description of x509issuerSerial can be found in [ldap-ac-schema]

For public directories of CAs that, besides the data stored in the certificates, do not hold any additional data about end entities the following DIT structure might be preferable. Entries for certificates are stored directly below the issuing CA’s entry. In this case these entries SHOULD be named by the certificate serial number. This is expressed in the following two name forms:

( 1.3.6.1.4.1.10126.1.5.5.10
  NAME "x509userCertificateSerialNumberNameForm"
  OC x509userCertificate
  MUST x509serialNumber )

( 1.3.6.1.4.1.10126.1.5.5.11
  NAME "x509caCertificateSerialNumberNameForm"
  OC x509caCertificate
  MUST x509serialNumber )

Care must be taken when encoding DNs that contain an x509issuer attribute. Such a value is a string representation according to
These strings contain RFC2253 special characters and must therefore be escaped. For example, the issuer name in a certificate may be:

x509issuer: OU=VeriSign Trust Network,OU=(c) 1998 VeriSign\2c Inc. - For authorized use only,OU=Class 1 Public Primary Certification Authority - G2,O=VeriSign\2c Inc.,C=US

When used in a DN, this will be appear as:

dn: x509serialNumber=123456+x509issuer=OU=VeriSign Trust Network\2c OU=(c) 1998 VeriSign\2c Inc. - For authorized use only,OU=Class 1 Public Primary Certification Authority - G2,O=VeriSign\2c Inc.,C=US,cn=Joe Example,...

6. Security Considerations

Attributes of directory entries are used to provide descriptive information about the real-world objects they represent which can be people, organizations, or devices. Most countries have privacy laws regarding the publication of information about people.

Without additional mechanisms such as Operation Signatures [RFC2649] which allow a client to verify the origin and integrity of the data contained in the attributes defined in this document, a client MUST NOT treat this data as authentic. Clients MUST only use - after proper validation - the data which they obtained directly from the certificate. Directory administrators MAY deploy ACLs which limit access to the attributes defined in this document to search filters.

Transfer of cleartext passwords is strongly discouraged where the underlying transport service cannot guarantee confidentiality and may result in disclosure of the password to unauthorized parties.

In order to protect the directory and its contents, strong authentication MUST have been used to identify the Client when an update operation is requested.

7. Open issues

There are still a number of todos with respect to this draft. Following work items will be dealt with in the next version of this draft:

- Section on IANA considerations
- Specification of an auxiliary object class with additional...
attributes for storing a Qualified certificates as defined in RFC3039

- Specification of an auxiliary object class with attributes standardized elsewhere (e.g. RFC 2798) for additional certificate search possibilities, e.g. for the attribute mail.

- complete alignment with [ldap-ac-schema] and [ldap-crl-schema]

8. Acknowledgments

This document borrows from a number of IETF documents, including [certinfo-schema].

The authors wish to thank David Chadwick, Russ Housley, Mikhail Sahalayev, Michael Stroeder, and Kurt Zeilenga for their contributions to this document.

This work is part of the DFN Project "Ausbau und Weiterbetrieb eines Directory Kompetenzzentrums" funded by the German Ministry of Research (BMBF).

This document has been written in XML according to the DTD specified in RFC2629. xml2rfc has been used to generate an RFC2033 compliant plain text form. The XML source and a HTML version are available on request.

9. References

Normative references


[pkix-ldap-schema] Chadwick, D. and S. Legg, "Internet X.509 Public Key Infrastructure - LDAP Schema and Syntaxes for PKIs", Internet Draft (work in progress), June 2002, <draft-ietf-pkix-ldap-pki-schema-
Non-normative references


Appendix A. Sample directory entries

A sample x509Certificate directory entry for an intermediate CA certificate in LDIF format:

dn: x509serialNumber=4903272,EMAILADDRESS=certify@pca.dfn.de,CN=DFN Toplevel Certification Authority,OU=DFN-PCA,OU=DFN-CERT GmbH,O=Deutsches Forschungsnetz,C=DE
objectclass: x509caCertificate
x509version: 2
x509issuer: EMAILADDRESS=certify@pca.dfn.de,CN=DFN Toplevel Certification Authority,OU=DFN-PCA,OU=DFN-CERT GmbH,O=Deutsches Forschungsnetz,C=DE
x509validityNotBefore: 20020110170112Z
x509validityNotAfter: 20060110170112Z
x509subject: EMAILADDRESS=ca@daasi.de,OU=DAASI CA,O=DAASI International GmbH,C=DE
x509subjectPublicKeyInfoAlgorithm: 1.2.840.113549.1.1.1
x509basicConstraintsCa: TRUE
x509keyUsage: keyCertSign
x509keyUsage: cRLSign
x509subjectKeyIdentifier:: 5nrZFpVK4RKfIglgQ4N4JXBS4Bk=
x509cLRdistributionPointURI: http://www.dfn-pca.de/certification/x509/g1/data/crls/root-ca-crl.crx
x509cLRdistributionPointURI: http://www.dfn-pca.de/certification/x509/g1/data/crls/root-ca-crl.crl
x509policyInformationIdentifier: 1.3.6.1.4.1.11418.300.1.1

x509caCert::

---

A sample x509certificate directory entry for an end identity certificate in LDIF format:
Appendix B. Sample searches

This section details how clients should access the certstore. The searches are presented in LDAP URL format.

Retrieval all certificates for an end entity from a certstore using the first DIT structure:

ldap://CN=Norbert%20Klasen,O=DAASI%20International%20GmbH,C=de?x509userCert?one?(objectClass=x509userCertificate)

Find a certificate in a trustcenter's certstore suitable for sending an encrypted S/MIME message to "norbert.klasen@daasi.de"
ldap:///O=TC%20TrustCenter%20for%20Security%20in%20Data%20Networks
%20GmbH,L=Hamburg,ST=Hamburg,C=de?x509userCert?sub?
((objectClass=x509userCertificate)
 (x509subjectRfc822Name=norbert.klasen@daasi.de) )
(|(x509keyUsage=keyEncipherment)(x509keyUsage=keyAgreement)
 (x509extendedKeyUsage=1.3.6.1.5.5.7.3.4)))

Find a CA certificate by its "subjectKeyIdentifier" obtained from the
"keyIdentifier" field of the "authorityKeyIdentifier" extension in an
end entity certificate:

ldap:///?caCertificate?sub?
((objectClass=x509caCertificate)(x509subjectKeyIdentifier=%5CE6
%5C7A%5C9D%5C16%5C95%5C4A%5CE1%5C12%5C9F%5C22%5C09%5C6A%5C83%
5C83%5C78%5C25%5C70%5C52%5CE0%5C19))

Appendix C
Changes from previous Drafts

C.1 Changes in Draft 01

- Included new Attributes x509authorityKeyIdentifier,
x509authorityCertissuer, x509authorityCertSerialNumber,
x509certificateLocation, x509certificateHolder, and new
objectclass x509certificateHolder
- Fixed bug in definition of objectclass x509certificate
- Changed references from RFC 2459 to RFC 3280 and included some
respective language in 3.2.
- Changed references from RFC 2251 to RFC 3377 and deleted all
references to LDAPv2.
- Deleted ";binary" in examples
- Included new section: Comparison with component matching approach
- Some changes in wording and section titles, and elimination of
typos
- Changed order of authors, and one author’s address

C.2 Changes in Draft 02

- abstract object class x509PKC
aligned to [ldap-ac-schema] and [ldap-crl-schema]

C.3 Changes in Draft 03

- Changed Matching Rules from caseIgnoreMatch to caseIgnoreIA5Match etc.
- moved the references to RFC 3280 from the DESC part of the attribute definition to the text
- added some additional text about CIP in Introduction
- reworded text in Section 4.1.7
- changed x509userCert and x509caCert to be inherited from userCertificate and caCertificate respectively
- added clarification about x509subject and subject alternative names in section Section 4.5
- added attribute type x509issuerSerial to x509PKC object class
- added attribute type x509basicConstraintsCa to x509PKC object class
- renamed attributetype x509cRLDistributionPointURI to x509FullcRLDistributionPointURI
- divided references in normative and non normative
- deleted attributetype mail from x509PKC objectclass
- created separate Name Forms for x509userCertificate and x509caCertificate object classes.
- changed attributetype x509SerialNumber to MULTI-VALUE
- adjusted examples to new schema
- Fixed more typos
Full Copyright Statement

Copyright (C) The Internet Society (2003). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.