An LDAPv3 Schema for X.509 Certificates

draft-klasen-ldap-x509certificate-schema-01

IETF 55, PKIX Meeting

November 20, 2002,

Atlanta, GA

Peter Gietz, Norbert Klasen

DAASI International / DFN

peter.gietz@daasi.de

Agenda

- Motivation and General Idea
- Changes in –01 ID
- Proposed changes from list discussions
- Open issues and future work

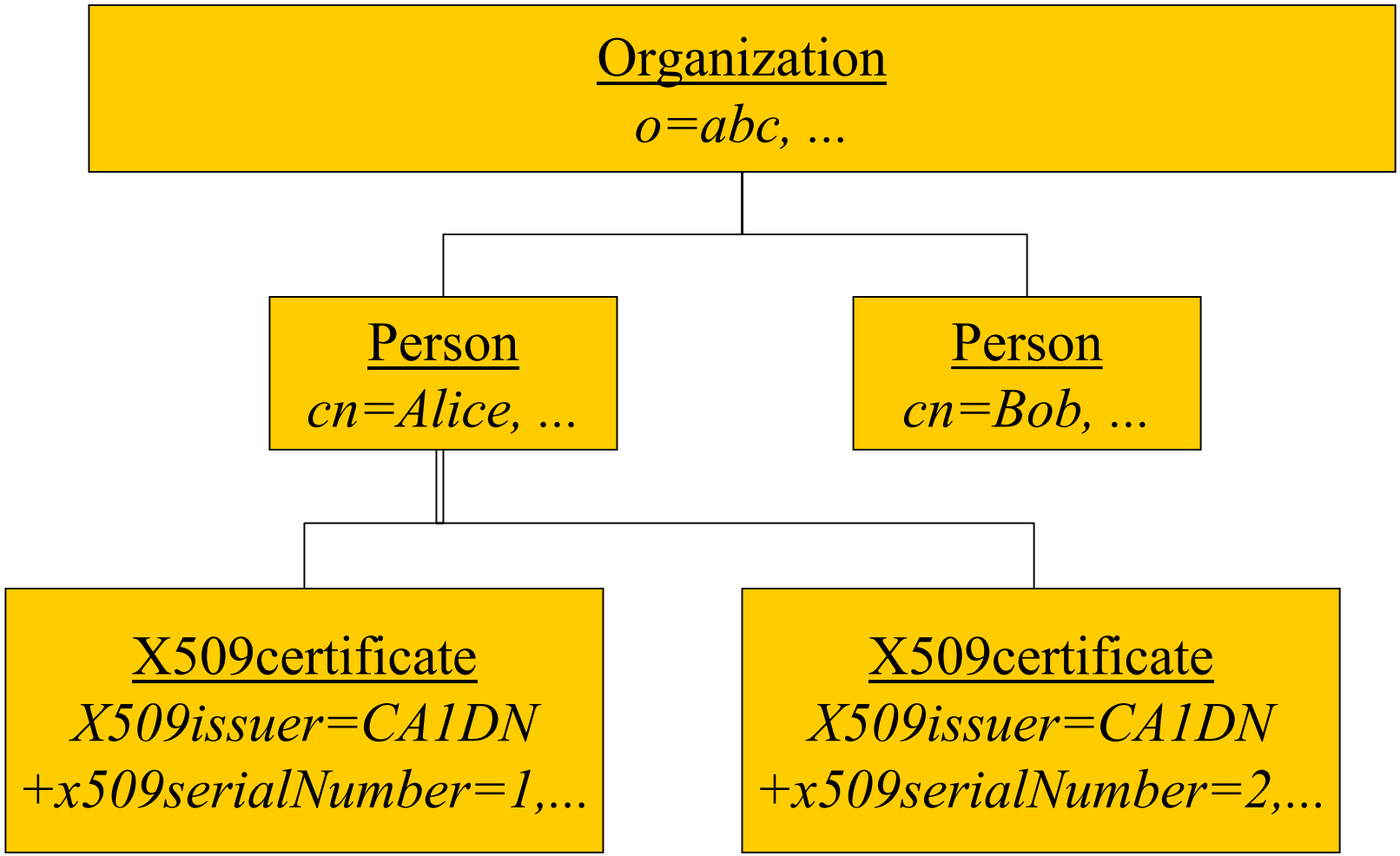
Motivation

- Address problem of multiple certificates for one entity
  - How can the client find the right certificate?
- Find a simple and easy to implement solution
- Solution should be usable in the frame of a large scale distributed LDAP / Common Indexing Protocol (CIP) based certificate repository

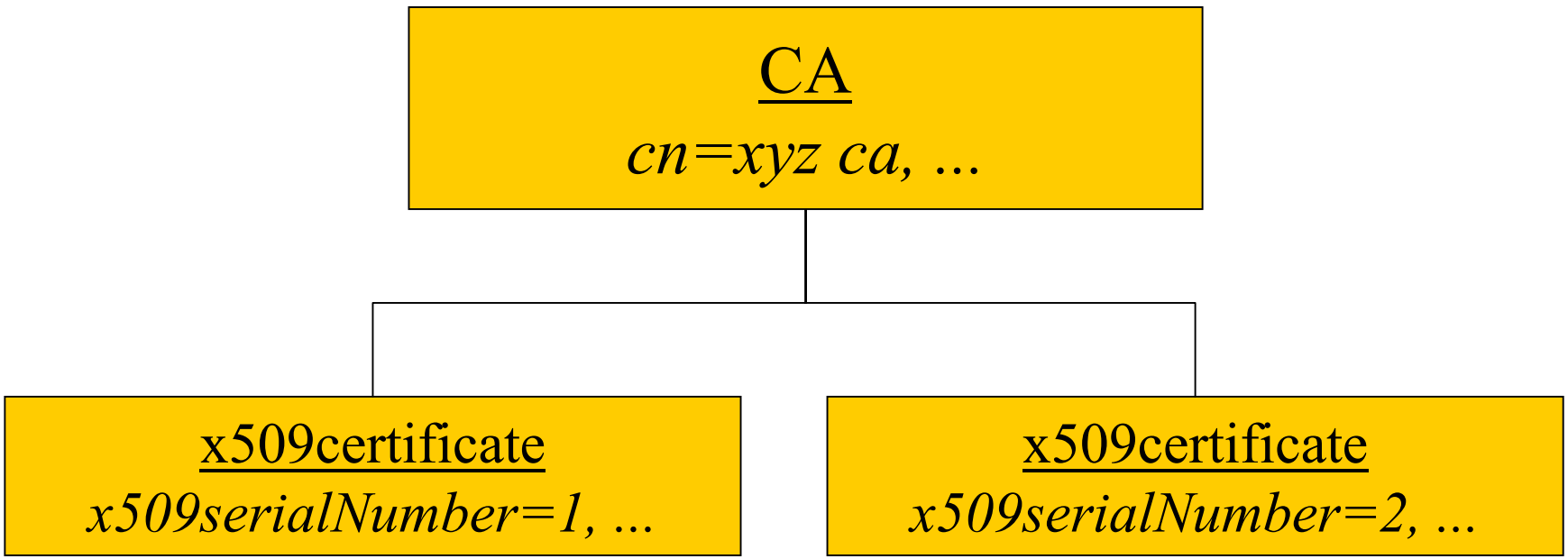
Schema as a simple solution

- Find a set of certificate fields and extensions that one might want to search upon
  - Meta-data approach
- Parse the certificate and store this set as LDAP attributes
- Advantages:
  - no new server features needed
  - easy to implement in clients
  - usable in a CIP environment

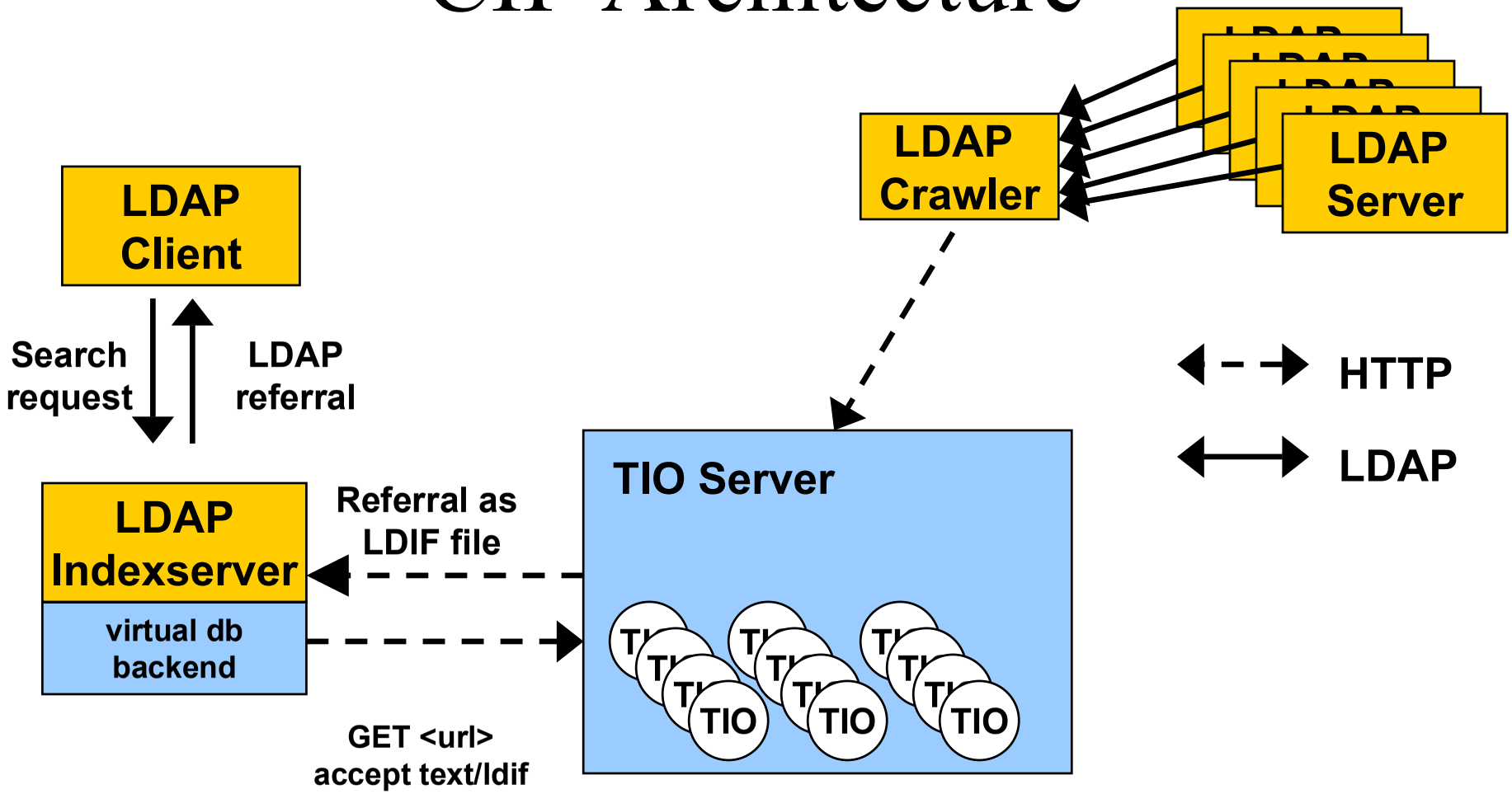
DIT Structure in white-pages services



DIT Structure in certificate repositories



CIP Architecture



Related work

- This approach:
  - Greenblatt, B., "LDAP Object Class for Holding Certificate Information", Internet Draft (work in progress, expired), Februar 2000, draft-greenblatt-ldap-certinfo-schema-02.txt
- The smarter but more complex solution:
  - Legg, S., "LDAP & X.500 Component Matching Rules", Internet Draft (work in progress), October 2002, draft-legg-ldapext-component-matching-09.txt
  - Chadwick, D. and S. Mullan, "Returning Matched Values with LDAPv3", Internet Draft (work in progress, expired), June 2002, draft-ietf-ldapext-matchedval-06.txt

Changes in Draft 01

- Fixed bug in definition of objectclass x509certificate
- updated references (RFC 3280, RFC 3377)
- new attributes
  - x509authorityKeyIdentifier
  - x509authorityCertissuer
  - x509authorityCertSerialNumber
  - x509certificateLocation
  - x509certificateHolder
- new objectclass
  - x509certificateHolder

Changes in Draft –01(cont´d)

attributetype (1.3.6.1.4.1.10126.1.5.4.73
NAME 'x509certificateHolder'
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)

Changes in Draft –01(cont´d)

attributetype (1.3.6.1.4.1.10126.1.5.4.71
NAME 'x509certificateLocation'
DESC 'Pointer to an x509certificate Entry'
EQUALITY distinguishedNameMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12)

objectclass (1.3.6.1.4.1.10126.1.5.4.2.2
NAME 'x509certificateHolder'
AUXILIARY
MAY (x509certificateLocation))

Changes in Draft –01(cont´d)

- Deleted ";binary" in examples (??)
- Included new section
  - Comparision with component matching approach
- Some minor changes
  - in wording
  - section titles
  - other editorial changes

Proposed new changes for next draft version

- Some valuable input from the list
- Thanks to Russ, Kurt and David

Abstract x509certificate object class

objectclass (1.3.6.1.4.1.10126.x..x..x

NAME 'x509certificate' **ABSTRACT**

MUST (x509serialNumber $ x509signatureAlgorithm
$ x509issuer $ x509validityNotBefore $ x509validityNotAfter
$ PublicKeyInfoAlgorithm)

MAY (mail $ x509authorityKeyIdentifier $ x509authorityCertIssuer
$ x509authorityCertSerialNumber $ x509subjectKeyIdentifier
$ x509keyUsage $ x509policyInformationIdentifier
$ x509subjectAltNameRfc822Name $ x509subjectAltNameDnsName
$ x509subjectAltNameDirectoryName $ x509subjectAltNameURI
$ x509subjectAltNameIpAddress $ x509subjectAltNameRegisteredID
$ x509isssuerAltNameRfc822Name $ x509isssuerAltNameDnsName
$ x509isssuerAltNameDirectoryName $ x509isssuerAltNameURI
$ x509isssuerAltNameIpAddress $ x509isssuerAltNameRegisteredID
$ x509extKeyUsage $ **x509FullcRLDistributionPoint**
$ **x509certHolder**))

New structural objectclasses

attributetype (x.x.x.x NAME '**x509userCert**'
EQUALITY certificateExactMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.8 **SINGLE-VALUE**)

attributetype (x.x.x.x NAME '**x509cACert**'
EQUALITY certificateExactMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.8 **SINGLE-VALUE**)

objectclass (x.x.x.x NAME '**x509userCertificate**'
SUP x509certificate
MUST x509userCert MAY x509subject)

objectclass (x.x.x.x NAME '**x509cACertificate**'
SUP x509certificate
MUST x509cACert $ x509subject)

Thus no more additional rules needed

- Following is now defined within the schema:
  - Entries MUST also have one of the two auxiliary object classes:
    - "pkiUser"
    - "pkiCA"
  - This way the entry will contain the binary certificate in on of the two attributes:
    - "userCertificate"
    - "caCertficate"

Open issues

- Support for implementations that can‘t do multi-valued RDNs
  - include a third name form with yet another naming attribute x509serialIssuer?
  - x509SerialIssuer=x509SerialNumber\3D12345\2Co\3D someCA\2Cc\3Dsomecountry,ou=somedepartment, o=someorg,c=somecountry
- Include some more clarifying language
  - 2: redundancy, consistency, transition
  - 2: CIP
  - 4.1.6: x509subject vs. x509subjectAltNames
  - 4.1.7: x509subjectPublicKeyInfoAlgorithm
  - 4.3.1: x509certLocation (missing)

Open Issues (contd.)

- “;binary”
- Bug in examples
- Include a use case chapter
- Include IANA consideration
- Make this part of PKIX work
- Publish as proposed or experimental RFC

Future work

- Include attributes for Qualified certificates (RFC 3039)
- New draft on CRLs
- New draft on Attribute Certificates