INTERNET-DRAFT
Document: draft-ietf-crisp-lw-asn-00.txt
Expires: January, 2003
Category: Experimental

Eric A. Hall
July 2002

Defining and Locating Autonomous System Numbers using the Internet Resource Query Service

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC 2026.

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.

1. Abstract

This document defines LDAP schema and searching rules for autonomous system numbers, in support of the Internet Resource Query Service described in [ldap-whois].

2. Definitions and Terminology

This document unites, enhances and clarifies several pre-existing technologies. Readers are expected to be familiar with the following specifications:

RFC 2247 - Using Domains in LDAP/X.500 DNs

RFC 2251 - Lightweight Directory Access Protocol (v3)

RFC 2252 - Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions.

RFC 2254 - The String Representation of LDAP Search Filters

[ir-dir-req] - <draft-newton-ir-dir-requirements-00.txt> - Internet Registry Directory Requirements

[ldap-whois] - <draft-ietf-crisp-lw-core-00.txt> - The Internet Resource Query Service and the Internet Resource Schema

The following abbreviations are used throughout this document:

DIT (Directory Information Tree) - A DIT is a contained branch of the LDAP namespace, having a root of a particular distinguished name. "dc=example,dc=com" is used throughout this document as one DIT, with many example entries being stored in this DIT.

DN (Distinguished Name) - A distinguished name provides a unique identifier for an entry, through the use of a multi-level naming syntax. Entries are named according to their location relevant to the root of their containing DIT. For example, "cn=inetResources,dc=example,dc=com" is a DN which uniquely identifies the "inetResources" entry within the "dc=example,dc=com" DIT.

RDN (Relative DN) - An RDN provides a locally-scoped unique identifier for an entry. A complete, globally-unique DN is formed by concatenating the RDNs of an entry together. For example, "cn=admins,cn=inetResources,dc=example,dc=com" consists of two RDNs ("cn=admins" and "cn=inetResources") within the "dc=example,dc=com" DIT. RDNs are typically only referenced within their local scope.

OID (Object Identifier) - An OID is a globally-unique, concatenated set of integers which provide a kind of "serial number" to attributes, object classes, syntaxes and other schema elements.

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 RFC 2119.

3. The inetAsNumber Object Class

The inetAsNumber object class is a structural object class which provides administrative information about a specific autonomous system (AS) number. AS numbers are used to identify routing domains, allowing multiple discontiguous IPv4 and IPv6 network blocks to be referenced with a single, globally-unique identifier.

3.1. Naming syntax

The naming syntax for AS number entries MUST follow the form of "cn=<inetAsNumberSyntax>,cn=inetResources,<dc-DIT>". Each AS number which is managed as a discrete LDAP-WHOIS network resource MUST have a dedicated entry in each of the DITs which provide public LDAP-WHOIS data regarding that autonomous system.

The inetAsNumberSyntax component of an entry is subject to DN rules, although the inetAsNumberSyntax is also used for search and compare operations, and is therefore subject to specific syntax rules. The AS number syntax uses the decimal equivalent of a 16-bit autonomous system number, with the non-affective leading zeroes removed. An augmented BNF for this syntax is as follows:

inetAsNumberSyntax = decimal value between "0" and "65535" inclusive, with the non-affective leading zeroes removed

For example, an entry for AS number "1" from the "dc=arin,dc=net" DIT would have a DN of "cn=1,cn=inetResources,dc=arin,dc=net", while an entry for AS number "65535" from the same DIT would have a DN of "cn=65535,cn=inetResources,dc=arin,dc=net".

3.2. Schema definition

AS number entries MUST exist with the top, inetResources and inetAsNumber object classes defined. If an entry exists as a referral, the entry MUST also be defined with the referral object class, in addition to the above requirements.

The inetAsNumber object class is a structural object class which is subordinate to the inetResources object class, and which MUST be treated as a container class capable of holding additional subordinate entries. The inetAsNumber object class has no mandatory attributes, although it does have several optional attributes.

The inetAsNumber object class defines attributes which are specific to autonomous systems and their associated routing domains, such as the delegation date, and the status of the delegation. The inetAsNumber object class is subordinate to the inetResources object class, so it inherits those attributes as well.

Some of the inetAsNumber object class attributes define contact-related referrals which provide LDAP URLs that refer to inetOrgPerson entries, and these entries will need to be queried separately if detailed information about a particular contact is required. The contact attribute values follow the same rules as the labeledURI attribute defined in RFC 2079, with additional restrictions described in [ldap-whois].

The various ModifiedBy and ModifiedDate attributes SHOULD be treated as operational attributes. Their values SHOULD be filled in automatically by the database management application, and SHOULD NOT be returned except when explicitly requested.

The network-specific attributes MUST only contain network addresses which are directly associated with the AS number, and MUST use the largest superior prefix delegated to those networks (using the inetIpv4NetworkSyntax and inetIpv6NetworkSyntax rules); these attributes MUST NOT contain host or subnet addresses which are subordinate to another value which is already listed, and these attributes MUST NOT contain network addresses of networks which are associated with any other AS number.

The schema definition for the inetAsNumber object class is as follows:

```
inetAsNumber
( 1.3.6.1.4.1.7161.1.4.0 NAME 'inetAsNumber' DESC 'Autonomous
   system attributes.' SUP inetResources STRUCTURAL MAY (
   inetAsnDelegationStatus $ inetAsnDelegationDate $
   inetAsnDelegationModifiedDate $
   inetAsnDelegationModifiedBy $ inetAsnContacts $
   inetAsnContactsModifiedBy $ inetAsnContactsModifiedDate $
   inetAsnRoutingContacts $ inetAsnRoutingContactsModifiedBy
   $ inetAsnRoutingContactsModifiedDate ) )
```

The attributes from the inetIpv4Network object class are described below:

```
inetAsnContacts
( 1.3.6.1.4.1.7161.1.4.2 NAME 'inetAsnContacts' DESC
   'Contacts for reporting problems with this routing
   domain.' EQUALITY caseExactMatch SYNTAX
   1.3.6.1.4.1.1466.115.121.1.15 )

inetAsnContactsModifiedBy
( 1.3.6.1.4.1.7161.1.4.3 NAME 'inetAsnContactsModifiedBy'
   DESC 'Person who last modified the inetAsnContacts
   attribute.' EQUALITY distinguishedNameMatch SYNTAX
   1.3.6.1.4.1.1466.115.121.1.12 SINGLE-VALUE USAGE
   distributedOperation )

inetAsnContactsModifiedDate
( 1.3.6.1.4.1.7161.1.4.4 NAME 'inetAsnContactsModifiedDate'
   DESC 'Last modification date of the inetAsnContacts
   attribute.' EQUALITY generalizedTimeMatch ORDERING
   generalizedTimeOrderingMatch SYNTAX
   1.3.6.1.4.1.1466.115.121.1.24 SINGLE-VALUE USAGE
   distributedOperation )

inetAsnDelegationDate
( 1.3.6.1.4.1.7161.1.4.5 NAME 'inetAsnDelegationDate' DESC
   'Date of original delegation.' EQUALITY
   generalizedTimeMatch ORDERING generalizedTimeOrderingMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.24 SINGLE-VALUE )
```

```
inetAsnDelegationModifiedBy
( 1.3.6.1.4.1.7161.1.4.6 NAME 'inetAsnDelegationModifiedBy'
   DESC 'Person who last modified the inetAsnDelegationStatus
   attribute.' EQUALITY distinguishedNameMatch SYNTAX
   1.3.6.1.4.1.1466.115.121.1.12 SINGLE-VALUE USAGE
   distributedOperation )

inetAsnDelegationModifiedDate
( 1.3.6.1.4.1.7161.1.4.7 NAME 'inetAsnDelegationModifiedDate'
   DESC 'Last modification date of the
   inetAsnDelegationStatus attribute.' EQUALITY
   generalizedTimeMatch ORDERING generalizedTimeOrderingMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.24 SINGLE-VALUE USAGE
   distributedOperation )

inetAsnDelegationStatus
( 1.3.6.1.4.1.7161.1.4.8 NAME 'inetAsnDelegationStatus' DESC
   'Current delegation status code for this AS number.'
   EQUALITY numericStringMatch SYNTAX
   1.3.6.1.4.1.1466.115.121.1.27{2} SINGLE-VALUE )
```

NOTE: In an effort to facilitate internationalization and programmatic processing, the current status of a delegation is identified by a 16-bit integer. The values and status mapping is as follows:

0	Reserved delegation (permanently inactive)
1	Assigned and active (normal state)
2	Assigned but not yet active (new delegation)
3	Assigned but on hold (disputed)
4	Assignment revoked (database purge pending)

Additional values for the inetIpv6DelegationStatus attribute are reserved for future use, and are to be administered by IANA. Note that there is no status code for "unassigned"; unassigned entries SHOULD NOT exist, and SHOULD NOT be returned as answers.

```
inetAsnRoutingContacts
( 1.3.6.1.4.1.7161.1.4.9 NAME 'inetAsnRoutingContacts' DESC
   'Contacts for routing issues with this IPv4 network.'
   EQUALITY caseExactMatch SYNTAX
   1.3.6.1.4.1.1466.115.121.1.15 )
```

```
inetAsnRoutingContactsModifiedBy
( 1.3.6.1.4.1.7161.1.4.10 NAME
   'inetAsnRoutingContactsModifiedBy' DESC 'Person who last
   modified the inetAsnRoutingContacts attribute.' EQUALITY
   distinguishedNameMatch SYNTAX
   1.3.6.1.4.1.1466.115.121.1.12 SINGLE-VALUE USAGE
   distributedOperation )

inetAsnRoutingContactsModifiedDate
( 1.3.6.1.4.1.7161.1.4.11 NAME
   'inetAsnRoutingContactsModifiedDate' DESC 'Last
   modification date of the inetAsnRoutingContacts
   attribute.' EQUALITY generalizedTimeMatch ORDERING
   generalizedTimeOrderingMatch SYNTAX
   1.3.6.1.4.1.1466.115.121.1.24 SINGLE-VALUE USAGE
   distributedOperation )
```

The inetAsNumberSyntax syntax is as follows:

```
inetAsNumberSyntax
( 1.3.6.1.4.1.7161.1.4.1 NAME 'inetAsNumberSyntax' DESC 'An
   autonomous system number.' )
```

3.3. Example

An example of the inetAsNumber object class is shown in Figure 1 below, with attributes from the inetResources object class also being used to provide administrative contacts. This data is a result of a query which was sent to the LDAP servers associated with the "arin.net" domain.

```
cn=65535,cn=inetResources,dc=arin,dc=net
[top object class]
[inetResources object class]
[inetAsNumber object class]
|
+-attribute: description
| value: "The example.net network"
|
+-attribute: inetAsnContacts
| value: "ldap://ldap.example.com/cn=hostmaster,ou=admins,
|           dc=example,dc=net"
|
+-attribute: inetGeneralContacts
  value: "ldap://ldap.example.com/cn=admins,ou=admins,
            dc=example,dc=net"
```

Figure 1: The inetAsNumber delegation entry for AS 65535.

4. inetAsNumber equalityMatch

The inetAsNumber object class can be searched using relatively simple equalityMatch filters.

In order to ensure that all of the relevant entries (including any referrals) are found, the search filters for these resources MUST specify two distinct elements: the object class of the resource being queried, and the naming element of the resource specified as a distinguished name attribute.

For example, a query for "(&(objectclass=inetAsNumber)(cn:dn:1))" with a search base of "cn=inetResources,dc=example,dc=com" would find all of the inetAsNumber object class entries associated with AS number "1" in the LDAP-WHOIS branch of "dc=example,dc=com".

Response entries MAY be fully-developed entries, or MAY be referrals generated from entries which have the referral object class defined. Any attribute values which are received MUST be displayed by the client. If a subordinate reference referral is received, the client MUST restart the query, using the provided data as the new search base. If any continuation reference referrals are received, the client SHOULD start new queries for each reference, and append the output of those queries to the original query's output.

5. Security Considerations

This document describes an application of the LDAPv3 protocol, and as such it inherits the security considerations associated with LDAPv3, as described in section 7 of RFC 2251.

By nature, LDAP is a read-write protocol, while the legacy WHOIS service has always been a read-only service. As such, there are significant risks associated with allowing unintended updates by unauthorized third-parties. Moreover, allowing the LDAP-WHOIS service to update the underlying delegation databases could result in network resources being stolen from their lawful operators. For example, if the LDAP front-end had update access to a domain delegation database, a malicious third-party could theoretically take ownership of that domain by exploiting an authentication weakness, thereby causing ownership of the domain to be changed to another party. For this reason, it is imperative that the LDAP-WHOIS service not be allowed to make critical modifications to delegated resources without ensuring that all possible precautions have been taken.

The query processing models described in this document make use of DNS lookups in order to locate the LDAP servers associated with a particular resource. DNS is susceptible to certain attacks and forgeries which may be used to redirect clients to LDAP servers which are not authoritative for the resource in question.

Some operators may choose to purposefully provide misleading or erroneous information in an effort to avoid responsibility for bad behavior. In addition, there are likely to be sporadic operator errors which will result in confusing or erroneous answers.

This document provides multiple query models which will cause the same query to be answered by different servers (one would be processed by a delegation entity, while another would be processed by an operational entity). As a result, each of the servers may provide different information, depending upon the query type that was originally selected.

For all of the reasons listed above, it is essential that applications and end-users not make critical decisions based on the information provided by the LDAP-WHOIS service without having reason to believe the veracity of the information. Users should limit unknown or untrusted information to routine purposes.

Finally, there are physical security issues associated with any service which provides physical addressing and delivery information. Although organizations are generally encouraged to provide as much information as they feel comfortable with, no information is required.

6. IANA Considerations

This document defines an application of the LDAPv3 protocol rather than a new Internet application protocol. As such, there are no protocol-related IANA considerations.

However, this document does define several LDAP schema elements, including object classes, attributes, syntaxes and extensibleMatch filters, and these elements should be assigned OID values from the IANA branch, rather than being assigned from a particular enterprise branch.

Finally, this document also describes several instances where public DNS and LDAP servers are queried. It is expected that IANA will establish and maintain these LDAP servers (and the necessary DNS SRV domain names and resource records) required for this service to operate. This includes providing SRV resource records in the generic TLDs and the root domain, and also includes administering the referenced LDAP servers.

7. Author's Addresses

Eric A. Hall
ehall@ehsco.com

8. References

RFC 2247 - Using Domains in LDAP/X.500 DNs

RFC 2251 - Lightweight Directory Access Protocol (v3)

RFC 2252 - Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions.

RFC 2254 - The String Representation of LDAP Search Filters

[ir-dir-req] - <draft-newton-ir-dir-requirements-00.txt> - Internet Registry Directory Requirements

[ldap-whois] - <draft-ietf-crisp-lw-core-00.txt> - The Internet Resource Query Service and the Internet Resource Schema

9. Acknowledgments

Portions of this work were funded by Network Solutions, Inc.