INTERNET-DRAFT
Document: draft-ietf-crisp-firs-ipv4-01.txt
Expires: December, 2003
Category: Experimental

Eric A. Hall

May 2003

Defining and Locating IPv4 Address Blocks in the Federated Internet Registry Service

Status of this Memo

Copyright Notice

Abstract

This document defines LDAP schema and searching rules for IPv4 address blocks, in support of the Federated Internet Registry Service (FIRS) described in [FIRS-ARCH] and [FIRS-CORE].

Table of Contents

1. Introduction

This specification defines the naming syntax, object classes, attributes, matching filters, and query processing rules for storing and locating IPv4 address blocks in the FIRS service. Refer to [FIRS-ARCH] for information on the FIRS architecture and [FIRS-CORE] for the schema definitions and rules which govern the FIRS service as a whole.

Note that reverse-lookup DNS domains for IPv4 address blocks are managed as DNS domain entries in [FIRS-DNS]. These are entirely different network resources, and should not be confused with IPv4 address block entries.

The definitions in this specification are intended to be used with FIRS. Their usage outside of FIRS is not prohibited, but any such usage is beyond this specification's scope of authority.

2. Prerequisites and Terminology

The complete set of specifications in the FIRS collection cumulative define a structured and distributed information service using LDAPv3 for the data-formatting and transport functions. This specification should be read in the context of the complete set of specifications, which currently include the following:

draft-ietf-crisp-firs-arch-01, "The Federated Internet Registry Service: Architecture and Implementation" [FIRS-ARCH]

draft-ietf-crisp-firs-core-01, "The Federated Internet Registry Service: Core Elements" [FIRS-CORE]

draft-ietf-crisp-firs-dns-01, "Defining and Locating DNS Domains in the Federated Internet Registry Service" [FIRS-DNS]

draft-ietf-crisp-firs-dnsrr-01, "Defining and Locating DNS Resource Records in the Federated Internet Registry Service" [FIRS-DNSRR]

draft-ietf-crisp-firs-contact-01, "Defining and Locating Contact Persons in the Federated Internet Registry Service" [FIRS-CONTCT]

draft-ietf-crisp-firs-asn-01, "Defining and Locating Autonomous System Numbers in the Federated Internet Registry Service" [FIRS-ASN]

draft-ietf-crisp-firs-ipv4-01, "Defining and Locating IPv4 Address Blocks in the Federated Internet Registry Service" (this document) [FIRS-IPV4]

draft-ietf-crisp-firs-ipv6-01, "Defining and Locating IPv6 Address Blocks in the Federated Internet Registry Service" [FIRS-IPV6]

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. Naming Syntax

The naming syntax for IPv4 address blocks in FIRS MUST follow the form of "cn=<inetIpv4NetworkSyntax>,cn=inetResources,<partition>", where <inetIpv4NetworkSyntax > is the IPv4 address block resource, and where <partition> is a sequence of domainComponent relative distinguished names which identifies the scope of authority for the selected directory partition.

The inetIpv4NetworkSyntax rules use the traditional "dotted-quad" notation, where each of four sub-components provide a decimal

value that represents one octet from a 32-bit IPv4 address, with the sub-components being separated by a full-stop (period) character, and with the four-part sequence being followed by a "/" character and a three-digit decimal "prefix" value.

Entries which use the inetIpv4NetworkSyntax MUST use the starting address from a range of inclusive addresses, and MUST use CIDR prefix notation. In this manner, it is possible to create an inetIpv4Network entry for a range of addresses of any size (including a single host).

The leading zeroes from each octet MUST be removed before the value is stored or used in a query. Octets which have a value of zero MUST be represented by the single-digit numeric value of "0".

If an input string does not match this syntax, a FIRS-aware application MAY attempt to manipulate the input string to form a valid value. For example, if a user enters a traditional IPv4 address without specifying a prefix value, the application MAY append "/32" to the end of the input string to form a valid assertion value. Similarly, if a user provides an octal or hexadecimal value, the client MAY attempt to convert the input string to the traditional dotted-quad IPv4 address notation.

An augmented BNF for this syntax is as follows:

```
inetIpv4NetworkSyntax = inetIpv4Octet "." inetIpv4Octet "."
  inetIpv4Octet "." inetIpv4Octet "/" inetIpv4Prefix

inetIpv4Octet = decimal value between "0" and "255"
  inclusive, with the non-affective leading zeroes removed

inetIpv4Prefix = decimal value between "1" and "32"
  inclusive, with the non-affective leading zeroes removed
```

The schema definition for inetIpv4NetworkSyntax is as follows:

```
inetIpv4NetworkSyntax
( 1.3.6.1.4.1.7161.1.2.1 NAME 'inetIpv4NetworkSyntax' DESC
  'An IPv4 address and prefix.' )
```

For example, an IPv4 address block with a range of addresses between "10.0.0.0" and "10.0.255.255" inclusive would be written as "cn=10.0.0.0/16", while a host address of "192.0.2.14" would be written as "cn=192.0.2.14/32".

Note that the entry name of "cn=0.0.0.0/0" encompasses the entire IPv4 address space.

Note that the use of "/" is illegal as data in URLs, and MUST be escaped before it is stored in a URL as data.

4. Object Classes and Attributes

IPv4 address block entries in FIRS MUST use the inetIpv4Network object class, in addition to the mandatory object classes defined in [FIRS-CORE]. IPv4 address block entries MUST be treated as containers capable of holding subordinate entries. If an entry exists as a referral source, the entry MUST also be defined with the referral object class, in addition to the above requirements.

The inetIpv4Network object class is a structural object class which is subordinate to the inetResources object class. The inetIpv4Network object class has no mandatory attributes, although it does have several optional attributes. The inetIpv4Network object class also inherits the attributes defined in the inetResources object class, including the "cn" naming attribute.

The schema definition for the inetIpv4Network object class is as follows:

```
inetIpv4Network
( 1.3.6.1.4.1.7161.1.2.0 NAME 'inetIpv4Network' DESC 'IPv4
  network attributes.' SUP inetResources STRUCTURAL MAY (
  inetIpv4DelegationStatus $ inetIpv4DelegationDate $
  inetIpv4Registrar $ inetIpv4Registry $ inetIpv4Contacts $
  inetIpv4RoutingContacts $ ) )
```

The attributes from the inetIpv4Network object class are described below:

```
inetIpv4Contacts
( 1.3.6.1.4.1.7161.1.2.2 NAME 'inetIpv4Contacts' DESC
  'Contacts for general administrative issues concerning this
  IPv4 address block.' EQUALITY caseIgnoreMatch SYNTAX
  inetContactSyntax )
```

inetIpv4DelegationDate
(1.3.6.1.4.1.7161.1.2.3 NAME 'inetIpv4DelegationDate' DESC
'Date this IPv4 address block was delegated.' EQUALITY
generalizedTimeMatch ORDERING generalizedTimeOrderingMatch
SYNTAX generalizedTime SINGLE-VALUE)

inetIpv4DelegationStatus
(1.3.6.1.4.1.7161.1.2.4 NAME 'inetIpv4DelegationStatus' DESC
'Delegation status of this IPv4 address block.' EQUALITY
numericStringMatch SYNTAX numericString{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 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.

inetIpv4Registrar
(1.3.6.1.4.1.7161.1.2.5 NAME 'inetIpv4Registrar' DESC
'Registrar who delegated this IPv4 address block.' EQUALITY
caseIgnoreMatch SYNTAX directoryString)

NOTE: The inetIpv4Registrar attribute uses a URL to indicate the registrar who delegated the address block. The attribute structure is identical to the labeledURI attribute, as defined in [RFC2798], including the URL and textual comments. The data can refer to any valid URL.

inetIpv4Registry
(1.3.6.1.4.1.7161.1.2.6 NAME 'inetIpv4Registry' DESC
'Registry where this IPv4 address block is managed.'
EQUALITY caseIgnoreMatch SYNTAX directoryString)

NOTE: The inetIpv4Registry attribute uses a URL to indicate the registry who is ultimately responsible for the address block. The attribute structure is identical to the labeledURI attribute, as defined in [RFC2798], including the URL and textual comments. The data can refer to any valid URL.

```
inetIpv4RoutingContacts
( 1.3.6.1.4.1.7161.1.2.7 NAME 'inetIpv4RoutingContacts' DESC
  'Contacts for routing-related problems with this IPv4
  address block.' EQUALITY caseExactMatch SYNTAX
  inetContactSyntax )
```

An example of the inetIpv4Network object class is shown in Figure 1 below. The example includes attributes from the inetIpv4Network, inetResources, and inetAssociatedResources object classes.

```
cn=192.0.2.0/24,cn=inetResources,dc=arin,dc=net
[top object class]
[inetResources object class]
[inetIpv4Network object class]
[inetAssociatedResources object class]
|
+-attribute: description
| value: "Example Hosting's IPv4 address block"
|
+-attribute: inetIpv4Contacts
| value: "hostmaster@example.com"
|
+-attribute: inetAssociatedAsNumbers
| value: "65535"
|
+-attribute: inetIpv4Registrar
  value: "http://www.arin.net/ (ARIN)"
```

Figure 1: The entry for the 192.0.2.0/24 address block in the dc=arin,dc=net partition.

5. Query Processing Rules

Queries for IPv4 address blocks have several special requirements, as discussed in the following sections.

Refer to [FIRS-CORE] for general information about FIRS queries.

5.1. Query Pre-Processing

Clients MUST ensure that the query input is normalized according to the rules specified in section 3 before the input is used as the assertion value to the resulting LDAP query.

The authoritative partition for an IPv4 address block is determined by mapping the normalized input to an associated reverse-lookup DNS domain name, and then mapping the resulting DNS domain name to a sequence of domainComponent labels.

The least-significant octet MUST include the subnet prefix in this mapping process, except in those cases where the address falls on an eight-bit boundary. In those cases where the address block specifies a 32-bit host address, the subnet prefix MUST be stripped from the input during the mapping process. In those cases where the address block specifies a legacy "address class", the least-significant octet and subnet prefix MUST both be stripped from the input during the mapping process. These steps are necessary in order to ensure that the reverse-pointer delegations in the public DNS are correctly matched to the authoritative partitions (note that these rules only apply to the mapping process by which an authoritative partition is constructed, and does not apply to the process by which the entry-specific relative distinguished name is constructed).

For example, a host-specific IPv4 address block of "192.0.2.14/32" would be mapped to the reverse-lookup DNS domain name of "14.2.0.192.in-addr.arpa." which would in turn be mapped to "dc=14,dc=2,dc=0,dc=192,dc=in-addr,dc=arpa". Meanwhile, the "Class C" block of "192.0.2.0/24" would be mapped to the reverse-lookup DNS domain name of "2.0.192.in-addr.arpa." which would in turn be mapped to "dc=2,dc=0,dc=192,dc=in-addr,dc=arpa". Finally, a classless IPv4 address block of "192.0.2.0/20" would be mapped to the reverse-lookup DNS domain name of "0/14.2.0.192.in-addr.arpa" which would in turn be mapped to the fully-qualified distinguished name of "dc=0/14,dc=2,dc=0,dc=192,dc=in-addr,dc=arpa".

5.2. Query Bootstrapping

FIRS clients MUST use the top-down bootstrap model by default for IPv4 address block queries. As such, the search base for default queries would be set to "dc=arpa" rather than being set to the fully-qualified distinguished name of the authoritative partition.

FIRS clients MAY use the targeted or bottom-up bootstrap models for queries if necessary or desirable. However, it is not likely that entries will be found for all IPv4 address block resources using these models. As such, the top-down bootstrap model will be the most useful in most cases, and MUST be used by default.

5.3. LDAP Matching

FIRS clients MUST use the inetIpv4NetworkMatch extensible matching filter in LDAP searches for IPv4 address block entries.

The inetIpv4NetworkMatch filter provides an identifier and search string format which collectively inform a queried server that a specific IPv4 address should be searched for, and that any matching inetIpv4network object class entries should be returned.

The inetIpv4NetworkMatch extensibleMatch filter is defined as follows:

```
inetIpv4NetworkMatch
( 1.3.6.1.4.1.7161.1.2.8 NAME 'inetIpv4NetworkMatch' SYNTAX
  inetIpv4NetworkSyntax )
```

The assertion value MUST be a normalized IPv4 address, using the inetIpv4NetworkSyntax defined in section 3.

A FIRS server MUST compare the assertion value against the RDN of all entries in the inetResources container of the partition specified in the search base which have an object class of inetIpv4Network. Any entry with an object class of inetIpv4Network and with a relative distinguished name which clearly encompasses the IPv4 address provided in the assertion value MUST be returned. Entries which do not clearly encompass the queried address MUST NOT be returned. Entries which do not have an object class of inetIpv4Network MUST NOT be returned.

In order to ensure that all of the relevant entries are found (including any referrals), the search filters for these resources MUST specify the inetIpv4Network object class along with the search criteria. For example, "(&(objectclass=inetIpv4Network) (1.3.6.1.4.1.7161.1.2.8:=192.0.2.0/24))" with a search base of "cn=inetResources,dc=arin,dc=net" would find all of the inetIpv4Network object class entries which were superior to the "192.0.2.0/24" address block in the "dc=arin,dc=net" partition.

Note that the entry name of "cn=0.0.0.0/0" encompasses the entire IPv4 address space. When used in conjunction with referrals, this entry MAY be used to redirect all inetIpv4NetworkMatch queries to another partition for subsequent processing.

The matching filters defined in this specification MUST be supported by FIRS clients and servers. FIRS servers MAY support additional sub-string filters, soundex filters, or any other filters they wish (these may be required to support generic LDAP clients), although FIRS clients MUST NOT expect any additional filters to be available.

5.4. Example Query

The following example assumes that the user has specified "192.0.2.14/32" as the query value:

a. Normalize the input, which is "192.0.2.14/32" in this case.

b. Determine the authoritative partition.

   1. Map the input sequence to the reverse-lookup domain name, which is "14.2.0.192.in-addr.arpa" in this case.

   2. Map the domain name to an authoritative partition, which is "dc=14,dc=2,dc=0,dc=192,dc=in-addr,dc=arpa" in this case. By default, queries for IPv4 address blocks use the top-down model, meaning that the right-most relative distinguished name of "dc=arpa" will be used as the authoritative partition.

c. Determine the search base for the query, which will be "cn=inetResources,dc=arpa" if the defaults are used.

d. Initiate a DNS lookup for the SRV resource records associated with "_ldap._tcp.arpa." For the purpose of this example, assume that this lookup succeeds, with the DNS response message indicating that "firs.iana.org" is the preferred LDAP server.

e. Submit an LDAPv3 query to the specified server, using "(&(objectClass=inetIpv4Network) (1.3.6.1.4.1.7161.1.2.8:=192.0.2.14/32))" as the matching filter, "cn=inetResources,dc=arpa" as the search base, and the global query defaults defined in [FIRS-CORE].

f. Assume that the queried server returns a continuation reference referral which points to "ldap:///cn=inetResources,dc=arin,dc=net". The distinguished name element of "cn=inetResources,dc=arin,dc=net" will be used as the new search base, while "dc=arin,dc=net" will be used as the new authoritative partition.

g. Initiate a DNS lookup for the SRV resource records associated with "_ldap._tcp. arin.net." For the purpose of this example, assume that this lookup succeeds, with the DNS response message indicating that "firs.arin.net" is the preferred LDAP server.

h. Submit an LDAPv3 query to the specified server, using "(&(objectClass=inetIpv4Network) (1.3.6.1.4.1.7161.1.2.8:=192.0.2.14/32)" as the matching filter, "cn=inetResources,dc=arin,dc=net" as the search base, and the global query defaults defined in [FIRS-CORE].

i. Assume that no other referrals are received. Display the answer data which has been received and exit the query.

6. Security Considerations

Security considerations are discussed in [FIRS-ARCH].

7. IANA Considerations

This specification uses the "dc=arpa" directory partition by default, with the expectation that FIRS-capable LDAP servers will be established, with this partition containing IPv4-specific entries which will provide referrals to the appropriate registrar's partitions. It is further expected that IANA will oversee the creation and management of the ARPA domain's LDAP SRV resource records, the "dc=arpa" LDAP partition, and the necessary LDAP servers.

The inetIpv4DelegationStatus attribute uses numeric code values. It is expected that IANA will manage the assignment of these values.

Additional IANA considerations are discussed in [FIRS-ARCH].

8. Author's Addresses

Eric A. Hall
ehall@ehsco.com

9. Normative References

[RFC2247] Kille, S., Wahl, M., Grimstad, A., Huber, R., and Sataluri, S. "Using Domains in LDAP/X.500 DNs", RFC 2247, January 1998.

[RFC2251] Wahl, M., Howes, T., and Kille, S. "Lightweight Directory Access Protocol (v3)", RFC 2251, December 1997.

[RFC2252] Wahl, M., Coulbeck, A., Howes, T., and Kille, S. "Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions", RFC 2252, December 1997.

[RFC2254] Howes, T. "The String Representation of LDAP Search Filters", RFC 2254, December 1997.

[FIRS-ARCH] Hall, E. "The Federated Internet Registry Service: Architecture and Implementation Guide", draft-ietf-crisp-firs-arch-01, May 2003.

[FIRS-ASN] Hall, E. "Defining and Locating Autonomous System Numbers in the Federated Internet Registry Service", draft-ietf-crisp-firs-asn-01, May 2003.

[FIRS-CONTCT] Hall, E. "Defining and Locating Contact Persons in the Federated Internet Registry Service", draft-ietf-crisp-firs-contact-01, May 2003.

[FIRS-CORE] Hall, E. "The Federated Internet Registry Service: Core Elements", draft-ietf-crisp-firs-core-01, May 2003.

[FIRS-DNS] Hall, E. "Defining and Locating DNS Domains in the Federated Internet Registry Service", draft-ietf-crisp-firs-dns-01, May 2003.

[FIRS-DNSRR] Hall, E. "Defining and Locating DNS Resource Records in the Federated Internet Registry Service", draft-ietf-crisp-firs-dnsrr-01, May 2003.

[FIRS-IPV4] Hall, E. "Defining and Locating IPv4 Address Blocks in the Federated Internet Registry Service", draft-ietf-crisp-firs-ipv4-01, May 2003.

[FIRS-IPV6] Hall, E. "Defining and Locating IPv6 Address Blocks in the Federated Internet Registry Service", draft-ietf-crisp-firs-ipv6-01, May 2003.

10. Acknowledgments

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

Portions of this document were funded by Verisign Labs.

The first version of this specification was co-authored by Andrew Newton of Verisign Labs, and subsequent versions continue to be developed with his active participation.

11. Changes from Previous Versions

draft-ietf-crisp-firs-ipv4-01:

* Several clarifications and corrections have been made.

draft-ietf-crisp-firs-ipv4-00:

* Restructured the document set.

* "Attribute references" have been eliminated from the specification. All referential attributes now provide

actual data instead of URL pointers to data. Clients that wish to retrieve these values will need to start new queries using the data values instead of URLs.

* The attribute-specific operational attributes have been eliminated as unnecessary.

* The inetIpv4Registrar and inetIpv4Registry attributes were added.

* Several attributes had their OIDs changed. NOTE THAT THIS IS AN INTERNET DRAFT, AND THAT THE OIDS ARE SUBJECT TO ADDITIONAL CHANGES AS THIS DOCUMENT IS EDITED.

* Several typographical errors have been fixed.

* Some unnecessary text has been removed.

12. Full Copyright Statement

THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.