Network Working Group
Request for Comments: 3795
Category: Informational

R. Sofia
P. Nesser, II
Nesser & Nesser Consulting

June 2004

Survey of IPv4 Addresses in Currently Deployed IETF Application Area Standards Track and Experimental Documents

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Abstract

This document describes IPv4 addressing dependencies in an attempt to clarify the necessary steps in re-designing and re-implementing specifications to become network address independent, or at least, to dually support IPv4 and IPv6. This transition requires several interim steps, one of them being the evolution of current IPv4 dependent specifications to a format independent of the type of IP addressing schema used. Hence, it is hoped that specifications will be re-designed and re-implemented to become network address independent, or at least to dually support IPv4 and IPv6.

To achieve that step, it is necessary to survey and document all IPv4 dependencies experienced by current standards (Full, Draft, and Proposed) as well as Experimental RFCs. Hence, this document describes IPv4 addressing dependencies that deployed IETF Application Area documented Standards may experience.

Table of Contents

1. Introduction

The exhaustive documentation of IPv4 addresses usage in currently deployed IETF documented standards has now been broken into seven documents conforming to current IETF main areas, i.e., Applications, Internet, Operations and Management, Routing, Sub-IP, and Transport. A general overview of the documentation, as well as followed methodology and historical perspective can be found in [1]. This document represents one of the seven blocks, and its scope is limited to surveying possible IPv4 dependencies in IETF Application Area documented Standards.

2. Document Organization

The remainder sections are organized as follows. Sections 3, 4, 5, and 6 describe, respectively, the raw analysis of Internet Standards [2]:

Full, Draft, and Proposed Standards, and Experimental RFCs. For each section, standards are analysed by their RFC number, in sequential order, i.e., from RFC 1 to RFC 3200. Exceptions to this are some RFCs above RFC 3200. They have been included, given that they obsoleted RFCs within the range 1-3200. Also, the comments presented for each RFC are raw in their nature, i.e., each RFC is simply analysed in terms of possible IPv4 addressing dependencies. Finally, Section 7 presents a global overview of the data described in the previous sections, and suggests possible future steps.

3. Full Standards

Internet Full Standards have attained the highest level of maturity on the standards track process. They are commonly referred to as "Standards", and represent fully technical mature specifications that are widely implemented and used throughout the Internet.

3.1. RFC854: Telnet Protocol Specifications

There are no IPv4 dependencies in this specification.

3.2. RFC 855: Telnet Option Specifications

There are no IPv4 dependencies in this specification.

3.3. RFC 856: Binary Transmission Telnet Option

There are no IPv4 dependencies in this specification.

3.4. RFC 857: Echo Telnet Option

There are no IPv4 dependencies in this specification.

3.5. RFC 858: Suppress Go Ahead Telnet Option

There are no IPv4 dependencies in this specification.

3.6. RFC 859: Status Telnet Option

There are no IPv4 dependencies in this specification.

3.7. RFC 860: Timing Mark Telnet Option

There are no IPv4 dependencies in this specification.

3.8. RFC 861: Extended Options List Telnet Option

There are no IPv4 dependencies in this specification.

3.9. RFC 862: Echo Protocol

There are no IPv4 dependencies in this specification.

3.10. RFC 863: Discard Protocol

There are no IPv4 dependencies in this specification.

3.11. RFC 864: Character Generator Protocol

There are no IPv4 dependencies in this specification.

3.12. RFC 865: Quote of the Day Protocol

There are no IPv4 dependencies in this specification.

3.13. RFC 866: Active Users Protocol

There are no IPv4 dependencies in this specification.

3.14. RFC 867: Daytime Protocol

There are no IPv4 dependencies in this specification.

3.15. RFC 868: Time Server Protocol

There are no IPv4 dependencies in this specification.

3.16. RFC 959: File Transfer Protocol

Section 4.1.2 (TRANSFER PARAMETER COMMANDS) describes the port command using the following format:

```
"A port command would be:
    PORT h1,h2,h3,h4,p1,p2
    where h1 is the high order 8 bits of the internet host
    address."
```

This is a clear reference to an IPv4 address. In sections 4.2.1 and 4.2.2, on reply codes, the code:

```
"227 Entering Passive Mode (h1,h2,h3,h4,p1,p2)"
```

also needs to be reworked for IPv6 addressing. Also, Section 5.3.2 (FTP COMMAND ARGUMENTS) contains:

```
"<host-number> ::= <number>,<number>,<number>,<number>
 <port-number> ::= <number>,<number>
 <number> ::= any decimal integer 1 through 255"
```

This needs to be solved to transition to IPv6.

3.17. RFC 1350: Trivial File Transfer Protocol

There are no IPv4 dependencies in this specification.

3.18. RFC 1870: SMTP Service Extension for Message Size Declaration

There are no IPv4 dependencies in this specification.

3.19. RFC 1939: Post Office Protocol - Version 3

There are no IPv4 dependencies in this specification.

3.20. RFC 2920: SMTP Service Extension for Command Pipelining

There are no IPv4 dependencies in this specification.

4. Draft Standards

Draft Standards is the nomenclature given to specifications that are on the penultimate maturity level of the IETF standards track process. They are considered to be final specifications, which may only experience changes to solve specific problems found. A specification is only considered to be a Draft Standard if there are at least two known independent and interoperable implementations. Hence, Draft Standards are usually quite mature and widely used.

4.1. RFC 954: NICNAME/WHOIS

There are no IPv4 dependencies in this specification.

4.2. RFC 1184: Telnet Linemode Option

There are no IPv4 dependencies in this specification.

4.3. RFC 1288: The Finger User Information Protocol

There are no IPv4 dependencies in this specification.

4.4. RFC 1305: Network Time Protocol (Version 3) Specification, Implementation

Section 3.2.1 (Common Variables) provides the following variable definitions:

> "Peer Address (peer.peeraddr, pkt.peeraddr), Peer Port (peer.peerport, pkt.peerport): These are the 32-bit Internet address and 16-bit port number of the peer.

Host Address (peer.hostaddr, pkt.hostaddr), Host Port (peer.hostport, pkt.hostport): These are the 32-bit Internet address and 16-bit port number of the host. They are included among the state variables to support multi-homing."

Section 3.4.3 (Receive Procedure) defines the following procedure:

"The source and destination Internet addresses and ports in the IP and UDP headers are matched to the correct peer. If there is no match a new instantiation of the protocol machine is created and the association mobilized."

Section 3.6 (Access Control Issues) proposes a simple authentication scheme in the following way:

"If a more comprehensive trust model is required, the design can be based on an access-control list with each entry consisting of a 32-bit Internet address, 32-bit mask and three-bit mode. If the logical AND of the source address (pkt.peeraddr) and the mask in an entry matches the corresponding address in the entry and the mode (pkt.mode) matches the mode in the entry, the access is allowed; otherwise an ICMP error message is returned to the requestor. Through appropriate choice of mask, it is possible to restrict requests by mode to individual addresses, a particular subnet or net addresses, or have no restriction at all. The access-control list would then serve as a filter controlling which peers could create associations."

Appendix B Section 3 (B.3 Commands) defines the following command:

"Set Trap Address/Port (6): The command association identifier, status and data fields are ignored. The address and port number for subsequent trap messages are taken from the source address and port of the control message itself. The initial trap counter for trap response messages is taken from the sequence field of the command. The response association identifier, status and data fields are not significant. Implementations should include sanity timeouts which prevent trap transmissions if the monitoring program does not renew this information after a lengthy interval."

The address clearly assumes the IPv4 version. Also, there are numerous places in sample code and in algorithms that use the above mentioned variables. It seems that there is no reason to modify the actual protocol. A small number of textual changes and an update to implementations, so they can understand both IPv4 and IPv6 addresses, will suffice to have a NTP version that works on both network layer protocols.

4.5. RFC 1575: An Echo Function for CLNP (ISO 8473)

There are no IPv4 dependencies in this specification.

4.6. RFC 1652: SMTP Service Extension for 8bit-MIME Transport

There are no IPv4 dependencies in this specification.

4.7. RFC 1832: eXternal Data Representation Standard

There are no IPv4 dependencies in this specification.

4.8. RFC 2045: Multipurpose Internet Mail Extensions (MIME), Part One: Format of Internet Message Bodies

There are no IPv4 dependencies in this specification.

4.9. RFC 2046: MIME, Part Two: Media Types

There are no IPv4 dependencies in this specification.

4.10. RFC 2047: MIME, Part Three: Message Header Extensions for Non-ASCII Text

There are no IPv4 dependencies in this specification.

4.11. RFC 2049: MIME Part Five: Conformance Criteria and Examples

There are no IPv4 dependencies in this specification.

4.12. RFC 2279: UTF-8, a transformation format of ISO 10646

There are no IPv4 dependencies in this specification.

4.13. RFC 2347: TFTP Option Extension

There are no IPv4 dependencies in this specification.

4.14. RFC 2348: TFTP Blocksize Option

Section "Blocksize Option Specification" gives the following example:

"For example:

```
+-------+--------+---+--------+---+--------+---+--------+---+
|   1   | foobar | 0 | octet  | 0 | blksize| 0 |  1428  | 0 |
+-------+--------+---+--------+---+--------+---+--------+---+
```

is a Read Request, for the file named "foobar", in octet (binary) transfer mode, with a block size of 1428 octets (Ethernet MTU, less the TFTP, UDP and IP header lengths)."

Clearly, the given blocksize example would not work with IPv6 header sizes, but it has no practical implications, since larger blocksizes are also available.

4.15. RFC 2349: TFTP Timeout Interval and Transfer Size Options

There are no IPv4 dependencies in this specification.

4.16. RFC 2355: TN3270 Enhancements

There are no IPv4 dependencies in this specification.

4.17. RFC 2396: Uniform Resource Identifiers (URI): Generic Syntax

Section 3.2.2. (Server-based Naming Authority) states:

"The host is a domain name of a network host, or its IPv4 address as a set of four decimal digit groups separated by ".". Literal IPv6 addresses are not supported.
...
Note: A suitable representation for including a literal IPv6 address as the host part of a URL is desired, but has not yet been determined or implemented in practice."

4.18. RFC 2616: Hypertext Transfer Protocol HTTP/1.1

Section 3.2.2 (http URL) states:

"The "http" scheme is used to locate network resources via the HTTP protocol. This section defines the scheme-specific syntax and semantics for http URLs.

```
http_URL = "http:" "//" host [ ":" port ] [ abs_path [ "?" query ]]
```

If the port is empty or not given, port 80 is assumed. The semantics are that the identified resource is located at the server listening for TCP connections on that port of that host, and the Request-URI for the resource is abs_path (section 5.1.2). The use of IP addresses in URLs SHOULD be avoided whenever possible (see RFC 1900 [24])."

The text is version neutral, but it is unclear whether individual implementations will support IPv6 addresses. In fact, the use of the ":"separator in IPv6 addresses will cause misinterpretation when parsing URI's. There are other discussions regarding a server recognizing its own IP addresses, spoofing DNS/IP address combinations, as well as issues regarding multiple HTTP servers running on a single IP interface. Again, the text is version neutral, but clearly, such statements represent implementation issues.

4.19. RFC 3191: Minimal GSTN address format in Internet Mail

There are no IPv4 dependencies in this specification.

4.20. RFC 3192: Minimal FAX address format in Internet Mail

There are no IPv4 dependencies in this specification.

4.21. RFC 3282: Content Language Headers

There are no IPv4 dependencies in this specification.

4.22. RFC 3461: Simple Mail Transfer Protocol (SMTP) Service Extension for Delivery Status Notifications

There are no IPv4 dependencies in this specification.

4.23. RFC 3462: The Multipart/Report Content Type for the Reporting of Mail System Administrative Messages

There are no IPv4 dependencies in this specification.

4.24. RFC 3463: Enhanced Mail System Status Codes

There are no IPv4 dependencies in this specification.

4.25. RFC 3464: An Extensible Message Format for Delivery Status Notifications

There are no IPv4 dependencies in this specification.

5. Proposed Standards

Proposed Standards represent initial level documents in the IETF standards track process. They are stable in terms of design, but do not require the existence of implementations. In several cases, these specifications are simply proposed as solid technical ideas, to be analysed by the Internet community, but are never implemented or advanced in the IETF standards process.

5.1. RFC 698: Telnet extended ASCII option

There are no IPv4 dependencies in this specification.

5.2. RFC 726: Remote Controlled Transmission and Echoing Telnet option

There are no IPv4 dependencies in this specification.

5.3. RFC 727: Telnet logout option

There are no IPv4 dependencies in this specification.

5.4. RFC 735: Revised Telnet byte macro option

There are no IPv4 dependencies in this specification.

5.5. RFC 736: Telnet SUPDUP option

There are no IPv4 dependencies in this specification.

5.6. RFC 749: Telnet SUPDUP-Output option

There are no IPv4 dependencies in this specification.

5.7. RFC 779: Telnet send-location option

There are no IPv4 dependencies in this specification.

5.8. RFC 885: Telnet end of record option

There are no IPv4 dependencies in this specification.

5.9. RFC 927: TACACS user identification Telnet option

There are no IPv4 dependencies in this specification.

5.10. RFC 933: Output marking Telnet option

There are no IPv4 dependencies in this specification.

5.11. RFC 946: Telnet terminal location number option

Section "TTYLOC Number" states:

> "The TTYLOC number is a 64-bit number composed of two (2) 32-bit numbers: The 32-bit official ARPA Internet host address (may be any one of the addresses for multi-homed hosts) and a 32-bit number representing the terminal on the specified host. The host address of [0.0.0.0] is defined to be "unknown", the terminal number of FFFFFFFF (hex, r or-1 in decimal) is defined to be "unknown" and the terminal number of FFFFFFFE (hex, or -2 in decimal) is defined to be "detached" for processes that are not attached to a terminal."

The clear reference to 32-bit numbers, and to the use of literal addresses in the form [0.0.0.0] is clearly an IPv4-dependency. Thus, the text above needs to be re-written.

5.12. RFC 977: Network News Transfer Protocol

There are no IPv4 dependencies in this specification.

5.13. RFC 1041: Telnet 3270 regime option

There are no IPv4 dependencies in this specification.

5.14. RFC 1043: Telnet Data Entry Terminal option: DODIIS implementation

There are no IPv4 dependencies in this specification.

5.15. RFC 1053: Telnet X.3 PAD option

There are no IPv4 dependencies in this specification.

5.16. RFC 1073: Telnet window size option

There are no IPv4 dependencies in this specification.

5.17. RFC 1079: Telnet terminal speed option

There are no IPv4 dependencies in this specification.

5.18. RFC 1091: Telnet terminal-type option

There are no IPv4 dependencies in this specification.

5.19. RFC 1096: Telnet X display location option

There are no IPv4 dependencies in this specification.

5.20. RFC 1274: The COSINE and Internet X.500 Schema

There are no IPv4 dependencies in this specification.

5.21. RFC 1276: Replication and Distributed Operations extensions to provide an Internet Directory using X.500

There are no IPv4 dependencies in this specification.

5.22. RFC 1314: A File Format for the Exchange of Images in the Internet

There are no IPv4 dependencies in this specification.

5.23. RFC 1328: X.400 1988 to 1984 downgrading

There are no IPv4 dependencies in this specification.

5.24. RFC 1372: Telnet Remote Flow Control Option

There are no IPv4 dependencies in this specification.

5.25. RFC 1415: FTP-FTAM Gateway Specification

Since this document defines a gateway for interaction between FTAM and FTP, the only possible IPv4 dependencies are associated with FTP, which has already been investigated above, in section 3.16.

5.26. RFC 1494: Equivalences between 1988 X.400 and RFC-822 Message Bodies

There are no IPv4 dependencies in this specification.

5.27. RFC 1496: Rules for downgrading messages from X.400/88 to X.400/84 when MIME content-types are present in the messages

There are no IPv4 dependencies in this specification.

5.28. RFC 1502: X.400 Use of Extended Character Sets

There are no IPv4 dependencies in this specification.

5.29. RFC 1572: Telnet Environment Option

There are no IPv4 dependencies in this specification.

5.30. RFC 1648: Postmaster Convention for X.400 Operations

There are no IPv4 dependencies in this specification.

5.31. RFC 1738: Uniform Resource Locators

Section 3.1. (Common Internet Scheme Syntax) states:

"host
The fully qualified domain name of a network host, or its IP address as a set of four decimal digit groups separated by ".". Fully qualified domain names take the form as described in Section 3.5 of RFC 1034 [13] and Section 2.1 of RFC 1123 [4]: a sequence of domain labels separated by ".", each domain label starting and ending with an alphanumerical character and possibly also containing "-" characters. The rightmost domain label will never start with a digit, though, which syntactically distinguishes all domain names from the IP addresses."

Clearly, this is only valid when using IPv4 addresses. Later in Section 5. (BNF for specific URL schemes), there is the following text:

```
"; URL schemeparts for ip based protocols:

 ip-schemepart  = "//" login [ "/" urlpath ]

 login          = [ user [ ":" password ] "@" ] hostport
 hostport       = host [ ":" port ]
 host           = hostname | hostnumber"
```

Again, this also has implications in terms of IP-version neutrality.

5.32. RFC 1740: MIME Encapsulation of Macintosh Files - MacMIME

There are no IPv4 dependencies in this specification.

5.33. RFC 1767: MIME Encapsulation of EDI Objects

There are no IPv4 dependencies in this specification.

5.34. RFC 1808: Relative Uniform Resource Locators

There are no IPv4 dependencies in this specification.

5.35. RFC 1835: Architecture of the WHOIS++ service

There are no IPv4 dependencies in this specification.

5.36. RFC 1913: Architecture of the WHOIS++ Index Service

Section 6.5. (Query referral) makes the following statement:

> "When referrals are included in the body of a response to a query, each referral is listed in a separate SERVER-TO-ASK block as shown below.

```
# SERVER-TO-ASK
 Version-number: // version number of index software, used to insure
                 // compatibility
 Body-of-Query: // the original query goes here
 Server-Handle: // WHOIS++ handle of the referred server
 Host-Name: // DNS name or IP address of the referred server
 Port-Number: // Port number to which to connect, if different from the
               // WHOIS++ port number"
```

The syntax used does not present specific IPv4 dependencies, but implementations should be modified to check, in incoming packets, which IP version was used by the original request, so they can determine whether or not to return an IPv6 address.

5.37. RFC 1914: How to Interact with a Whois++ Mesh

Section 4 (Caching) states the following:

> "A client can cache all information it gets from a server for some time. For example records, IP-addresses of Whois++ servers, the Directory of Services server etc.
>
> A client can itself choose for how long it should cache the information.
>
> The IP-address of the Directory of Services server might not change for a day or two, and neither might any other information."

Also, subsection 4.1. (Caching a Whois++ servers hostname) contains:

"An example of cached information that might change is the cached hostname, IP-address and portnumber which a client gets back in a servers-to-ask response. That information is cached in the server since the last poll, which might occurred several weeks ago. Therefore, when such a connection fails, the client should fall back to use the serverhandle instead, which means that it contacts the Directory of Services server and queries for a server with that serverhandle. By doing this, the client should always get the last known hostname.

An algorithm for this might be:

```
response := servers-to-ask response from server A
IP-address := find ip-address for response.hostname in DNS
connect to ip-address at port response.portnumber
if connection fails {
   connect to Directory of Services server
   query for host with serverhandle response.serverhandle
   response := response from Directory of Services server
   IP-address := find ip-address for response.hostname in DNS
   connect to ip-address at port response.portnumber
   if connection fails {
       exit with error message
   }
 }
 Query this new server"
```

The paragraph does not contain IPv4 specific syntax. Hence, IPv6 compliance will be implementation dependent.

5.38. RFC 1985: SMTP Service Extension for Remote Message Queue Starting

There are no IPv4 dependencies in this specification.

5.39. RFC 2017: Definition of the URL MIME External-Body Access-Type

There are no IPv4 dependencies in this specification.

5.40. RFC 2034: SMTP Service Extension for Returning Enhanced Error Codes

There are no IPv4 dependencies in this specification.

5.41. RFC 2056: Uniform Resource Locators for Z39.50

There are no IPv4 dependencies in this specification.

5.42. RFC 2077: The Model Primary Content Type for Multipurpose Internet Mail Extensions

There are no IPv4 dependencies in this specification.

5.43. RFC 2079: Definition of an X.500 Attribute Type and an Object Class to Hold Uniform Resource Identifiers (URIs)

There are no IPv4 dependencies in this specification.

5.44. RFC 2086: IMAP4 ACL extension

There are no IPv4 dependencies in this specification.

5.45. RFC 2087: IMAP4 QUOTA extension

There are no IPv4 dependencies in this specification.

5.46. RFC 2088: IMAP4 non-synchronizing literals

There are no IPv4 dependencies in this specification.

5.47. RFC 2122: VEMMI URL Specification

Section 3 (Description of the VEMMI scheme) states:

> "The VEMMI URL scheme is used to designate multimedia interactive services conforming to the VEMMI standard (ITU/T T.107 and ETS 300 709).
>
> A VEMMI URL takes the form:
>
> ```
> vemmi://<host>:<port>/<vemmiservice>;
> <attribute>=<value>
> ```
>
> as specified in Section 3.1. of RFC 1738. If :<port> is omitted, the port defaults to 575 (client software may choose to ignore the optional port number in order to increase security). The <vemmiservice> part is optional and may be omitted."

IPv4 dependencies may relate to the possibility of the <host> portion containing an IPv4 address, as defined in RFC 1738 (see section 5.31. above). Once the problem is solved in the context of RFC 1738, this issue will be automatically solved.

5.48. RFC 2141: URN Syntax

There are no IPv4 dependencies in this specification.

5.49. RFC 2142: Mailbox Names for Common Services, Roles and Functions

There are no IPv4 dependencies in this specification.

5.50. RFC 2156: MIXER (Mime Internet X.400 Enhanced Relay): Mapping between X.400 and RFC 822/MIME

There are no IPv4 dependencies in this specification.

5.51. RFC 2157: Mapping between X.400 and RFC-822/MIME Message Bodies

There are no IPv4 dependencies in this specification.

5.52. RFC 2158: X.400 Image Body Parts

There are no IPv4 dependencies in this specification.

5.53. RFC 2159: A MIME Body Part for FAX

There are no IPv4 dependencies in this specification.

5.54. RFC 2160: Carrying PostScript in X.400 and MIME

There are no IPv4 dependencies in this specification.

5.55. RFC 2163: Using the Internet DNS to Distribute MIXER Conformant Global Address Mapping

There are no IPv4 dependencies in this specification.

5.56. RFC 2164: Use of an X.500/LDAP directory to support MIXER address mapping

There are no IPv4 dependencies in this specification.

5.57. RFC 2165: Service Location Protocol

Section 7. (Service Type Request Message Format) and Section 9. (Service Registration Message Format) have an 80-bit field from addr-spec (see below) which cannot support IPv6 addresses. Also, Section 20.1. (Previous Responders’ Address Specification) states:

"The previous responders’ Address Specification is specified as

```
  &lt;Previous Responders’ Address Specification&gt; ::=
         &lt;addr-spec&gt; |
         &lt;addr-spec&gt;, &lt;Previous Responders’ Address Specification&gt;
```

i.e., a list separated by commas with no intervening white space. The Address Specification is the address of the Directory Agent or Service Agent which supplied the previous response. The format for Address Specifications in Service Location is defined in section 20.4. The comma delimiter is required between each <addr-spec>. The use of dotted decimal IP address notation should only be used in environments which have no Domain Name Service."

Later, in Section 20.4. (Address Specification in Service Location) there is also the following reference to addr-spec:

"The address specification used in Service Location is:

```
<addr-spec> ::= [<user>:<password>@]<host>[:<port>]

  <host>      ::= Fully qualified domain name |
                  dotted decimal IP address notation
```

When no Domain Name Server is available, SAs and DAs must use dotted decimal conventions for IP addresses. Otherwise, it is preferable to use a fully qualified domain name wherever possible as renumbering of host addresses will make IP addresses invalid over time."

The whole Section 21. (Protocol Requirements) defines the requirements for each of the elements of this protocol. Several IPv4 statements are made, but the syntax used is sufficiently neutral to apply to the use of IPv6.

Section 22. (Configurable Parameters and Default Values) states:

"There are several configuration parameters for Service Location. Default values are chosen to allow protocol operation without the need for selection of these configuration parameters, but other values may be selected by the site administrator. The configurable parameters will allow an implementation of Service Location to be more useful in a variety of scenarios.

Multicast vs. Broadcast
All Service Location entities must use multicast by default. The ability to use broadcast messages must be configurable for UAs and SAs. Broadcast messages are to be used in environments where not all Service Location entities have hardware or software which supports multicast.

Multicast Radius
Multicast requests should be sent to all subnets in a site. The default multicast radius for a site is 32. This value must be configurable. The value for the site’s multicast TTL may be obtained from DHCP using an option which is currently unassigned."

Once again, nothing here precludes IPv6, Section 23.

(Non-configurable Parameters) states:

"IP Port number for unicast requests to Directory Agents:

UDP and TCP Port Number: 427

Multicast Addresses

Service Location General Multicast Address: 224.0.1.22
Directory Agent Discovery Multicast Address: 224.0.1.35

A range of 1024 contiguous multicast addresses for use as Service Specific Discovery Multicast Addresses will be assigned by IANA."

Clearly, the statements above require specifications related to the use of IPv6 multicast addresses with equivalent functionality.

5.58. RFC 2177: IMAP4 IDLE command

There are no IPv4 dependencies in this specification.

5.59. RFC 2183: Communicating Presentation Information in Internet Messages: The Content-Disposition Header Field

There are no IPv4 dependencies in this specification.

5.60. RFC 2192: IMAP URL Scheme

The specification has IPv4 dependencies, as RFC 1738, which is integral to the document, is not IPv6 aware.

5.61. RFC 2193: IMAP4 Mailbox Referrals

Section 6. (Formal Syntax) presents the following statement:

```
"referral_response_code = "[" "REFERRAL" 1*(SPACE <url>) "]"; See
[RFC-1738] for <url> definition"
```

The above presents dependencies on RFC 1738 URL definitions, which have already been mentioned in this document, section 5.31.

5.62. RFC 2218: A Common Schema for the Internet White Pages Service

There are no IPv4 dependencies in this specification.

5.63. RFC 2221: IMAP4 Login Referrals

Section 4.1. (LOGIN and AUTHENTICATE Referrals) provides the following example:

```
"Example:  C: A001 LOGIN MIKE PASSWORD
           S: A001 NO [REFERRAL IMAP://MIKE@SERVER2/] Specified
                   user is invalid on this server. Try SERVER2."
```

Even though the syntax "user@SERVER2" is presented often, there are no specifications related to the format of "SERVER2". Hence, it is up to individual implementations to determine acceptable values for the hostname. This may or not include explicit IPv6 addresses.

5.64. RFC 2227: Simple Hit-Metering and Usage-Limiting for HTTP

There are no IPv4 dependencies in this specification.

5.65. RFC 2231: MIME Parameter Value and Encoded Word Extensions: Character Sets, Languages, and Continuations

There are no IPv4 dependencies in this specification.

5.66. RFC 2234: Augmented BNF for Syntax Specifications: ABNF

There are no IPv4 dependencies in this specification.

5.67. RFC 2244: Application Configuration Access Protocol

There are no IPv4 dependencies in this specification.

5.68. RFC 2247: Using Domains in LDAP/X.500 Distinguished Names

There are no IPv4 dependencies in this specification.

5.69. RFC 2251: Lightweight Directory Access Protocol (v3)

There are no IPv4 dependencies in this specification.

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

There are no IPv4 dependencies in this specification.

5.71. RFC 2253: Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names

Section 7.1. (Disclosure) states:

> "Distinguished Names typically consist of descriptive information about the entries they name, which can be people, organizations, devices or other real-world objects. This frequently includes some of the following kinds of information:
>
> - the common name of the object (i.e., a person's full name)
> - an email or TCP/IP address
> - its physical location (country, locality, city, street address)
> - organizational attributes (such as department name or affiliation)"

This section requires the caveat "Without putting any limitations on the version of the IP address.", to avoid ambiguity in terms of IP version.

5.72. RFC 2254: The String Representation of LDAP Search Filters

There are no IPv4 dependencies in this specification.

5.73. RFC 2255: The LDAP URL Format

The specification has IPv4 dependencies, as RFC 1738, which is integral to the document, is not IPv6 aware.

5.74. RFC 2256: A Summary of the X.500(96) User Schema for use with LDAPv3

There are no IPv4 dependencies in this specification.

5.75. RFC 2293: Representing Tables and Subtrees in the X.500 Directory

There are no IPv4 dependencies in this specification.

5.76. RFC 2294: Representing the O/R Address hierarchy in the X.500 Directory Information Tree

There are no IPv4 dependencies in this specification.

5.77. RFC 2298: An Extensible Message Format for Message Disposition Notifications

There are no IPv4 dependencies in this specification.

5.78. RFC 2301: File Format for Internet Fax

There are no IPv4 dependencies in this specification.

5.79. RFC 2305: A Simple Mode of Facsimile Using Internet Mail

There are no IPv4 dependencies in this specification.

5.80. RFC 2334: Server Cache Synchronization Protocol

Appendix B, part 2.0.1 (Mandatory Common Part) states:

> "Cache Key
> This is a database lookup key that uniquely identifies a piece of data which the originator of a CSA Record wishes to synchronize with its peers for a given "Protocol ID/Server Group ID" pair. This key will generally be a small opaque byte string which SCSP will associate with a given piece of data in a cache. Thus, for example, an originator might assign a particular 4 byte string to the binding of an IP address with that of an ATM address. Generally speaking, the originating server of a CSA record is responsible for generating a Cache Key for every element of data that the given server originates and which the server wishes to synchronize with its peers in the SG."

The statement above is simply meant as an example. Hence, any IPv4 possible dependency of this protocol is an implementation issue.

5.81. RFC 2342: IMAP4 Namespace

There are no IPv4 dependencies in this specification.

5.82. RFC 2359: IMAP4 UIDPLUS extension

There are no IPv4 dependencies in this specification.

5.83. RFC 2368: The mailto URL scheme

There are no IPv4 dependencies in this specification.

5.84. RFC 2369: The Use of URLs as Meta-Syntax for Core Mail List Commands and their Transport through Message Header Fields

There are no IPv4 dependencies in this specification.

5.85. RFC 2371: Transaction Internet Protocol Version 3.0

In section 7. (TIP Transaction Manager Identification and Connection Establishment):

"The <hostport> component comprises:

```
<host>[:<port>]
```

where <host> is either a <dns name> or an <ip address>; and <port> is a decimal number specifying the port at which the transaction manager (or proxy) is listening for requests to establish TIP connections. If the port number is omitted, the standard TIP port number (3372) is used.

A <dns name> is a standard name, acceptable to the domain name service. It must be sufficiently qualified to be useful to the receiver of the command.

An <ip address> is an IP address, in the usual form: four decimal numbers separated by period characters."

This section has to be re-written to become IP-version neutral. Besides adding a reference to the use of IPv6 addresses, the "host" field should only be defined as a "dns name". However, if the use of literal IP addresses is to be included, the format specified in RFC 2372 has to be followed.

Later in section 8. (TIP Uniform Resource Locators):

"A TIP URL takes the form:

```
tip://<transaction manager address>?<transaction string>
```

where <transaction manager address> identifies the TIP transaction manager (as defined in Section 7 above); and <transaction string> specifies a transaction identifier, which may take one of two forms (standard or non-standard):

i. "urn:" <NID> ":" <NSS>

A standard transaction identifier, conforming to the proposed Internet Standard for Uniform Resource Names (URNs), as specified by RFC2141; where <NID> is the Namespace Identifier, and <NSS> is the Namespace Specific String. The Namespace ID determines the syntactic interpretation of the Namespace Specific String. The Namespace Specific String is a sequence of characters representing a transaction identifier (as defined by <NID>). The rules for the contents of these fields are specified by [6] (valid characters, encoding, etc.).

This format of <transaction string> may be used to express global transaction identifiers in terms of standard representations. Examples for <NID> might be <iso> or <xopen>. e.g.,

```
tip://123.123.123.123/?urn:xopen:xid
```

Note that Namespace Ids require registration. See [7] for details on how to do this."

There are other references in section 8, regarding the use of literal IP addresses. Therefore, this section also needs to be re-written, and special care should be taken to avoid the use of IP (either IPv4 or IPv6) literal addresses. However, if such use is exemplified, the format specified in RFC 2732 has to be respected.

5.86. RFC 2384: POP URL Scheme

Section 3. (POP Scheme) states:

"A POP URL is of the general form:

```
pop://<user>;auth=<auth>@<host>:<port>
```

Where <user>, <host>, and <port> are as defined in RFC 1738, and some or all of the elements, except "pop://" and <host>, may be omitted."

RFC 1738 (please refer to section 5.31) has a potential IPv4 limitation. Hence, RFC 2384 will only be IPv6 compliant when RFC 1738 becomes properly updated.

5.87. RFC 2387: The MIME Multipart/Related Content-type

There are no IPv4 dependencies in this specification.

5.88. RFC 2388: Returning Values from Forms: multipart/form-data

There are no IPv4 dependencies in this specification.

5.89. RFC 2389: Feature negotiation mechanism for the File Transfer Protocol

There are no IPv4 dependencies in this specification.

5.90. RFC 2392: Content-ID and Message-ID Uniform Resource Locators (CIDMID-URL)

There are no IPv4 dependencies in this specification.

5.91. RFC 2397: The "data" URL scheme

There are no IPv4 dependencies in this specification.

5.92. RFC 2421: Voice Profile for Internet Mail - version 2

There are no IPv4 dependencies in this specification.

5.93. RFC 2422: Toll Quality Voice - 32 kbit/s ADPCM MIME Sub-type Registration

There are no IPv4 dependencies in this specification.

5.94. RFC 2423: VPIM Voice Message MIME Sub-type Registration

There are no IPv4 dependencies in this specification.

5.95. RFC 2424: Content Duration MIME Header Definition

There are no IPv4 dependencies in this specification.

5.96. RFC 2425: A MIME Content-Type for Directory Information

There are no IPv4 dependencies in this specification.

5.97. RFC 2426: vCard MIME Directory Profile

There are no IPv4 dependencies in this specification.

5.98. RFC 2428: FTP Extensions for IPv6 and NATs

This RFC documents an IPv6 extension and hence, it is not considered in the context of the current discussion.

5.99. RFC 2445: Internet Calendaring and Scheduling Core Object Specification (iCalendar)

Section 4.8.4.7 (Unique Identifier) states:

"Property Name: UID

Purpose: This property defines the persistent, globally unique identifier for the calendar component.

Value Type: TEXT

Property Parameters: Non-standard property parameters can be specified on this property.

Conformance: The property MUST be specified in the "VEVENT", "VTODO", "VJOURNAL" or "VFREEBUSY" calendar components.

Description: The UID itself MUST be a globally unique identifier. The generator of the identifier MUST guarantee that the identifier is unique. There are several algorithms that can be used to accomplish this. The identifier is RECOMMENDED to be the identical syntax to the [RFC 822] addr-spec. A good method to assure uniqueness is to put the domain name or a domain literal IP address of the host on which the identifier was created on the right hand side of the "@", and on the left hand side, put a combination of the current calendar date and time of day (i.e., formatted in as a DATE-TIME value) along with some other currently unique (perhaps sequential) identifier available on the system (for example, a process id number). Using a date/time value on the left hand side and a domain name or domain literal on the right hand side makes it possible to guarantee uniqueness since no two hosts should be using the same domain name or IP address at the same time. Though other algorithms will work, it is RECOMMENDED that the right hand side contain some domain identifier (either of the host itself or otherwise) such that the generator of the message identifier can guarantee the uniqueness of the left hand side within the scope of that domain."

Although the above does not explicitly state the use of IPv4 addresses, it addresses the explicit use of RFC 822 (obsoleted by RFC 2822). To become IPv6 compliant it should follow the guidelines for RFC 2822 (see section 5.129).

5.100. RFC 2446: iCalendar Transport-Independent Interoperability Protocol (iTIP) Scheduling Events, BusyTime, To-dos and Journal Entries

There are no IPv4 dependencies in this specification.

5.101. RFC 2447: iCalendar Message-Based Interoperability Protocol (iMIP)

There are no IPv4 dependencies in this specification.

5.102. RFC 2449: POP3 Extension Mechanism

There are no IPv4 dependencies in this specification.

5.103. RFC 2476: Message Submission

This RFC contains several discussions on the usage of IP Address authorization schemes, but it does not limit those addresses to IPv4.

5.104. RFC 2480: Gateways and MIME Security Multiparts

There are no IPv4 dependencies in this specification.

5.105. RFC 2518: HTTP Extensions for Distributed Authoring

There are no IPv4 dependencies in this specification.

5.106. RFC 2530: Indicating Supported Media Features Using Extensions to DSN and MDN

There are no IPv4 dependencies in this specification.

5.107. RFC 2532: Extended Facsimile Using Internet Mail

There are no IPv4 dependencies in this specification.

5.108. RFC 2533: A Syntax for Describing Media Feature Sets

There are no IPv4 dependencies in this specification.

5.109. RFC 2534: Media Features for Display, Print, and Fax

There are no IPv4 dependencies in this specification.

5.110. RFC 2554: SMTP Service Extension for Authentication

There are no IPv4 dependencies in this specification.

5.111. RFC 2557: MIME Encapsulation of Aggregate Documents, such as HTML

There are no IPv4 dependencies in this specification.

5.112. RFC 2589: Lightweight Directory Access Protocol (v3): Extensions for Dynamic Directory Services

There are no IPv4 dependencies in this specification.

5.113. RFC 2595: Using TLS with IMAP, POP3 and ACAP

There are no IPv4 dependencies in this specification.

5.114. RFC 2596: Use of Language Codes in LDAP

There are no IPv4 dependencies in this specification.

5.115. RFC 2608: Service Location Protocol, Version 2

Section 8.1. (Service Request) contains the following:

```
"
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Service Location header (function = SrvRqst = 1)        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      length of <PRList>       |        <PRList> String        \
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   length of <service-type>    |    <service-type> String      \
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    length of <scope-list>     |     <scope-list> String       \
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  length of predicate string   |  Service Request <predicate>  \
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  length of <SLP SPI> string   |       <SLP SPI> String        \
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

   ...
```

<PRList> is the Previous Responder List. This <string-list> contains dotted decimal notation IP (v4) addresses, and is iteratively multicast to obtain all possible results (see Section 6.3). UAs SHOULD implement this discovery algorithm. SAs MUST use this to discover all available DAs in their scope, if they are not already configured with DA addresses by some other means."

And later:

"A SA silently drops all requests which include the SA’s address in the <PRList>. An SA which has multiple network interfaces MUST check if any of the entries in the <PRList> equal any of its interfaces. An entry in the PRList which does not conform to an IPv4 dotted decimal address is ignored: The rest of the <PRList> is processed normally and an error is not returned."

To become IPv6 compliant, this protocol requires a new version.

5.116. RFC 2609: Service Templates and Service: Schemes

Section 2.1. (Service URL Syntax) defines:

"The ABNF for a service: URL is:

```
hostnumber      =   ipv4-number
ipv4-number     =   1*3DIGIT 3("." 1*3DIGIT)"
```

This document presents many other references to hostnumber, which requires an update to support IPv6.

5.117. RFC 2640: Internationalization of the File Transfer Protocol

There are no IPv4 dependencies in this specification.

5.118. RFC 2645: ON-DEMAND MAIL RELAY (ODMR) SMTP with Dynamic IP Addresses

There are no IPv4 dependencies in this specification.

5.119. RFC 2646: The Text/Plain Format Parameter

There are no IPv4 dependencies in this specification.

5.120. RFC 2651: The Architecture of the Common Indexing Protocol (CIP)

There are no IPv4 dependencies in this specification.

5.121. RFC 2652: MIME Object Definitions for the Common Indexing Protocol

There are no IPv4 dependencies in this specification.

5.122. RFC 2653: CIP Transport Protocols

There are no IPv4 dependencies in this specification.

5.123. RFC 2732: Format for Literal IPv6 Addresses in URL's

This document defines an IPv6 specific protocol and hence, it is not discussed in this document.

5.124. RFC 2738: Corrections to "A Syntax for Describing Media Feature Sets"

There are no IPv4 dependencies in this specification.

5.125. RFC 2739: Calendar Attributes for vCard and LDAP

There are no IPv4 dependencies in this specification.

5.126. RFC 2806: URLs for Telephone Calls

There are no IPv4 dependencies in this specification.

5.127. RFC 2821: Simple Mail Transfer Protocol

The specification discusses A records at length, and the MX record handling with the different combinations of A and AAAA records and IPv4/IPv6-only nodes might cause several kinds of failure modes.

5.128. RFC 2822: Internet Message Format

Section 3.4.1 (Addr-spec specification) contains:

> "The domain portion identifies the point to which the mail is delivered. In the dot-atom form, this is interpreted as an Internet domain name (either a host name or a mail exchanger name) as described in [STD3, STD13, STD14]. In the domain-literal form, the domain is interpreted as the literal Internet address of the particular host. In both cases, how addressing is used and how messages are transported to a particular host is covered in the mail transport document [RFC2821]. These mechanisms are outside of the scope of this document.
>
> The local-part portion is a domain dependent string. In addresses, it is simply interpreted on the particular host as a name of a particular mailbox."

Literal IP addresses should be avoided. However, in case they are used, there should be a reference to the format described in RFC 2732.

5.129. RFC 2846: GSTN Address Element Extensions in E-mail Services

There are no IPv4 dependencies in this specification.

5.130. RFC 2849: The LDAP Data Interchange Format (LDIF) - Technical Specification

There are no IPv4 dependencies in this specification.

5.131. RFC 2852: Deliver By SMTP Service Extension

There are no IPv4 dependencies in this specification.

5.132. RFC 2879: Content Feature Schema for Internet Fax (V2)

There are no IPv4 dependencies in this specification.

5.133. RFC 2891: LDAP Control Extension for Server Side Sorting of Search Results

There are no IPv4 dependencies in this specification.

5.134. RFC 2910: Internet Printing Protocol/1.1: Encoding and Transport

There are no IPv4 dependencies in this specification.

5.135. RFC 2911: Internet Printing Protocol/1.1: Model and Semantics

There are no IPv4 dependencies in this specification.

5.136. RFC 2912: Indicating Media Features for MIME Content

There are no IPv4 dependencies in this specification.

5.137. RFC 2913: MIME Content Types in Media Feature Expressions

There are no IPv4 dependencies in this specification.

5.138. RFC 2919: List-Id: A Structured Field and Namespace for the Identification of Mailing Lists

There are no IPv4 dependencies in this specification.

5.139. RFC 2938: Identifying Composite Media Features

There are no IPv4 dependencies in this specification.

5.140. RFC 2965: HTTP State Management Mechanism

This document includes several references to host IP addresses, but there is no explicit mention to a particular protocol version. A caveat similar to "Without putting any limitations on the version of the IP address." should be added, so that there will remain no doubts about possible IPv4 dependencies.

5.141. RFC 2971: IMAP4 ID extension

There are no IPv4 dependencies in this specification.

5.142. RFC 2987: Registration of Charset and Languages Media Features Tags

There are no IPv4 dependencies in this specification.

5.143. RFC 3009: Registration of parityfec MIME types

There are no IPv4 dependencies in this specification.

5.144. RFC 3017: XML DTD for Roaming Access Phone Book

Section 6.2.1. (DNS Server Address) states:

> "The dnsServerAddress element represents the IP address of the Domain Name Service (DNS) server which should be used when connected to this POP.
>
> The address is represented in the form of a string in dotted-decimal notation (e.g., 192.168.101.1).
>
> Syntax:

```
   <!-- Domain Name Server IP address -->
   <!ELEMENT dnsServerAddress (#PCDATA)>
   <!ATTLIST dnsServerAddress
           value NOTATION (IPADR) #IMPLIED>"
```

Additionally, it is stated in Section 6.2.9. (Default Gateway Address):

"The defaulttGatewayAddress element represents the address of the default gateway which should be used when connected to this POP. The address is represented in the form of a string in dotted-decimal notation (e.g., 192.168.101.1).

Syntax:

```
<!-- Default Gateway IP address (in dotted decimal notation) -->
<!ELEMENT defaultGatewayAddress (#PCDATA)>
<!ATTLIST defaultGatewayAddress
        value NOTATION (IPADR) #IMPLIED>"
```

It should be straightforward to implement elements that are IPv6 aware.

5.145. RFC 3023: XML Media Types

There are no IPv4 dependencies in this specification.

5.146. RFC 3028: Sieve: A Mail Filtering Language

There are no IPv4 dependencies in this specification.

5.147. RFC 3030: SMTP Service Extensions for Transmission of Large and Binary MIME Messages

There are no IPv4 dependencies in this specification.

5.148. RFC 3049: TN3270E Service Location and Session Balancing

There are no IPv4 dependencies in this specification.

5.149. RFC 3059: Attribute List Extension for the Service Location Protocol

There are no IPv4 dependencies in this specification.

5.150. RFC 3080: The Blocks Extensible Exchange Protocol Core (BEEP)

There are no IPv4 dependencies in this specification.

5.151. RFC 3081: Mapping the BEEP Core onto TCP

There are no IPv4 dependencies in this specification.

5.152. RFC 3111: Service Location Protocol Modifications for IPv6

This is an IPv6 related document and is not discussed in this document.

5.153. RFC 3302: Tag Image File Format (TIFF) - image/tiff MIME Sub-type Registration

There are no IPv4 dependencies in this specification.

5.154. RFC 3404: Dynamic Delegation Discovery System (DDDS) Part Four: The Uniform Resource Identifiers (URI) Resolution Application

This specification has no explicit dependency on IPv4. However, when referring to the URI format specified in RFC 2396 (see section 4.3. flags, first paragraph), a reference to RFC 2732 should be also added.

5.155. RFC 3501: Internet Message Access Protocol - Version 4rev1

There are no IPv4 dependencies in this specification.

6. Experimental RFCs

Experimental RFCs belong to the category of "non-standard" specifications. This group involves specifications considered "off-track", e.g., specifications that haven’t yet reach an adequate standardization level, or that have been superseded by more recent specifications.

Experimental RFCs represent specifications that are currently part of some research effort, and that are often propriety in nature, or used in limited arenas. They are documented to the Internet community in order to allow potential interoperability or some other potential useful scenario. In a few cases, they are presented as alternatives to the mainstream solution of an acknowledged problem.

6.1. RFC 887: Resource Location Protocol

Section 3.1 (Request Messages) contains:

"<Who-Anywhere-Provides?>
This message parallels the <Who-Provides?> message with the "third-party" variant described above. The confirming host is required to return at least its own IP address (if it provides the named resource) as well as the IP addresses of any other hosts it believes may provide the named resource. The confirming host

though, may never return an IP address for a resource which is the same as an IP address listed with the resource name in the request message. In this case it must treat the resource as if it was unsupported at that IP address and omit it from any reply list.

<Does-Anyone-Provide?>
This message parallels the <Do-You-Provide?> message again with the "third-party" variant described above. As before, the confirming host is required to return its own IP address as well as the IP addresses of any other hosts it believes may provide the named resource and is prohibited from returning the same IP address in the reply resource specifier as was listed in the request resource specifier. As in the <Do-You-Provide?> case and for the same reason, this message also may not be broadcast."

Throughout this section, there are several other references to IP address. To avoid ambiguity, a reference to IPv6 addressing should be added.

Section 4.1. (Resource Lists) presents the following qualifier format:

"In addition, resource specifiers in all <Who-Anywhere-Provides?>, <Does-Anyone-Provide?> and <They-Provide> messages also contain an additional qualifier following the <Protocol-ID>. This qualifier has the format

```
+--------+--------+--------+--------+---//---+
|        |                                    |
|IPLength|          IP-Address-List           |
|        |                                    |
+--------+--------+--------+--------+---//---+
```

where

<IPLength>
is the number of IP addresses containing in the following <IP-Address-List> (the <IP-Address-List> field thus occupies the last 4*<IPLength> octets in its resource specifier). In request messages, this is the maximum number of qualifying addresses which may be included in the corresponding reply resource specifier. Although not particularly useful, it may be 0 and in that case provides no space for qualifying the resource name with IP addresses in the returned specifier. In reply messages, this is the number of qualifying addresses known to provide the resource. It may not exceed the number specified in the corresponding request specifier. This field may not be 0 in a reply message unless it was supplied as 0 in

the request message and the confirming host would have returned one or more IP addresses had any space been provided.

<IP-Address-List>
is a list of four-octet IP addresses used to qualify the resource specifier with respect to those particular addresses. In reply messages, these are the IP addresses of the confirming host (when appropriate) and the addresses of any other hosts known to provide that resource (subject to the list length limitations). In request messages, these are the IP addresses of hosts for which resource information may not be returned. In such messages, these addresses should normally be initialized to some "harmless" value (such as the address of the querying host) unless it is intended to specifically exclude the supplied addresses from consideration in any reply messages."

This section requires re-writing considering the 128-bit length of IPv6 addresses, and will clearly impact implementations.

6.2. RFC 909: Loader Debugger Protocol (LDP)

There are no IPv4 dependencies in this specification.

6.3. RFC 1143: The Q Method of Implementing TELNET Option Negotiation

There are no IPv4 dependencies in this specification.

6.4. RFC 1153: Digest message format (DMF-MAIL)

There are no IPv4 dependencies in this specification.

6.5. RFC 1165: Network Time Protocol (NTP) over the OSI Remote Operations Service

The only dependency this protocol presents is included in Appendix A (ROS Header Format):

```
"ClockIdentifier ::= CHOICE {
                  referenceClock[0] PrintableString,
                  inetaddr[1] OCTET STRING,
                  psapaddr[2] OCTET STRING
  }"
```

6.6. RFC 1176: Interactive Mail Access Protocol: Version 2

There are no IPv4 dependencies in this specification.

6.7. RFC 1204: Message Posting Protocol

There are no IPv4 dependencies in this specification.

6.8. RFC 1235: Coherent File Distribution Protocol

Section "Protocol Specification" provides the following example, for the Initial Handshake:

"The ticket server replies with a "This is Your Ticket" (TIYT) packet containing the ticket. Figure 2 shows the format of this packet.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      'T'      |      'I'      |      'Y'      |      'T'      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            "ticket"                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        BLKSZ (by default 512)                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                              FILSZ                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            IP address of CFDP server (network order)          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   client UDP port# (cfdpcln)  |   server UDP port# (cfdpsrv)  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Fig. 2: "This Is Your Ticket" packet."

This protocol assumes IPv4 multicast, but could be converted to IPv6 multicast with a little effort.

6.9. RFC 1279: X.500 and Domains

This protocol specifies a protocol that assumes IPv4, but does not actually have any limitations which would limit its operation in an IPv6 environment.

6.10. RFC 1312: Message Send Protocol 2

There are no IPv4 dependencies in this specification.

6.11. RFC 1339: Remote Mail Checking Protocol

There are no IPv4 dependencies in this specification.

6.12. RFC 1440: SIFT/UFT: Sender-Initiated/Unsolicited File Transfer

There are no IPv4 dependencies in this specification.

6.13. RFC 1459: Internet Relay Chat Protocol

There are only two specific IPv4 addressing references. The first is presented in Section 6.2. (Command Response):

```
"203     RPL_TRACEUNKNOWN
                 "???? <class> [<client IP address in dot form>]""
```

The second appears in Section 8.12 (Configuration File):

> "In specifying hostnames, both domain names and use of the ’dot’ notation (127.0.0.1) should both be accepted."

After correcting the above, IPv6 support can be added straightforwardly.

6.14. RFC 1465: Routing Coordination for X.400 MHS Services Within a Multi Protocol / Multi Network Environment Table Format V3 for Static Routing

There are no IPv4 dependencies in this specification.

6.15. RFC 1505: Encoding Header Field for Internet Messages

There are no IPv4 dependencies in this specification.

6.16. RFC 1528: Principles of Operation for the TPC.INT Subdomain: Remote Printing -- Technical Procedures

There are no IPv4 dependencies in this specification.

6.17. RFC 1608: Representing IP Information in the X.500 Directory

There are no IPv4 dependencies in this specification.

6.18. RFC 1609: Charting Networks in the X.500 Directory

There are no IPv4 dependencies in this specification.

6.19. RFC 1639: FTP Operation Over Big Address Records

This document defines a method for overcoming FTP IPv4 limitations and is therefore both IPv4 and IPv6 aware.

6.20. RFC 1641: Using Unicode with MIME

There are no IPv4 dependencies in this specification.

6.21. RFC 1756: Remote Write Protocol - Version 1.0

There are no IPv4 dependencies in this specification.

6.22. RFC 1801: MHS use of the X.500 Directory to support MHS Routing

There are no IPv4 dependencies in this specification.

6.23. RFC 1804: Schema Publishing in X.500 Directory

There are no IPv4 dependencies in this specification.

6.24. RFC 1806: Communicating Presentation Information in Internet Messages: The Content-Disposition Header

There are no IPv4 dependencies in this specification.

6.25. RFC 1845: SMTP Service Extension for Checkpoint/Restart

There are no IPv4 dependencies in this specification.

6.26. RFC 1846: SMTP 521 Reply Code

There are no IPv4 dependencies in this specification.

6.27. RFC 1873: Message/External-Body Content-ID Access Type

There are no IPv4 dependencies in this specification.

6.28. RFC 1874: SGML Media Types

There are no IPv4 dependencies in this specification.

6.29. RFC 1986: Experiments with a Simple File Transfer Protocol for Radio Links using Enhanced Trivial File Transfer Protocol

This protocol is IPv4 dependent, as can be seen from the segment presented below, taken from Section 2. (PROTOCOL DESCRIPTION):

```
"Table 3: ETFTP Data Encapsulation

+------------+------------+------------+------------+-----------+
|Ethernet(14)|            |            |ETFTP/      |           |
|SLIP(2)     |IP(20)      |UDP(8)      |NETBLT(24)  |DATA(1448) |
|AX.25(20)   |            |            |            |           |
+------------+------------+------------+------------+-----------+"
```

6.30. RFC 2016: Uniform Resource Agents (URAs)

There are no IPv4 dependencies in this specification.

6.31. RFC 2066: TELNET CHARSET Option

There are no IPv4 dependencies in this specification.

6.32. RFC 2075: IP Echo Host Service

There are no IPv4 dependencies in this specification.

6.33. RFC 2090: TFTP Multicast Option

This protocol is limited to IPv4 multicast. It is expected that a similar functionality could be implemented on top of IPv6 multicast.

6.34. RFC 2120: Managing the X.500 Root Naming Context

There are no IPv4 dependencies in this specification.

6.35. RFC 2161: A MIME Body Part for ODA

There are no IPv4 dependencies in this specification.

6.36. RFC 2162: MaXIM-11 - Mapping between X.400 / Internet mail and Mail-11 mail

There are no IPv4 dependencies in this specification.

6.37. RFC 2169: A Trivial Convention for using HTTP in URN Resolution

There are no IPv4 dependencies in this specification.

6.38. RFC 2217: Telnet Com Port Control Option

There are no IPv4 dependencies in this specification.

6.39. RFC 2295: Transparent Content Negotiation in HTTP

There are no IPv4 dependencies in this specification.

6.40. RFC 2296: HTTP Remote Variant Selection Algorithm RVSA/1.0

There are no IPv4 dependencies in this specification.

6.41. RFC 2307: An Approach for Using LDAP as a Network Information Service

This protocol assumes IPv4 addressing in its schema, as shown in Section 3. (Attribute definitions):

```
"( nisSchema.1.19 NAME 'ipHostNumber'
   DESC 'IP address as a dotted decimal, eg. 192.168.1.1,
         omitting leading zeros'
   EQUALITY caseIgnoreIA5Match
   SYNTAX 'IA5String{128}' )

 ( nisSchema.1.20 NAME 'ipNetworkNumber'
   DESC 'IP network as a dotted decimal, eg. 192.168,
         omitting leading zeros'
   EQUALITY caseIgnoreIA5Match
   SYNTAX 'IA5String{128}' SINGLE-VALUE )

 ( nisSchema.1.21 NAME 'ipNetmaskNumber'
   DESC 'IP netmask as a dotted decimal, eg. 255.255.255.0,
         omitting leading zeros'
   EQUALITY caseIgnoreIA5Match
   SYNTAX 'IA5String{128}' SINGLE-VALUE )"
```

The document does try to provide some IPv6 support as in Section 5.4. (Interpreting Hosts and Networks):

"Hosts with IPv6 addresses MUST be written in their "preferred" form as defined in section 2.2.1 of [RFC1884], such that all components of the address are indicated and leading zeros are omitted. This provides a consistent means of resolving ipHosts by address."

However, the defined format mentioned above has been replaced, hence it is no longer valid.

6.42. RFC 2310: The Safe Response Header Field

There are no IPv4 dependencies in this specification.

6.43. RFC 2483: URI Resolution Services Necessary for URN Resolution

There are no IPv4 dependencies in this specification.

6.44. RFC 2567: Design Goals for an Internet Printing Protocol

There are no IPv4 dependencies in this specification.

6.45. RFC 2568: Rationale for the Structure of the Model and Protocol for the Internet Printing Protocol

There are no IPv4 dependencies in this specification.

6.46. RFC 2569: Mapping between LPD and IPP Protocols

There are no IPv4 dependencies in this specification.

6.47. RFC 2649: An LDAP Control and Schema for Holding Operation Signatures

There are no IPv4 dependencies in this specification.

6.48. RFC 2654: A Tagged Index Object for use in the Common Indexing Protocol

There are no IPv4 dependencies in this specification.

6.49. RFC 2655: CIP Index Object Format for SOIF Objects

There are no IPv4 dependencies in this specification.

6.50. RFC 2656: Registration Procedures for SOIF Template Types

There are no IPv4 dependencies in this specification.

6.51. RFC 2657: LDAPv2 Client vs. the Index Mesh

There are no IPv4 dependencies in this specification.

6.52. RFC 2756: Hyper Text Caching Protocol

This specification claims to be both IPv4 and IPv6 aware, but in Section 2.8. (An HTCP/0.0 AUTH has the following structure), it makes the following statement:

```
"SIGNATURE   is a COUNTSTR [3.1] which holds the HMAC-MD5 digest
             (see [RFC 2104]), with a B value of 64, of the
             following elements, each of which is digested in its
             "on the wire" format, including transmitted padding
             if any is covered by a field's associated LENGTH:

             IP SRC ADDR                            [4 octets]
             IP SRC PORT                            [2 octets]
             IP DST ADDR                            [4 octets]
             IP DST PORT                            [2 octets]
             HTCP MAJOR version number              [1 octet]
             HTCP MINOR version number              [1 octet]
             SIG-TIME                               [4 octets]
             SIG-EXPIRE                             [4 octets]
             HTCP DATA                              [variable]
             KEY-NAME (the whole COUNTSTR [3.1])   [variable]"
```

The given SIGNATURE calculation should be expanded to support IPv6 16 byte addresses.

6.53. RFC 2774: An HTTP Extension Framework

There are no IPv4 dependencies in this specification.

6.54. RFC 2974: Session Announcement Protocol

This protocol is both IPv4 and IPv6 aware and needs no changes.

6.55. RFC 3018: Unified Memory Space Protocol Specification

In section 3.4 (Address Formats), there are explicit references to IPv4 addressing:

"The following address format numbers are definite for nodes, immediately connected to the global IPv4 network:

N 4-0-0 (4)
N 4-0-1 (4-1)
N 4-0-2 (4-2)

The appropriate formats of 128-bit addresses:

```
Octets:
   +0              +1              +2              +3
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
0: |0 1 0 0|0 0|0 0|                 Free                          |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
4: |                             Free                              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
8: |            Free               |          IP address           |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
12:|           IP address          |      Local memory address     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+


   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
0: |0 1 0 0|0 0|0 1|                 Free                          |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
4: |                             Free                              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
8: |     Free      |                 IP address                    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
12:|   IP address  |             Local memory address              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+


   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
0: |0 1 0 0|0 0|1 0|                 Free                          |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
4: |                           Free                                |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
8: |                        IP address                             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
12:|                    Local memory address                       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Free

It is not used by the protocol.

IP address

It sets the node address in the global IPv4 network."

This section needs to be re-written, so that the specification becomes IPv6 compliant.

6.56. RFC 3082: Notification and Subscription for SLP

This protocol is both IPv4 and IPv6 aware, and thus requires no changes.

6.57. RFC 3088: OpenLDAP Root Service An experimental LDAP referral service

Section 5. (Using the Service) states:

> "The service supports LDAPv3 and LDAPv2+ [LDAPv2+] clients over TCP/IPv4. Future incarnations of this service may support TCP/IPv6 or other transport/internet protocols."

7. Summary of Results

This survey contemplates 257 RFCs, having 34 (12.84%) been identified as having some form of IPv4 dependency. Results are broken down as follows:

Standards:	1 out of 20 or 5.00%
Draft Standards:	4 out of 25 or 16.00%
Proposed Standards:	19 out of 155 or 12.26%
Experimental RFCs:	10 out of 57 or 17.54%

Of the 33 identified, the majority simply require minor actions, such as adding a caveat to IPv6 addressing that would avoid ambiguity, or re-writing a section to avoid IP-version dependent syntax. The remaining instances are documented below. The authors have attempted to organize the results in a format that allows easy referencing by other protocol designers.

7.1. Full Standards

7.1.1. RFC 959: STD 9 File Transfer Protocol

Problems have already been fixed in [5].

7.2. Draft Standards

7.2.1. RFC 1305: Network Time Protocol (version 3): Specification, Implementation and Analysis

As documented in Section 4.4. above, there are too many specific references to the use of 32-bit IPv4 addresses. An updated specification to support NTP over IPv6 is needed. However, there has been some work related with this issue, as an already expired work in progress, allegedly documents. Also, there is at least one IPv6 NTP implementation.

7.2.2. RFC 2396: URI Syntax

URI's allow the literal use of IPv4 addresses but have no specific recommendations on how to represent literal IPv6 addresses. This problem has already been addressed in [3].

7.2.3. RFC 2616: Hypertext Transfer Protocol HTTP/1.1

HTTP allows the literal use of IPv4 addresses, but has no specific recommendations on how to represent literal IPv6 addresses. This problem has already been addressed in [3].

7.3. Proposed Standards

7.3.1. RFC 946: Telnet Terminal LOC

There is a dependency in the definition of the TTYLOC Number which would require an updated version of the protocol. However, since this functionality is of marginal value today, an updated version might not make sense.

7.3.2. RFC 1738: URLs

URL's with IPv4 dependencies have already been addressed in [3].

Note that these dependencies affect other specifications as well, such as RFC 2122, RFC 2192, RFC 2193, RFC 2255, RFC 2371, and RFC 2384. All of these protocols have to revisited, and are not described separately in this memo.

7.3.3. RFC 2165: Service Location Protocol

The problems of this specification have already been addressed in [4].

7.3.4. RFC 2384: POP3 URL Scheme

POP URL IPv4 dependencies have already been addressed in [3].

7.3.5. RFC 2608: Service Location Protocol v2

The problems of this specification have already been addressed in [4].

7.3.6. RFC 2821: Simple Mail Transfer Protocol

Some textual updates and clarifications to MX processing would likely be useful. The operational scenarios and guidelines to avoid the problems have been described in [6].

7.3.7. RFC 3017: XML DTP For Roaming Access Phone Books

Extensions should be defined to support IPv6 addresses.

7.4. Experimental RFCs

7.4.1. RFC 1235: The Coherent File Distribution Protocol

The packet format of this protocol depends on IPv4, and would require updating to add IPv6 support. However, the protocol is not believed to be in use, so such an update may not be warranted.

7.4.2. RFC 1459: Internet Relay Chat Protocol

This specification only requires a text update to become IPv6 compliant.

7.4.3. RFC 1986: Simple File Transfer Using Enhanced TFTP

This specification only requires a text update to become IPv6 compliant.

7.4.4. RFC 2090: TFTP Multicast Option

This protocol relies on IPv4 IGMP Multicast. To become IPv6 compliant, a new version should be produced.

7.4.5. RFC 2307: Using LDAP as a NIS

This document tries to provide IPv6 support but it relies on an outdated format for IPv6 addresses. Thus, there is the need for an IPv6 compliant version.

8. Acknowledgements

Phil would like to acknowledge the support of the Internet Society in the research and production of this document. Additionally, Phil would like to thank his partner in all ways, Wendy M. Nesser.

9. Security Considerations

This document provides an exhaustive documentation of current IETF documented standards IPv4 address dependencies. Such process does not have security implications in itself.

10. References

10.1. Normative References

[1] Nesser, II, P. and A. Bergstrom, Editor, "Introduction to the Survey of IPv4 Addresses in Currently Deployed IETF Standards", RFC 3789, June 2004.

[2] Bradner, S., "The Internet Standards Process - version 3", BCP 9, RFC 2026, October 1996.

10.2. Informative References

[3] Hinden, R., Carpenter, B. and L. Masinter, "Format for Literal IPv6 Addresses in URL's", RFC 2732, December 1999.

[4] Guttman, E., "Service Location Protocol Modifications for IPv6", RFC 3111, May 2001.

[5] Allman, M., Ostermann, S. and C. Metz, "FTP Extensions for IPv6 and NATs", RFC 2428, September 1998.

[6] Hagino, J. and M. Nakamura, "SMTP operational experience in mixed IPv4/IPv6 environements", Work in Progress.

11. Authors' Addresses

Rute Sofia
FCCN
Av. Brasil, 101
1700 Lisboa, Portugal

Phone: +351 91 2507372
EMail: rsofia@zmail.pt

Philip J. Nesser II
Principal
Nesser & Nesser Consulting
13501 100th Ave NE, #5202
Kirkland, WA 98034

Phone: +1 425 481 4303
Fax: +1 425 482 9721
EMail: phil@nesser.com

12. Full Copyright Statement

Intellectual Property

Acknowledgement

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