

Survey of IPv4 Addresses in Currently Deployed IETF Application Area Standards

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Status of this Memo

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

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

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

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

Abstract

This document seeks to document all usage of IPv4 addresses in currently deployed IETF Application Area documented standards. In order to successfully transition from an all IPv4 Internet to an all IPv6 Internet, many interim steps will be taken. One of these steps is the evolution of current protocols that have IPv4 dependencies. It is hoped that these protocols (and their implementations) will be redesigned to be network address independent, but failing that will at least dually support IPv4 and IPv6. To this end, all Standards (Full, Draft, and Proposed) as well as Experimental RFCs will be surveyed and any dependencies will be documented.

[1.0](#) Introduction

This work began as a megolithic document [draft-ietf-ngtrans-ipv4survey-XX.txt](#). In an effort to rework the information into a more manageable form, it has been broken into 8 documents conforming to the current IETF areas (Application, General, Internet, Management & Operations, Routing, Security, Sub-IP and Transport).

1.1 Short Historical Perspective

There are many challenges that face the Internet Engineering community. The foremost of these challenges has been the scaling issue. How to grow a network that was envisioned to handle thousands of hosts to one that will handle tens of millions of networks with billions of hosts. Over the years this scaling problem has been overcome with changes to the network layer and to routing protocols. (Ignoring the tremendous advances in computational hardware)

The first "modern" transition to the network layer occurred in during the early 1980's from the Network Control Protocol (NCP) to IPv4. This culminated in the famous "flag day" of January 1, 1983. This version of IP was documented in [RFC 760](#). This was a version of IP with 8 bit network and 24 bit host addresses. A year later IP was updated in [RFC 791](#) to include the famous A, B, C, D, & E class system.

Networks were growing in such a way that it was clear that a need for breaking networks into smaller pieces was needed. In October of 1984 [RFC 917](#) was published formalizing the practice of subnetting.

By the late 1980's it was clear that the current exterior routing protocol used by the Internet (EGP) was not sufficient to scale with the growth of the Internet. The first version of BGP was documented in 1989 in RFC 1105.

The next scaling issues to become apparent in the early 1990's was the exhaustion of the Class B address space. The growth and commercialization of the Internet had organizations requesting IP addresses in alarming numbers. In May of 1992 over 45% of the Class B space was allocated. In early 1993 [RFC 1466](#) was published directing assignment of blocks of Class C's be given out instead of Class B's. This solved the problem of address space exhaustion but had significant impact of the routing infrastructure.

The number of entries in the "core" routing tables began to grow exponentially as a result of [RFC 1466](#). This led to the implementation of BGP4 and CIDR prefix addressing. This may have solved the problem for the present but there are still potential scaling issues.

Current Internet growth would have long overwhelmed the current address space if industry didn't supply a solution in Network Address Translators (NATs). To do this the Internet has sacrificed the underlying "End-to-End" principle.

In the early 1990's the IETF was aware of these potential problems and began a long design process to create a successor to IPv4 that would address these issues. The outcome of that process was IPv6.

The purpose of this document is not to discuss the merits or problems of IPv6. That is a debate that is still ongoing and will eventually be decided on how well the IETF defines transition mechanisms and how

industry accepts the solution. The question is not "should," but "when."

1.2 A Brief Aside

Throughout this document there are discussions on how protocols might be updated to support IPv6 addresses. Although current thinking is that IPv6 should suffice as the dominant network layer protocol for the lifetime of the author, it is not unreasonable to contemplate further upgrade to IP. Work done by the IRTF Interplanetary Internet Working Group shows one idea of far reaching thinking. It may be a reasonable idea (or may not) to consider designing protocols in such a way that they can be either IP version aware or independent. This idea must be balanced against issues of simplicity and performance. Therefore it is recommended that protocol designer keep this issue in mind in future designs.

Just as a reminder, remember the words of Jon Postel:

"Be conservative in what you send; be liberal in what you accept from others."

2.0 Methodology

To perform this study each class of IETF standards are investigated in order of maturity: Full, Draft, and Proposed, as well as Experimental. Informational RFC are not addressed. RFCs that have been obsoleted by either newer versions or as they have transitioned through the standards process are not covered.

Please note that a side effect of this choice of methodology is that some protocols that are defined by a series of RFC's that are of different levels of standards maturity are covered in different spots in the document. Likewise other natural groupings (i.e. MIBs, SMTP extensions, IP over FOO, PPP, DNS, etc.) could easily be imagined.

2.1 Scope

The procedure used in this investigation is an exhaustive reading of the applicable RFC's. This task involves reading approximately 25000 pages of protocol specifications. To compound this, it was more than a process of simple reading. It was necessary to attempt to understand the purpose and functionality of each protocol in order to make a proper determination of IPv4 reliability. The author has made every effort to make this effort and the resulting document as complete as possible, but it is likely that some subtle (or perhaps not so subtle) dependence was missed. The author encourage those familiar (designers, implementers or anyone who has an intimate knowledge) with any protocol to review the appropriate sections and make comments.

[2.2](#) Document Organization

The rest of the document sections are described below.

Sections [3](#), [4](#), [5](#), and [6](#) each describe the raw analysis of Full, Draft, and Proposed Standards, and Experimental RFCs. Each RFC is discussed in its turn starting with [RFC 1](#) and ending with [RFC 3247](#). The comments for each RFC is "raw" in nature. That is, each RFC is discussed in a vacuum and problems or issues discussed do not "look ahead" to see if the problems have already been fixed.

[Section 7](#) is an analysis of the data presented in Sections [3](#), [4](#), [5](#), and [6](#). **It is here that all of the results are considered as a whole and the problems that have been resolved in later RFCs are correlated.**

[3.0](#) Full Standards

Full Internet Standards (most commonly simply referred to as "Standards") are fully mature protocol specification that are widely implemented and used throughout the Internet.

[3.01](#) Telnet Protocol. [RFC0854](#), [RFC0855](#)

[3.01.1](#) [RFC 854](#)

There are no IPv4 dependencies in [RFC 854](#).

[3.01.2](#) [RFC 855](#)

There are no IPv4 dependencies in [RFC 855](#).

[3.02](#) [RFC 959](#) File Transfer Protocol

[Section 4.1.2](#) "TRANSFER PARAMETER COMMANDS" the port command has the following format: "PORT h1,h2,h3,h4,p1,p2" where h1 is the high order 8 bits of the internet host address. This needs to be reworked to transition to IPv6 addressing.

In sections [4.2.1](#) & 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.

[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 is clearly an IPv4 address reference.

3.03 SMTP Service Extensions. [RFC821](#), [RFC1869](#)

3.03.1 RFC 821

[Section 4.1.2](#) "Command Syntax" contains the following reference:

`<dotnum> ::= <snum> "." <snum> "." <snum> "." <snum>`

The is clearly an IPv4 address reference. There is also the following paragraph:

Sometimes a host is not known to the translation function and communication is blocked. To bypass this barrier two numeric forms are also allowed for host "names". One form is a decimal integer prefixed by a pound sign, "#", which indicates the number is the address of the host. Another form is four small decimal integers separated by dots and enclosed by brackets, e.g., "[123.255.37.2]", which indicates a 32-bit ARPA Internet Address in four 8-bit fields.

3.03.2 RFC 1869

There are no IPv4 dependencies in [RFC 1869](#).

3.04 RFC 822 Standard for the format of ARPA Internet text messages

6.2.3. "DOMAIN TERMS" contains the following text:

A domain-ref must be THE official name of a registry, network, or host. It is a symbolic reference, within a name sub-domain. At times, it is necessary to bypass standard mechanisms for resolving such references, using more primitive information, such as a network host address rather than its associated host name.

To permit such references, this standard provides the domain-literal construct. Its contents must conform with the needs of the sub-domain in which it is interpreted.

Domain-literals which refer to domains within the ARPA Internet specify 32-bit Internet addresses, in four 8-bit fields noted in decimal, as described in Request for Comments #820, "Assigned Numbers." For example:

[10.0.3.19]

Note: THE USE OF DOMAIN-LITERALS IS STRONGLY DISCOURAGED. It is permitted only as a means of bypassing temporary system limitations, such as name tables which are not

complete.

3.05 RFC 1305 Network Time Protocol (Version 3)

Section 3.2.1 Common Variables defines 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 Procedures 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 as follows:

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 (i.e. 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 an IPv4 address.

There are numerous places in sample code and algorithms use the above mentioned variables. It seems that there is no reason to modify the actual protocol. A small number of text changes and an update to implementations to understand both IPv4 and IPv6 addresses can achieve an NTP that works on both network layer protocols.

[3.06 RFC 974](#) Mail Routing and the Domain System

The examples section of this RFC uses the well established A records which have previously been discussed.

[3.07 RFC 862](#) Echo Protocol

There are no IPv4 dependencies in this protocol.

[3.08 RFC 863](#) Discard Protocol

There are no IPv4 dependencies in this protocol.

[3.09 RFC 864](#) Character Generator Protocol

There are no IPv4 dependencies in this protocol.

[3.10 RFC 865](#) Quote of the Day Protocol

There are no IPv4 dependencies in this protocol.

[3.11 RFC 866](#) Active Users Protocol

There are no IPv4 dependencies in this protocol.

[3.12 RFC 867](#) Daytime Protocol

There are no IPv4 dependencies in this protocol.

[3.13 RFC 868](#) Time Server Protocol

There are no IPv4 dependencies in this protocol.

[3.14 RFC 856](#) Binary Transmission Telnet Option

There are no IPv4 dependencies in this protocol.

[3.15 RFC 857](#) **Echo Telnet Option**

There are no IPv4 dependencies in this protocol.

[3.16 RFC 858](#) **Suppress Go Ahead Telnet Option**

There are no IPv4 dependencies in this protocol.

[3.17 RFC 859](#) **Status Telnet Option**

There are no IPv4 dependencies in this protocol.

[3.18 RFC 860](#) **Timing Mark Telnet Option**

There are no IPv4 dependencies in this protocol.

[3.19 RFC 861](#) **Extended Options List Telnet Option**

There are no IPv4 dependencies in this protocol.

[3.20 RFC 1350](#) **Trivial File Transfer Protocol**

There are no IPv4 dependencies in this protocol.

[3.21 RFC 1939](#) **Post Office Protocol - Version 3**

There are no IPv4 dependencies in this protocol.

[3.22 RFC 2920](#) **SMTP Service Extension for Command Pipelining (SMTP-pipe)**

There are no IPv4 dependencies in this protocol.

[4.0](#) **Draft Standards**

Draft Standards represent the penultimate standard level in the IETF. A protocol can only achieve draft standard when there are multiple, independent, interoperable implementations. Draft Standards are usually quite mature and widely used.

[4.01 RFC 954](#) **NICNAME/WHOIS (NICNAME)**

There are no IPv4 dependencies in this protocol.

[4.02 RFC 1184](#) Telnet Linemode Option (TOPT-LINE)

There are no IPv4 dependencies in this protocol.

[4.03 RFC 1288](#) The Finger User Information Protocol (FINGER)

There are no IPv4 dependencies in this protocol.

[4.04 RFC 1305](#) Network Time Protocol (Version 3) Specification, Implementation (NTPV3)

There are no new issues than those presented in [Section 3.12](#) of this document.

[4.05 RFC 1575](#) An Echo Function for CLNP (ISO 8473) (ISO-TS-ECH)

There are no IPv4 dependencies in this protocol.

[4.06 RFC 1652](#) SMTP Service Extension for 8bit-MIMEtransport

There are no IPv4 dependencies in this protocol.

[4.07 RFC 1777](#) Lightweight Directory Access Protocol

There are no IPv4 dependencies in this protocol.

[4.08 RFC 1778](#) The String Representation of Standard Attribute Syntaxes

There are no IPv4 dependencies in this protocol.

[4.09 RFC 1832](#) XDR: External Data Representation Standard (XDR)

There are no IPv4 dependencies in this protocol.

[4.10 RFC 2045](#) Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies (MIME)

There are no IPv4 dependencies in this protocol.

[4.11 RFC 2046](#) Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types (MIME-MEDIA)

There are no IPv4 dependencies in this protocol.

[4.12 RFC 2047](#) **MIME (Multipurpose Internet Mail Extensions) Part Three: Message Header Extensions for Non-ASCII Text (MIME-MSG)**

There are no IPv4 dependencies in this protocol.

[4.13 RFC 2049](#) **Multipurpose Internet Mail Extensions (MIME) Part Five: Conformance Criteria and Examples (MIME-CONF)**

There are no IPv4 dependencies in this protocol.

[4.14 RFC 2279](#) **UTF-8, a transformation format of ISO 10646 (UTF-8)**

There are no IPv4 dependencies in this protocol.

[4.15 RFC 2347](#) **TFTP Option Extension (TFTP-Ext)**

There are no IPv4 dependencies in this protocol.

[4.16 RFC 2348](#) **TFTP Blocksize Option (TFTP-Blk)**

In the Section Blocksize Option Specification the following example is given:

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 example blocksize would not work with IPv6 header sizes, but it has no real significance on since larger block sizes are available.

[4.17 RFC 2349](#) **TFTP Timeout Interval and Transfer Size Options (TFTP-Opt)**

There are no IPv4 dependencies in this protocol.

[4.18 RFC 2355](#) **TN3270 Enhancements (TN3270E)**

There are no IPv4 dependencies in this protocol.

[4.19 RFC 2396](#) Uniform Resource Identifiers (URI): Generic Syntax (URI-GEN)

[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.

[4.20 RFC 2616](#) Hypertext Transfer Protocol -- HTTP/1.1 (HTTP)

[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 the issues regarding multiple HTTP servers running on a single IP interface. The text is version neutral, but clearly remains an implementation issue.

[5.0](#) Proposed Standards

Proposed Standards are introductory level documents. There are no requirements for even a single implementation. In many cases Proposed

are never implemented or advanced in the IETF standards process. They therefore are often just proposed ideas that are presented to the Internet community. Sometimes flaws are exposed or they are one of many competing solutions to problems. In these later cases, no discussion is presented as it would not serve the purpose of this discussion.

[5.001 RFC 698](#) Telnet extended ASCII option (TOPT-EXT)

There are no IPv4 dependencies in this protocol.

[5.002 RFC 726](#) Remote Controlled Transmission and Echoing Telnet option (TOPT-REM)

There are no IPv4 dependencies in this protocol.

[5.003 RFC 727](#) Telnet logout option (TOPT-LOGO)

There are no IPv4 dependencies in this protocol.

[5.004 RFC 735](#) Revised Telnet byte macro option (TOPT-BYTE)

There are no IPv4 dependencies in this protocol.

[5.005 RFC 736](#) Telnet SUPDUP option (TOPT-SUP)

There are no IPv4 dependencies in this protocol.

[5.006 RFC 749](#) Telnet SUPDUP-Output option (TOPT-SUP0)

There are no IPv4 dependencies in this protocol.

[5.007 RFC 779](#) Telnet send-location option (TOPT-SNDL)

There are no IPv4 dependencies in this protocol.

[5.008 RFC 885](#) Telnet end of record option (TOPT-EOR)

There are no IPv4 dependencies in this protocol.

[5.009 RFC 927](#) TACACS user identification Telnet option (TOPT-TACAC)

There are no IPv4 dependencies in this protocol.

[5.010 RFC 933](#) Output marking Telnet option (TOPT-OM)

There are no IPv4 dependencies in this protocol.

[5.011 RFC 946](#) Telnet terminal location number option (TOPT-TLN)

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, 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.

Although there is a dependency here, it is unlikely to be of any major significance.

[5.012 RFC 977](#) Network News Transfer Protocol (NNTP)

There are no IPv4 dependencies in this protocol.

[5.013 RFC 1041](#) Telnet 3270 regime option (TOPT-3270)

There are no IPv4 dependencies in this protocol.

[5.014 RFC 1043](#) Telnet Data Entry Terminal option: DODIIS implementation (TOPT-DATA)

There are no IPv4 dependencies in this protocol.

[5.015 RFC 1053](#) Telnet X.3 PAD option (TOPT-X.3)

There are no IPv4 dependencies in this protocol.

[5.016 RFC 1073](#) Telnet window size option (TOPT-NAWS)

There are no IPv4 dependencies in this protocol.

[5.017 RFC 1079](#) Telnet terminal speed option (TOPT-TS)

There are no IPv4 dependencies in this protocol.

[5.018](#) [RFC 1091](#) Telnet terminal-type option (TOPT-TERM)

There are no IPv4 dependencies in this protocol.

[5.019](#) [RFC 1096](#) Telnet X display location option (TOPT-XDL)

There are no IPv4 dependencies in this protocol.

[5.020](#) [RFC 1274](#) The COSINE and Internet X.500 Schema

There are no IPv4 dependencies in this protocol.

[5.021](#) [RFC 1276](#) Replication and Distributed Operations extensions to provide an Internet Directory using X.500

There are no IPv4 dependencies in this protocol.

[5.022](#) [RFC 1314](#) A File Format for the Exchange of Images in the Internet (NETFAX)

There are no IPv4 dependencies in this protocol.

[5.023](#) [RFC 1328](#) X.400 1988 to 1984 downgrading

There are no IPv4 dependencies in this protocol.

[5.024](#) [RFC 1372](#) Telnet Remote Flow Control Option (TOPT-RFC)

There are no IPv4 dependencies in this protocol.

[5.025](#) [RFC 1415](#) FTP-FTAM Gateway Specification (FTP)

This document defines a gateway for interaction between FTAM and FTP. As long as the problems associated with the FTP protocol identified above are addressed, this protocol specification does not add any other dependencies.

[5.026](#) [RFC 1494](#) Equivalences between 1988 X.400 and [RFC-822](#) Message Bodies (Equiv)

There are no IPv4 dependencies in this protocol.

[5.027](#) [RFC 1496](#) Rules for downgrading messages from X.400/88 to X.400/84 when MIME content-types are present in the messages

(HARPOON)

There are no IPv4 dependencies in this protocol.

[5.028 RFC 1502](#) X.400 Use of Extended Character Sets

There are no IPv4 dependencies in this protocol.

[5.029 RFC 1572](#) Telnet Environment Option (TOPT-ENVIR)

There are no IPv4 dependencies in this protocol.

[5.030 RFC 1648](#) Postmaster Convention for X.400 Operations

There are no IPv4 dependencies in this protocol.

[5.031 RFC 1738](#) Uniform Resource Locators (URL) (URL)

[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](#) [5]: 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 the following text is presented:

; URL schemeparts for ip based protocols:

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

login = [user [":" password] "@"] hostport

hostport = host [":" port]

host = hostname | hostnumber

[5.032 RFC 1740](#) MIME Encapsulation of Macintosh Files - MacMIME (MacMIME)

There are no IPv4 dependencies in this protocol.

[5.033 RFC 1767](#) MIME Encapsulation of EDI Objects (MIME-EDI)

There are no IPv4 dependencies in this protocol.

[5.034 RFC 1781](#) Using the OSI Directory to Achieve User Friendly Naming (OSI-Dir)

There are no IPv4 dependencies in this protocol.

[5.035 RFC 1798](#) Connection-less Lightweight X.500 Directory Access Protocol (CLDAP)

[Section 5.2](#). Client Implementations makes the following observation:

For simple lookup applications, use of a retry algorithm with multiple servers similar to that commonly used in DNS stub resolver implementations is recommended. The location of a CLDAP server or servers may be better specified using IP addresses (simple or broadcast) rather than names that must first be looked up in another directory such as DNS.

It is not specific to IPv4 and compliance with IPv6 will be implementation dependent.

[5.036 RFC 1808](#) Relative Uniform Resource Locators (URL)

There are no IPv4 dependencies in this protocol.

[5.037 RFC 1835](#) Architecture of the WHOIS++ service (WHOIS++)

There are no IPv4 dependencies in this protocol.

[5.038 RFC 1891](#) SMTP Service Extension for Delivery Status Notifications (SMTP-DSN)

There are no IPv4 dependencies in this protocol.

[5.039 RFC 1892](#) The Multipart/Report Content Type for the Reporting of Mail System Administrative Messages (MIME-RPT)

There are no IPv4 dependencies in this protocol.

[5.040 RFC 1893](#) Enhanced Mail System Status Codes (EMS-CODE)

There are no IPv4 dependencies in this protocol.

[5.041 RFC 1894](#) An Extensible Message Format for Delivery Status Notifications (DSN)

There are no IPv4 dependencies in this protocol.

[5.042 RFC 1913](#) Architecture of the Whois++ Index Service,WHOIS++A

[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
```

This syntax does not necessarily have any IPv4 dependencies but implementations should be modified to check the incoming packet to see which IP version the original request used in order to determine whether returning an IPv6 address is reasonable.

[5.043 RFC 1914](#) How to Interact with a Whois++ Mesh (WHOIS++)

This document states beginning in [Section 4](#):

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.

[4.1](#). Caching a Whois++ servers hostname

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
```

but these statements are IP version neutral.

[5.044 RFC 1985](#) SMTP Service Extension for Remote Message Queue Starting (SMTP-ETRN)

There are no IPv4 dependencies in this protocol.

[5.045 RFC 2017](#) Definition of the URL MIME External-Body Access-Type (URL-ACC)

There are no IPv4 dependencies in this protocol.

[5.046 RFC 2034](#) SMTP Service Extension for Returning Enhanced Error Codes (SMTP-ENH)

There are no IPv4 dependencies in this protocol.

[5.047 RFC 2056](#) Uniform Resource Locators for Z39.50 (URLZ39.50)

There are no IPv4 dependencies in this protocol.

[5.048 RFC 2060](#) Internet Message Access Protocol - Version 4rev1 (IMAPV4)

There are no IPv4 dependencies in this protocol.

[5.049 RFC 2077](#) The Model Primary Content Type for Multipurpose Internet Mail Extensions (MIME-MODEL)

There are no IPv4 dependencies in this protocol.

[5.050 RFC 2079](#) Definition of an X.500 Attribute Type and an Object Class to Hold Uniform Resource Identifiers (URIs) (URI-ATT)

There are no IPv4 dependencies in this protocol.

[5.052 RFC 2086](#) IMAP4 ACL extension (IMAP4-ACL)

There are no IPv4 dependencies in this protocol.

[5.053 RFC 2087](#) IMAP4 QUOTA extension (IMAP4-QUO)

There are no IPv4 dependencies in this protocol.

[5.054 RFC 2088](#) IMAP4 non-synchronizing literals (IMAP4-LIT)

There are no IPv4 dependencies in this protocol.

[5.055 RFC 2122](#) VEMMI URL Specification (VEMMI-URL)

[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.

It is possible that the <host> portion to contain an IPv4 only address as defined in [RFC 1738](#) (see above). Once the problem is clarified for [RFC 1738](#), this issue will automatically be resolved.

[5.056 RFC 2141](#) URN Syntax (URN-SYNTAX)

There are no IPv4 dependencies in this protocol.

[5.057 RFC 2142](#) "Mailbox Names for Common Services, Roles and Functions" (MAIL-SERV)

There are no IPv4 dependencies in this protocol.

[5.058 RFC 2156](#) MIXER (Mime Internet X.400 Enhanced Relay): Mapping between X.400 and [RFC 822](#)/MIME (MIXER)

There are no IPv4 dependencies in this protocol.

[5.059 RFC 2157](#) Mapping between X.400 and [RFC-822](#)/MIME Message Bodies

There are no IPv4 dependencies in this protocol.

[5.060 RFC 2158](#) X.400 Image Body Parts

There are no IPv4 dependencies in this protocol.

[5.061 RFC 2159](#) A MIME Body Part for FAX

There are no IPv4 dependencies in this protocol.

[5.062 RFC 2160](#) Carrying PostScript in X.400 and MIME

There are no IPv4 dependencies in this protocol.

[5.063 RFC 2163](#) Using the Internet DNS to Distribute MIXER Conformant Global Address Mapping (MCGAM) (DNS-MCGAM)

There are no IPv4 dependencies in this protocol.

[5.064 RFC 2164](#) Use of an X.500/LDAP directory to support MIXER address mapping

There are no IPv4 dependencies in this protocol.

[5.065 RFC 2165](#) Service Location Protocol (SLP)

Sections [7](#). Service Type Request Message Format, and 9. Service Registration Message Format each have a 80 bit field from addr-spec (see below) which would not support IPv6 addresses.

In [Section 20.1](#). Previous Responders' Address Specification

The previous responders' Address Specification is specified as

```
<Previous Responders' Address Specification> ::=  
    <addr-spec> |  
    <addr-spec>, <Previous Responders' Address Specification>
```

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.

Example:

```
RESOLVO.NEATO.ORG,128.127.203.63
```

and further in [Section 20.4](#). Address Specification in Service Location

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.

All of [Section 21](#) defines the requirements for each of the elements of this protocol. The discussion makes many statements that seem to imply IPv4, but the statements are general enough that they can still operate on 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 there needs to be equivalent IPv6 multicast addresses,

[5.066 RFC 2177](#) **IMAP4 IDLE command (IMAP4-IDLE)**

There are no IPv4 dependencies in this protocol.

[5.067 RFC 2183](#) **Communicating Presentation Information in Internet Messages: The Content-Disposition Header Field**

There are no IPv4 dependencies in this protocol.

[5.068 RFC 2192](#) **IMAP URL Scheme (IMAP-URL)**

There are no IPv4 dependencies in this protocol.

[5.069 RFC 2193](#) **IMAP4 Mailbox Referrals (IMAP4MAIL)**

In [Section 6](#). Formal Syntax

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

leaves a dependency on [RFC 1738](#) URL definitions. Presuming the issues discussed for that RFC are resolved, this protocol becomes IPv6 aware.

[5.070 RFC 2218](#) A Common Schema for the Internet White Pages Service

There are no IPv4 dependencies in this protocol.

[5.071 RFC 2221](#) IMAP4 Login Referrals (IMAP4LOGIN)

In the referral syntax presented in this document the string USER@SERVER2 is presented. No specifications on the formatting of "SERVER2" is presented. It is up to individual implementations to decide acceptable values for the hostname. This may or may not include explicit IPv6 addresses.

[5.072 RFC 2227](#) Simple Hit-Metering and Usage-Limiting for HTTP

There are no IPv4 dependencies in this protocol.

[5.073 RFC 2231](#) MIME Parameter Value and Encoded Word Extensions: Character Sets, Languages, and Continuations (MIME-EXT)

There are no IPv4 dependencies in this protocol.

[5.074 RFC 2234](#) Augmented BNF for Syntax Specifications: ABNF (ABNF)

There are no IPv4 dependencies in this protocol.

[5.075 RFC 2244](#) ACAP -- Application Configuration Access Protocol (ACAP)

There are no IPv4 dependencies in this protocol.

[5.076 RFC 2254](#) The String Representation of LDAP Search Filters (STR-LDAP)

There are no IPv4 dependencies in this protocol.

[5.077 RFC 2255](#) The LDAP URL Format (LDAP-URL)

There are no IPv4 dependencies in this protocol.

[5.078 RFC 2247](#) Using Domains in LDAP/X.500 Distinguished Names

There are no IPv4 dependencies in this protocol.

**[5.079 RFC 2251](#) Lightweight Directory Access Protocol (v3)
(LDAPV3)**

There are no IPv4 dependencies in this protocol.

**[5.080 RFC 2252](#) Lightweight Directory Access Protocol (v3):
Attribute Syntax Definitions (LDAP3-ATD)**

There are no IPv4 dependencies in this protocol.

**[5.081 RFC 2253](#) Lightweight Directory Access Protocol (v3):
UTF-8 String Representation of Distinguished Names
(LDAP3-UTF8)**

[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)

Without putting any limitations on the version of the IP address.
With that single caveat, there are no IPv4 dependencies in this protocol.

**[5.082 RFC 2256](#) A Summary of the X.500(96) User Schema for use
with LDAPv3**

There are no IPv4 dependencies in this protocol.

**[5.083 RFC 2293](#) Representing Tables and Subtrees in the X.500
Directory (SUBTABLE)**

There are no IPv4 dependencies in this protocol.

[5.084 RFC 2294](#) Representing the O/R Address hierarchy in the X.500 Directory Information Tree (OR-ADD)

There are no IPv4 dependencies in this protocol.

[5.085 RFC 2298](#) An Extensible Message Format for Message Disposition Notifications (EMF-MDN)

There are no IPv4 dependencies in this protocol.

[5.086 RFC 2301](#) File Format for Internet Fax (FFIF)

There are no IPv4 dependencies in this protocol.

[5.087 RFC 2302](#) Tag Image File Format (TIFF) - image/tiff MIME Sub-type Registration (TIFF)

There are no IPv4 dependencies in this protocol.

[5.088 RFC 2303](#) Minimal PSTN address format in Internet Mail (MIN-PSTN)

There are no IPv4 dependencies in this protocol.

[5.089 RFC 2304](#) Minimal FAX address format in Internet Mail (MINFAX-IM)

There are no IPv4 dependencies in this protocol.

[5.090 RFC 2305](#) A Simple Mode of Facsimile Using Internet Mail (SMFAX-IM)

There are no IPv4 dependencies in this protocol.

[5.091 RFC 2334](#) Server Cache Synchronization Protocol (SCSP) (SCSP)

The only reference to IPv4 specific issues is shown below:

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.

Since this is only an example, it does not preclude the use of IPv6 addresses as well. It is most likely an implementation issue.

[5.092 RFC 2342](#) IMAP4 Namespace (IMAP4NAME)

There are no IPv4 dependencies in this protocol.

[5.093 RFC 2359](#) IMAP4 UIDPLUS extension (IMAP4UIDPL)

There are no IPv4 dependencies in this protocol.

[5.094 RFC 2368](#) The mailto URL scheme (URLMAILTO)

There are no IPv4 dependencies in this protocol.

[5.095 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 protocol.

[5.096 RFC 2384](#) POP URL Scheme (POP-URL)

The following dependencies are in this document:

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.

Since [RFC 1738](#) has a potential IPv4 limitation, this protocol will be IPv6 compliant when [RFC 1738](#) is updated.

[5.097 RFC 2387](#) The MIME Multipart/Related Content-type (MIME-RELAT)

There are no IPv4 dependencies in this protocol.

[5.098 RFC 2388](#) Returning Values from Forms: multipart/form-data

There are no IPv4 dependencies in this protocol.

[5.099 RFC 2389](#) Feature negotiation mechanism for the File Transfer Protocol

There are no IPv4 dependencies in this protocol.

[5.100 RFC 2392](#) Content-ID and Message-ID Uniform Resource Locators (CIDMID-URL)

There are no IPv4 dependencies in this protocol.

[5.101 RFC 2397](#) The "data" URL scheme (DATA-URL)

There are no IPv4 dependencies in this protocol.

[5.102 RFC 2421](#) Voice Profile for Internet Mail - version 2 (MIME-VP2)

There are no IPv4 dependencies in this protocol.

[5.103 RFC 2422](#) Toll Quality Voice - 32 kbit/s ADPCM MIME Sub-type Registration (MIME-ADPCM)

There are no IPv4 dependencies in this protocol.

[5.104 RFC 2423](#) VPIM Voice Message MIME Sub-type Registration (MIME-VPIM)

There are no IPv4 dependencies in this protocol.

[5.105 RFC 2424](#) Content Duration MIME Header Definition (CONT-DUR)

There are no IPv4 dependencies in this protocol.

[5.106 RFC 2425](#) A MIME Content-Type for Directory Information (TXT-DIR)

There are no IPv4 dependencies in this protocol.

[5.107 RFC 2426](#) vCard MIME Directory Profile (MIME-VCARD)

There are no IPv4 dependencies in this protocol.

[5.108 RFC 2428](#) FTP Extensions for IPv6 and NATs

This RFC documents an IPv6 extension and is not considered in this discussion.

[5.109 RFC 2445](#) Internet Calendaring and Scheduling Core Object Specification (iCalendar) (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 it does not explicitly state the use of IPv4 addresses, they are explicit in [RFC 822](#).

It should explicitly disallow the use of IPv4 addresses.

**[5.110 RFC 2446](#) iCalendar Transport-Independent Interoperability Protocol
(iTIP) Scheduling Events, BusyTime, To-dos and Journal Entries (iTIP)**

There are no IPv4 dependencies in this protocol.

**[5.111 RFC 2447](#) iCalendar Message-Based Interoperability Protocol (iMIP)
(iMIP)**

There are no IPv4 dependencies in this protocol.

[5.112 RFC 2449](#) POP3 Extension Mechanism (POP3-EXT)

There are no IPv4 dependencies in this protocol.

[5.113 RFC 2476](#) Message Submission

There are several discussions of using IP Address authorization schemes, but the protocol does not limit those addresses to IPv4.

[5.114 RFC 2480](#) Gateways and MIME Security Multiparts

There are no IPv4 dependencies in this protocol.

**[5.115 RFC 2518](#) HTTP Extensions for Distributed Authoring --
WEBDAV (WEBDAV)**

There are no IPv4 dependencies in this protocol.

**[5.116 RFC 2530](#) Indicating Supported Media Features Using
Extensions to DSN and MDN**

There are no IPv4 dependencies in this protocol.

[5.117 RFC 2532](#) Extended Facsimile Using Internet Mail

There are no IPv4 dependencies in this protocol.

[5.118 RFC 2533](#) A Syntax for Describing Media Feature Sets

There are no IPv4 dependencies in this protocol.

[5.119 RFC 2534](#) Media Features for Display, Print, and Fax

There are no IPv4 dependencies in this protocol.

[5.120 RFC 2554](#) SMTP Service Extension for Authentication

There are no IPv4 dependencies in this protocol.

[5.121 RFC 2557](#) MIME Encapsulation of Aggregate Documents, such as HTML (MHTML) (MHTML)

There are no IPv4 dependencies in this protocol.

[5.122 RFC 2589](#) Lightweight Directory Access Protocol (v3): Extensions for Dynamic Directory Services (LDAPv3)

There are no IPv4 dependencies in this protocol.

[5.123 RFC 2595](#) Using TLS with IMAP, POP3 and ACAP

There are no IPv4 dependencies in this protocol.

[5.124 RFC 2596](#) Use of Language Codes in LDAP

There are no IPv4 dependencies in this protocol.

[5.125 RFC 2608](#) Service Location Protocol, Version 2 (SLP)

In [Section 8.1](#). Service Request

```

      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.

A new version of the protocol must be defined to support IPv6 environments.

[5.126 RFC 2609](#) Service Templates and Service: Schemes

This document defines:

The ABNF for a service: URL is:

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

There are many other references to the hostnumber in the document.

There should be an update to support IPv6.

[5.127 RFC 2640](#) Internationalization of the File Transfer Protocol

There are no IPv4 dependencies in this protocol.

[5.128 RFC 2645](#) ON-DEMAND MAIL RELAY (ODMR) SMTP with Dynamic IP Addresses (ODMR-SMTP)

There are no IPv4 dependencies in this protocol.

[5.129 RFC 2646](#) The Text/Plain Format Parameter

There are no IPv4 dependencies in this protocol.

[5.130 RFC 2651](#) The Architecture of the Common Indexing Protocol (CIP) (CIP)

There are no IPv4 dependencies in this protocol.

[5.131 RFC 2652](#) MIME Object Definitions for the Common Indexing

Protocol (CIP)

There are no IPv4 dependencies in this protocol.

[5.132 RFC 2653](#) CIP Transport Protocols

There are no IPv4 dependencies in this protocol.

[5.133 RFC 2732](#) Format for Literal IPv6 Addresses in URL's

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

[5.134 RFC 2738](#) Corrections to "A Syntax for Describing Media Feature Sets"

There are no IPv4 dependencies in this protocol.

[5.135 RFC 2739](#) Calendar Attributes for vCard and LDAP

There are no IPv4 dependencies in this protocol.

[5.136 RFC 2806](#) URLs for Telephone Calls

There are no IPv4 dependencies in this protocol.

[5.137 RFC 2846](#) GSTN Address Element Extensions in E-mail Services

There are no IPv4 dependencies in this protocol.

[5.138 RFC 2849](#) The LDAP Data Interchange Format (LDIF) - Technical Specification (LDIF)

There are no IPv4 dependencies in this protocol.

[5.139 RFC 2852](#) Deliver By SMTP Service Extension

There are no IPv4 dependencies in this protocol.

[5.140 RFC 2879](#) Content Feature Schema for Internet Fax (V2)

There are no IPv4 dependencies in this protocol.

[5.141](#) [RFC 2891](#) LDAP Control Extension for Server Side Sorting of Search Results

There are no IPv4 dependencies in this protocol.

[5.142](#) [RFC 2910](#) Internet Printing Protocol/1.1: Encoding and Transport (IPP-E-T)

There are no IPv4 dependencies in this protocol.

[5.143](#) [RFC 2911](#) Internet Printing Protocol/1.1: Model and Semantics (IPP-M-S)

There are no IPv4 dependencies in this protocol.

[5.144](#) [RFC 2912](#) Indicating Media Features for MIME Content

There are no IPv4 dependencies in this protocol.

[5.145](#) [RFC 2913](#) MIME Content Types in Media Feature Expressions

There are no IPv4 dependencies in this protocol.

[5.146](#) [RFC 2919](#) List-Id: A Structured Field and Namespace for the Identification of Mailing Lists

There are no IPv4 dependencies in this protocol.

[5.147](#) [RFC 2938](#) Identifying Composite Media Features

There are no IPv4 dependencies in this protocol.

[5.148](#) [RFC 2965](#) HTTP State Management Mechanism

This document makes many references to the IP addresses of hosts but has no fundamental reasons why they could not be either IPv4 or IPv6 addresses.

[5.149](#) [RFC 2971](#) IMAP4 ID extension

There are no IPv4 dependencies in this protocol.

[5.150](#) [RFC 2987](#) Registration of Charset and Languages Media Features Tags

There are no IPv4 dependencies in this protocol.

[5.151 RFC 3009](#) **Registration of parityfec MIME types**

There are no IPv4 dependencies in this protocol.

[5.152 RFC 3017](#) **XML DTD for Roaming Access Phone Book**

The following 2 sections show an IPv4 limitation.

[6.2.1.](#) **DNS Server Address**

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>
```

[6.2.9.](#) **Default Gateway Address**

The defaultGatewayAddress 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 fairly straightforward to implement elements that are IPv6 aware.

[5.153 RFC 3023](#) **XML Media Types**

There are no IPv4 dependencies in this protocol.

[5.154 RFC 3028](#) **Sieve: A Mail Filtering Language**

There are no IPv4 dependencies in this protocol.

[5.155 RFC 3030](#) SMTP Service Extensions for Transmission of Large and Binary MIME Messages

There are no IPv4 dependencies in this protocol.

[5.156 RFC 3049](#) TN3270E Service Location and Session Balancing

There are no IPv4 dependencies in this protocol.

[5.157 RFC 3059](#) Attribute List Extension for the Service Location Protocol (SLPv2)

There are no IPv4 dependencies in this protocol.

[5.158 RFC 3080](#) The Blocks Extensible Exchange Protocol Core (BEEP)

There are no IPv4 dependencies in this protocol.

[5.159 RFC 3081](#) Mapping the BEEP Core onto TCP

There are no IPv4 dependencies in this protocol.

[5.160 RFC 3111](#) Service Location Protocol Modifications for IPv6

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

[6.0](#) Experimental RFCs

Experimental RFCs typically define protocols that do not have widescale implementation or usage on the Internet. They 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 to an acknowledged problem.

[6.01 RFC 909](#) Loader Debugger Protocol (LDP)

There are no IPv4 dependencies in this protocol.

[6.02 RFC 1143](#) The Q Method of Implementing TELNET Option Negotiation

There are no IPv4 dependencies in this protocol.

[6.03 RFC 1153](#) Digest message format (DMF-MAIL)

There are no IPv4 dependencies in this protocol.

[6.04 RFC 1159](#) Message Send Protocol

There are no IPv4 dependencies in this protocol.

[6.05 RFC 1165](#) Network Time Protocol (NTP) over the OSI Remote Operations Service (NTP-OSI)

The only dependency is in the implementation Appendix:

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

and depending on the implementation this might not even be an issue.

[6.06 RFC 1176](#) Interactive Mail Access Protocol: Version 2 (IMAP2)

There are no IPv4 dependencies in this protocol.

[6.07 RFC 1204](#) Message Posting Protocol (MPP) (MPP)

There are no IPv4 dependencies in this protocol.

[6.08 RFC 1235](#) Coherent File Distribution Protocol (CFDP)

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

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.

[6.09 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 (MSP2)

There are no IPv4 dependencies in this protocol.

[6.11 RFC 1339](#) Remote Mail Checking Protocol (RMCP)

There are no IPv4 dependencies in this protocol.

[6.12 RFC 1440](#) SIFT/UFT: Sender-Initiated/Unsolicited File Transfer (SIFT)

There are no IPv4 dependencies in this protocol.

[6.13 RFC 1459](#) Internet Relay Chat Protocol (IRCP)

There are two spots in this document that are limited to IPv4 addressing.

```

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

```

and

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

It should be relatively simple to add support for IPv6.

[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 protocol.

[6.15 RFC 1505](#) Encoding Header Field for Internet Messages (EHF-MAIL)

There are no IPv4 dependencies in this protocol.

**[6.16 RFC 1528](#) Principles of Operation for the TPC.INT Subdomain: Remote
Printing -- Technical Procedures (REM-PRINT)**

There are no IPv4 dependencies in this protocol.

**[6.17 RFC 1608](#) Representing IP Information in the X.500 Directory
(X500-DIR)**

There are no IPv4 dependencies in this protocol.

[6.18 RFC 1609](#) Charting Networks in the X.500 Directory (X500-CHART)

There are no IPv4 dependencies in this protocol.

**[6.19 RFC 1639](#) FTP Operation Over Big Address Records (FOOBAR)
(FOOBAR)**

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 (MIME-UNI)

There are no IPv4 dependencies in this protocol.

[6.21 RFC 1756](#) Remote Write Protocol - Version 1.0 (RWP)

There are no IPv4 dependencies in this protocol.

[6.22 RFC 1801](#) MHS use of the X.500 Directory to support MHS Routing

There are no IPv4 dependencies in this protocol.

[6.23 RFC 1804](#) Schema Publishing in X.500 Directory

There are no IPv4 dependencies in this protocol.

[6.24 RFC 1806](#) **Communicating Presentation Information in Internet Messages: The Content-Disposition Header**

There are no IPv4 dependencies in this protocol.

[6.25 RFC 1845](#) **SMTP Service Extension for Checkpoint/Restart**

There are no IPv4 dependencies in this protocol.

[6.26 RFC 1846](#) **SMTP 521 Reply Code**

There are no IPv4 dependencies in this protocol.

[6.27 RFC 1873](#) **Message/External-Body Content-ID Access Type (CONT-MT)**

There are no IPv4 dependencies in this protocol.

[6.28 RFC 1874](#) **SGML Media Types (SGML-MT)**

There are no IPv4 dependencies in this protocol.

[6.29 RFC 1986](#) **Experiments with a Simple File Transfer Protocol for Radio Links using Enhanced Trivial File Transfer Protocol (ETFTP) (ETFTP)**

This protocol is IPv4 dependent. See below:

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) (URAS)**

There are no IPv4 dependencies in this protocol.

[6.31 RFC 2066](#) **TELNET CHARSET Option (TOPT-CHARS)**

There are no IPv4 dependencies in this protocol.

[6.32 RFC 2075](#) IP Echo Host Service (IP-Echo)

There are no IPv4 dependencies in this protocol.

[6.33 RFC 2090](#) TFTP Multicast Option (TFTP-MULTI)

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 (X.500-NAME)

There are no IPv4 dependencies in this protocol.

[6.35 RFC 2161](#) A MIME Body Part for ODA (MIME-ODA)

There are no IPv4 dependencies in this protocol.

[6.36 RFC 2162](#) MaXIM-11 - Mapping between X.400 / Internet mail and Mail-11 mail (MAP-MAIL)

There are no IPv4 dependencies in this protocol.

[6.37 RFC 2168](#) Resolution of Uniform Resource Identifiers using the Domain Name System

There are no IPv4 dependencies in this protocol.

[6.38 RFC 2169](#) A Trivial Convention for using HTTP in URN Resolution

There are no IPv4 dependencies in this protocol.

[6.39 RFC 2217](#) Telnet Com Port Control Option (TOPT-COMPO)

There are no IPv4 dependencies in this protocol.

[6.40 RFC 2295](#) Transparent Content Negotiation in HTTP (TCN-HTTP)

There are no IPv4 dependencies in this protocol.

[6.41 RFC 2296](#) HTTP Remote Variant Selection Algorithm -- RVSA/1.0 (HTTP-RVSA)

There are no IPv4 dependencies in this protocol.

[6.42 RFC 2307](#) An Approach for Using LDAP as a Network Information Service (LDAP-NIS)

This protocol assumes IPv4 addressing in its schema. As is:

```
( 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:

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.

but the format defines has been replaced and it is no longer valid.

[6.43 RFC 2310](#) The Safe Response Header Field

There are no IPv4 dependencies in this protocol.

[6.44 RFC 2483](#) URI Resolution Services Necessary for URN Resolution

There are no IPv4 dependencies in this protocol.

[6.45 RFC 2567](#) Design Goals for an Internet Printing Protocol (IPP-DG)

There are no IPv4 dependencies in this protocol.

[6.46 RFC 2568](#) Rationale for the Structure of the Model and Protocol for the Internet Printing Protocol (IPP-RAT)

There are no IPv4 dependencies in this protocol.

[6.47 RFC 2569](#) Mapping between LPD and IPP Protocols

There are no IPv4 dependencies in this protocol.

[6.48 RFC 2649](#) An LDAP Control and Schema for Holding Operation Signatures

There are no IPv4 dependencies in this protocol.

[6.49 RFC 2654](#) A Tagged Index Object for use in the Common Indexing Protocol

There are no IPv4 dependencies in this protocol.

[6.50 RFC 2655](#) CIP Index Object Format for S0IF Objects

There are no IPv4 dependencies in this protocol.

[6.51 RFC 2656](#) Registration Procedures for S0IF Template Types

There are no IPv4 dependencies in this protocol.

[6.52 RFC 2657](#) LDAPv2 Client vs. the Index Mesh

There are no IPv4 dependencies in this protocol.

[6.53 RFC 2756](#) Hyper Text Caching Protocol (HTCP/0.0) (HTCP)

This protocol claims to be aware of both IPv4 & IPv6 addresses but it does make 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]

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

[6.54 RFC 2774](#) **An HTTP Extension Framework**

There are no IPv4 dependencies in this protocol.

[6.55 RFC 2974](#) **Session Announcement Protocol (SAP)**

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

[6.56 RFC 3018](#) **Unified Memory Space Protocol Specification**

This protocol seems to be extensible to support IPv6 but only has definitions for IPv4 addresses in this specification.

[6.57 RFC 3082](#) **Notification and Subscription for SLP**

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

[6.58 RFC 3088](#) **OpenLDAP Root Service An experimental LDAP referral service**

>From the document:

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.

[7.0](#) **Summary of Results**

In the initial survey of RFCs 17 positives were identified out of a total of 262, broken down as follows:

Standards	4 of 24 or 16.67%
Draft Standards	3 of 20 or 15.00%
Proposed Standards	5 of 160 or 3.13%
Experimental RFCs	5 of 58 or 8.62%

Of those identified many require no action because they document outdated and unused protocols, while others are document protocols that are actively being updated by the appropriate working groups. Additionally there are many instances of standards that SHOULD be updated but do not cause any operational impact if they are not updated. The remaining instances are documented below.

The author has attempted to organize the results in a format that allows easy reference to other protocol designers. The following recommendations uses the documented terms "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" described in [RFC 2119](#). They should only be interpreted in the context of [RFC 2119](#) when they appear in all caps. That is, the word "should" in the previous SHOULD NOT be interpreted as in [RFC 2119](#).

The assignment of these terms has been based entirely on the authors perceived needs for updates and should not be taken as an official statement.

[7.1](#) Standards

[7.1.1](#) STD 9 File Transfer Protocol ([RFC 959](#))

Problems have been fixed in [RFC 2428](#) FTP Extensions for IPv6 and NATs

[7.1.2](#) STD 10 Simple Mail Transfer Protocol ([RFC 821](#))

The use of literal IP addresses as part of email addresses (i.e. phil@10.10.10.10) is depreciated and therefore no additional specifications for using literal IPv6 addresses SHOULD NOT be defined.

[7.1.3](#) STD 11 Standard for the format of ARPA Internet Text Messages ([RFC 822](#))

See the above [section 7.1.6](#).

[7.1.4](#) STD 12 Network Time Protocol ([RFC 1305](#))

As documented in [Section 3.12](#) above, there are many specific steps that assume the use of 32-bit IPv4 addresses. An updated specification to support NTP over IPv6 packets MUST be created.

[7.2](#) Draft Standards

[7.2.1](#) Network Time Protocol ([RFC 1305](#))

See [Section 7.1.8](#).

[7.2.2](#) URI ([RFC 2396](#))

URI's allow the literal use of IPv4 addresses but have no specific recommendations on how to represent literal IPv6 addresses. This problem is addressed in [RFC 2732](#), IPv6 Literal Addresses in URL's.

[7.2.3](#) HTTP ([RFC 2616](#))

HTTP allows the literal use of IPv4 addresses but have no specific recommendations on how to represent literal IPv6 addresses. This problem is addressed in [RFC 2732](#), IPv6 Literal Addresses in URL's.

[7.3](#) Proposed Standards

[7.3.1](#) Telnet Terminal LOC ([RFC 946](#))

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, a newer version MAY be created.

[7.3.2](#) Uniform Resource Locators ([RFC 1738](#))

This problem is addressed in [RFC 2732](#), IPv6 Literal Addresses in URL's.

[7.3.3](#) POP3 URL Scheme ([RFC 2384](#))

The problem is addressed in [RFC 2732](#), IPv6 Literal Addresses in URL's.

[7.3.4](#) SLP v2 ([RFC 2608](#))

The problems have been addressed in [RFC 3111](#), Service Location Protocol Modifications for IPv6.

[7.3.5](#) XML DTP For Roaming Access Phone Books ([RFC 3017](#))

Extensions SHOULD be defined to support IPv6 addresses.

7.4 Experimental RFCs

7.4.1 The Coherent File Distribution Protocol ([RFC 1235](#))

This protocol relies on IPv4 and a new protocol standard SHOULD NOT be produced.

7.4.2 IRC Protocol ([RFC 1459](#))

This protocol relies on IPv4 and a new protocol standard SHOULD be produced.

7.4.3 Simple File Transfer Using Enhanced TFTP ([RFC 1986](#))

This protocol relies on IPv4 and a new protocol standard MAY be produced.

7.4.4 TFTP Multicast Option ([RFC 2090](#))

This protocol relies on IPv4 IGMP Multicast and a new protocol standard MAY be produced.

7.4.5 Using LDAP as a NIS ([RFC 2307](#))

This document tries to provide IPv6 support but it relies on an outdated format for IPv6 addresses. A new specification MAY be produced.

8.0 Acknowledgements

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

9.0 Authors Address

Please contact the author with any questions, comments or suggestions at:

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

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

This draft expires in August 2003.