Network Working Group
Request for Comments: 1545
Category: Experimental

D. Piscitello
Bellcore

November 1993

FTP Operation Over Big Address Records (FOOBAR)

Status of this Memo

This memo defines an Experimental Protocol for the Internet community. This memo does not specify an Internet standard of any kind. Discussion and suggestions for improvement are requested. Distribution of this memo is unlimited.

Abstract

This paper describes a convention for specifying longer addresses in the PORT command.

Introduction

This RFC specifies a method for assigning long addresses in the HOST-PORT specification for the data port to be used in establishing a data connection for File Transfer Protocol, FTP (STD 9, RFC 959). This is a general solution, applicable for all "next generation" IP alternatives, and can also be extended to allow FTP operation over transport interfaces other than TCP.

Acknowledgments

Many thanks to all the folks in the IETF who casually mentioned how to do this, but who left it to me to write this RFC. Special thanks to Rich Colella, Bob Ullmann, Shawn Ostermann, Steve Lunt, and Brian Carpenter who had the time and decency to comment on the initial draft. :-)

1. Background

The PORT command of File Transfer Protocol allows users to specify an address other than the default data port for the transport connection over which data are transferred. The PORT command syntax is:

```
PORT <SP> <host-port> <CRLF>
```

The <host-port> argument is the concatenation of a 32-bit internet <host-address> and a 16-bit TCP <port-address>. This address information is broken into 8-bit fields and the value of each field is transmitted as a decimal number (in character string

representation). The fields are separated by commas. A port command is thus of the general form "PORT h1,h2,h3,h4,p1,p2", where h1 is the high order 8 bits of the internet host address.

To accommodate larger network addresses anticipated for all IP "next generation" alternatives, new commands and reply codes are needed for FTP. This memo addresses these needs.

2. The LPRT Command

The LPRT command allows users to specify a "long" address for the transport connection over which data are transferred. The LPRT command syntax is:

LPRT <SP> <long-host-port> <CRLF>

The <long-host-port> argument is the concatenation of the following fields;

- an 8-bit <address-family> argument (af)
- an 8-bit <host-address-length> argument (hal)
- a <host-address> of <host-address-length> (h1, h2, ...)
- an 8-bit <port-address-length> (pal)
- a <port-address> of <port-address-length> (p1, p2, ...)

The <address-family> argument takes the value of the version number of IP (see Assigned Numbers, STD 2, RFC 1340), or generally speaking, an Internet layer protocol. Relevant assigned IPng version numbers are:

Decimal	Keyword
0	reserved
1-3	unassigned
4	Internet Protocol (IP)
5	ST Datagram Mode
6	SIP
7	TP/IX
8	PIP
9	TUBA
10-14	unassigned
15	reserved

The value of each field is broken into 8-bit fields and the value of each field is transmitted as an unsigned decimal number (in character string representation, note that negative numbers are explicitly not permitted). The fields are separated by commas.

A LPRT command is thus of the general form

LPRT af,hal,h1,h2,h3,h4...,pal,p1,p2...

where h1 is the high order 8 bits of the internet host address, and p1 is the high order 8 bits of the port number (transport address).

3. The LPSV Command

The L(ONG) PASSIVE command requests the server-DTP to listen on a data port other than its default data port and to wait for a connection rather than initiate one upon receipt of a transfer command. The response to this command includes the address family, host address length indicator, host address, port address length, and port address this server is listening on. The reply code and text for entering the passive mode using a long address is 228 (Interpretation according to FTP is: positive completion reply 2yz, connections x2z, passive mode entered using long address xy8). The suggested textual message to accompany this reply code is:

228 Entering Long Passive Mode (af,hal,h1,h2,h3,h4...,pal,p1,p2...)

4. Permanent Negative Completion Reply Codes

The negative completion reply codes that are associated with syntax errors in the PORT and PASV commands are appropriate for the LPRT and LPSV commands (500, 501). An additional negative completion reply code is needed to distinguish the case where a host supports the LPRT or LPSV command, but does not support the address family specified. Of the FTP function groupings currently defined for reply codes (syntax, information, connections, authentication and accounting, and file system), "connections" seems the most logical choice; thus, an additional negative command completion reply code, 521 is added, with the following suggested textual message:

521 Supported address families are (af1, af2, ..., afn)

Where (af1, af2, ..., afn) are the values of the version numbers of the "next generation" or other protocol families supported. IP address noted earlier.

5. Rationale

An explicit address family argument in the LPRT command and LPSV reply allows the Internet community to experiment with a variety of "next generation IP" alternatives within a common FTP implementation framework. (It also allows the use of a different address family on the command and data connections.) An explicit length indicator for the host address is necessary because some of the IPNG alternatives make use of variable length addresses. An explicit host address is necessary because FTP says it's necessary.

The decision to provide a length indicator for the port number is not as obvious, and certainly goes beyond the necessary condition of having to support TCP port numbers. Currently, at least one IPng alternative (TP/IX) supports longer port addresses. And given the increasingly "multi-protocol" nature of the Internet, it seems reasonable that someone, somewhere, might wish to operate FTP operate over Appletalk, IPX, and OSI networks as well as TCP/IP networks. (In theory, FTP should operate over *any* transport protocol that offers the same service as TCP.) Since some of these transport protocols may offer transport selectors or port numbers that exceed 16 bits, a length indicator may be desirable. If FTP must indeed be changed to accommodate larger network addresses, it may be prudent to determine at this time whether the same flexibility is useful or necessary with respect to transport addresses.

6. Conclusions

The mechanism defined here is simple, extensible, and meets both IPNG and possibly multi-protocol internet needs.

7. References

STD 9, RFC 959 Postel, J., and J. Reynolds, "File Transfer Protocol", STD 9, RFC 959, USC/Information Sciences Institute, October 1985.

STD 2, RFC 1340 Reynolds, J., and J. Postel, "Assigned Numbers", STD 2, RFC 1340, USC/Information Sciences Institute, July 1992. (Does not include recently assigned IPv7 numbers).

STD 3, RFC 1123 Braden, R., Editor, "Requirements for Internet Hosts - Application and Support", STD 3, RFC 1123, USC/Information Sciences Institute, October 1989.

8. Security Considerations

Security issues are not discussed in this memo.

9. Author's Address

David M. Piscitello
Bell Communications Research
NVC 1C322
331 Newman Springs Road
Red Bank, NJ 07701

EMail: dave@mail.bellcore.com