IPv6和NATs的FTP扩展

本备忘录状态

本文档为Internet community说明了一种Internet标准追踪协议。它需要进一步进行讨论和建议以得到改进。请查询“Intetnet 正式协议标准” (STD 1)的最新版本，以获得这个协议的标准化状况。发布本备忘录不受限制。

版权声明

概述

FTP规范假定在下面的网络协议中使用32位的网络地址（在IP version 4中定义）。根据IP协议第6版中的规定，网络地址不再是32位。本文指定的FTP扩展将允许协议工作在IPv4和IPv6之上。另外，该架构定义也能支持未来的附加网络协议。

1. 介绍

本文中能看到的像MUST和SHOULD这样的关键字，都已在RFC 2119 [Bra97]做过定义。

文件传输协议[PR85]仅仅提供了使用IPv4数据连接进行通信的能力。FTP假设网络地址的长度是32位。不过，在IP协议第6版[DH96]的规定中，地址长度将不再是32位，RFC 1639 [Pis94]说明了FTP能够在不同的网络协议中使用的扩展。不幸的是，该机制在多协议环境中是失败的。在IPv4 and IPv6两者之间传输期间，FTP需要能使数据传输以便过渡网络协议的能力。

本文提供了描述了一种方法，它让FTP在不同于IPv4的网络协议下，可以通信临界点信息的数据连接。在本文的描述中，FTP指令PORT和PSAV分别被EPRT和EPSV替代。第二节概述了EPRT指令，第三节概述了EPSV指令。第四节定义了这两种新的FTP指令的使用方法。第五节简要地介绍了安全方面的考虑。最后，第六节得出结论。

2. EPRT指令

EPRT指令允许为数据连接指定扩展的地址说明。扩展地址必须由像网络和传输地址这样的网络协议组成。EPRT的格式是:

EPRT<space><d><net-prt><d><net-addr><d><tcp-port><d>

EPRT关键字之后必须跟随一个空格（ASCII 32）。空格之后，必须定义个分界符（<d>）。该分界符必须是一个包含在ASCII 码33-126范围内的

字符。推荐使用字符“|”（ ASCII 124），除非该字符在网络地址的编码中需要使用。

<net-prt>参数必须是一个由IANA在最新编号的RFC中定义的地址族数字（RFC1700 [RP94]就是这样的文档）。该数字指出要使用的协议（而且还隐含地指出了地址的长度）。本文的例子中将使用两个地址族数字，根据下表的定义：

AF数字	协议
1	IP协议版本4[Pos81a]
2	IP协议版本6 [DH96]

<net-addr>是一个协议特定字串，用来表示网络地址。这两个地址族如上所指出的（AF 数字1和2），地址必须使用下列格式：

AF 数字	地址格式	例
1	用点来分隔的十进制数	132.235.1.2
2	在[HD96]中定义的 IPv6 字串表示法	1080::8:800:200C:417A

<tcp-port>参数必须是主机监听数据连接所在的端口号的字串表示。以下是EPRT指令的例子：

EPRT |1|132.235.1.2|6275|

EPRT |2|1080::8:800:200C:417A|5282|

第一条指令说明服务器可以使用IPv4打开一个数据连接到主机“132.235.1.2”在TCP的端口6275。第二条指令说明服务器可以使用IPv6网络协议，以网络地址“1080::8:800:200C:417A”的5282端口打开一个TCP数据连接。

在收到一个有效的EPRT指令之后，服务器必须返回一个200（指令正常）代码。标准的否认错误代码500和501 [PR85]足够用来处理包括EPRT指令在内的大多数的错误（比如语法错误）。不过，还需要其他错误代码。应答码522指出服务器不能指出网络协议的请求。新的错误码的解释是：

5yz 拒绝完成
x2z 连接
xy2 扩展端口失败——未知的网络协议

应答文本的一部分必须指明服务器支持的网络协议。如果网络协议不被支持，应答码文本的格式必须是：

<text stating that the network protocol is unsupported> \
(prot1,prot2,...,protn)

上面指定的数字代码和字符“(”和“)”之间的协议信息都是为了软件自动接收应答码而设置的。数字代码和“(”之间的文本信息是为了人类用户而设置的，该文本信息可以是任意的文本，但不能包括字符“(”和“)”。在上面的情况中，文本应指出服务器不支持的EPRT指令的网络协议。括号内的网络协议地址族数字列表必须用逗号来分隔。下面是两个应答字串的例子：

Network protocol not supported, use (1)

Network protocol not supported, use (1,2)

3. EPSV指令

EPSV指令要求服务器监听一个数据端口，等待数据连接。EPSV有一个可选择的参数。该指令的应答仅含有监听连接的TCP端口号。应答信息的格式和EPRT指令的参数类似。另外，该格式为EPSV回应的未来需求保留了一个网络协议或网络地址的位置。为了使用扩展的地址进入被动模式，回应码必须是229。代码的解释，依照[PR85]：

2yz 确定完成
x2z 连接
xy9 已进入扩展被动模式

对于EPSV指令的回应文本必须是如下格式：

<text indicating server is entering extended passive mode> \
(<d><d><d><tcp-port><d>)

圆括号内的一部分附加字串，必须是根据EPRT指令打开的数据连接得到的精确字串，像上面说明的那样。

圆括号内包含的前两个域必须是空的。第三个域必须是主机监听数据连接所在的端口号的字串表示。另外，建立数据连接的网络地址必须和控制连接的网络地址是同一个。下面是一个应答字串的例子：

Entering Extended Passive Mode (|||6446|)

标准的否认错误代码500和501 [PR85]足够用来处理包括EPSV指令在内的大多数的错误（比如语法错误）。

如果EPSV指令不带参数，服务器将选择基于控制连接使用的协议作为数据连接的网络协议。不过，在代理FTP的情况下，该协议不适合在两个服务器之间通信。因此，客户端需要能指定一个确定的协议。如果服务器返回一个即将连接到端口的主机不支持的协议，客户端必须发出一个ABOR（abort）指令以允许服务器关闭监听连接。然后客户端可以发送EPSV指令请求一个特殊的网络，格式如下:

EPSV<space><net-prt>

如果服务器支持被请求的协议，它将使用这个协议。如果不支持，服务器必须返回一个第二节中描述的522错误。

最后，EPSV指令可以使用参数“ALL”通知网络地址译码器，该EPRT指令（也可以是其他指令）将不再使用。下面是这个指令的例子:

EPSV<space>ALL

在接收到EPSV ALL指令后，服务器必须拒绝除EPSV以外的所有数据连接设置指令（也就是，EPRT， PORT， PASV，等）。第四节中将对EPSV指令的使用做进一步的说明。

4. 指令的用处

对于所有的FTP传输，将在两台相同的机器间建立控制和数据连接。使用EPSV指令有利于执行通过防火墙或网络地址译码器（NATs）的传输。RFC 1579 [Bel94]推荐在防火墙之后使用被动的指令，因为防火墙一般不允许引入的连接（当使用PORT（EPRT）指令时需要）。另外，使用本文中定义的EPSV指令在传输过程中不需要NATs转换网络地址。如果使用EPRT指令，NAT将转换地址。最后，如果客户端发出了一个“EPSV ALL”指令，NATs 可以提出一个用“快速路径”通过译码器的连接，因为EPRT指令不会再被用到，所以，数据的一部分的转换就不需要了。如果客户端只是希望双路的FTP传输，它将一可能就发出这个指令。如果一个客户端稍侯发现在发出EPSV ALL指令之后，必须进行三路的传输，将开始进行一个新的FTP传输过程。

5. 安全问题

作者不认为这些FTP的改变会引入新的安全问题。为配合问题的进展[AO98]，应该有更多的关于FTP安全问题和技术的讨论，以减少这些安全问题。

6. 结论

本文的扩展说明能使FTP在不同的网络协议上运行。

参考书目

[AO98]
Allman, M., and S. Ostermann, "FTP安全考虑", Work in Progress。

[Bel94]
Bellovin, S., "防火墙——友好FTP", RFC 1579, 1994年2月。

[Bra97]
Bradner, S., "使用RFCs的关键字来指出需求级别", BCP 14, RFC 2119, 1997年3月。

[DH96]
Deering, S., and R. Hinden, "IP协议版本6 (IPv6) 规范", RFC 1883, 1995年12月。

[HD96]
Hinden, R., and S. Deering, "IP协议版本6地址结构体系", RFC 2373, 1998年7月。

[Pis94]
Piscitello, D., " FTP 在大地址记录上的操作(FOOBAR)", RFC 1639, 1994年6月。

[Pos81a]
Postel, J., "IP协议", STD 5, RFC 791, 1981年9月。

[Pos81b]
Postel, J., "传输控制协议", STD 7, RFC 793, 1981年9月。

[PR85]
Postel, J., and J. Reynolds, "文件传输协议 (FTP)", STD 9, RFC 959, 1985年10月。

[RP94]
Reynolds, J., and J. Postel, "分配编号", STD 2, RFC 1700, October 1994. 参看: http://www.iana.org/numbers.html

作者地址

（略——译者注）

完整的版权声明

原文：RFC 2428《FTP Extensions for IPv6 and NATs》
译者：张偶 2002年5月