



链滴

FTP 的 PORT 模式, PASV 模式, FXP 协议 和在 java 中使用时的一些问题

作者: [turbidsoul](#)

原文链接: <https://ld246.com/article/1356709266044>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

这是最近一礼拜遇到的一个很头疼的问题，由于我们的系统需要需要用到ftp的fxp进行两个ftp服务器间进行数据传输。但是就是在这上出现了让我很郁闷的问题，连着几天，一个坑接一个坑，连续解决5, 6个问题，到今天下午才算是把所有问题解决。在讲述我遇到的问题之前我们先了解下几个概念：

首先是PORT模式和PASV模式，也就是FTP的主动模式和被动模式：> * 主动模式的FTP连接建

要遵循以下步骤：>> 1. 客户端打开一个随机的端（端口号大于1024，在这里，我们称它为x），同时一个FTP进程连接至服务器的21号命令端口。此，源端口为随机端口x，在客户端，远程端口为21，在服务器。>

gt; 2. 客户端开始监听端口（x+1），同时向服务器发送一个端口命令（通过服务器的21号命令端口，此命令告诉服务器客户端正在监听的端口号并且已准备好从此端口接收数据。这个端口就是我们所的数据端口。>> 3. 服务器打开20号源端口并且建立和客户

数据端口的连接。此时，源端口为20，远程数据端口为（x+1）。>

gt; 4. 客户端通过本地的数据端口建立一个和服务器20号端口的连接，然后向服务器发送一个应答，诉服务器它已经建立好了一个连接。>

gt; > * 被动模式FTP: 为了解决服务器发起到客户的连接的问题，人们开发了一种不同的FT连接方式。这就是所谓的被动方式，或者叫做PASV，当客户端通知服务器它处于被动模式时才启用。

gt;> 在被动方式FTP中，命令连接和数据连接都由客户端发起，这样就可以解决从服务器到客端的数据端口的入方向连接被防火墙过滤掉的问题。>

gt; 当开启一个FTP连接时，客户端打开两个任意的非特权本地端口（N >

1024和N+1）。第一个端口连接服务器的21端口，但与主动方式的FTP不同，客户端不会提交PORT令并允许服务器来回连它的数据端口，而是提交 PASV命令。这样做的结果是服务器会开启一个任意非特权端口（P > 1024），并发送PORT P命令给客户端。然后客户端发起从本地

口N+1到服务器的端口P的连接用来传送数据。>

gt; 对于服务器端的防火墙来说，必须允许下面的通讯才能支持被动方式的FTP：>

gt; >> 1. 从任何大于1024的端口到服务器的21端口（客户端的初始化连接）

gt;> 2. 服务器的21端口到任何大于1024的端口（服务器响应到客户端的控制端口的连接）

gt;> 3. 从任何大于1024端口到服务器的大于1024端口（客户端初始化数据连接到服务器指定任意端口）>> 4. 服务器的大于1024端口到远程的大于1024的端

（服务器发送ACK响应和数据到客户端的数据端口） * FXP全称：File Exchange Protocol（文件交换协议）（貌似很人对这个全称持有不赞同的意见）> F

P是一个服务器之间传输文件的协议，这个协议控制着两个支持FXP协议的服务器，在无需人工干预情况下，自动地完成传输文件的操作。在我们的客户机上，可以简单的发送一个传输的命令，即可控服务器从另一个FTP服务器上下载一个文件，下载过程中，无须客户机干预，客户机甚至可以断网关

。这种协议通常只适用于管理员作管理的用途，在一般的公开FTP服务器上，是不会允许FXP的，因这样会浪费服务器资源，而且有可能出现安全问题。以上这些都是我从百度百科扒出来的，也只是给

位简单的了解先FTP和我这几天遇到的问题的技术背景 我们的系统环境是这样的，数据是从A服务器(统是window server 2003, ftp服务器serv-u, 使用的默认模式配置，应该是被动模式，但是貌似是

被动都支持) 先通过FXP传送到B服务器(中转服务器，是linux, ftp服务器是vsftpd, 被动模式，不能被动同时开启，会出现5XX错误)，然后在使用fxp分别传送到c1,c2服务器(windows server 2003, ftp

务器是serv-u, 同样是使用的默认配置模式)，我们在项目中使用的是Apache的net包的ftp功能进行F P操作，下面是我们使用FXP功能的代码： FTPClient srcServer = connect(source, ftpLog); FTPClient

argetServer = connect(target, ftpLog); // Let's just assume success for now. srcServer.enterRe otePassiveMode(); targetServer.enterRemoteActiveMode(InetAddress.getByNam

(srcServer.g tPassiveHost()), srcServer.getPassivePort()); // Although you would think the store command hould be sent to // server2 // first, in reality, ftp servers like wu-ftp

d start accepting data // co nections right after entering passive mode. Additionally, they // don't even send the positive reliminary reply until after the // transfer is completed (in the case of passive mode transfers).

// Therefore, calling store first would hang waiting for a // preliminary // reply. if (targetServer remoteStoreUnique(encodepath(target.getPath())) && srcServer.remoteRetrieve(so

rce.getPath())) { // if(ftp1.remoteRetrieve(file1) && ftp2.remoteStore(file2)) { // We ave to fetch the positive completion reply. srcServer.completePendingCommand(); targetServ

r.completePendingCommand(); } 这段代码是我们头从apache网站中扒下来，稍作改动，目的就是了从src服务器向target服务器发送一个文件，文件在10G左右。代码很简单创建两个FTP连接，分别

接源服务器和目标服务器，告诉源服务器使用被动模式，并告诉目标服务器使用主动模式，并告诉目服务器源服务器的ip和端口。最后面的几句就简单的，if判断条件里&

amp;的左边的语句 `targetServer.remoteStoreUnique(encodepath(target.getPath()))`的意思，目标服务器接收数据，`&`；
边`srcServer.remoteRetrieve(source.getPath())`是告诉源服务器发送数据，这样两个服务器就会进行数据传输，直到传输完成，如果这两个服务器不支持FXP，那么if语句里面的两句话就会返回false。Apache的代码中不难看出，进行FXP需要两个FTP服务器一个是被动模式一个是主动模式，将被动模式的服务器的地址和端口告诉主动模式的服务器，有主动模式服务器连接被动模式服务器进行数据传输。这里不一定要源服务器是主动模式，目标服务器是主动模式，反过来也是完全可以的，我们这里之的代码是目标服务器是被动模式，源服务器是主动模式，但是由于系统架构的变化所以这里的代码也了变化。**这里需要注意的是：**`* srcServer.enterRemotePassiveMode();`这句话需要在前面，`targetServer.enterRemoteActiveMode(InetAddress.getByAddress(srcServer.getPassiveHost()), srcServer.getPassivePort());`一定要在后面，因为主动模式的服务器需要被动模式的服务器的地址和端口。`* targetServer.remoteStoreUnique(encodepath(target.getPath())) & & srcServer.remoteRetrieve(source.getPath())`这两句也是有顺序的，最开始的时候`srcServer.remoteRetrieve(source.getPath())`在`targetServer.remoteStoreUnique(encodepath(target.getPath()))`之前，在两个都是serv-u的情况下是没有问题的，但是后面我们的源ftp服务器换成了vsftpd这样就不可以，在内网中进行测试时候，只有20字节的速度，而在公网中这两个就干脆连不上，但是改成现在的样子速度就能跑满。我在调试中还发现，vsftpd是不能同时启动两种模式的，不然在执行命令的时候会返回一个5XX的错误只能启动主动或者被动模式，网上的一些文章发的配置信息很多都是主被动同时启动的，我认为这是的，但是我有一点很疑惑我们的serv-u使用的是默认配置，这种的配置貌似既是主动也是被动，有点理解，也就是说客户端连接的时候需要什么模式就会切换到什么模式。还想提醒一下不论什么程序，要是在程序中使用FTP操作的，需要注意FTP是否支持一些命令，比方说vsftpd就不支持mlst这个命令，还有serv-u默认配置下，是不能使用list命令的，会返回425错误，但是serv-u支持mlst命令，等等种问题。下午在闲下来的时候在和同时讨论这个问题的时候，得到结论是：FTP的协议规范可能太简单，很多服务器的开发者，对协议支持的都不太一样，虽然大部分功能是一样的，但是在一些细节上有大的差别，这些差别和可能就是我们需要进行攻关的地方。当然我们也在哪里拌了好几天，一个问题一个问题的解决，虽然耽误了很多时间但是收获也很多。