

第26章 文件传输协议

作者：Anne Carasik

本章内容包括：

- FTP和TFTP在网络世界中的作用
- 使用FTP传输文件
- 使用TFTP

将文件从一个系统发送到另一个系统是网络功能的重要组成部分。用户可以使用 email 完成这一任务，但要获得较快的传输速度，则可以使用文件传输协议，它包括文件传输协议 (File Transfer Protocol, FTP) 和普通文件传输协议 (Trivial File Transfer Protocol, TFTP)。远程拷贝将在第28章中讨论。

26.1 FTP和TFTP在网络世界中的作用

大多数站点都使用 Web 在服务器间传送文件，文件传输协议 FTP 和 TFTP 已不如过去那么流行。然而，并不是所有系统都运行了 Web 服务器。因此，在文件传输时，仍需要使用 FTP 和 TFTP。

即使安装了 Web，将文件传送到服务器也比较困难。许多 Web 命令不如 FTP 命令健壮，不支持许多开关和功能。

许多 FTP 应用仍被广泛使用。这些命令行方式的应用允许用户在没有安装 Web 服务器的系统间传输文件。大多数 UNIX 和 Linux 系统，均安装了 FTP 应用。

因为大多数 FTP 应用都不基于图形界面 (有一些应用包含图形接口)，本章使用 UNIX 系统中基于命令行方式的 FTP 应用程序以便更好地说明 FTP 协议。

虽然 TFTP 不如 FTP 常见，但本章还是对它进行了一定的阐述。

26.2 使用FTP传输文件

FTP 是 Internet 和 IP 网络上传输文件的通用方法。在万维网 (WWW) 出现以前，用户使用命令行方式传输文件，最通用的应用程序是 FTP。虽然，目前大多数用户选择使用 Email 和 Web 传输文件，以代替命令行方式的接口，但是 FTP 仍然被使用。

FTP 是 TCP/IP 应用，它工作在 OSI 模型的第七层，TCP 模型的第四层上，在 RFC 959 中定义。FTP 使用 TCP 传输而不是 UDP。FTP 位于 OSI 和 TCP 模型的应用层 (见图 26-1)。

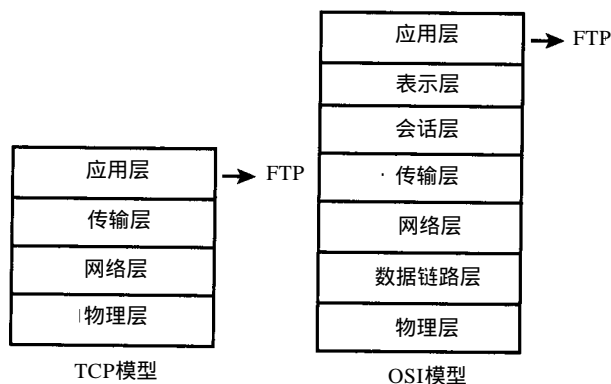


图26-1 FTP在OSI和TCP模型中的设置

起初，FTP是ARPANET网络中计算机间的文件传输协议，ARPANET是美国国防部组建的老网络，于1960~1980年使用。在那时，FTP的主要功能是在主机间高速可靠地传输文件。目前FTP仍然保持其可靠性，即使在今天，它还允许文件远程存取。

这使得用户可以在某个系统上工作，而将文件存贮在别的系统。例如，如果某用户运行Web服务器，需要从远程主机上取得HTML文件和CGI程序在本机上工作，他需要从远程存储站点获取文件(远程站点也需安装Web服务器)。

当用户完成工作后，可使用FTP将文件传回到Web服务器。采用这种方法，用户无需使用Telnet登录到远程主机进行工作。

26.2.1 FTP连接

当用户建立FTP连接时，将与两个端口建立联系：端口20和21。这两个端口有不同的功能——端口20是数据端口，21是控制端口。

图26-2说明FTP客户端是如何与远程服务器FTP守护进程建立连接的。

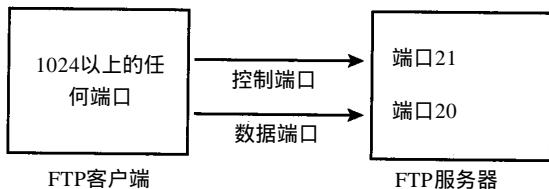


图26-2 FTP客户端与FTP服务器建立连接的过程

1. 控制端口

FTP控制端口用于交换命令信息及对命令的响应信息。其工作过程类似于军官和新兵。军官发出命令，新兵迅速响应命令。

在FTP中，如果用户发送命令如get this file，其响应如下：

```
200 PORT command successful.
150 Opening ASCII mode data connection for .message (127 bytes).
226 Transfer complete.
local: .message remote: .message
135 bytes received in 1.4 seconds (0.09 Kbytes/s)
```

控制端口的命令以PORT开始，这表示命令通过控制端口21而不是数据端口进行传输。命令也允许通过设置被动方式(PASV)发送到数据端口。

控制连接用于显示数据的状态、所处的模型、传输状态及数据传输总量。控制连接使用Telnet协议，在RFC 495中定义。

PASV命令允许原始FTP连接，它不使用用户熟悉的命令。PASV连接仅使用一个端口进行文件传输(端口20)，而不像其他连接那样使用端口20和21。

用户也可以使用相同的命令进行PASV连接，但是，使用控制命令的“cooked”连接更加安全并且便于管理，因为管理员可以设置用户可以使用的命令。

2. 数据端口

数据端口是FTP数据文件发送到FTP服务器(ftp)时使用的端口。

注意 FTP服务器，ftp是UNIX系统服务器守护进程，它可能与其他操作系统不同。

与控制端口不同，控制端口是用户发送命令的端口，数据端口是用户传输文件的端口，一般为端口号为20。

与数据端口的连接一般称为被动(passive)连接，它有自己的一组命令。所有数据连接均知

道文件的模式、文件类型和文件传输量。用户还可以仅传送文件的一部分，虽然并没有太多很实际的应用要求这么做。

一般情况下，命令通过控制端口发送，但用户也可直接使用被动 FTP命令控制数据端口。

被动FTP命令

PASV命令不直接被用户使用，其用户接口为 PORT命令。虽然不是每一台FTP服务器均支持被动命令(RFC 1123)，但大多数服务器都支持RFC 1123。

用户可以通过telnet到端口21调用这些命令：

```
tigerlair:/home/stripes- telnet localhost 21
然后将会看到：

Trying 1.2.3.4...
Connected to localhost.
Escape character is '^]'

220 mysystem.tigerlair.com FTP server (SunOS 5.6) ready.
```

接着，所有用户敲入的命令都是被动命令。

现在，PWD命令将显示用户当前路径：

```
PWD

257 '/home/stripes' is current directory.
```

如果用户使用 ‘ CDUP ’ 被动FTP命令，将进入到用户当前目录的上一级目录。

```
CDUP

250 CWD command successful.
```

现在可用PWD显示当前路径：

```
PWD

257 '/home' is current directory.
```

用户可以看到其他用户的目录。

在每个用户命令前并未加上 PORT命令，因为命令并未使用控制端口，而是直接访问了数据端口。

26.2.2 使用FTP客户端建立连接

使用FTP命令打开连接，在FTP命令后加入主机名即可。FTP的语法格式如下：

```
ftp [ 开关 ] [ 主机名称 ]
```

表26-1 ftp命令行属性

属 性	功 能
-d	启用调试功能
-i	关闭提示功能
-n	关闭自动登录
-v	显示远程服务器响应
-g	关闭通配符功能

例如，用户需要与ftp.tigerlair.com建立连接，可以敲入如下命令：

```
tigerlair:/home/stripes -ftp ftp.example.org
```

如果用户需要FTP客户端完成平时不常用的任务，可以使用一些命令属性。表 26-1列出了ftp客户方可以使用的命令属性。

当用户需要使用FTP客户方关闭自动登录功能（FTP客户端自动提示用户名及登录口令），并且在多文件传输时不显示提示功能，可用以下命令：

```
ftp -i -n ftp.example.org
```

当我们不使用任何命令属性时，即使用最初的ftp命令：

```
tigerlair:/home/stripes-ftp ftp.example.org
```

将获得以下消息：

```
Connected to ftp.example.org.  
220 ftp.example.org FTP Server (Version wu-2.4.2-VR16(1)  
Wed Apr 7 15:59:03 PDT 1999) ready.  
Name (ftp.example.org:stripes):
```

由以上消息可以看到，登录到远程节点的缺省用户名为本地帐号名（在本例中为stripes），如果登录用户名不同，必须敲入正确的用户名：

```
Connected to ftp.example.org.  
220 ftp.example.org FTP Server (Version wu-2.4.2-VR16(1)  
Wed Apr 7 15:59:03 PDT 1999) ready.  
Name (ftp.example.org:stripes): ahc  
Password:
```

注意 不要使用example.org作为FTP站点。它仅用于示例说明。如果用户需要使用FTP客户方而不愿登录到远程站点，可以使用FTP客户端连接本地主机。

用户将被提示输入口令，确认口令输入正确。口令不在屏幕上显示，甚至敲入时不显示星号。

下面，用户将获得FTP提示符，如下：

```
ftp>
```

提示符表明用户可以输入命令传输文件，这通常称为命令解释器（command interpreter）。

1. FTP 命令解释器

ftp命令解释器使用户可与FTP客户方进行交互，它使用户可以通过命令打开/关闭特定连接，传输文件、改变传输文件类型等，而不需要断开FTP。

表26-2列出了用户在交互模式下使用FTP时可使用的命令。

表26-2 交互式模式下的FTP命令

命 令	功 能
! <local command>	在本地系统运行命令
\$<macro>	运行宏
account [password]	应用其他用户名/口令以获取额外的对远程系统的访问权限
ascii	将文件传输模式设为ASCII
append localfile [remote file]	将本地文件添加到远程文件中（文件在远程计算机中）
bell	文件传输完成后响铃

(续)

命 令	功 能
binary	将文件传输模式设为二进制方式
bye, exit	退出，结束远程连接
case	打开case功能
close, disconnect	关闭远程连接
cd <directory>	改变远程系统目录名
cd up	进入父目录
debug <level>	设置调试级别
dir	打印路径名
delete <filename>	在远程计算机中删除文件
get <filename>	将文件从远程系统下载到本地系统
glob	在文件传输时使用通配符
hash	每传输 1024 字节打印一个 “#”
help	打印帮助文件
lcd<directory>	改变本地系统路径
ls	打印远程主机目录
lpwd	打印本地主机工作路径
macdef <macroname>	定义宏
mdel <filename (s)>	删除多个宏文件
mdir <filename (s)>	打印多个文件的路径
mget <filename (s)>	将多个文件从远程系统下载到本地系统
mkdir <directory>	在远程计算机上创建路径
mput <filename (s)>	将多个文件上载到远程计算机
open <site>	打开到<site>的连接
prompt	交互式提示
put <filename>	将文件上载到远程计算机
pwd	显示当前工作路径
user [user] [password]	登录到远程计算机系统
verbose	激活冗余模式

通过对上述命令的运用，用户可以使用 FTP 完成多种功能，为了使用户有一个列形象具体的认识，下面一小节将列举一个具体的 FTP 会话实例。

2. FTP 会话示例

并非所有的 FTP 会话均包含在以下示例之内，示例仅对用户使用 FTP 经常碰到的情况加以说明。首先，用户需要与远程 FTP 站点建立连接。

```
tigerlair:/home/stripes- ftp ftp.example.org
Connected to ftp.example.org.
220 ftp.example.org FTP Server (Version wu-2.4.2-VR16(1)
Wed Apr 7 15:59:03 PDT 1999) ready.
Name (ftp.example.org:stripes): ahc
Password:
230 User ahc logged in.
ftp>
```

至此，用户可以访问远程节点。当需要查看远程节点的文件时，有两种列出文件的方法，

ls和dir。注意FTP列的是UNIX系统。下面是ls命令的输出：

```
ftp> ls
200 PORT command successful.
150 ASCII data connection for /bin/ls (1.2.3.4,52262) (0 bytes).
ISS-SOLARIS.TAR
Mail
ar302sol.tar.Z
c-files.tar.gz
debug.ssh
deshadow.c
e205-pdf.pdf

226 ASCII Transfer complete.
468 bytes received in 0.0077 seconds (59.35 Kbytes/s)
ftp>
```

dir命令的输出如下：

```
ftp> dir
200 PORT command successful.
150 ASCII data connection for /bin/ls (1.2.3.4,52263) (0 bytes).
total 56220
drwxr-xr-x 16 ahc users 1536 May 25 09:22 .
drwxr-xr-x 6 root other 512 Oct 27 1998 ..
-rw-r--r-- 1 ahc users 168 May 7 11:26 .Xauthority
-rw-r--r-- 1 ahc users 424 Dec 2 14:43 .alias
-rw-r--r-- 1 ahc users 313 Jun 2 1998 .cshrc
drwxr-xr-x 11 ahc users 512 May 7 11:27 .dt
-rwxr-xr-x 1 ahc users 5111 Nov 2 1998 .dtprofile
drwx----- 2 ahc users 512 Jan 27 15:04 .elm
-rw-r--r-- 1 ahc users 174 Dec 2 14:45 .login
-rw-r--r-- 1 ahc users 556 Dec 2 15:32 .tcshrc
-rw----- 1 ahc users3655680 Jan 4 08:27 ISS-SOLARIS.TAR
drwx----- 2 ahc users 512 Dec 8 16:01 Mail
-rw-r--r-- 1 ahc users5919933 Nov 3 1998 ar302sol.tar.Z
-rw----- 1 ahc users 14605 May 13 10:43 c-files.tar.gz
-rw----- 1 ahc users 1818 Mar 8 12:09 debug.ssh
-rw-r--r-- 1 ahc users 2531 Jan 7 10:07 deshadow.c
-rw-r--r-- 1 ahc users 54532 Nov 19 1998 e205-pdf.pdf
-rw-r--r-- 1 ahc users 898279 Jan 6 09:50 neotr122.zip
226 ASCII Transfer complete.
3830 bytes received in 0.047 seconds (79.06 Kbytes/s)
ftp>
```

dir命令不仅显示出ls命令列举出的文件的更详细信息，而且它还列出了UNIX的隐含文件（以.开头的文件）。

注意 在UNIX系统中，ls和dir均可列举文件，但dir可显示文件的信息更为详细，包括最近修改时间、文件所有者、权限和大小等。如果用户FTP登录到NT或Windows系统，ls将不能工作。仅dir可用来列举文件，但它不能列出文件的详细信息。

列出系统当前目录中的文件后，可以从远程系统获取文件并将它存放入本地系统。用户

在下载文件前，可以先检查本地工作路径，以确定文件被下载在正确的位置。

```
ftp> !pwd
/home/stripes
ftp>
```

如果用户想要将文件下载到 Home 目录，则不需做任何修改就可下载文件。

```
ftp> get notes
200 PORT command successful.
150 ASCII data connection for notes (1.2.3.4,52264) (210 bytes).
226 ASCII Transfer complete.
local: notes remote: notes
226 bytes received in 0.019 seconds (11.81 Kbytes/s)
ftp>
```

从远程系统下载到本地计算机的文件名将不会发生改变，除非用户指定新的文件名。要修改文件名，需使用以下语法格式：

`get 远程文件名 用户指定的本地文件名`

因此，当用户需要下载另一个名为 notes 的文件时，需将文件更名。

在本地系统中，用户已有一个文件名为 notes，因此，用户必须修改本地文件名；或下载时修改远地文件名。本例中，我们将第二个下载的文件更名为 notes2。用户可从 FTP 输出中看到下载文件的本地文件名和远程文件名。

```
ftp> get notes notes2
200 PORT command successful.
150 ASCII data connection for notes (1.2.3.4,52264) (210 bytes).
226 ASCII Transfer complete.
local: notes2 remote: notes
226 bytes received in 0.019 seconds (11.81 Kbytes/s)
ftp>
```

上述两个 notes 文件均为文本文件。用户可以看到 FTP 使用 ASCII 方式传输文本文件。它确保传输的文件为文本文件。如果需要传输二进制文件，需要修改文件类型：

```
ftp> bin
200 Type set to I.
```

类型 I 表示二进制，类型 A 表示 ASCII。二进制文件包括可执行文件、压缩文件和库文件。基本上，二进制文件是不可读文件。

用户可将文件类型设为二进制文件类型，以传输二进制文件。

```
ftp> get vr40a.exe windowsfile.exe
200 PORT command successful.
150 Binary data connection for vr40a.exe (1.2.3.4,52265) (2338635 bytes).
226 Binary Transfer complete.
local: windowsfile.exe remote: vr40a.exe
2338635 bytes received in 1.4 seconds (1593.23 Kbytes/s)
ftp>
```

注意，FTP 会显示文件传输方式。如果用户需要将传输类型设为 ASCII 方式，只需敲入 ascii 即可。

```
ftp> ascii
200 Type set to A.
```

```
Insert code
```

当用户下载单个文件时，采用上述方法，当用户需要一次下载多个文件呢？

例如，用户需要下载以“s”开头的所有文件，可能会尝试使用s*，但FTP将显示失败信息。

```
ftp> get s*
550 s*: No such file or directory.
```

当用户需要FTP识别通配符时，需要使用mget命令下载多个文件。

```
ftp> mget s*
mget s3-Solaris.tar?
```

当用户使用mget或mput下载或上载多个文件时，缺省情况下，FTP将询问用户文件是否需要下载或上载。当用户需要下载或上载时，敲入“y”。

```
mget s3-Solaris.tar? y
200 PORT command successful.
150 ASCII data connection for s3-Solaris.tar
(127.0.0.1,52452) (0 bytes).
226 ASCII Transfer complete.
mget sendmail.8.9.2.tar.gz? y
200 PORT command successful.
150 ASCII data connection for sendmail.8.9.2.tar.gz
(127.0.0.1,52453) (0 bytes).
226 ASCII Transfer complete.
mget solsniffer.c? y
200 PORT command successful.
150 ASCII data connection for solsniffer.c
(127.0.0.1,52454) (0 bytes).
226 ASCII Transfer complete.
mget spade110.exe? y
200 PORT command successful.
150 ASCII data connection for spade110.exe
(127.0.0.1,52455) (8192 bytes).
226 ASCII Transfer complete.
mget sun-sniff.c? y
200 PORT command successful.
150 ASCII data connection for sun-sniff.c
(127.0.0.1,52456) (8192 bytes).
226 ASCII Transfer complete.
ftp>
```

每下载一个文件都需敲入“y”，是一件十分枯燥的工作，尤其当下大量文件时。为了除去提示信息，可敲入prompt。

```
ftp> prompt
Interactive mode off.
```

除去提示信息后，用户敲入相同的命令，将不需每下载一个文件均需确认。因此，文件

传输速度快得多。

```
ftp> mget s*
200 PORT command successful.
150 ASCII data connection for s3-Solaris.tar
(127.0.0.1,52458) (0 bytes).
226 ASCII Transfer complete.
200 PORT command successful.
150 ASCII data connection for sendmail.8.9.2.tar.gz
(127.0.0.1,52459) (0 bytes).
226 ASCII Transfer complete.
200 PORT command successful.
150 ASCII data connection for solsniffer.c
(127.0.0.1,52460) (0 bytes).
226 ASCII Transfer complete.
200 PORT command successful.
150 ASCII data connection for spade110.exe
(127.0.0.1,52461) (0 bytes).
226 ASCII Transfer complete.
200 PORT command successful.
150 ASCII data connection for sun-sniff.c
(127.0.0.1,52462) (0 bytes).
226 ASCII Transfer complete.
ftp>
```

这比使用提示模式快捷得多。

警告 如果用户打算覆盖一个文件(在上述例子中,使用相同文件),FTP将不会在覆盖之前收到警告信息。

现在,我们已经示例了如何下载文件,下面举例说明如何将文件上传到远程服务器。上传文件需要使用命令 put。

```
ftp> mput h2obj.jpg
200 PORT command successful.
150 Binary data connection for h2obj.jpg (1.2.3.4,52270).
226 Transfer complete.
local: h2obj.jpg remote: h2obj.jpg
1194 bytes sent in 0.019 seconds (60.82 Kbytes/s)
ftp>
```

它将本地文件传送到远程服务器。如果用户需要一次传输多个文件,可使用 mput命令。

```
ftp> mput *
mput debug.ssh? y
200 PORT command successful.
150 ASCII data connection for debug.ssh (1.2.3.4,52266).
226 Transfer complete.
mput deshadow.c? y
200 PORT command successful.
150 ASCII data connection for deshadow.c (1.2.3.4,52267).
226 Transfer complete.
ftp>
```

与mget类似，用户首先需要关闭提示功能，以避免每传输一个文件都要敲入“Y”或“N”确认。

```
ftp> prompt
Interactive mode off.
ftp> bin
200 Type set to I.
ftp> mput k*
200 PORT command successful.
150 Binary data connection for kayaking.jpg (1.2.3.4,52268).
226 Transfer complete.
200 PORT command successful.
150 Binary data connection for key.tar (1.2.3.4,52269).
226 Transfer complete.
ftp>
```

注意 在与传输文件相关的命令(get、put或删除)前加上“m”可对多个文件进行操作。

如果用户需要删除远程系统上的文件(用户有权限完成此操作)，用户可以非常简便地完成删除文件的任务。delete命令专门用于删除文件，它还可简写为del。

```
ftp> del h2obg.jpg
250 DELE command successful.
ftp>
```

如果用户需要一次删除多个文件，可使用 mdel。同时，prompt命令对于mdel的作用与mget或mput相同。

```
ftp> mdel s*
250 DELE command successful.
250 DELE command successful.
250 DELE command successful.
ftp>
```

警告 与覆盖文件类似，用户在远程系统删除文件时也不会得到警告信息。

用户不仅可以在本地系统与远程系统间进行文件复制、删除操作，而且也可以创建和删除目录。mkdir命令用于创建一个目录。

```
ftp> mkdir morestuff
257 MKD command successful.
```

用户也可以在不同的目录间移动。与UNIX和Windows命令行(其命令解释器称为DOS)相似，可以使用cd改变当前路径。

```
ftp> cd morestuff
250 CWD command successful.
```

如果用户不知道当前路径或需要查看当前所处的路径，可以使用pwd命令。

```
ftp> pwd
257 '/home/ahc/morestuff' is current directory.
```

如果用户需要对大文件进行传输，可能需要使能 hash，它将显示FTP的工作状态，即使FTP空闲几分钟甚至数小时。激活这一功能，仅需敲入 hash。

```
ftp> hash
Hash mark printing on (8192 bytes/hash mark).
如果用户在hash模式下put或get一个文件，可以看到文件传输的速度。

ftp> put sendmail.8.9.2.tar.gz
200 PORT command successful.
150 Binary data connection for sendmail.8.9.2.tar.gz (1.2.3.4,52271).
#####
226 Transfer complete.
local: sendmail.8.9.2.tar.gz remote: sendmail.8.9.2.tar.gz
1063534 bytes sent in 0.56 seconds (1870.03 Kbytes/s)
ftp> pwd
```

现在，我们已经完成了对一个站点的操作，如果不想退出 FTP而只想关闭连接，可以敲入 close命令，它仅断开与远程系统的连接但仍处在 FTP命令解释器中。

```
ftp> close
221 Goodbye.
ftp>
```

注意 我们仍有ftp>提示符，当需要打开到另外一主机的连接时，使用open命令：

```
ftp> open another-example.org
Connected to another-example.org.
220 another-example.org FTP server (SunOS 5.6) ready.
Name (localhost:ahc):
331 Password required for ahc.
Password:
230 User ahc logged in.
ftp>
```

现在，已经处在另外一个系统中。用户可以使用相同的命令，但得到的信息将与上一个系统不同。

完成了对FTP的操作后，可以敲入quit或bye命令。

```
ftp> quit
221 Goodbye.
```

26.2.3 FTP安全

虽然FTP可以对文件进行访问，但没有一个用户会允许别人在自己的系统制造混乱。FTP可以实现基本的访问控制，但它不能解决FTP基本的安全问题。

FTP一直使用明文认证。这意味着用户的口令未经加密就在网上传输，这是一个很严重的安全问题。不恰当的匿名FTP服务器的配置将使用户进入他们不能访问的目录，FTP服务需要两个端口而不像大多数TCP/IP网络服务仅需一个端口也是FTP的一个安全隐患。

1. /etc/ftpusers文件

如果用户需要控制FTP服务器上客户对特定帐号的访问权限，可以使用/etc/ftpusers文件。文件中列出了不能使用ftp的用户名。而且它不限制用户对其他网络应用如 rsh、telnet和Secure shell (SSH)的访问。

一个合理的/etc/ftpusers文件如下：

```
tigerlair:/home/stripes- cat /etc/ftpusers
root
uucp
bin
```

它限制管理员帐号及相关帐号不能使用FTP。

2. 使用.netrc

.netrc文件用于FTP客户自动登录到远程主机，该文件位于用户根目录，并且可在不需要键入用户名和口令的情况下进入不同的系统。该文件的权限为 700(除了所有者外，没有人可读/写该文件)。

.netrc文件的格式如下：

```
machine <机器名> login<用户名> password<用户口令>
```

如果用户要设置到example.org的访问，文件如下：

```
tigerlair:/home/stripes- cat .netrc
machine example.org login ahc password stuff
```

.netrc通常用于登录帐号，因此它可以很安全地用于匿名FTP站点，它仅需要用户的电子邮件地址，而无需用户口令。同时，匿名FTP站点仅向用户提供公用信息——即无需口令就可获取的信息。

警告 在当今社会，设置.netrc文件不是一个好主意。它包含可被任何人获取的口令，因此，帐号可被轻易地破解。

3. 使用不同的ftp客户端软件

前述的案例中，我们使用的是操作系统如UNIX、Windows或VMS自带的标准FTP客户端软件。此外，因用户使用的操作系统不同，FTP客户端软件的功能也略有差异。下面简单介绍这些差异。

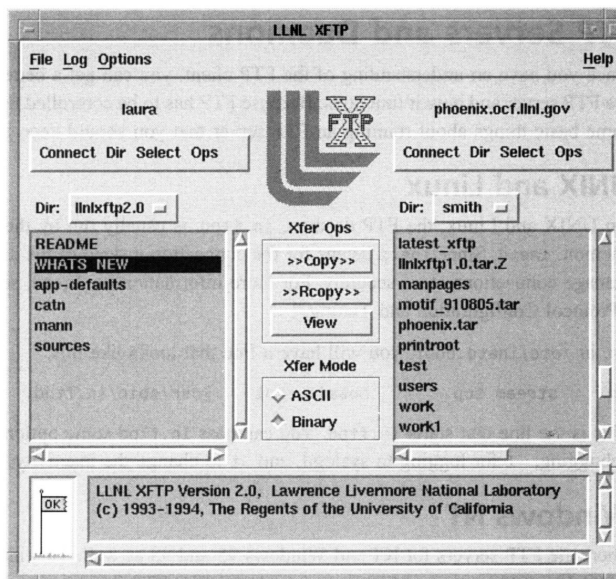


图26-3 激活的xftp会话

4. UNIX

在UNIX系统中，有两个不同的FTP客户端软件可以提供比标准FTP客户软件更复杂的功能：ncftp和xftp(见图26-3)。

ncftp自动登录匿名ftp，因此，用户无需敲入口令(可能是用户邮件地址)或不必担心保存.netrc文件。同时，它也提供一些基于文本的图形界面，以显示文件传输速度及其他命令行编辑功能。ncftp可从<http://www.ncftp.com>中获得。

xftp软件提供图形化用户接口，它使用户无需与命令解释器直接接触即可完成FTP传输，图26-3显示xftp窗口。xftp软件可从<http://www.llnl.gov/ia/xftp/xftp.html>中获得。

5. Windows和Macintosh客户端软件

有大量FTP软件可用于Windows和Macintosh平台，其中大部分都有与xftp类似的图形界面，它允许用户在远程系统和本地计算机间拖放文件。

同时，也有一些FTP客户端软件支持命令行方式。其他的客户端软件将激活浏览器，基于浏览器的FTP仅提供部分FTP功能。

表26-3列出了部分可用于Windows和Macintosh的FTP客户端软件。

表26-3 支持Windows和Macintosh平台的FTP客户端软件

FTP客户端	站 点
WS-FTP Pro	http://www.ipswitch.com
fetch	http://www.dartmouth.edu/pages/softdev/fetch.html
cuteFTP	http://www.cuteftp.com
FTPPro2000	http://www.ftppro.com
Email	

无论相信与否，用户可以通过电子邮件使用FTP客户端。FTPmail可使得仅能通过电子邮件访问互联网的用户使用FTP。这种方法可使用户不需直接使用FTP而只需通过email就可从FTP站点获取文件。

互联网上的某些服务器为用户提供FTPmail服务。这些服务器提供FTPmail帐号，用户可将包含FTP请求的消息发送到FTPmail帐号(FTP请求使用与FTP命令解释器类似的命令)。

FTP会话通过发送的FTP请求进行，结果发送到用户电子邮件地址。如果FTPmail连接FTP服务器失败，返回给用户的邮件将解释失败的原因。

许多站点提供FTPmail服务，任何可以使用电子邮件的用户均可使用这些服务。建议大家最好使用本国站点的服务。

表26-4列出了部分FTPmail服务器及其所处的国家。

表26-4 FTPmail服务器

服 务 器	国 家
ftpmail@grasp.insalyon.fr	法国
ftpmail@doc.ic.ac.uk	英国
ftpmail@decwr1.dec.com	美国

26.2.4 FTP服务器及守护进程

理解FTP客户端的工作过程，将有助于用户理解 FTP服务器及其工作过程。因为 FTP由服务器方控制，所以用户需要了解FTP服务器基本工作流程。

1. UNIX和Linux

在UNIX和Linux系统中，FTP守护进程 in.ftpd通常由 Internet超级守护进程 inetd运行。inetd监听连接，它可以更安全地管理连接。inetd更详细的信息见35章。

因此，在/etc/inetd.conf文件中，包含如下行：

```
ftp stream tcp      nowait root    /usr/sbin/in.ftpd.  in.ftpd
```

这一行启动 in.ftpd，用户可以添加一些属性如 -d调试、-l记录日志及 -t修改活跃时间域等。

2. Windows NT

同时也存在一些可运行于 NT、Windows 95和Windows 98的FTP服务器，包括FTP Serv-u(可从<http://www.ftpserv-u.com>下载)和War FTP daemon(可从<http://www.jgaa.com/tftpd.htm>下载)。

26.2.5 匿名FTP访问

即使当前Web站点蓬勃发展，匿名FTP站点仍然是Internet的重要组成部分。在众多的匿名FTP站点上，我们可以获取各种各样的信息、文件和软件。其中最著名的匿名 FTP站点之一为wuarchive.wustl.edu。

大量匿名FTP站点对所有用户公开文件访问权，它们旨在发布软件和信息，因而免去了需要用户名和口令所带来的不便。

1. 匿名FTP服务器

匿名FTP服务器是提供文件访问的公用站点。在 Web出现前，gopher(菜单驱动的互联网应用)和FTP是人们从互联网获取信息和文件的主要手段。

因此，匿名FTP日益流行，至今仍有许多匿名 FTP站点运行。匿名FTP站点需要单独的环境和硬盘且完全与系统文件分离。我们通常称之为 chrooted环境。

例如，UNIX系统有/usr和/etc目录。UNIX系统中的匿名FTP服务器也包含这两个目录。并且，存储于UNIX系统/etc目录下的文件与匿名FTP服务器中/etc下的文件不能相同。

系统文件不能存贮在匿名 FTP目录下。FTP的运行环境一定要与系统的运行环境彻底分开。

2. 匿名FTP客户端

为了便于使用，匿名FTP允许用户使用“anonymous”或ftp作为用户名，邮件地址作为口令(例如：stripes @ tigerlair.com)。

匿名FTP也控制用户的访问权限。通常用户仅能访问公用目录下的文件，而不能访问系统及配置文件，这可使系统更加安全。

以下示例说明一个匿名FTP的会话过程：

```
tigerlair:/home/stripes- ftp wuarchive.wustl.edu
Connected to wuarchive.wustl.edu.
220 wuarchive.wustl.edu FTP server (Version wu-2.4.2-academ[BETA-16])
(1) Wed Apr 1 08:28:10 CST 1998) ready.
Name (wuarchive.wustl.edu:stripes): anonymous
```

```
331 Guest login ok, send your complete e-mail address as password.
Password:
230- If your FTP client crashes or hangs shortly after login please try
230- using a dash (-) as the first character of your password. This
230- will turn off the informational messages that may be confusing your
230- FTP client.
230-
230- This system may be used 24 hours a day, 7 days a week. The local
230- time is Wed May 26 12:02:37 1999. You are user number 23 out of a
230- possible 500. All transfers to and from wuarchive are logged. If
230- you don't like this then please disconnect.
230-
230- Wuarchive is currently a Sun Ultra Enterprise 2 Server.
230-
230- Wuarchive is connected to the Internet over a T3 (45Mb/s) line from
230- STARnet and MCI. Thanks to both of these groups for their support.
230-
230- Welcome to wuarchive.wustl.edu, a public service of Washington
230- University in St. Louis, Missouri USA.
230-
230-
230-
230 Guest login ok, access restrictions apply.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

因为Web的日益普及，许多匿名FTP站点可通过Web浏览器直接访问而无需FTP客户软件。

提示 用户在敲入口令时，不需要输入整个邮件地址，仅需要键入：

用户名 @

例如，用户可敲入：

stripes @

代替：

stripes @ tigerlair.com

26.3 使用TFTP

在网络发展早期，TFTP是局域网传输文件的通用方式。TFTP是普通文件传输协议，它也用于文件传输，缺省端口为69。

TFTP不需要任何形式的用户登录认证，TFTP有共享安全问题。目前，它常用于工作站向路由器发送路由器访问列表及无盘工作站的启动中。

警告 除非特别需要，否则不要启动TFTP服务，因为如果用户疏忽，它可带来许多安全问题，包括允许客户访问口令文件。

TFTP目前已开发出第2版，在RFC 1350中定义。与FTP不同，FTP使用TCP而TFTP使用UDP。因此，TFTP简单并且紧凑。与FTP类似，TFTP也由inetd启动。

每一个客户和服务器的报文交换均开始于客户向服务器发送的文件读/写请求。

文件以octet(八进制)或netascii方式传输。在netascii方式下,数据为以返回符加上行结束符(CR/LF)结尾的ASCII码文本。

26.3.1 FTP与TFTP的区别

TFTP的功能比FTP少。例如,TFTP不能使用通配符,不能创建和删除目录,不能删除文件。同时,它也不提供用户名与口令认证,其访问由其他手段限制(TCP Wrapper等)。

因为TFTP的简洁性,所以路由器使用TFTP传输访问列表及路由配置信息。因为路由器无需登录(除非用户运行TACACS+或RADIUS实施访问控制),在已有文本上进行编辑远比一行行敲入简单。

26.3.2 TFTP命令

如果用户熟悉FTP,TFTP将不难理解。TFTP命令较少,表26-5中列出了所有命令。其中大部分与FTP类似。

表26-5 TFTP命令

命 令	描 述
ascii	将传输模式设为netascii(文本)
binary	将传输模式设为octet(非文本)
connect	连接远程TFTP
mode	设置文件传输模式
put	发送文件到远程站点
get	从远程站点下载文件
quit	退出TFTP
rexmt <value>	设置每个报文的超时值
status	显示服务器的状态
timeout <seconds>	设置传输超时(秒)
trace	打开报文路径
verbose	激活冗余模式

26.4 小结

文件传输对于在TCP/IP网络不同系统间传输文件至关重要。TCP/IP使用文件传输协议(FTP)和普通文件传输协议(TFTP)作为TCP/IP网络发送文件的重要方式。

使用TCP传输文件的FTP需要打开两个端口:端口20为数据端口,端口21为控制端口。数据端口传输文件,控制端口用于传输命令和消息。

TFTP使用UDP传输文件,它仅需要端口69。TFTP远比FTP简单,其功能也较FTP少。TFTP通常用于路由器管理访问列表和配置信息。它也用于无盘工作站的启动。

在下一章,我们将讨论Telnet,学习另一种类型的交互式TCP/IP网络服务。