

怎样使用匿名 FTP

本备忘录状态

本备忘录提供了 Internet community 的信息，但不说明任何一种类型的 Internet 标准。发布本备忘录不受限制。

概要

本文档向 Internet 初学者们提供了文件传输协议（FTP）的相关信息。文档解释了 FTP 是什么、匿名 FTP 是什么、以及什么是 FTP 文件存档站点。文中给出了一个匿名 FTP 登陆过程的范例。文中还介绍了为了高效的存储和传输而经常采用的文件打包方法。

感谢

（略——译者注）

什么是 FTP？

FTP 是 File Transfer Protocol 的缩写，它是 Internet 上使用的 TCP/IP 协议一组里的其中一个协议。文件传输协议使 Internet 上从一台计算机（或主机）向另一台计算机（或主机）传输文件成为可能。已经有许多建立在 FTP 协议规范之上的现实应用。为了传输文件从一台主机到另一台，FTP 程序的用户必须登陆到双方的主机。

对于一个在多于一台主机上存有文件的用户来说，使用 FTP 程序在主机间传送文件是经常的。在这种情况下，他必须在每台主机上都有一个帐号和密码。

不过，Internet 用户也可以使用一个叫做“匿名 FTP”的公用帐号来从 FTP 存档站点获得大量有效信息。

什么是存档站点？

存档站点就是有大量电子信息的主机，很像传统的图书馆。存放在这些 Internet 主机上的信息可以被用户利用来传输到他们的本地站点。用户运行软件来识别信息并将它传输到自己的主机。这样的通过程序来进行的传输就是文件传输协议（FTP）在应用方面的实现。

什么是匿名 FTP？

匿名 FTP 的意思就是允许用户对存档站点的信息进行一般性访问。这些站点建立了一个特殊的用户“anonymous”，用户“anonymous”拥有有限的站点内访问权限，即操作权受限制。实际上，这仅有的操作允许用户登录到 FTP 站点，列出有限的目录内容和下载文件。一些站点还限制了匿名用户能看到的目录内容。注意，“匿名”用户通常不允许向存档站点上传文件，而仅允许从存档站点下载文件。

一般地，这个特殊的匿名用户帐号允许使用任何字符串作为密码，但是通常它被限定为使用“guest”字串或一个电子邮件（e-mail）地址。一些存档站点还明确地要求用户输入电子邮件地址作为密码，而不允许输入

“guest”。倘若提供了有效的电子邮件地址，存档站点的操作员就能知道是谁在使用他们的服务了。

你需要知道什么信息？

为了得到一个指定的文件，用户需要知道它所在的主机及文件的路径名。路径名指出了文件所在的目录（也可能是子目录），和文件的名称。可利用文件的讨论通常不会明确指明：“这个文件可以在 X 主机的 Y 路径通过匿名 FTP 下载”。不过，如果一个文件被公布在类似 `nisc.sri.com` 网站的 `pub/good-stuff` 目录下，就有很大的可能能被下载。

你还需要知道你的机器使用的是 ASCII 码, EBCDIC 码, 或其他的字符集，也要知道传输是以怎样的二进制编码信息工作的，或者是否传输会要求其他的关键字，比如 TENEX 就是这样。

通常情况下，你采用 ASCII 码来传输纯文本文件都是正确的。不过，更多更多的信息是以压缩过的格式存储的（这点将在后文中讨论），所以知道你的机器的二进制编码特性是非常重要的。

一个 FTP 传输过程的例子

为了在 UNIX 或 VMS 主机上开始 FTP 传输，首先键入你想要连接的主机名或主机地址。例如，如果你想访问 NASA 网络应用程序和信息中心的存档站点，你通常需要在 UNIX 主机的提示符下执行下面的命令：

```
ftp naic.nasa.gov  
或  
ftp 128.102.128.6
```

注意，第一行的格式使用的是完整的域名，第二行使用的是同一主机的 Internet 地址。

下面的例子是连接到 `naic.nasa.gov` 主机，下载 STD 9，RFC 959，“文件传输协议（FTP）” [1]。

在传输过程需要注意几点：

1. 每一个在存档站点上的 FTP 程序应答之前都会给出一个数字，这些数字称为应答编码，它们是在 RFC 959——FTP 规范中定义的。通常，和应答编码一起给出的文本会根据不同的 FTP 程序而改变。

还要注意一些 FTP 客户端程序（比如 MVS 系统）不会显示从远程主机传来的应答编码或伴随文本。它们可能显示自己的状态行或隐藏对你来说不重要的回应。为了达到本文的目的，我们介绍更为流行的 UNIX 界面作为 FTP 客户端。

2. 你键入的密码不会被显示在屏幕上。

3. 可能在存档站点上浏览，但更多的用户已经知道了他们想要的文件的路径名。RFC 959 在主机上的路径名是 `files/rfc/rfc959.txt`。在本例中，我们先连接到“`files/rfc`”目录（`cd files/rfc`），然后获得我们已知的想要得到的文件。如果你不知道你想要的文件的文件名，可能需要先下载一个叫做 `README` 或者类似这样的文件（`00README.1ST`，`AAREAD.ME`，`INDEX`，等等）。

```

atlas.arc.nasa.gov% ftp naic.nasa.gov
Connected to naic.nasa.gov.
220 naic.nasa.gov FTP server (Wed May 4 12:15:15 PDT 1994) ready.
Name (naic.nasa.gov:amarine): anonymous
331 Guest login ok, send your complete e-mail address as password.
Password:
230-----
230-Welcome to the NASA Network Applications and Info Center Archive
230-
230-      Access to NAIC's online services is also available through:
230-
230-      Gopher          - naic.nasa.gov (port 70)
230-      World-Wide-Web - http://naic.nasa.gov/naic/naic-home.html
230-
230-      If you experience any problems please send email to
230-
230-                      naic@nasa.gov
230-
230-                      or call +1 (800) 858-9947
230-----
230-
230-Please read the file README
230-  it was last modified on Fri Dec 10 13:06:33 1993 - 165 days ago
230 Guest login ok, access restrictions apply.
ftp> cd files/rfc
250-Please read the file README.rfc
250-  it was last modified on Fri Jul 30 16:47:29 1993 - 298 days ago
250 CWD command successful.
ftp> get rfc959.txt
200 PORT command successful.
150 Opening ASCII mode data connection for rfc959.txt (147316 bytes).
226 Transfer complete.
local: rfc959.txt remote: rfc959.txt
151249 bytes received in 0.9 seconds (1.6e+02 Kbytes/s)
ftp> quit
221 Goodbye.
atlas.arc.nasa.gov%

```

变化

上面的例子在 UNIX 系统中是有效的。在其他操作系统也利用 FTP 程序。使用其他的程序，实际命令会稍有变化。然而，一般地，在每个 FTP 程序中你都会做如下操作：

- 登录到你的本地主机，调用 FTP 程序。
- 连接到远程主机（使用主机名或者它的 IP 地址）。
- 一旦连接到远程主机，使用用户名“anonymous”登录。
- 用“guest”或者根据站点的要求提供密码。
- 执行你需要的 FTP 命令，比如像改变目录或下载文件。
- 任务完成时，退出 FTP 程序，关闭和存档主机的连接。

友好的服务器

目前，很多站点采用了一种这样的 FTP 方式，它允许显示几行解释性文本以帮助和指导用户访问他们的文档。naic.nasa.gov 就是这样进行列表选择服务的一个例子。如果这些友好的服务在你使用时导致客户端混乱，试着在登录时你的密码前加一个连字符（-）。这样会屏蔽服务器的详细模式。

其他 FTP 命令

我们已经示范了一些 FTP 程序的可用命令。许多其他的命令也会用到。例如，一旦你登录到了一台远程主机：

- 以不带参数的方式调用 FTP 程序，然后键入“help”，可以让 FTP 程序显示可用的命令列表。
- 如果要查看你所连接的目录的内容，键入“dir”或者“ls”。
- 使用“get”命令，在命令行中远程文件名之后键入任意的本地文件名，就可以为文件改名。当远程文件名超出你本地文件系统的命名规则时，你就要对文件改名，比如文件名太长。传输文件过程中一个使用“get”命令改名的例子：“get really-long-named-file.txt short.txt”。
- 你可以键入“binary”来设置用二进制模式传输可执行文件或数据文件。通常 FTP 程序默认文件的每个字节只使用 7 个二进制位，这是 ASCII 编码规定的标准。“binary”命令允许你以每字节 8 个二进制位的方式来传输文件而不发生错误。不过，以何种方式传输回你的本地系统，会给以提示。如果你不能确认一个文件是何种格式，在第一次传输错误之后，你需要用另一种格式（BINARY 格式 或 ASCII 格式）进行第二次传输。文件的扩展名可以给你点线索。文件的扩展名将在下文中描述。因为一些机器存储文本文件的方式与其他机器不同，如果你不知道一个文件是何种格式，那就只好碰运气了。一个好的猜测方法是，如果你有比较大的把握认为这是一个文本文件时，先尝试用 ASCII 模式，否则用 BINARY 模式，最后再用 TENEX 模式。
- 你可能一次传输多个文件。用“mget”来设置这种模式。你也可以提供一个有远程系统可以理解的有通配符的文件名，远程系统会试着传回每一个文件。如果你的本地 FTP 程序不能以远程文件名的形式将传输的文件命名为本地文件名，或者如果其中有些文件必须使用 ASCII 模式而另一些必须使用 BINARY 模式传输，你就不能使用这个简单有效的命令了。在你的系统的 FTP 文档中，有全部命令和有效操作的详细说明。你也可以在 FTP 命令提示符下键入“help”来显示命令操作的列表。

UNIX 版的 FTP 文档可以从在线手册中获得。如果你的 UNIX 站点已经安装了这些手册，在 UNIX 提示符下键入以下命令：

% man ftp

文件的打包和命名

一些广泛使用的约定，允许存档站点上存储的文件以有效的方式存储和传输。

存储在存档站点的信息经常使用三种常用的方式来“转换”。“压缩”（减少文件尺寸）被存储的信息，使得存档站点拥有了更多的可用空间，也减少了大量的网络间的实际传输数据。“打包”将几个文件合并成一个更大的文件，但还维持其内部的目录结构，也能让用户一次只传输一个大的目标文件而不是多个（有时是数百个）小文件。

另外，二进制文件经常转换成 ASCII 格式来传输，这个过程在本文中称为“格式转换”。一般地，基于 Internet RFC 822 的电子邮件和 USENET 协议不允许用“二进制”（8 位）数据来传输，二进制文件在传输之前必须转换成可打印的 7 位 ASCII 格式。

在许多系统中，不同的文件名约定规则可以帮助远程用户确定存储的信息的格式而不必先把文件下载回去。下面我们列出了更多的在 Internet 上使用的压缩、打包、格式转换方面的约定。列出的约定可能会有遗漏。在网络上有所有与这些机制有关的公共领域或自由软件。

1) compress/uncompress

文件名以“.Z”结尾的文件一般表示该文件已被标准的 UNIX Lempel-Ziv “压缩”程序压缩过。还有一种对应的叫做“解压缩”程序，它把压缩过的文件做反向处理还原成文件初始的状态。压缩程序不提供打包机制，不管初始的文件是什么格式，生成的文件总是二进制格式的。

2) atob/btoa

执行从 ASCII 格式到二进制格式的转换（atob），或者对应的从二进制格式到 ASCII 格式的转换（btoa）。经过这样的转换，文件名经常会以“.atob”结尾。不使用打包或压缩机制。

3) atox/xtoa

此数据标准用来把二进制文件转换成可供传输的 ASCII 格式文件。因为它能提供更多的有效空间，所以有时候和其他机制相比它更优先地被选用，尽管它本质上不是一种压缩机制。文件从一种格式到另一种格式的转换它更有效。这种格式的文件经常以“.atox”作为扩展名。

4) uuencode/uudecode

从二进制到 ASCII 的转换（“uuencode”）和相反的转换（“uudecode”）是一种标准的方式。最初用于 UUCP（“Unix to Unix CoPy”）的 mail/USENET 系统。不使用打包或压缩机制。命名的习惯经常是在文件名结尾加上“.uu”。

5) tar/untar

最初基于 UNIX 的程序提供打包（和解包）多个文件和目录到一个单独的文件（取“Tape ARchive”的首字母缩写）。标准的格式没有提供压缩机制。生成的文件通常是二进制格式的，不管组成的文件是否是二进制文件。命名的习惯经常是保留原有的文件名并加上“.tar”或“-tar”。

6) zip/unzip

经常用于 IBM PC 环境中，这些补充程序提供打包和压缩机制。生成的文件通常是二进制格式的。用“zip”程序生成的文件约定以“.zip”作为

扩展名。

7) arc/unarc

经常用于 IBM PC 环境中，这些补充程序提供打包和压缩机制。生成的文件通常是二进制格式的。这种格式的文件通常以“.arc”作为扩展名。

8) binhex

使用在苹果的 MacIntosh 环境中，binhex 提供类似二进制到 ASCII 格式数据转换的打包机制。这种格式的文件约定以“.hqx”作为扩展名。

9) shar

证券交易所打包文本文件或二进制文件的外壳文档。它生成一个单独的更大的文件，当执行时，会生成组成的文件。因为这种格式在误用时易于受到攻击，所以大多数用户使用一个特殊的叫做 unshar 的程序来解开这样的文档。这种格式的文档约定以“.shar”作为扩展名。

10) VMS_SHARE

DCL 打包文本文件或二进制文件的外壳文档。它生成一个单独的更大的文件，当执行时，会生成组成的文件。因为这种格式在误用时易于受到攻击，在运行之前必须对这样的文件进行认真的检查。这种格式的文档约定以“.shar”作为扩展名。

11) Multipart shar/vms_share files

有时这些外壳文档文件被分割成多个小的局部文件，以便简单地在具有相同的目录树结构的文件服务器之间传输。在这种情况下，文件名通常使用一个序列号码（比如：xyz.01 xyz.02 xyz.03 ...或者如：.01-of-05）。收集到所有的局部文件后，在你的本地系统中进行连接，用你已经连接好的文件，应用上述过程，生成一个简单的 shar 或者 vms_share 文件。

12) zoo

zoo 程序在一个单一的程序里执行压缩/解压缩和打包/解包。支持 zoo 格式文件的应用程序在许多不同的操作系统里都存在，包括 Unix，MS-DOS，Macintosh，OS/2，AtariST，and VAX VMS。用“zoo”程序创建的文件约定以“.zoo”作为扩展名。Zoo 是一种广为流传的文件格式，原因在于它的应用程序在许多操作系统下都是免费的（包括可执行文件和源代码）。

13) gzip/gunzip

免费软件基金会 GNU 计划采用了一个不同的 zip 压缩机制以取代 compress/uncompress 命令。生成的文件通常是二进制格式的。用“gzip”程序生成的文件约定以“.z”或者“.gz”作为扩展名。gunzip 程序也能识别“.tgz”和“.taz”作为扩展名的文件，将它们作为“.tag.z”或“.tag.Z”的缩写。同样，gunzip 能够识别并解压缩由 gzip，zip，compress 或 pack 命令压缩而生成的文件。

GUN 计划最近开始发布和使用 gzip/gunzip 应用程序。甚至最近他们还把缺省的后缀从.Z 改为.gz，尝试减少使用.Z 造成的混乱，并消除像 MS-DOS 这样对大小写不加区分的文件系统造成的问题。Gzip 软件是自由发布的，并且已经在大多数的 UNIX 系统中实现，也在 Amiga，Atari，MSDOS，OS2，和 VMS 系统中实现了。

在一些情况下，为了生成最后供存档的文件，需要执行上面的一系列

处理。在进行了多个格式转换过程以后，通常保留最初的（基本的）文件名，为了反映进行的转换过程，相关的文件扩展名被依次加入以表明进行过那些转换。例如，一般的过程是先对文件和目录使用“tar”进行打包处理，然后用“compress”压缩打包的文件。开始的基本文件名是“foobar”，生成的文档的文件名会变为“foobar.tar.Z”。如果这是一个二进制文件，为了便于传输和在电子邮件和 USENET 中使用，它还需要更进一步地用比如“uuencode”程序转换到可供打印的 ASCII 格式，所以它最后可能会变为“foobar.tar.Z.uu.”。

有些操作系统不能处理含有多个扩展名的文件；在这种情况下，它们经常用连字符（-），下划线（_）来替代，或者在目录中加入一个详细描述“read me”文件。

Compress and Tar

下面是一个关于使用“compress/uncompress”和“tar/untar”的例子。

假设“补丁”是一个对已有程序进行修补和更新的公共域程序。你发现该程序在一个存档站点上，名为“patch.tar.Z”。现在你知道“.Z”表示该文件被 UNIX 的“compress”命令压缩过，“.tar”表示它被 UNIX 的“tar”命令处理过。

首先，使用匿名 FTP 把这个文件下载到你的机器里，为了解开该程序，首先要键入解压缩的命令：

```
uncompress patch.tar.Z
```

将会解压缩这个文件，并在处理过程中将文件改名为“patch.tar”。你可以执行“tar”命令再把里面每个文件释放出来。

在对 patch.tar 这个文件例子中，你可以调用如下命令：

```
%tar xvf patch.tar
```

文件将会从文件 patch.tar（因为加了“f”参数）被释放出来（因为加了“x”参数）。因为使用了“v”（verbose）参数，当执行完毕以后，所有的文件名会被打印出来。当 tar 完成时，原来组成这个“patch”程序的文件就完整地存在于你的工作目录中了。

礼节

不是每一个站点都支持 FTP 的匿名传输。试图从一个没有公开匿名 FTP 服务的站点获取文件是不对的。

记住，Internet 存档站点的管理员们是为了更有效地利用 Internet 社区而建立的他们的系统，管理员为了管理站点所付出的时间和精力很少得到充分的补偿。有些做法可以使得管理员们稍微轻松一些，比如问题发生时，在请求站点管理员帮助之前，首先询问一下本地的技术支持人员。

大多数的存档服务器能够很好地完成其他功能。请尊重他们的首要用户的需要，并且尽可能约束自己在非黄金时段（一般是站点所在地的本地时间 19:00 到 6:00）访问 FTP 服务器。特别重要的是对于远在另一个大陆

或大洋彼岸的站点，大多数这样的链接在下载时会非常地慢。

另外，一些站点限制同时进行匿名 FTP 登录的用户数量。如果你在登录一个站点时收到一个当前在线匿名 FTP 用户太多的错误信息时，在再次尝试连接之前稍等一会儿比马上再次连接的效果要好。

为了减少多余的存储，你应该想到如何有效地和你团队的成员共享你从 FTP 站点下载回来的有用的文件。一旦你发现某个软件确实很有用处，你可以请求你的本地网络管理员，让这个软件在本地共享。这样可以避免每个人都在自己的用户目录里安装同样的软件包，从而节省了存储空间。

如果你在一个存档站点发现了一个有趣的文件或程序，把这个发现告诉其他人。不要把它复制到你自己的档案站点上，除非你愿意保持对它进行同步更新。

参考书目

[1] Postel, J., and J. Reynolds, “文件传输协议 (FTP)”, STD 9, RFC 959. USC/信息科学学会, 1985 年 10 月。

安全方面的考虑

本文的讨论没有涉及安全方面的问题。

作者地址

(略——译者注)

原文: RFC 1635 《How to Use Anonymous FTP》

译者: comehope 2002 年 5 月

博客: <http://www.comehope.com>