

§ 1.3 验证实验：手工试验 HTTP 事务

1.3.1 实验要求

- 1) 学习配置 Web 服务器 Tomcat 基本方法与步骤；
- 2) 通过手工测试与 Web 服务器通讯事务，理解 HTTP 协议的工作原理，交互命令格式与消息头部内容。

1.3.2 实验分析

(1) Web 服务器安装与配置

在安装 Tomcat 之前必须先安装 JDK（参照本书第 2.4.1 节）。Tomcat 选择 apache-tomcat-5.5.20，下载网址为 <http://tomcat.apache.org/>。

安装 apache-tomcat-5.5.20 后，首先要启动 Tomcat 服务器，以测试 Tomcat 服务器是否安装成功。然后在浏览器地址栏中输入 <http://localhost:8080>（或 <http://127.0.0.1:8080>）进行访问。如果出现如图 Error! No text of specified style in document.-1 所示界面，则说明 Tomcat 安装成功。

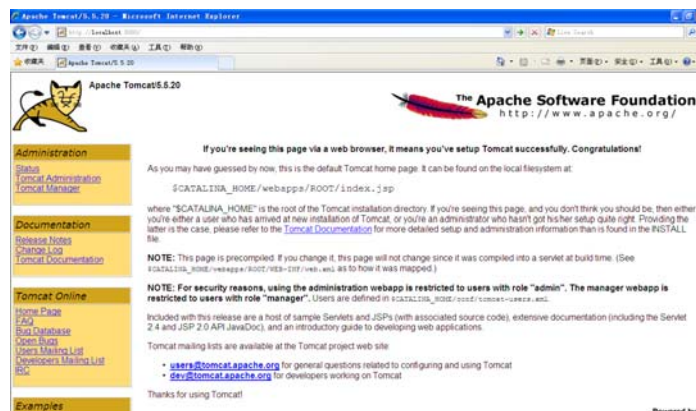


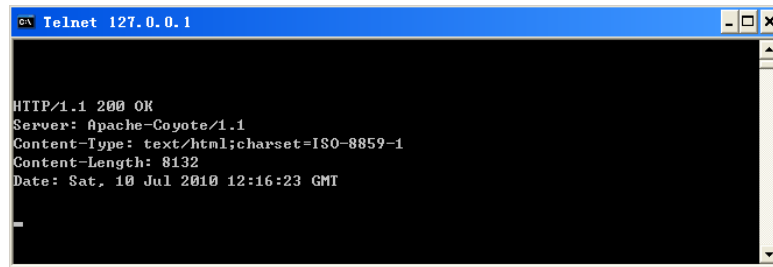
图 Error! No text of specified style in document.-1 Tomcat默认主页

【提示】Tomcat 是 Sun 的 JSWDK (Java Server Web Development Kit) 中的 Servlet 容器，属于 Apache 软件基金会 (Apache Software Foundation) 的 Jakarta 项目中的一个核心项目。Tomcat 既是一个 Web 服务器软件，同时又是一个开放源码、免费支持 JSP 和 Servlet 技术的容器。

(2) 手工测试与 Web 服务器通讯事务

运行 Telnet 后，输入命令 open 指定 Web 服务器主机名和端口号（默认为 80），然后可输入 HTTP 命令直接与 Web 服务器进行交互。

```
prompt> telnet
prompt> set localecho
prompt> open 127.0.0.1 8080
prompt> HEAD / HTTP/1.1
prompt> Host:127.0.0.1
```

A screenshot of a Telnet window titled 'Telnet 127.0.0.1'. The window has a blue title bar and standard Windows window controls. The main area is black with white text showing the following HTTP headers: HTTP/1.1 200 OK, Server: Apache-Coyote/1.1, Content-Type: text/html; charset=ISO-8859-1, Content-Length: 8132, and Date: Sat, 10 Jul 2010 12:16:23 GMT. A cursor is visible at the bottom left.

```
Telnet 127.0.0.1

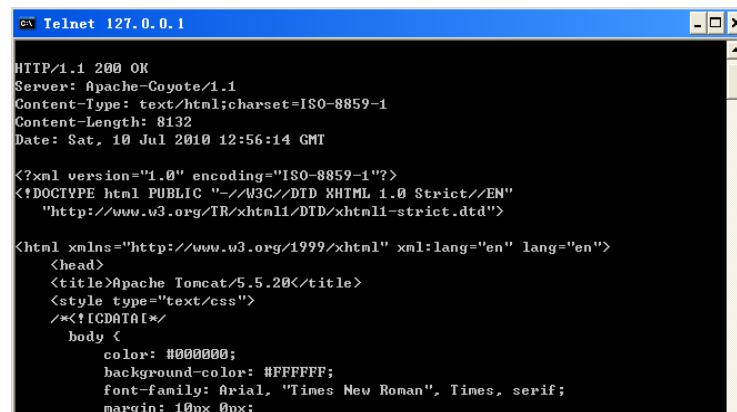
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Content-Type: text/html; charset=ISO-8859-1
Content-Length: 8132
Date: Sat, 10 Jul 2010 12:16:23 GMT
-
```

图 Error! No text of specified style in document.-2 请求获取响应消息报头内容

利用软件工具 ieHTTPHeaders 从 Web 浏览器 Internet Explorer 中查看相同服务器返回的消息头部，所看到的消息头部内容应该是相同的。

变换请求方法，请求 <http://127.0.0.1> 主页的内容

```
prompt> open 127.0.0.1 8080
prompt> GET / HTTP/1.1
prompt> Host:127.0.0.1
```

A screenshot of a Telnet window titled 'Telnet 127.0.0.1'. The window shows the same HTTP headers as the previous image, followed by the start of an HTML document. The HTML content includes an XML declaration, a DOCTYPE declaration, an XML namespace declaration, a head section with a title 'Apache Tomcat/5.5.20' and a style section, and the beginning of a body section with CSS styles for color, background-color, font-family, and margin.

```
Telnet 127.0.0.1

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Content-Type: text/html; charset=ISO-8859-1
Content-Length: 8132
Date: Sat, 10 Jul 2010 12:56:14 GMT

<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
  <head>
    <title>Apache Tomcat/5.5.20</title>
    <style type="text/css">
/**/
    body {
      color: #000000;
      background-color: #FFFFFF;
      font-family: Arial, "Times New Roman", Times, serif;
      margin: 10px 0px;</pre></div><div data-bbox="221 554 774 569" data-label="Caption"><p>图 Error! No text of specified style in document.-3 用GET方法向服务器获取资源</p></div>
```

§ 1.4 验证实验：手工试验 FTP 事务

1.4.1 实验要求

- 1) 学习配置 FTP 服务器 Serv-u FTP Server 基本方法与步骤;
- 2) 通过手工测试与 FTP 服务器通讯事务, 理解 FTP 协议的工作原理, 主动与被动工作模式、FTP 交互命令格式与消息头部内容。

1.4.2 实验步骤

通过手工输入 FTP 请求与几个 FTP 服务器进行交互, 并观察这些服务器的响应结果。试验输入 FTP 的请求既包括合法请求也包括不合法请求, 理解 FTP 请求的格式。

为理解 FTP 服务器传回的响应消息的格式与语义, 应特别注意响应的消息头部的内容。

例如: 通过消息头部的 server 域观察不同 FTP 服务器的响应是否有差别。

手工与 FTP 服务器交互时, 可使用 windows 提供的 telnet 或 ftp 命令。运行 telnet 或 ftp 后, 输入命令指定 ftp.zsu.edu.cn 服务器主机名和端口号 (默认为 21), 然后可输入原始的 telnet 或 ftp 命令直接与服务器进行交互。另一种指定服务器主机和端口号的方法是在启动 telnet 或 ftp 时给出命令行参数。例如运行以下命令连接到中山大学的 FTP 服务器。

```
telnet ftp.zsu.edu.cn 21
ftp ftp.zsu.edu.cn 21
```

一个 FTP 会话过程中, 始终有一个控制连接, 如果客户端请求文件, 则会有一个数据连接, 但 FTP 协议规定: 只要关闭了控制连接, 数据连接 (如果有) 也必须关闭。

不同的 FTP 服务器对 FTP 命令的支持程度可能不同, 但是 TCP 标准定义了所有 FTP 服务器都必须实现的命令, 我们的目标就是构建一个实现这个最小命令集的 FTP 服务器。

1.4.3 实验分析

(1) FTP 服务器安装与配置

用 Serv-u FTP Server 构建 FTP 服务器, 下载网址为 <http://www.serv-u.com/>。

安装 Serv-u FTP Server 后, 首先要启动 Serv-u FTP Server 服务器, 以测试 Tomcat 服务器是否安装成功。然后在浏览器地址栏中输入 <http://localhost:8080> (或 <http://127.0.0.1:8080>)。如果出现如图 Error! No text of specified style in document.-1 所示界面, 则说明 Tomcat 安装成功。



图 Error! No text of specified style in document.-4 Serv-U配置界面

(2) 手工试验 FTP 事务

```
C:\WINDOWS\system32\cmd.exe - ftp - ftp 127.0.0.1 - ftp - ftp 127.0.0.1...
D:\>ftp 127.0.0.1
Connected to 127.0.0.1.
220 Serv-U FTP Server v10.1 ready...
User (127.0.0.1:(none)): anonymous
331 User name okay, please send complete E-mail address as password.
Password:
230 User logged in, proceed.
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
drwx-rw-rw-  1 user  group      0 Jul 10 23:34 Chapter03 HTTP
drwx-rw-rw-  1 user  group      0 Jul 10 23:34 Chapter04 FTP
drwx-rw-rw-  1 user  group      0 Jul 10 23:34 Chapter05 RMI
drwx-rw-rw-  1 user  group      0 Jul 10 23:34 Chapter06 Corba To-Do Lis
```

图 Error! No text of specified style in document.-5 FTP客户使用FTP命令建立于服务器的连接

当 FTP 客户以 PORT 模式连接服务器时，动态地选择一个端口号（本次试验是 6015）连接服务器的 21 端口，注意这个端口号一定是 1024 以上的，因为 1024 以前的端口都已经预先被定义好，被一些典型的服务使用，当然有的还没使用，保留给以后会用到这些端口的资源服务。当经过 TCP 的三次握手后，连接（控制信道）被建立（如图 Error! No text of specified style in document.-5 和图 Error! No text of specified style in document.-6）。

【提示】Netstat 命令用于监视网络状态。可以通过“netstat-?”命令列出 netstat 的所有参数和用法。

```
C:\WINDOWS\system32\cmd.exe
D:\>netstat -na | find "127.0.0.1:21"
TCP    127.0.0.1:21          0.0.0.0:0             LISTENING
TCP    127.0.0.1:21          127.0.0.1:3778        ESTABLISHED
TCP    127.0.0.1:21          127.0.0.1:3928        ESTABLISHED
TCP    127.0.0.1:22          0.0.0.0:0             LISTENING
TCP    127.0.0.1:3778        127.0.0.1:21          ESTABLISHED
TCP    127.0.0.1:3928        127.0.0.1:21          ESTABLISHED
UDP    127.0.0.1:2248        *:*
```

图 Error! No text of specified style in document.-6 NetStat 显示服务器控制信道监听状况

如果用户要列出服务器上的目录结构（使用 ls 或 dir 命令），那么首先就要建立一个数据通道，因为只有数据通道才能传输目录和文件列表，此时用户会发出 PORT 指令告诉服务器连接自己的什么端口来建立一条数据通道（这个命令由控制信道发送给服务器），当服务器接到这一指令时，服务器会使用 20 端口连接用户在 PORT 指令中指定的端口号，用以发送目录的列表（如图 Error! No text of specified style in document.-7）。

```
D:\WINNT\System32\cmd.exe - ftp 127.0.0.1
230 Anonymous user logged in.
ftp> ls
200 PORT command successful.
150 Opening ASCII mode data connection for file list.
Book
Incoming
Incoming_WebMaster
jo14.zip
MP3
Software
上传文件请到Incoming目录.txt
会员专区
使用说明.txt
在校学习
如果有问题请联系 alone0@163.com.txt
教程
桌面背景
每日新鲜
226 Transfer complete.
ftp: 188 bytes received in 0.02Seconds 9.40Kbytes/sec.
ftp>
```

图 Error! No text of specified style in document.-7 ls 命令建立数据传输信道过程

经验证本次试验客户机使用 6044 端口。当完成这一操作时，FTP 客户也许要下载一个文件，那么就会发出 get 指令，请注意，这时客户会再次发送 PORT 指令，告诉服务器连接他的哪个“新”端口，可以先用 netstat -na 这个命令验证，上一次使用的 6044 已经处于 TIME_WAIT

状态（如图 Error! No text of specified style in document.-8）。

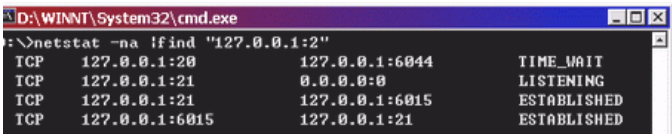


图 Error! No text of specified style in document.-8 netstat 命令验证 ls 命令建立的数据传输通道已关闭

当这个新的数据传输通道建立后（在微软的系统中，客户端通常会使用连续的端口，也就是说这一次客户端会用 6045 这个端口），就开始文件传输的工作。当 FTP 客户以 PASV 模式连接服务器时，情况有所不同。在初始化连接这个过程即连接服务器这个过程和 PORT 模式是一样的，不同的是，当 FTP 客户发送 ls、dir、get 等这些要求数据返回的命令时，他不向服务器发送 PORT 指令而是发送 PASV 指令，在这个指令中，用户告诉服务器自己要连接服务器的某一个端口，如果这个服务器上的这个端口是空闲的可用的，那么服务器会返回 ACK 的确认信息，之后数据传输通道被建立并返回用户所要的信息（根据用户发送的指令，如 ls、dir、get 等）；如果服务器的这个端口被另一个资源所使用，那么服务器返回 UNACK 的信息，那么这时，FTP 客户会再次发送 PASV 命令，这也就是所谓的连接建立的协商过程。为了验证这个过程我们不得不借助 CuteFTP Pro 这个大家经常使用的 FTP 客户端软件，因为微软自带的 FTP 命令客户端，不支持 PASV 模式。虽然可以使用 QUOTE PASV 这个命令强制使用 PASV 模式，但是当你用 ls 命令列出服务器目录列表，你会发现它还是使用 PORT 方式来连接服务器的。现在使用 CuteFTP Pro 以 PASV 模式连接服务器（如图 Error! No text of specified style in document.-9）。

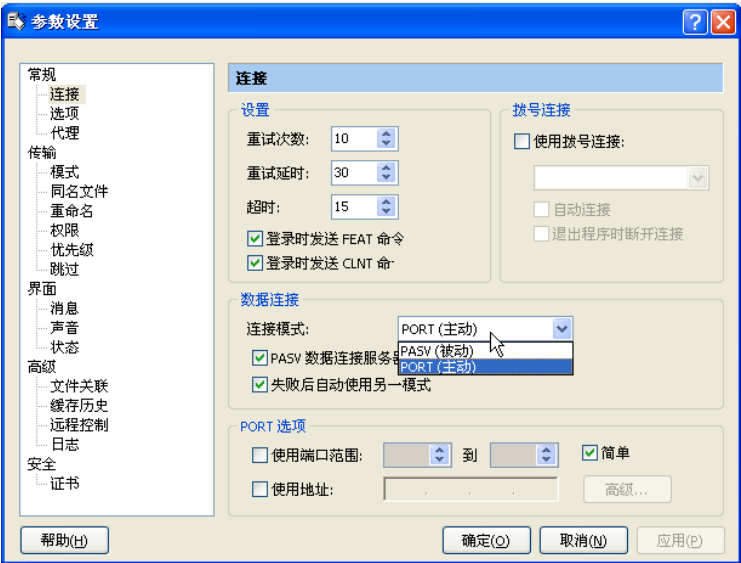


图 Error! No text of specified style in document.-9 使用 CUTEFTP Pro 以 PASV 模式连接服务器