Telnet协商

表2.1  TELNET命令

|  |  |  |
| --- | --- | --- |
| 名 称 | 编 码 | 说 明 |
| EOF | 236 | 文件结束符 |
| SUSP | 237 | 挂起当前进程 |
| ABORT | 238 | 中止进程 |
| EOR | 238 | 记录结束符 |
| SE | 240 | 子选项结束 |
| NOP | 241 | 空操作 |
| DM | 242 | 数据标记 |
| BRK | 243 | 终止符（break） |
| IP | 244 | 终止进程 |
| AO | 245 | 终止输出 |
| AYT | 246 | 请求应答 |
| EC | 247 | 终止符 |
| EL | 248 | 擦除一行 |
| GA | 249 | 继续 |
| SB | 250 | 子选项开始 |
| WILL | 251 | 选项协商 |
| WONT | 252 | 选项协商 |
| DO | 253 | 选项协商 |
| DON’T | 254 | 选项协商 |
| IAC | 255 | 字符0xFF |

表2.2  TELNET选项协商的六种情况

|  |  |  |
| --- | --- | --- |
| 发 送 者 | 接 收 者 | 说 明 |
| WILL | DO | 发送者想激活某选项，接受者接收该选项请求 |
| WILL | DON’T | 发送者想激活某选项，接受者拒绝该选项请求 |
| DO | WILL | 发送者希望接收者激活某选项，接受者接受该请求 |
| DO | DON’T | 发送者希望接收者激活某选项，接受者拒绝该请求 |
| WONT | DON’T | 发送者希望使某选项无效，接受者必须接受该请求 |
| DON’T | WONT | 发送者希望接受者使某选项无效，接受者必须接收该请求 |

表2.3  TELNET选项代码

|  |  |  |
| --- | --- | --- |
| 选项标识 | 名 称 | RFC |
| 1 | 回应（echo） | 857 |
| 3 | 禁止继续 | 858 |
| 5 | 状态 | 859 |
| 6 | 时钟标识 | 860 |
| 24 | 终端类型 | 1019 |
| 31 | 窗口大小 | 1073 |
| 32 | 终端速率 | 1079 |
| 33 | 远程流量控制 | 1372 |
| 34 | 行模式 | 1184 |

选项协商需要3个字节：IAC，然后是WILL、DO、WONT或DONT；最后一个标识字节用来指明操作的选项。

由于Telnet规则规定，对于激活选项请求（如WILL和DO），有权同意或者不同意。而对于使选项失效请求（如WONT和DONT），必须同意。这样，4种请求就会组合出 6种情况，如图26-9所示。选项协商需要3个字节：一个IAC字节，接着一个字节是WILL,DO,WONT和DONT这四者之一，最后一个ID字节指 明激活或禁止选项。

二、数据包分析：

过程概述：

第一步：172.16.1.2发出arp包；172.16.1.3回复arp包；

第二步：用三次握手建立tcp连接；

        SYN=1，seq=x；SYN=1，ACK=1，seq=y，ack=x+1；ACK=1，seq=x+1，ack=y+1；

第三步：进行telnet连接的建立，进行远程控制；

第四步：四次握手（两个二次握手）释放tcp连接；

        1.FIN=1，seq=u；2.ACK=1，seq=v，ack=u+1；2.FIN=1，ACK=1，seq=w，ack=u+1；1.ACK=1，seq=u+1，ack=w+1；

telnet包的分析：

在第18个数据包中开始出现数据，Red Hat Linux release 9 (Shrike)

这是172.16.1.3传送给172.16.1.2的第一条显示信息。之后的数据包则传送了登录名和登录密码，以明文的形式。

1、每击打键盘一次，就发送一个数据包，所以登录名yanji就发送了5个数据包；解释：

Telnet工作模式有四种：

1） 半双工：客户端在接收用户输入之前，必须从服务器进程获得GO AHEAD (G A）命令. 现在已很少使用.

2）一次一个字符：客户端把用户输入的每个字符都单独发送给服务器，服务器回显字符给客户端. 是目前大多数Telnet程序的默认方式. SUPPRESS GO AHEAD选项和ECHO选项必须同时有效.

3） 准行方式（kludge line mode): 用户每键入一行信息，客户端向服务器发送一次. 当上面两个选项其中之一无效时采用此模式.

4） 行方式：类似准行方式，纠正了准行方式的缺点. 较新的Telnet程序支持这种方式.

2、为什么每次服务器会会送客户机每次发送的数据；解释：

telnet的工作过程

使用Telnet协议进行远程登录时需要满足以下条件：在本地计算机上必须装有包含Telnet协议的客户程序；必须知道远程主机的Ip地址或域名；必须知道登录标识与口令。

Telnet远程登录服务分为以下4个过程：

1）本地与远程主机建立连接。该过程实际上是建立一个TCP连接，用户必须知道远程主机的Ip地址或域名；

2）将本地终端上输入的用户名和口令及以后输入的任何命令或字符以NVT（Net Virtual Terminal）格式传送到远程主机。该过程实际上是从本地主机向远程主机发送一个IP数据包；

3）将远程主机输出的NVT格式的数据转化为本地所接受的格式送回本地终端，包括输入命令回显和命令执行结果；

4）最后，本地终端对远程主机进行撤消连接。该过程是撤销一个TCP连接。

3、为什么说telnet的效率不高；解释：

Telnet中的数据流向：

数据信息被用户从本地键盘键入并通过操作系统传到客户机程序，客户机程序将其处理后返回操作系统，并由操作系统经过网络传送到远程机器，远程操作系统将所 接收数据传给服务器程序，并经服务器程序再次处理后返回到操作系统上的伪终端入口点，最后，远程操作系统将数据传送到用户正在运行的应用程序，这便是一次 完整的输入过程；输出将按照同一通路从服务器传送到客户机。　因为每一次的输入和输出，计算机将切换进程环境好几次，这个开销是很昂贵的。还好用户的键入 速率并不算高，这个缺点我们仍然能够接受。

4、 SUPPRESS GO AHEAD和ECHO选项作用和意义：

如果激活"GO AHEAD"选项，则是一个半双工模式，为了工作在“一次一个字符”，需要抑制此选项，即激活"SUPPRESS GO AHEAD".ECHO选项使回显功能有效.两者同时激活表示采用“一次一个字符” 工作方式.

5、telnet中<IAC,WILL,24><IAC,DO,24><IAC,SB,24,1,IAC,SE>选项的含义：

子选项协商

有些选项不是仅仅用“激活”或“禁止”就能够表达的。指定终端类型就是一个例子，客户进程必须发送用一个A S C I I字符串来表示终端类型。为了处理这种选项，我们必须定义子选项协商机制。

在R F C 1 0 9 1 [ VanBokkelen 1989]中定义了如何表示终端类型这样的子选项协商机制。首先连接的某一方（通常是客户进程）发送3个字节的字符序列来请求激活该选项。

<IAC,WILL,24>

这里的2 4（十进制）是终端类型选项的I D号。如果收端（通常是服务器进程）同意，那么响应数据是：

<IAC,DO,24>

然后服务器进程再发送如下的字符串：

<IAC,SB,24,1,IAC,SE>

该字符串询问客户进程的终端类型。其中S B是子选项协商的起始命令标志。下一个字节的“2 4”代表这是终端类型选项的子选项（通常S B后面的选项值就是子选项所要提交的内容）。下一个字节的“ 1”表示“发送你的终端类型”。子选项协商的结束命令标志也是I A C，就像S B是起始命令标志一样。如果终端类型是i b m p c，客户进程的响应命令将是：

第4个字节“0”代表“我的终端类型是”（在Assigned Numbers RFC文档中有正式的关于终端类型的数值定义，但是最起码在U n i x系统之间，终端类型可以用任何对方可理解的数据进行表示。只要这些数据在t e r m c a p或t e r m i n f o[数据库](http://lib.csdn.net/base/14" \o "MySQL知识库" \t "http://blog.csdn.net/frank_jb/article/details/_blank)中有定义）。在Te l n e t子选项协商过程中，终端类型用大写表示，当服务器收到该字符串后会自动转换为小写字符。

6、在所创建的文件shiyan中的内容是以八进制ascii码来传送

适应异构

为了使多个操作系统间的Telnet交互操作成为可能，就必须详细了解异构计算机和操作系统。比如，一些操作系统需要每行文本用ASCⅡ回车控制符 （CR）结束，另一些系统则需要使用ASCⅡ换行符（LF），还有一些系统需要用两个字符的序列回车-换行（CR-LF）；再比如，大多数操作系统为用户 提供了一个中断程序运行的快捷键，但这个快捷键在各个系统中有可能不同（一些系统使用CTRL+C，而另一些系统使用ESCAPE）。如果不考虑系统间的 异构性，那么在本地发出的字符或命令，传送到远地并被远程系统解释后很可能会不准确或者出现错误。因此，Telnet协议必须解决这个问题。　为了适应异 构环境，Telnet协议定义了数据和命令在Internet上的传输方式，此定义被称作网络虚拟终端NVT（Net Virtual Terminal）。它的应用过程如下：　对于发送的数据：客户机软件把来自用户终端的按键和命令序列转换为NVT格式，并发送到服务器，服务器软件将收 到的数据和命令，从NVT格式转换为远地系统需要的格式；　对于返回的数据：远地服务器将数据从远地机器的格式转换为NVT格式，而本地客户机将接收到的 NVT格式数据再转换为本地的格式。

7、关于telnet的基本内容

Telnet是位于OSI模型的第7层---应用层上的一种协议，是一个通过创建虚拟终端提供连接到远程主机终端仿真的TCP/IP协议。这一协议需要通 过用户名和口令进行认证，是Internet远程登陆服务的标准协议。应用Telnet协议能够把本地用户所使用的计算机变成远程主机系统的一个终端。它 提供了三种基本服务：

1）Telnet定义一个网络虚拟终端为远程系统提供一个标准接口。客户机程序不必详细了解远程系统，他们只需构造使用标准接口的程序；

2）Telnet包括一个允许客户机和服务器协商选项的机制，而且它还提供一组标准选项；　.

3）Telnet对称处理连接的两端，即Telnet不强迫客户机从键盘输入，也不强迫客户机在屏幕上显示输出。

8、为什么会有malformed packet

有可能数据在转发传输的过程中某些字节发生错误。查看一下数据的各层校验和是否有错误。或者就是发出数据的设备的协议栈程序出错。