

组织：中国互动出版网 (<http://www.china-pub.com/>)

RFC 文档中文翻译计划 (<http://www.china-pub.com/compters/emook/aboutemook.htm>)

E-mail: ouyang@china-pub.com

译者：seamansl (seamansl seamansl@263.net)

译文发布时间：2001-6-5

版权：本中文翻译文档版权归中国互动出版网所有。可以用于非商业用途自由转载，但必须保留本文档的翻译及版权信息。

Network Working Group
Request for Comments: 1091
Obsoletes: RFC 930

J. VanBokkelen
FTP Software, Inc.
February 1989

TELNET 终端类型选项

(RFC1091 Telnet Terminal-Type Option)

Status of This Memo

This RFC specifies a standard for the Internet community. Hosts on the Internet that exchange terminal type information within the Telnet protocol are expected to adopt and implement this standard.

This standard supersedes RFC 930. A change is made to permit cycling through a list of possible terminal types and selecting the most appropriate.

Distribution of this memo is unlimited.

1. 命令名称及编号

TERMINAL-TYPE 24

2. 命令含义

IAC WILL TERMINAL-TYPE

发送者希望在接下来的子选项协商中发送终端类型信息。

IAC WON' T TERMINAL-TYPE

发送者拒绝发送终端类型信息。

IAC DO TERMINAL-TYPE

发送者希望在接下来的子选项协商中接收终端类型信息。

IAC DON' TREMINAL-TYPE

发送者拒绝接受终端类型信息。

IAC SB TERMINAL-TYPE SEND IAC SE

服务器要求客户方传送它的（客户方的）下一个终端类型，并切换到仿真模式（如果它支持多个终端类型）。SEND 的编码为 1。（见下面）

IAC SB TERMINAL-TYPE IS ... IAC SE

客户方声明自己当前的(或仅有的)终端类型。

4. 使用选项的原因

在大部分机器的位图显示器上(如 PC 机和图形工作站),客户方终端仿真程序被用来模拟传统的 ASCII 终端。这些程序大部分都有多种仿真模式,通常特征也多种多样。同时,现在的主机系统软件和应用能处理很多终端类型。因此,需要有一种方法:客户向服务器提交一系列可用的终端仿真模式,服务器从中选出一个合适的模式(无论何种原因)。另外,还需要一种机制:能够在会话过程中,可能是根据应用程序的需要,去改变仿真模式。

Telnet 中现有的终端类型传送机制在设计时没有考虑多仿真模式。虽然允许多个名字是,但它们被看作同义词。仿真模式的变化没有定义,并且所有的模式只能被检查一次。

本文档定义了对现存机制的简单扩充,它满足上面的标准。假定执行动作按以前的标准编码,目的是获得充分的向后兼容性。

5. 选项描述

在 Telnet 中,规定通过传统的 Telnet 选项协商来自动交换终端类型信息。WILL 和 DO 仅被用请求和授权许可,来为后面的协商做准备。真正的状态信息交换是在后面的子选项协商命令中进行的(IAC SB TERMINAL-TYPE...)。

一旦两台主机交换过 WILL 和 DO 后,发送 DO TERMINAL-TYPE 命令的主机(服务器)可以任意的询问终端类型信息。只有服务器能够发送询问命令(IAC SB TERMINAL-TYPE SEND SE)也只有客户方能够传送真正的类型信息(用 IAC SE TERMINAL-TYPE IS ... IAC SE 命令)。终端类型信息不能自发的被传送,只能是为了响应请求而被发出。

终端类型信息是一个 NVT ASCII 字符串。在这个字符串中,不区分大小写。所有有效的终端类型名都能够在最新的“数字分配标识”RFC 文档中找到。

当 Telnet 客户为响应服务器的询问传送终端类型信息时,就意味着客户必须同步地改变仿真模式,除非传送的终端类型是前一个类型的同义词,或者进入新的模式有其它的先决条件(例如已约定 Telnet 的二进制选项)。在收到请求信息时,并不表示服务器会立即改变它的处理过程。但是,这种信息可能被传递给一个进程。此进程可以调整发送的数据以适应不同终端的不同特点。例如,某些操作系统有一个终端驱动器,它能通过接收一些代码从而指出哪种终端类型正在被驱动。在这样的系统中,通过使用 TERMINAL TYPE 和 BINARY 选项, Telnet 服务程序能够安排终端的驱动过程(包括一些标准网络虚拟终端不具有的特殊功能),就像它们被直接连接在一起。

注意,这个规范是特意不对称的。它假定服务方操作系统和应用不能在一个连接的任意点改变终端类型。因此,只有在服务器要求时,客户方才能发送新的终端类型(并潜在的改变仿真模式)。

6. 执行问题

“终端类型”信息可以是任意的 NVT ASCII 字符串。只要字符串对协商的两端都有意义。在数字分配标识文档的终端类型列表中,试图将由于终端类型写法的可选择性而引起的混淆降到最小。例如,当一方将一个终端称作“IBM3278-2”,而另一方称它为“IBM-3278/2”时,就会出现混淆。对于一个不能识别的终端类型不会有否定的确认,但是当没有指定一个有效的终端类型名时,有些选项(例如切换到 BINARY 模式)可能会被拒绝。

在一些情况下,或者某些特殊的终端可能有不止一个名字,例如有一个特殊的类型和几个普通的类型,或者客户方是一个带有复合式显示器的工作站,它能仿真多种终端模式。在这种情况下,发送 TERMINAL-TYPE IS 命令的一方应该用不同的名字来响应接下来的 TERMINAL-TYPE SEND 命令。用这种方法,当不能解释第一次响应的类型时, Telnet 服务器

能够提示进行选择。如果客户方支持多种不同的终端仿真，除非某些特殊的仿真需要其他的 Telnet 选项（如 BINARY）作为先决条件（在这种条件下，当先决条件被执行后，仿真模式将转换到最后发送的类型），否则必须将仿真模式转换到最后的类型。当类型名是同义词时，它们应该按从细节最多的名字到细节最少的顺序被发送。

当服务器（收到 TERMINAL-TYPE IS 的一方）在连续的时间收到相同的类型时，这说明此为有效类型列表的末尾。同样，客户端应该通过重复发送最后一个类型以表示它已经传了所有有效的类型名。如果客户端收到一个额外的请求，它表示服务方（发送 IS 的一方）希望返回到有效类型列表的顶部（可能希望选择最小的 *Nevals*）。

以前的标准规定，当在两个连续的时间收到相同的响应时，服务器将停止发送 TERMINAL-TYPE SEND 命令，并将根据原来的标准工作。假定客户方的执行将照以前的标准去发送列表中的最后一个类型，以响应第三次请求（和第二次）。新式的服务器必须能识别这一点并且不再发送更多的请求。

当终端类型是未知的或是一方不能识别时，将使用“未知”类型。

最新的全部终端类型名列表被保存在“数字分配标识”中。一个终端类型的最大长度是 40 字节。

7. 用户接口

符合此规范的 Telnet 客户和服务器应该在它们的用户接口中提供以下的功能：

支持多仿真模式的客户应该允许用户能够在连接建立以前指定首选的模式（即哪个名字首先被发送）。当协商开始后，发送类型的顺序就不能再更改。当服务器不支持 TERMINAL-TYPE 时，这种模式也将成为它的默认模式。

服务器应该用某种方式存储当前的终端类型名和有效的名字列表。只有这样，它才能获得所需要的用户（通过键盘命令）和任何应用程序的信息。另外，也需要一种相应的机制（通过建立一系列的 SEND/IS 子协商）去请求改变终端类型。

8. 例子

在这个例子中，服务器查找第一个可接受的类型。

```
Server: IAC DO TERMINAL-TYPE
```

```
Client: IAC WILL TERMINAL-TYPE
```

（现在，服务器可在任意时刻请求终端类型。）

```
Server: IAC SB TERMINAL-TYPE SEND IAC SE
```

```
Client: IAC SB TERMINAL-TYPE IS IBM-3278-2 IAC SE
```

在这个例子中，服务器请求额外的终端类型，并接受响应的第二种类型（列表中的最后一个）

```
Server: IAC DO TERMINAL-TYPE
```

```
Client: IAC WILL TERMINAL-TYPE
```

（现在，服务器可在任意时刻请求终端类型）

```
Server: IAC SB TERMINAL-TYPE SEND IAC SE
```

Client: IAC SB TERMINAL-TYPE IS ZENITH-H19 IAC SE

Server: IAC SB TERMINAL-TYPE SEND IAC SE

Client: IAC SB TERMINAL-TYPE IS UNKNOWN IAC SE

Server: IAC SB TERMINAL-TYPE SEND IAC SE

Client: IAC SB TERMINAL-TYPE IS UNKNOWN IAC SE

在这个例子中，服务器请求额外的终端类型，并超出了类型列表的末尾。它选择客户提供的第一种类型（新型的客户端和服务端）。

Server: IAC DO TERMINAL-TYPE

Client: IAC WILL TERMINAL-TYPE

（现在，服务器可在任意时刻请求终端类型。）

Server: IAC SB TERMINAL-TYPE SEND IAC SE

Client: IAC SB TERMINAL-TYPE IS DEC-VT220 IAC SE

Server: IAC SB TERMINAL-TYPE SEND IAC SE

Client: IAC SB TERMINAL-TYPE IS DEC-VT100 IAC SE

Server: IAC SB TERMINAL-TYPE SEND IAC SE

Client: IAC SB TERMINAL-TYPE IS DEC-VT52 IAC SE

Server: IAC SB TERMINAL-TYPE SEND IAC SE

Client: IAC SB TERMINAL-TYPE IS DEC-VT52 IAC SE

Server: IAC SB TERMINAL-TYPE SEND IAC SE

Client: IAC SB TERMINAL-TYPE IS DEC-VT220 IAC SE

9. References:

- [1] Postel, J., and J. Reynolds, "Telnet Protocol Specification", [RFC 854](#), USC Information Sciences Institute, May 1983.

- [2] Postel, J., and J. Reynolds, "Telnet Option Specification", [RFC 855](#), USC Information Sciences Institute, May 1983.
- [3] Solomon, M., and E. Wimmers, "Telnet Terminal Type Option", [RFC 930](#), University of Wisconsin - Madison, January 1985.
- [4] Reynolds, J., and J. Postel, "Assigned Numbers", [RFC 1010](#), USC Information Sciences Institute, May 1987.

Reviser's note:

I owe much of this text to RFCs 884 and 930, by Marvin Solomon and Edward Wimmers of the University of Wisconsin - Madison, and I owe the idea of the extension to discussions on the "tn3270" mailing list in the Summer of 1987.

Author's Address

James VanBokkelen
FTP Software, Inc.
26 Princess Street
Wakefield, MA 01880-3004

Phone: (617) 246-0900

Email: jbvb@ftp.com