

The TELNET Protocol

Reference: RFC 854

Netprog: TELNET

1

TELNET vs. telnet

- TELNET is a *protocol* that provides “a general, bi-directional, eight-bit byte oriented communications facility”.
- **telnet** is a *program* that supports the TELNET protocol over TCP.
- Many application protocols are built upon the TELNET protocol.

Netprog: TELNET

2

The TELNET Protocol

- TCP connection
- data and control over the same connection.
- Network Virtual Terminal
- negotiated options

Netprog: TELNET

3

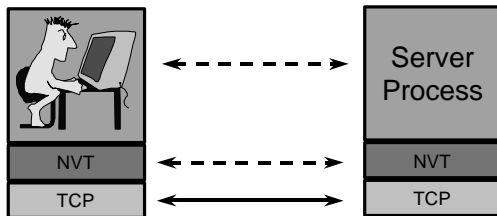
Network Virtual Terminal

- intermediate representation of a generic terminal.
- provides a standard language for communication of terminal control functions.

Netprog: TELNET

4

Network Virtual Terminal



Netprog: TELNET

5

Negotiated Options

- All NVTs support a minimal set of capabilities.
- Some terminals have more capabilities than the minimal set.
- The 2 endpoints negotiate a set of mutually acceptable options (character set, echo mode, etc).

Netprog: TELNET

6

Negotiated Options

- The protocol for requesting optional features is well defined and includes rules for eliminating possible negotiation “loops”.
- The set of options is not part of the TELNET protocol, so that new terminal features can be incorporated without changing the TELNET protocol.

Netprog: TELNET

7

Option examples

- Line mode vs. character mode
- echo modes
- character set (EBCDIC vs. ASCII)

Netprog: TELNET

8

Control Functions

- TELNET includes support for a series of control functions commonly supported by servers.
- This provides a uniform mechanism for communication of (the supported) control functions.

Netprog: TELNET

9

Control Functions

- Interrupt Process (IP)
 - suspend/abort process.
- Abort Output (AO)
 - process can complete, but send no more output to user's terminal.
- Are You There (AYT)
 - check to see if system is still running.

Netprog: TELNET

10

More Control Functions

- Erase Character (EC)
 - delete last character sent
 - typically used to edit keyboard input.
- Erase Line (EL)
 - delete all input in current line.

Netprog: TELNET

11

Command Structure

- All TELNET commands and data flow through the same TCP connection.
- Commands start with a special character called the Interpret as Command *escape* character (IAC).
- The IAC code is 255.
- If a 255 is sent as data - it must be followed by another 255.

Netprog: TELNET

12

Looking for Commands

- Each receiver must look at each byte that arrives and look for IAC.
- If IAC is found and the next byte is IAC - a single byte is presented to the application/terminal (a 255).
- If IAC is followed by any other code - the TELNET layer interprets this as a command.

Netprog: TELNET

13

Command Codes

- | | | | |
|-------|-----|---------|-----|
| ■ IP | 243 | ■ WILL | 251 |
| ■ AO | 244 | ■ WON'T | 252 |
| ■ AYT | 245 | ■ DO | 253 |
| ■ EC | 246 | ■ DON'T | 254 |
| ■ EL | 247 | ■ IAC | 255 |

Netprog: TELNET

14

Playing with TELNET

- You can use the `telnet` program to play with the TELNET protocol.
- `telnet` is a *generic* TCP client.
 - Sends whatever you type to the TCP socket.
 - Prints whatever comes back through the TCP socket.
 - Useful for testing TCP servers (ASCII based protocols).

Netprog: TELNET

15

Some TCP Servers you can play with

- Many Unix systems have these servers running (by default):

- `echo` port 7
- `discard` port 9
- `daytime` port 13
- `chargen` port 19

Netprog: TELNET

16

telnet hostname port

```
> telnet rcs.rpi.edu 7
Trying 128.113.113.33...
Connected to cortex.sss.rpi.edu
(128.113.113.33).
Escape character is '^]'.
Hi dave
Hi dave
stop it
stop it
^]
telnet> quit
Connection closed.
```

Netprog: TELNET

17

telnet vs. TCP

- Not all TCP servers talk TELNET (most don't)
- You can use the `telnet` program to play with these servers, but the fancy commands won't do anything.
 - type `^]`, then "help" for a list of fancy TELNET stuff you can do in `telnet`.

Netprog: TELNET

18
