NAME

telnetd – DARPA TELNET protocol server

SYNOPSIS

/usr/libexec/telnetd [**-a** *authmode*] [**-B**] [**-D** *debugmode*] [**-edebug**] [**-h**] [**-I***initid*] [**-l**] [**-k**] [**-n**] [**-r***lowpty-highpty*] [**-s**] [**-S** *tos*] [**-u** *len*] [**-U**] [**-X** *authtype*] [**-debug** [*port*]]

DESCRIPTION

The **telnetd** command is a server which supports the DARPA standard TELNET virtual terminal protocol. **Telnetd** is normally invoked by the internet server (see inetd(8)) for requests to connect to the TELNET port as indicated by the /etc/services file (see services(5)). The **-debug** option may be used to start up **telnetd** manually, instead of through inetd(8). If started up this way, *port* may be specified to run **telnetd** on an alternate TCP port number.

The **telnetd** command accepts the following options:

-a *authmode*
: This option may be used for specifying what mode should be used for authentication. Note that this option is only useful if **telnetd** has been compiled with support for the AUTHENTICATION option. There are several valid values for *authmode:*

- debug — Turns on authentication debugging code.
- user — Only allow connections when the remote user can provide valid authentication information to identify the remote user, and is allowed access to the specified account without providing a password.
- valid — Only allow connections when the remote user can provide valid authentication information to identify the remote user. The login(1) command will provide any additional user verification needed if the remote user is not allowed automatic access to the specified account.
- other — Only allow connections that supply some authentication information. This option is currently not supported by any of the existing authentication mechanisms, and is thus the same as specifying **-a valid**.
- none — This is the default state. Authentication information is not required. If no or insufficient authentication information is provided, then the login(1) program will provide the necessary user verification.
- off — This disables the authentication code. All user verification will happen through the login(1) program.

-B
: Specifies bftp server mode. In this mode, **telnetd** causes login to start a bftp(1) session rather than the user's normal shell. In bftp daemon mode normal logins are not supported, and it must be used on a port other than the normal TELNET port.

-D *debugmode*
: This option may be used for debugging purposes. This allows **telnetd** to print out debugging information to the connection, allowing the user to see what **telnetd** is doing. There are several possible values for *debugmode:*

- **options** — Prints information about the negotiation of TELNET options.

report Prints the **options** information, plus some additional information about what processing is going on.

netdata Displays the data stream received by **telnetd.**

ptydata Displays data written to the pty.

exercise Has not been implemented yet.

−debug Enables debugging on each socket created by **telnetd** (see `SO_DEBUG` in `socket`(2)).

−edebug If **telnetd** has been compiled with support for data encryption, then the **−edebug** option may be used to enable encryption debugging code.

−h Disables the printing of host-specific information before login has been completed.

−I *initid* This option is only applicable to UNICOS systems prior to 7.0. It specifies the `ID` from `/etc/inittab` to use when init starts login sessions. The default `ID` is `fe.`

−k This option is only useful if **telnetd** has been compiled with both linemode and kludge linemode support. If the **−k** option is specified, then if the remote client does not support the `LINEMODE` option, then **telnetd** will operate in character at a time mode. It will still support kludge linemode, but will only go into kludge linemode if the remote client requests it. (This is done by by the client sending `DONT SUPPRESS-GO-AHEAD` and `DONT ECHO`.) The **−k** option is most useful when there are remote clients that do not support kludge linemode, but pass the heuristic (if they respond with `WILL TIMING-MARK` in response to a `DO TIMING-MARK)` for kludge linemode support.

−l Specifies line mode. Tries to force clients to use line- at-a-time mode. If the `LINEMODE` option is not supported, it will go into kludge linemode.

−n Disable `TCP` keep-alives. Normally **telnetd** enables the TCP keep-alive mechanism to probe connections that have been idle for some period of time to determine if the client is still there, so that idle connections from machines that have crashed or can no longer be reached may be cleaned up.

−r *lowpty−highpty*
This option is only enabled when **telnetd** is compiled for `UNICOS.` It specifies an inclusive range of pseudo-terminal devices to use. If the system has sysconf variable `_SC_CRAY_NPTY` configured, the default pty search range is 0 to `_SC_CRAY_NPTY;` otherwise, the default range is 0 to 128. Either *lowpty* or *highpty* may be omitted to allow changing either end of the search range. If *lowpty* is omitted, the - character is still required so that **telnetd** can differentiate *highpty* from *lowpty*.

−s This option is only enabled if **telnetd** is compiled with support for SecurID cards. It causes the **−s** option to be passed on to `login`(1), and thus is only useful if `login`(1) supports the **−s** flag to indicate that only SecurID validated logins are allowed, and is usually useful for controlling remote logins from outside of a firewall.

−S *tos*

−u *len* This option is used to specify the size of the field in the `utmp` structure that holds the remote host name. If the resolved host name is longer than *len*, the dotted decimal value will be used instead. This allows hosts with very long host names that overflow this field to still be uniquely identified. Specifying **−u0** indicates that only dotted decimal addresses should be put into the `utmp` file.

−U This option causes **telnetd** to refuse connections from addresses that cannot be mapped back into a symbolic name via the `gethostbyaddr`(3) routine.

-X *authtype*

This option is only valid if **telnetd** has been built with support for the authentication option. It disables the use of *authtype* authentication, and can be used to temporarily disable a specific authentication type without having to recompile **telnetd**.

Telnetd operates by allocating a pseudo-terminal device (see `pty`(4)) for a client, then creating a login process which has the slave side of the pseudo-terminal as `stdin`, `stdout` and `stderr`. **Telnetd** manipulates the master side of the pseudo-terminal, implementing the TELNET protocol and passing characters between the remote client and the login process.

When a TELNET session is started up, **telnetd** sends TELNET options to the client side indicating a willingness to do the following TELNET options, which are described in more detail below:

```
DO AUTHENTICATION
WILL ENCRYPT
DO TERMINAL TYPE
DO TSPEED
DO XDISPLOC
DO NEW-ENVIRON
DO ENVIRON
WILL SUPPRESS GO AHEAD
DO ECHO
DO LINEMODE
DO NAWS
WILL STATUS
DO LFLOW
DO TIMING-MARK
```

The pseudo-terminal allocated to the client is configured to operate in cooked mode, and with `XTABS and CRMOD` enabled (see `tty`(4)).

Telnetd has support for enabling locally the following TELNET options:

WILL ECHO
: When the `LINEMODE` option is enabled, a `WILL ECHO` or `WONT ECHO` will be sent to the client to indicate the current state of terminal echoing. When terminal echo is not desired, a `WILL ECHO` is sent to indicate that telnetd will take care of echoing any data that needs to be echoed to the terminal, and then nothing is echoed. When terminal echo is desired, a `WONT ECHO` is sent to indicate that telnetd will not be doing any terminal echoing, so the client should do any terminal echoing that is needed.

WILL BINARY
: Indicates that the client is willing to send a 8 bits of data, rather than the normal 7 bits of the Network Virtual Terminal.

WILL SGA
: Indicates that it will not be sending `IAC GA,` go ahead, commands.

WILL STATUS
: Indicates a willingness to send the client, upon request, of the current status of all TELNET options.

WILL TIMING-MARK
: Whenever a `DO TIMING-MARK` command is received, it is always responded to with a `WILL TIMING-MARK`

WILL LOGOUT
: When a `DO LOGOUT` is received, a `WILL LOGOUT` is sent in response, and the

TELNET session is shut down.

WILL ENCRYPT
: Only sent if **telnetd** is compiled with support for data encryption, and indicates a willingness to decrypt the data stream.

Telnetd has support for enabling remotely the following TELNET options:

DO BINARY
: Sent to indicate that telnetd is willing to receive an 8 bit data stream.

DO LFLOW
: Requests that the client handle flow control characters remotely.

DO ECHO
: This is not really supported, but is sent to identify a 4.2BSD `telnet`(1) client, which will improperly respond with `WILL ECHO.` If a `WILL ECHO` is received, a `DONT ECHO` will be sent in response.

DO TERMINAL-TYPE
: Indicates a desire to be able to request the name of the type of terminal that is attached to the client side of the connection.

DO SGA
: Indicates that it does not need to receive `IAC GA,` the go ahead command.

DO NAWS
: Requests that the client inform the server when the window (display) size changes.

DO TERMINAL-SPEED
: Indicates a desire to be able to request information about the speed of the serial line to which the client is attached.

DO XDISPLOC
: Indicates a desire to be able to request the name of the X windows display that is associated with the telnet client.

DO NEW-ENVIRON
: Indicates a desire to be able to request environment variable information, as described in RFC 1572.

DO ENVIRON
: Indicates a desire to be able to request environment variable information, as described in RFC 1408.

DO LINEMODE
: Only sent if **telnetd** is compiled with support for linemode, and requests that the client do line by line processing.

DO TIMING-MARK
: Only sent if **telnetd** is compiled with support for both linemode and kludge linemode, and the client responded with `WONT LINEMODE.` If the client responds with `WILL TM,` the it is assumed that the client supports kludge linemode. Note that the [**-k**] option can be used to disable this.

DO AUTHENTICATION
: Only sent if **telnetd** is compiled with support for authentication, and indicates a willingness to receive authentication information for automatic login.

DO ENCRYPT
: Only sent if **telnetd** is compiled with support for data encryption, and indicates a willingness to decrypt the data stream.

ENVIRONMENT

FILES

`/etc/services`
`/etc/inittab` (UNICOS systems only)
`/etc/iptos` (if supported)
`/usr/ucb/bftp` (if supported)

SEE ALSO

telnet(1), login(1), bftp(1) (if supported)

STANDARDS

RFC−854 TELNET PROTOCOL SPECIFICATION
RFC−855 TELNET OPTION SPECIFICATIONS
RFC−856 TELNET BINARY TRANSMISSION
RFC−857 TELNET ECHO OPTION
RFC−858 TELNET SUPPRESS GO AHEAD OPTION
RFC−859 TELNET STATUS OPTION
RFC−860 TELNET TIMING MARK OPTION
RFC−861 TELNET EXTENDED OPTIONS - LIST OPTION
RFC−885 TELNET END OF RECORD OPTION
RFC−1073 Telnet Window Size Option
RFC−1079 Telnet Terminal Speed Option
RFC−1091 Telnet Terminal-Type Option
RFC−1096 Telnet X Display Location Option
RFC−1123 Requirements for Internet Hosts -- Application and Support
RFC−1184 Telnet Linemode Option
RFC−1372 Telnet Remote Flow Control Option
RFC−1416 Telnet Authentication Option
RFC−1411 Telnet Authentication: Kerberos Version 4
RFC−1412 Telnet Authentication: SPX
RFC−1571 Telnet Environment Option Interoperability Issues
RFC−1572 Telnet Environment Option

BUGS

Some TELNET commands are only partially implemented.

Because of bugs in the original 4.2 BSD telnet(1), **telnetd** performs some dubious protocol exchanges to try to discover if the remote client is, in fact, a 4.2 BSD telnet(1).

Binary mode has no common interpretation except between similar operating systems (Unix in this case).

The terminal type name received from the remote client is converted to lower case.

Telnetd never sends TELNET IAC GA (go ahead) commands.