GNU Netcat

The famous networking tool

by Giovanni Giacobbi

1 Overview

Netcat is a simple Unix utility which reads and writes data across network connections, using TCP or UDP protocol. It is designed to be a reliable "back-end" tool that can be used directly or easily driven by other programs and scripts. At the same time, it is a feature-rich network debugging and exploration tool, since it can create almost any kind of connection you would need and has several interesting built-in capabilities. Netcat, or "nc" as the original program was named, should have been supplied long ago as another one of those cryptic but standard Unix tools.

2 Invoking

Netcat has three main modes of functionality. These are the connect mode, the listen mode, and the tunnel mode.

The most common mode is the connect mode, which for example allows the output of a locally called command to be redirected for example to a remote netcat listening or to any other kind of daemon waiting for a connection.

On the other hand, the listen mode can be used to obtain some kind of stream of data from a remote site.

The most new feature is the tunnel mode, which is a powerful and reliable mode that allows tunneling a remote site towards any other remote site, allowing to specify for example from which interface create the connection and from which port.

2.1 Basic Startup Options

'`-V`'
'`--version`'
: Display the version of netcat and exit.

'`-h`'
'`--help`'
: Print a help message describing most common netcat's command-line switches and a short description.

'`-v`'
'`--verbose`'
: Prints status messages, usually needed for using netcat as user front-end. All messages are printed to stderr in order not to affect the data stream.

  Use this option double to get more messages.

2.2 Protocol and Interface Options

'`-t`'
'`--tcp`'
: Selects the TCP protocol, this is the default. It may be useful (see Tunnel Mode) to specify this option after for example the UDP option in order to allow a cross-protocol bridge between TCP and UDP.

'`-u`'
'`--udp`'
: Selects the UDP protocol. See the –tcp option.

'`-p NUM`'
'`--local-port=NUM`'
: Selects the local port. In listen and tunnel mode, it specifies which port to use for listening, while in connect mode it specifies the source port (the port from which originating the connection).

  If this option is not specified, the OS will assign a random available port.

'`-s ADDRESS`'
'`--source=ADDRESS`'
: Specifies the source address used for creating sockets. In listen mode and tunnel mode this switch specifies the bound address, and it is generally a good idea not to specify this, which causes netcat to bind to a generic interface. In the connect mode, this switch is used to specify the source address for connecting to the outside

world. Again, if it's not specified a proper address for the destination route will be used.

'`-P NUM`'
'`--tunnel-port=NUM`'
: Same as –port, but affects only the connect phase (thus this option has no effect in listen mode). This switch is useful in tunnel mode for specifying the source port for the connecting socket.

'`-S ADDRESS`'
'`--tunnel-source=ADDRESS`'
: Same as –source, but affects only the connect phase (thus this has no effects in listen mode). This switch is useful in tunnel mode for specifying the source address for the connecting socket.

2.3 Advanced Options

'`-i SECS`'
'`--interval SECS`'
: sets the buffering output delay time. This affects all the current modes and makes the connection sock to buffer outgoing data. This means that in tunnel mode everything received from the listening socket is buffered for the connect socket.

'`-n`'
'`--dont-resolve`'
: Don't do DNS lookups on any of the specified addresses or hostnames, or names of port numbers from /etc/services.

'`-r`'
'`--randomize`'
: Randomizes the target remote ports ranges. If more than one range is specified it will randomize the ports in the whole global range.

'`-w`'
'`--wait=SECS`'
: Specifies the starting inactivity delay after which netcat will exit with an error status. In connect mode and in tunnel mode this specifies the timeout for the connecting socket, while in listen mode it specifies the time to wait for a VALID incoming connection (see listen mode).

'`-T`'
'`--telnet`'
: Answers the telnet codes as described in RFC0854. This makes possible to use netcat to script telnet sessions. The incoming telnet codes are parsed inside the receiving queue and are stripped off before forwarding the data as they were never received, so the application doesn't have to parse the codes itself (this behaviour can be disabled at compile time with –enable-oldtelnet or with –enable-compat).

'`-z`'
'`--zero`'
: Sets the zero I/O flag for the selected mode. In connect mode it means that as soon as the port is open it is immediately shutdown and closed. This may be useful for probing or scanning (even if there are faster portscanners out there, but this may be useful for scripting purposes). In listen mode, it makes netcat refusing all the incoming connections thus running in timeout (if set), or waiting forever. In both cases, no data is transfered.

  This option is incompatible with the tunnel mode.

3 The Connect Mode

In the connect mode, netcat connects to a remote host and simply links his stdin and stdout to the remote data stream.

Basic usage is:

netcat [options] hostname ports . . .

The specified ports indicates actually all ports to connect to. With normal operations netcat will sequentially (or randomly, if you specified the '-r' option) connect to all the specified ports and links stdin/stdout. No more than one port is used at once.

Particular options: In this mode, the '-p' option specifies the source port for connection, but it should never be specified unless you know what you are doing. The '-w' (timeout) option specifies how long to wait for the connection to succeed (if the remote host connects but doesn't send any data, the timeout DOESN'T apply).

4 The Listen Mode

In listen mode, netcat stays idle listening on a port, specified by the '-p' switch, until some remote host connects. At this point, the basic behaviour is the same of the connect mode.

Basic usage is:

netcat -l -p port [remote_hostname] [remote_ports] . . .

The remote hostname specifies which host is allowed to connect and from which ports. Usually these parameters are not specified, but if you want to sort out a special connection.

5 The Tunnel Mode

The tunnel mode ...

6 Examples

under writing ...

netcat [options] hostname port [port] . . .

netcat -l -p port [options] [hostname] [port] . . .

netcat -L hostname:port -p port [options] . . .