NAME

tcpspy − TCP/IP Connection Monitor

SYNOPSIS

tcpspy [**-dp**] [**-e** *rule*]... [**-f** *rulefile*]... [**-F** *facility*] [**-I** *interval*] [**-U** *user*] [**-G** *group*]

DESCRIPTION

tcpspy logs information about selected incoming and outgoing TCP/IP connections to syslog. The following information is logged: username, local address and port, remote address, port, and optionally the filename of the executable. At present, only the IPv4 protocol is supported.

Options

-e *'rule'*

Log only connections matching the specified rule. Rule syntax is outlined below. If this option is specified more than once, connections matching any of the specified rules are logged. You should quote the rule, as shown above.

-f *rulefile*

Read rules from *rulefile*. Each rule is on a new line. The '#' character may be used to add comments; everything from this character to the end of the line is ignored.

The **-e** and **-f** options may be used together.

-F *facility*

Log to syslog facility *facility* instead of the compile-time default setting. See the **syslog.conf**(5) manual page for a list of facilities.

-I *interval*

Update the internal state every *interval* milliseconds, instead of the default of 1000 ms. Connections that last less than *interval* milliseconds may be missed, so you should experiment to find a value small enough that it catches most connections, but not so small that it causes tcpspy to use too much CPU time.

-U *user* Switch to the specified user after startup. *user* may be a numeric user id or a user name from the system password file.

-G *group*

Switch to the specified group after startup. *group* may be a numeric group id or a group name from the system group file. If a username to switch to with the **-U** option is specified but **-G** is omitted, tcpspy will switch to that specified user's primary group.

-d Debugging mode; if this option is specified, tcpspy will not detach from the console after initialisation, and will log connections to standard output instead of syslog.

-p Log the filename of the executable that created/accepted the connection. You may require superuser privileges to obtain this information for processes you do not own (this is a kernel limitation).

This option can greatly increase the amount of CPU time required to process each connection/disconnection.

Rule Syntax

A rule may be specified with the **-e** option to log information about connections matching this rule, overriding the default of logging all connections.

The following comparison operations are defined:

user *uid*

True if the local user initiating or accepting the connection has the **effective** user id *uid*.

user *"username"*
Same as above, but using a username instead of a user id.

lport *port*
True if the local end of the connection has port number *port*.

lport *[low] - [high]*
True if the local end of the connection has a port number greater than or equal to *low* and less than or equal to *high*. If the form *low-* is used, high is assumed to be 65535. If the form *-high* is used, low is assumed to be 0. It is an error to omit both *low* and *high*.

lport *"service"*
Same as above, but using a service name from */etc/services* instead of a port number.

rport Same as **lport** but compares the port number of the remote end of the connection.

laddr *n.n.n.n[/m.m.m.m]*
Interpreted as a "net/mask" expression; true if "net" is equal to the bitwise AND of the local address of the connection and "mask". If no mask is specified, a default mask with all bits set (255.255.255.255) is used.

raddr Same as **laddr** but compares the remote address.

exe *"pattern"*
True if the full filename (including directory) of the executable that created/accepted the connection matches *pattern*, a **glob**(7)-style wildcard pattern.

The pattern "" (an empty string) matches connections created/accepted by processes whose executable filename is unknown.

If the **-p** option is not specified, a warning message will be printed, and the result of this comparison will always be true.

Expressions (including the comparisons listed above) may be joined together with the following logical operations:

expr1 **or** *expr2*
True if either of *expr1* or *expr2* are true (logical OR).

expr1 **and** *expr2*
True if both *expr1* and *expr2* are true (logical AND).

not *expr*
True if *expr* is false (logical NOT).

Rules are evaluated from left to right. Whitespace (space, tab and newline) characters are ignored between "words". Rules consisting of only whitespace match no connections, but do not cause an error. Parentheses, '(' and ')' may be placed around expressions to affect the order of evaluation.

The Examples section contains some sample rules which further demonstrate how they are constructed.

EXIT STATUS

0 The daemon was successfully started

>0 An error occurred

SIGNALS

TERM Shut down at most *interval* milliseconds from now.

INT (Debugging mode only) Handled identically to *TERM*.

All other signals retain their default behaviour, which is documented in **signal**(7).

EXAMPLES

tcpspy -e ’user "joe" and rport "ssh"’
: Log connections made by user "joe" for the service "ssh".

tcpspy -e ’not raddr 10.0.0.0/255.0.0.0 and rport 25 and (user "bob" or user "joe")’
: Log connections made by users "bob" and "joe" to remote port 25 on machines not on a fictional "intranet".

tcpspy -e ’exe "/usr/bin/irc"’
: Log connections made by /usr/bin/irc (probably ircII).

BUGS

Empty rule files cause **tcpspy** to log no connections instead of all connections.

AUTHOR

Tim J. Robbins <tim@robbins.dropbear.id.au>

SEE ALSO

glob(7), **proc**(5), **services**(5), **signal**(7), **syslog**(3), **syslog.conf**(5)