G. Pape
ipsvd

Name

udpsvd - UDP/IP service daemon

Synopsis

udpsvd [-hpvv] [-u *user]* [-l *name]* [-i *dir*|-x *cdb*] [-t *sec] host port prog*

Description

udpsvd creates an UDP/IP socket, binds it to the address *host:port*, and listens on the socket for incoming datagrams.

If a datagram is available on the socket, **udpsvd** conditionally starts a program, with standard input reading from the socket, and standard output redirected to standard error, to handle this, and possibly more datagrams. **udpsvd** does not start the program if another program that it has started before still is running. If the program exits, **udpsvd** again listens to the socket until a new datagram is available. If there are still datagrams available on the socket, the program is restarted immediately.

udpsvd optionally checks for special intructions depending on the IP address or hostname of the client sending the datagram which not yet was handled by a running program, see *ipsvd-instruct(5)* for details.

Attention:

UDP is a connectionless protocol. Most programs that handle user datagrams, such as *talkd(8)*, keep running after receiving a datagram, and process subsequent datagrams sent to the socket until a timeout is reached. **udpsvd** only checks special instructions for a datagram that causes a startup of the program; not if a program handling datagrams already is running. It doesn't make much sense to restrict access through special instructions when using such a program.

On the other hand, it makes perfectly sense with programs like *tftpd(8)*, that fork to establish a separate connection to the client when receiving the datagram. In general it's adequate to set up special instructions for programs that support being run by tcpwrapper.

Options

host
>   *host* either is a hostname, or a dotted-decimal IP address, or 0. If *host* is 0, **udpsvd** accepts datagrams to any local IP address.

port
>   **udpsvd** accepts datagrams to *host*:*port*. *port* may be a name from /etc/services or a number.

prog
>   *prog* consists of one or more arguments. **udpsvd** normally runs *prog* to handle a datagram, and possibly more, that is sent to the socket, if there is no program that was started before by **udpsvd** still running and handling datagrams.

-i *dir*
>   read instructions for handling new connections from the instructions directory *dir*. See *ipsvd-instruct(5)* for details.

-x *cdb*
>   read instructions for handling new connections from the constant database *cdb*. The constant database normally is created from an instructions directory by running *ipsvd-cdb(8)*.

-t *sec*

> timeout. This option only takes effect if the -i option is given. While checking the instructions directory, check the time of last access of the file that matches the clients address or hostname if any, discard and remove the file if it wasn't accessed within the last *sec* seconds; **udpsvd** does not discard or remove a file if the user's write permission is not set, for those files the timeout is disabled. Default is 0, which means that the timeout is disabled.

-l *name*

> local hostname. Do not look up the local hostname in DNS, but use *name* as hostname. By default **udpsvd** looks up the local hostname once at startup.

-u *[:]user[:group]*

> drop permissions. Set uid and gid to the *user*'s uid and gid, as found in */etc/passwd*, before running *prog*. If *user* is followed by a colon and a *group*, set the gid to *group*'s gid, as found in */etc/group*, instead of *user*'s gid. If *group* consists of a colon-separated list of group names, set the group ids of all listed groups. If *user* is prefixed with a colon, the *user* and all *group* arguments are interpreted as uid and gids respectively, and not looked up in the password or group file. All supplementary groups are removed.

-h

> Look up the client's hostname in DNS.

-p

> paranoid. After looking up the client's hostname in DNS, look up the IP addresses in DNS for that hostname, and forget the hostname if none of the addresses match the client's IP address. You should set this option if you use hostname based instructions. The -p option implies the -h option.

-v

> verbose. Print verbose messages to standard output.

-vv

> more verbose. Print more verbose messages to standard output.

See Also

ipsvd(7), *tcpsvd(8)*, *sslsvd(8)*, *ipsvd-instruct(5)*, *ipsvd-cdb(8)*

http://smarden.org/ipsvd/

Author

Gerrit Pape <pape@smarden.org>

Table of Contents