Rocky Enterprise Linux 9.2 Manual Pages on command 'sealert.8'

$ man sealert.8

sealert(8) sealert(8)

NAME

sealert - setroubleshoot client tool

SYNOPSIS

sealert [-b] [-f local_id] [-h] [-s] [-S] [-l id] [-a file] [-u] [-p]

[-P plugin_name]

DESCRIPTION

This manual page describes the sealert program.

sealert is the user interface component (either GUI or command line) to

the setroubleshoot system. setroubleshoot is used to diagnose SELinux

denials and attempts to provide user friendly explanations for a

SELinux denial (e.g. AVC) and recommendations for how one might adjust

the system to prevent the denial in the future.

In a standard configuration setroubleshoot is composed of two compo?

nents, setroubleshootd and sealert.

setroubleshootd is a system daemon which runs with root privileges and

listens for audit events emitted from the kernel related to SELinux.

The audit daemon must be running. The audit daemon sends a dbus mes?

sage to the setroubleshootd daemon when the system gets an SELinux AVC denial. The setroubleshootd daemon then runs a series of analysis plugins which examines the audit data related to the AVC. It records the results of the analysis and signals any clients which have attached to the setroubleshootd daemon that a new alert has been seen.

sealert can be run in either a GUI mode or a command line mode. In both instances sealert run as a user process with the privileges associated with the user. In GUI mode it attaches to a setroubleshootd server in? stance and listens for notifications of new alerts. When a new alert arrives it alerts the desktop user via a notification in the status icon area. The user may then click on the alert notification which will open an alert browser. In addition to the current alert sealert communicates with the setroubleshootd daemon to access all prior alerts stored in the setroubleshoot database.

The user may elect to tag any given alert as "ignore" in the browser which prevents any future notification for the given alert. This is useful when a user is already aware of a reoccurring problem.

sealert may also be run in command line mode. The two most useful com? mand line options are -l to "lookup" an alert ID and -a to "analyze" a log file. When setroubleshootd generates a new alert it assigns it a local ID and writes this as a syslog message. The -l lookup option may then be used to retrieve the alert from the setroubleshootd alert data? base and write it to stdout. This is most useful when setroubleshootd is being run on a headless system without the GUI desktop alert facil? ity. The -a analyze option is equivalent to the "Scan Logfile" command in the browser. The log file is scanned for audit messages, analysis is performed, alerts generated, and then written to stdout.

LOG FILE SCANNING

You may ask sealert to parse a file accumulating all the audit messages it finds in that file. As each audit event is recognized it is pre? sented for analysis which may generate an alert report if the analysis was successful. If the same type of event is seen multiple times re? sulting in the same report the results are coalesced into a single re?

port. The report count field will indicate the number of times the tool thought it saw the same issue. The report will also include a list of every line number on which it found an audit record which contributed to the coalesced report. This will allow you to coordinate the contents of the file with the analysis results if need be.

Log file scanning may be initiated from the sealert browser via the File::ScanLogFile menu or from the command line via 'sealert -a file? name'. Please note that sealert runs as a user level process with the permissions of the user running it. Many system log files are readable by root only. To work around this if you have root access one can copy the file as root to a temporary file and change its permissions. This is a good solution when scanning via the GUI as a normal user. Or you might consider su'ing to root and run the analysis via the command line (e.g. sealert -a filename).

The audit records in the log file must be valid syntactically correct audit messages or the parser will ignore them.

OPTIONS

-b --browser

Launch the browser

-f --fix

Execute the fix command for the avc with the given uuid and plugin, requires --plugin option.

-h --help

Show this message

-s --service

Start sealert service, Usually used by dbus.

-S --noservice

Start sealert without dbus service as stand alone app

-l --lookupid id

Lookup alert by id, if id is wildcard * then return all alerts

-a --analyze file

Scan a log file, analyze its AVCs

-u --user

logon as user

-p --password

set user password

-P --plugin

Set plugin name associated with the --fix option

AUTHOR

This man page was written by John Dennis <jdennis@redhat.com> and Dan Walsh <dwalsh@redhat.com>.

SEE ALSO

selinux(8),

20061121 sealert(8)