Full credit is given to the above companies including the OS that this PDF file was generated!

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

$ man semodule.8

SEMODULE(8) NSA SEMODULE(8)

NAME

semodule - Manage SELinux policy modules.

SYNOPSIS

semodule [option]... MODE...

DESCRIPTION

semodule is the tool used to manage SELinux policy modules, including installing, upgrading, listing and removing modules. semodule may also be used to force a rebuild of policy from the module store and/or to force a reload of policy without performing any other transaction. se? module acts on module packages created by semodule_package. Conven? tionally, these files have a .pp suffix (policy package), although this is not mandated in any way.

MODES

-R, --reload

force a reload of policy

-B, --build

force a rebuild of policy (also reloads unless -n is used)

--refresh

Like --build, but reuses existing linked policy if no changes to module files are detected (by comparing with checksum from the last transaction). One can use this instead of -B to ensure that any changes to the module store done by an external tool (e.g. a package manager) are applied, while automatically skip?ping the module re-linking if there are no module changes.

-D, --disable_dontaudit

Temporarily remove dontaudits from policy. Reverts whenever policy is rebuilt

-i,--install=MODULE_PKG

install/replace a module package

-u,--upgrade=MODULE_PKG

deprecated, alias for --install

-b,--base=MODULE_PKG

deprecated, alias for --install

-r,--remove=MODULE_NAME

remove existing module at desired priority (defaults to -X 400)

-l[KIND],--list-modules[=KIND]

display list of installed modules (other than base)

KIND:

standard

list highest priority, enabled, non-base modules

full list all modules

-X,--priority=PRIORITY

set priority for following operations (1-999)

-e,--enable=MODULE_NAME

enable module

-d,--disable=MODULE_NAME

disable module

-E,--extract=MODULE_PKG

Extract a module from the store as an HLL or CIL file to the current directory. A module is extracted as HLL by default. The name of the module written is <module-name>.<lang_ext>

OPTIONS

-s,--store

name of the store to operate on

-n,--noreload,-N

do not reload policy after commit

-h,--help

prints help message and quit

-P,--preserve_tunables

Preserve tunables in policy

-C,--ignore-module-cache

Recompile CIL modules built from HLL files

-p,--path

Use an alternate path for the policy root

-S,--store-path

Use an alternate path for the policy store root

-v,--verbose

be verbose

-c,--cil

Extract module as a CIL file. This only affects the --extract option and only modules listed in --extract after this option.

-H,--hll

Extract module as an HLL file. This only affects the --extract option and only modules listed in --extract after this option.

-m,--checksum

Add SHA256 checksum of modules to the list output.

EXAMPLE

```
# Install or replace a base policy package.

$ semodule -b base.pp

# Install or replace a non-base policy package.

$ semodule -i httpd.pp

# Install or replace all non-base modules in the current directory.

# This syntax can be used with -i/u/r/E, but no other option can be entered after the module names
```

```
$ semodule -i *.pp

# Install or replace all modules in the current directory.

$ ls *.pp | grep -Ev "base.pp|enableaudit.pp" | xargs /usr/sbin/semodule -b base.pp -i

# List non-base modules.

$ semodule -l

# List all modules including priorities

$ semodule -lfull

# Remove a module at priority 100

$ semodule -X 100 -r wireshark

# Turn on all AVC Messages for which SELinux currently is "dontaudit"ing.

$ semodule -DB

# Turn "dontaudit" rules back on.

$ semodule -B

# Disable a module (all instances of given module across priorities will be disabled).

$ semodule -d alsa

# Install a module at a specific priority.

$ semodule -X 100 -i alsa.pp

# List all modules.

$ semodule --list=full

# Set an alternate path for the policy root

$ semodule -B -p "/tmp"

# Set an alternate path for the policy store root

$ semodule -B -S "/tmp/var/lib/selinux"

# Write the HLL version of puppet and the CIL version of wireshark

# modules at priority 400 to the current working directory

$ semodule -X 400 --hll -E puppet --cil -E wireshark

# Check whether a module in "localmodule.pp" file is same as installed module "localmodule"

$ /usr/libexec/selinux/hll/pp localmodule.pp | sha256sum

$ semodule -l -m | grep localmodule
```

SEE ALSO

checkmodule(8), semodule_package(8)

AUTHORS

This manual page was written by Dan Walsh <dwalsh@redhat.com>.

The program was written by Karl MacMillan <kmacmillan@tresys.com>, Joshua Brindle <jbrindle@tresys.com>, Jason Tang <jtang@tresys.com>

Security Enhanced Linux Nov 2005 SEMODULE(8)