SELinux Mandatory access control

CONCEPTS

Never disable SELinux to make something work. Switch to permissive, reproduce, read the denial, then fix the label, boolean or port.

Process domain, e.g. httpd_t
↓ wants access to
Object file or port type
↓ checked by
Policy allow rules + booleans
↓ no rule
Denied AVC in audit.log

Reading a context

Part	Example	Matters for
user	`system_u`	confined users
role	`object_r`	rarely
type	`etc_t`	almost every rule
level	`s0`	MLS/MCS, containers

MODE AND STATUS

`getenforce`
Show the current mode

`sestatus`
Full status: mode, config, policy

`setenforce 0`
Switch to permissive until reboot

`setenforce 1`
Switch back to enforcing

`semanage permissive -a` *`<domain>`*
Make one domain permissive

`semanage permissive -d` *`<domain>`*
Enforce that domain again

`SELINUX=enforcing`
Boot mode in /etc/selinux/config

Modes

Mode	Effect
`enforcing`	denies and logs
`permissive`	logs only, allows
`disabled`	no checks, no labels kept

CONTEXTS

`ls -Z` *`<path>`*
Show file contexts

CONTEXTS (cont.)

`ps -eZ`
Show the domain of every process

mv keeps a file's old label; cp creates a new one from the target directory. Run restorecon after moving files into place.

FILE LABELS

`semanage fcontext -a -t` *`<type>`* **`'`***`<path>`***`(/.*)?'`**
Add a persistent label rule

`restorecon -Rv` *`<path>`*
Relabel files to match the rules

`restorecon -Rnv` *`<path>`*
Dry run: show what would change

`semanage fcontext -l -C`
List your local label rules

chcon or semanage?

Tool	After relabel
`chcon`	label lost: tests only
`semanage fcontext`	label kept: the real fix

BOOLEANS

`getsebool -a`
List all booleans and values

`setsebool -P` *`<bool>`* **`on`**
Set a boolean, persistently

`semanage boolean -l`
Booleans with descriptions

PORTS

`semanage port -l`
List port types

`semanage port -a -t` *`<type>`* **`-p tcp`** *`<port>`*
Allow a service on a port

DENIALS

`ausearch -m AVC -ts recent`
Show denials of the last 10 minutes

`ausearch -m AVC -ts recent | audit2why`
Explain why access was denied

`sealert -l "*"`
Analysis with suggested fixes