SELinux (Security Enhanced)

==========

What is SELinux?

SELinux is a kernel-level access control system. SELinux acts like a gatekeeper, enforcing rules about what users, programs, and services can access on a system. SELinux is a complex but effective security tool. While it might seem like overkill for some users, it offers a strong layer of defense for those who need to seriously tighten up system security.

SELinux Enforcement Modes

SELinux comes pre-installed on most new RHEL systems (most likely not enabled, or set into an inactive mode).

Check SELinux status

```
sestatus
```

SELinux has 3 modes:

enforcing	the strictest security setting. When enabled, SELinux actively enforces the security policies it has been configured with.
permissive	SELinux logs attempted violations of the security policy but doesn't block them. This can be useful for troubleshooting purposes or when initially configuring SELinux policies for new applications.
disabled	SELinux is disabled and it is not having any impact.

Changing SELinux mode

```
setenforce chosenmode
```

Check SELinux enforcement mode

```
getenforce
```

Access Levels

In SELinux, every process and system resource has a security label called a context. This context is like an ID card that defines the security properties of that process or resource. The SELinux policy uses these contexts along with a set of rules to dictate how processes can interact with each other and access system resources.

Here's a breakdown of the key aspects of access levels for processes in SELinux:

- **SELinux Context:** This context contains multiple fields, including user, role, type, and a security level.
  - **SELinux Type:** This is a crucial part of the context, often ending in "_t". For instance, a web server process might have a type of "httpd_t". SELinux policy rules primarily rely on these types to define allowed interactions between processes and resources.

- **SELinux Policy Rules:** These rules define what a process with a certain type is allowed to do with other processes and resources based on their types. By default, all interaction is denied unless a rule explicitly grants permission. DAC (Discretionary Access Controls - traditional file permission/ownership) rules are checked first, and SELinux rules only come into play if DAC allows access.

Check context of a process:

```
ps axfuZ | grep -i processname
```

Show context of a file

```
ls -lZ
```

Changing context of a file

```
chcon --type=servicetype_t /path/to/change
```

Ports

List all ports being monitored by SELinux

```
semanage port -l
```

Change port management

```
semanage -a -t portname_t -p TCP portnumber
```

example:

```
semanage -a -t http_port_t -p TCP 8080
```

In this context, we're wanting to enable Apache to access port 8080. Apache has a context specifically for setting port access. So this command is adding http_port_t to the allow configuration on port 8080.

-a	add
-d	delete
-t	define SELinux type
-p	protocol ie TCP or udp

Logging

SELinux logs all activity that it detects into the audit log (`/var/log/audit/audit.log`) when in enforcing or permissive mode.

Revision #6
Created 2024-06-12 22:30:30 UTC by Daniel
Updated 2024-06-24 21:10:54 UTC by Daniel