Contents

[**SELinux- Security Enhanced Linux** 2](#_Toc476475759)

[**Need of SELinux:** 2](#_Toc476475760)

[**SELinux Modes:** 2](#_Toc476475761)

[**Features of SELinux:** 4](#_Toc476475762)

[**SELinux Policy:** 4](#_Toc476475763)

[ **Users: (SELinux User)** 4](#_Toc476475764)

[ **Roles:** 4](#_Toc476475765)

[ **Subject:** 4](#_Toc476475766)

[ **Object:** 4](#_Toc476475767)

[ **Domain:** 4](#_Toc476475768)

[ **Types of Objects:** 4](#_Toc476475769)

[ **Type Enforcement (TE):** 5](#_Toc476475770)

[ **Programs & Process:** 5](#_Toc476475771)

[ **Graphical Representation:** 6](#_Toc476475772)

[**Difference between Standard Linux & SELinux:** 6](#_Toc476475773)

[**SELinux Context:** 7](#_Toc476475774)

[**Commands for modifying context:** 8](#_Toc476475775)

[ **chcon (change context):** 8](#_Toc476475776)

[ **restorecon (restore context):** 8](#_Toc476475777)

[ **semanage fcontext:** 8](#_Toc476475778)

[ **matchpathcon:** 8](#_Toc476475779)

[**Files keep context of everything in Linux:** 8](#_Toc476475780)

# **SELinux- Security Enhanced Linux**

* It is an advanced access control mechanism that implements MAC (Mandatory Access Control).
* In traditional access control scheme, DAC (Discretionary [you get what you choose] Access Control) was used.
* With SElinux, a process/application has only access rights it needs to function and nothing more.

## **Need of SELinux:**

* In DAC, user can set the permissions (rwx) of the files without any restrictions. So according to permissions set, user owner, group owner and others can access the files/directory. Even user owner change the user of file as well. DAC leaves too much access control in the hands of end users. In such cases, system administrator can’t restrict certain users from accessing some certain processes.
* With MAC, system administrator restricts users to access certain processes only. For an example, System admin allows users to run some scripts, to view some log files but restricts to use sudo or su commands, to restrict scripts to be run from their home directory.
* With MAC, even a user opens files/directories to world access, it doesn’t mean everyone can access files/directory. MAC constrains access what users can do. For an example, users can see files/directories and those are visible for everyone. But system admin constrains users to read those files/directories.

## **SELinux Modes:**

1. Enforcing : SELinux is operating
2. Permissive : SELinux is active but only displaying warnings
3. Disabled : SELinux is turned off entirely
4. Enforcing:

* SELinux enforce (make something happen forcefully) its policy (set of rules) on the system.
* Any unauthorized access attempt by users/processes are denied.
* Access denials are also written to relevant log files.

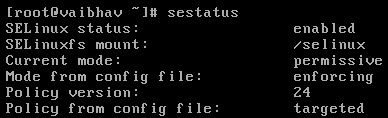
1. Permissive:

* SELinux doesn’t apply its policy.
* No access is denied.
* Policy violation is still written to relevant log files.

1. Disabled:

* SELinux is entirely turned off. No policy is applied and no enhanced security is running.
* No access is denied, But it does not provide any information on what would have been denied.
* To see mode of SELinux.

Command: sestatus OR getenforce





* To change SELinux mode either enforcing or permissive temporary,

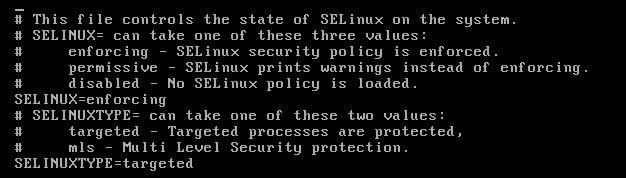
Command: setenforce 1: for setting enforcing mode

Command: setenforce 0: for setting permissive mode

Note: We can’t use setenforce command when SELinux in disabled mode.

* To change SELinux modes(any) permanently,

Command: vi /etc/selinux/config [preferable] OR vi /etc/sysconfig/selinux



* Change the value of SELINUX.
* To reflect changes made, restart of system is must. Without restarting the system, SELinux will not change permanently.
* To restart the system,

Command: reboot

* Setting the status to permissive first is necessary because every file in the system needs to have its context labelled before SELinux can be enforced.

## **Features of SELinux:**

* It allows applications to query the policy.
* It allows to change policy without having to reboot (in order to activate the changes) the system.
* It allows to write flexible policies according to user’s need.
* It has control over the files, directories, file descriptors, sockets (TCP/UDP), messaging interfaces and network interfaces.

## **SELinux Policy:**

It is a set of rules that define the security and access rights for the system.

### **Users: (SELinux User)**

* SELinux has a set of pre-built users. It has identifier to system administrator that can be used to limit roles can be used.
* The purpose of SELinux user is to avoid problem occurred in DAC (end users can easily change security contents).
* Every Linux user account is mapped to one or more SELinux users by SELinux policy.
* SELinux user can be linked to one or multiple roles.
* To list out SElinux users,

Command: semanage user -l

* To list out SELinux login users,

Command: semanage login –l

### **Roles:**

* Roles define privileges of a SELinux user.
* Role defines which users can access which processes.

### **Subject:**

* Subject is a process that can affect an object.

### **Object:**

* Object is anything that can be acted upon.
* Object can be files, directories, ports, sockets and so on.

### **Domain:**

* It is a context within which process (subject) is supposed to run.
* It is just like a boundary around the process that confines it.
* It tells the process what it can do and what it cannot do.
* It ensures that each process on a domain can run certain types of files and nothing more.
* It is security context of subject (process).

### **Types of Objects:**

* Type of object defines the purpose of the object.
* It is security context of object (files/directories).
* It is just like a group of files having similar characteristics or of having same origin.

### **Type Enforcement (TE):**

* Process running within a particular domain can perform certain operations on certain types of objects is called type enforcement.

### **Programs & Process:**

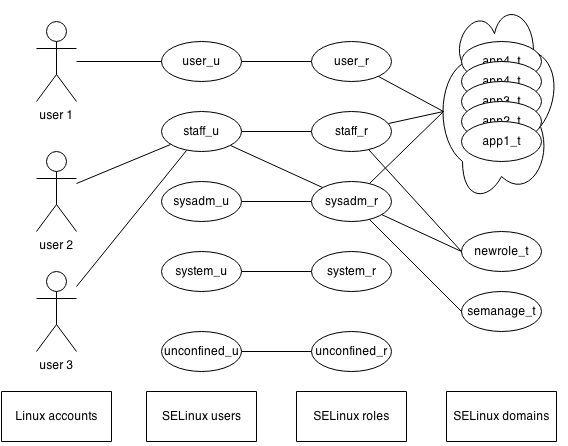
#### **Program:**

* Program is an executable file containing set of instructions written to perform predefined task.
* E.g., notepad.exe is a program (executable file) which allows to read/edit/print the text file.
* Programs are not stored on the primary memory (RAM), they are stored on secondary memory (hard disks). Program is read into primary memory and executed by kernel and so it is referred as passive entry.

#### **Process:**

* Process is an executing instance of a program.
* E.g., when you double click on notepad icon, a process is started that will run the notepad program.
* Multiple processes can be run. For an example, multiple notepad window/instances can be opened and each instance is referred as process.
* A process is referred as active entry as it resides on the primary memory and leave the memory if the system is booted.

### **Graphical Representation:**



* Each Linux user account is mapped to one or multiple SELinux users that limits the roles to be used.
* SElinux user can be linked to one or multiple roles that defines privileges of SElinux user.
* According to domain, SElinux user is restricted to access only certain type of files.

# **Difference between Standard Linux & SELinux:**

|  |  |  |
| --- | --- | --- |
|  | **Standard Linux** | **SELinux** |
| Process(Subject) security attributes | Real user and group IDs | Security Context |
| File(Object) security attributes | Access modes and file user & group IDs | Security Context |
| Basis for access control | Process user/group ID and file’s access modes based on file’s user/group ID | Permission allowed between process type and file type |

## **SELinux Context:**

* SELinux context is a collection of security related information that helps SELinux to make access control decisions.
* Everything in Linux has security context.
* To display security context of the shell,

Command: id -Z



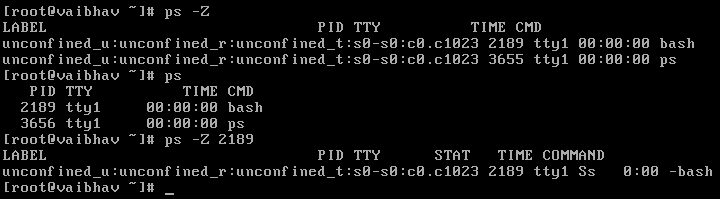
* To display security context of the object(files/directories),

Command: ls -Z <file name>



* To display security context of the subject(process),

Command: ps -Z OR ps -Z <PID>



* **Label:**
* Label is like a file that contains security contexts.
* In Linux, every entity is attached to label.
* A file is said to be have labelled with security context if security information is available with file.
* Security context is divided into four parts and each of them are separated by comma (:).

It is generally in following format:

*user****:****role****:****type/domain****:****sensitivity*

1. SElinux user
2. Roles
3. type(for objects)/domain(for subjects)
4. sensitivity of the resource

* **Naming conventions:**

SELinux users are suffixed by “\_u”, roles are suffixed by “\_r” and types/domains are suffixed by “\_t”.

## **Commands for modifying context:**

### **chcon (change context):**

* This command allows to change security context of files.
* This command changes the context temporarily.

### **restorecon (restore context):**

* This command allows to restore security context.
* This command will revert file context back to original file context.

### **semanage fcontext:**

* This command will write new context to the one of those two files that maintain context of each and every file/directory.
* With the help of this command, file relabeling can also be done.

### **matchpathcon:**

* This command looks at the current context of a resource and compare it with those 2 files.
* If contexts are different then it will suggest the changes required.

## **Files keep context of everything in Linux:**

1. /etc/selinux/targeted/contexts/files/file\_contexts
2. /etc/selinux/targeted/contexts/files/file\_contexts.local

Additional

SELinux-kernel level security

Firewalld/iptables- port level security

sestatus: mode from config file: permanent mode; current mode: current mode