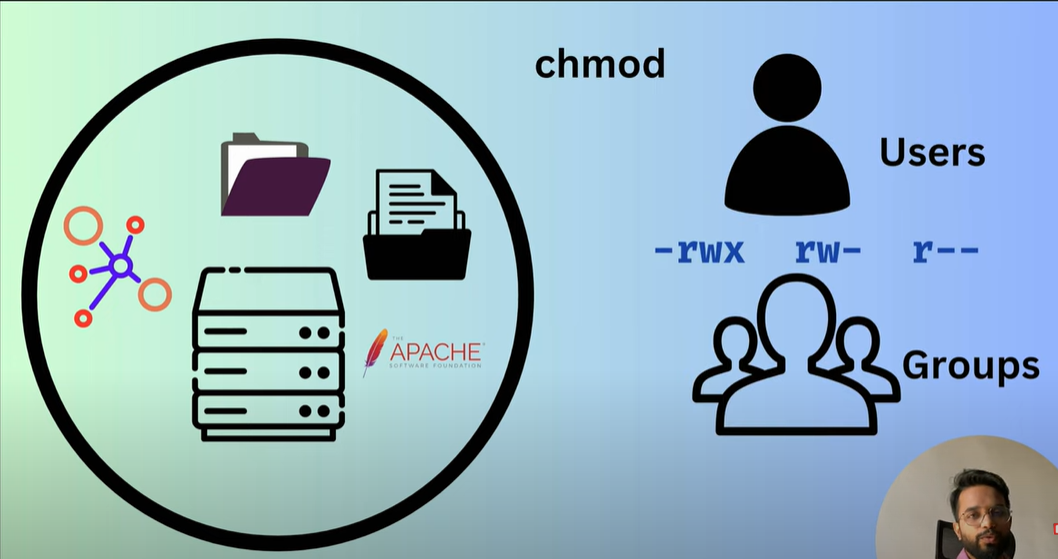
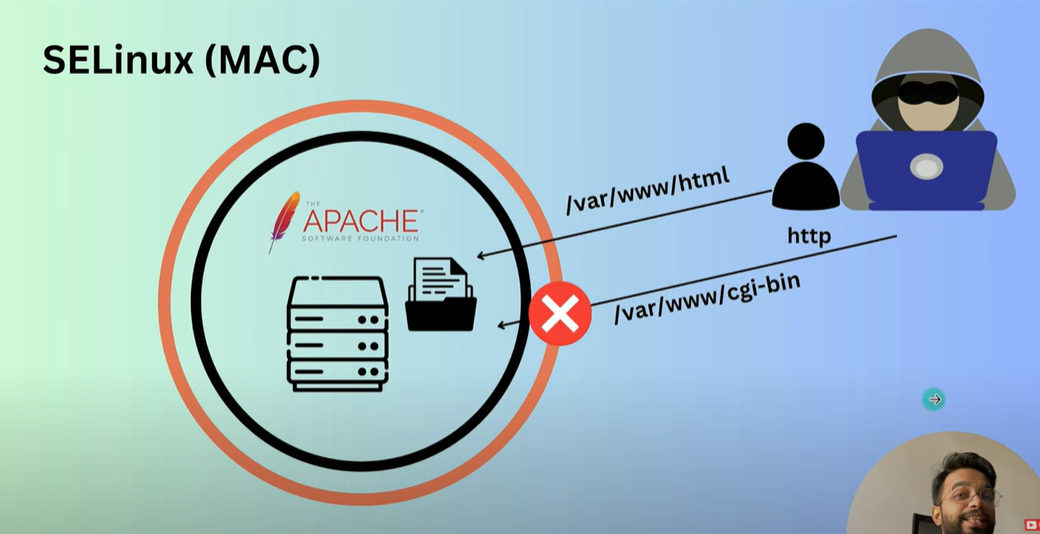
**SELinux in Linux [HINDI] | MPrashant**

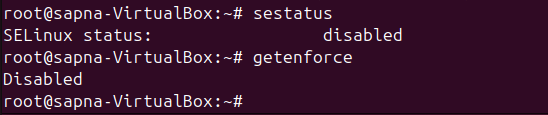
* SELinux – Security enhanced linux
* SELinux is used to enhance the security of linux machine
* SELinux is a linux kernel security module or framework
* SELinux provides additional layer of access control and mandatory access control(MAC) to enhance the security
* It prevents unauthorized access
* It also limits the impact of vulnerabilities
* SELinux was developed as a project of US NSA(National security agency) with SELinux community in 2003
* **DAC vs MAC**
* **DAC -**



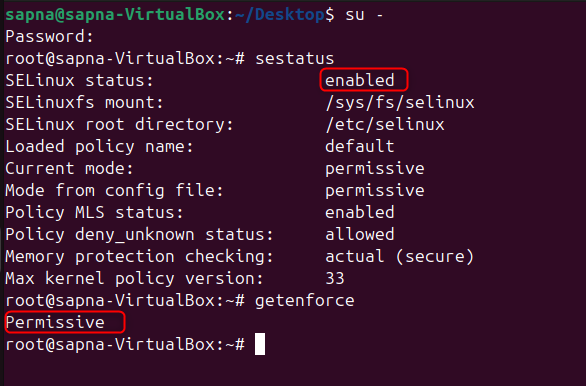
* Say we have a simple linux server with networking, files and folders, web server
* Now when we talk about security of this linux server, we usually do it via permissions for the files and folders using chmod command
* This type of permission based security is referred as DAC(Discretionary access control)
* Discretionary means something is up to the user to decide or choose
* **SELinux (MAC)**



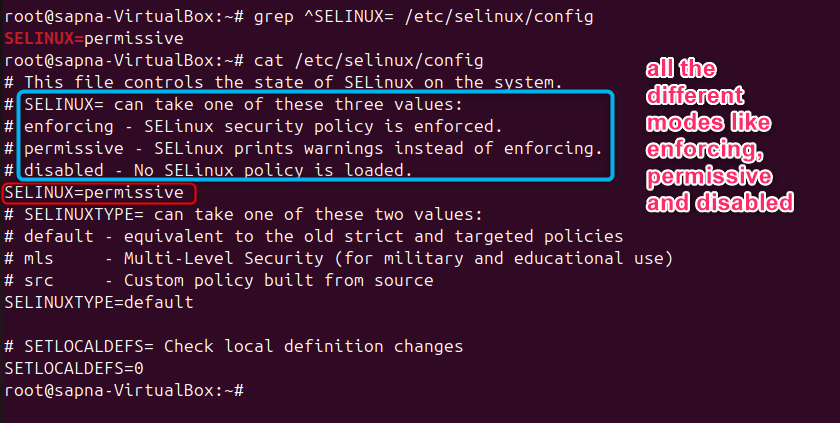
* Here the black circle is the normal DAC (meaning permission bases security)
* Then above that we have an orange circle which is the MAC (Mandatory access control) security
* Say our linux server has web server like apache installed
* Since web server is installed we can access the linux server via internet
* Say a user wants to access this web server
* Now the problem is the linux server is compromised like a hacker has hacked our web server
* Now the hacker has the control for the web server http
* Now the hacker also wants to access other folders like var, logs, sys, boot
* But the hacker is not able to access any other files and folders other than http
* Now this could be due to permission issues that the hacker is not able to access any other files and folders
* But in our scenario the hacker is able to access the /var/www/html folder but not the /var/www/cgi-bin folder
* This is weird because both html and cgi-bin are inside the www folder
* So this is clearly because of the MAC security and not DAC
* **SELinux Options –** 
  + **Enforcing (**by default enabled in redhat)
  + **Permissive**(disabled but logs the activity)
  + **Disable**(Stop SELinux)
* Command to check if SELinux status – sestatus or getenforce



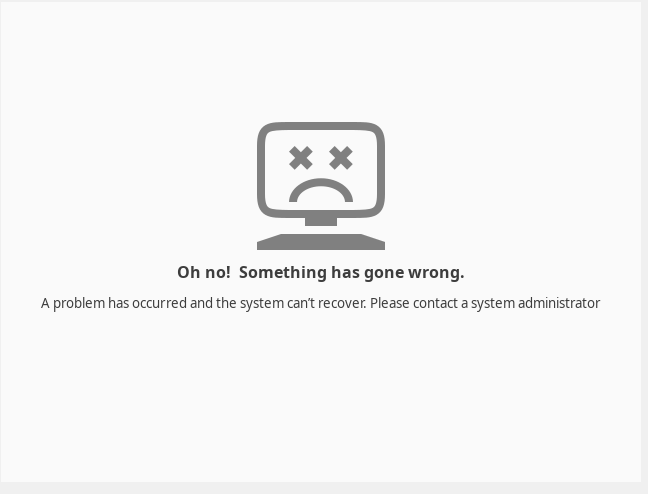
* To run sestatus command make sure policycoreutils package is installed using command – apt install policycoreutils
* To change the SElinux setting from disabled to enabled run – setenforce 1
* To disable it again – setenforce 0
* This doesn’t work on ubuntu machine
* To make sure SELinux works on ubuntu follow the below steps –(Do as root user)
  + apt update
  + apt install selinux-basics selinux-policy-default auditd policycoreutils -y
  + selinux-activate
  + reboot
  + su – (login as root again after reboot)
  + getenforce (o/p should be enforcing or permissive)
  + sestatus



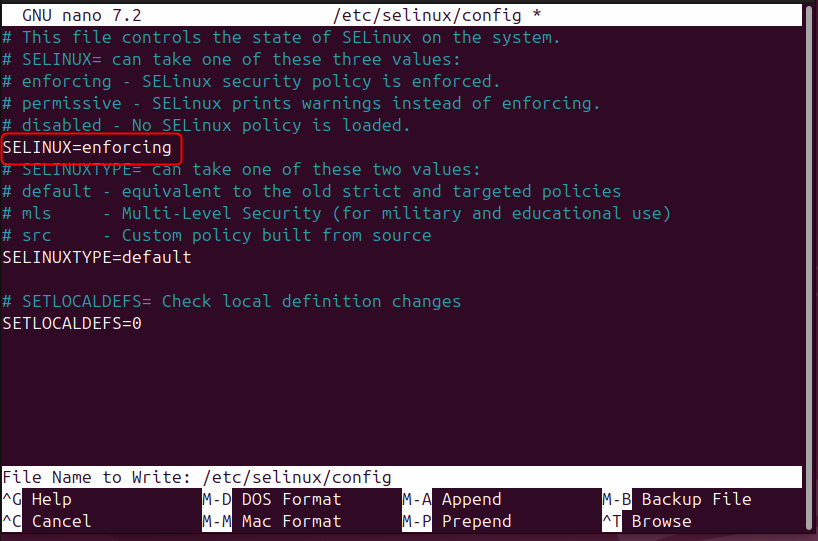
* currently when we run selinux-activate command the SELinux setting is permanent
* and this command ensures the kernel boots with SELinux support everytime
* and the selinux-activate command also writes the setting into /etc/selinux/config file to set the default mode (SELinux to permissive)
* so every time kernel boots it checks the default mode and sets it
* to confirm we can follow the below steps –
* run command – grep ^SELINUX= /etc/selinux/config
* o/p should be permissive



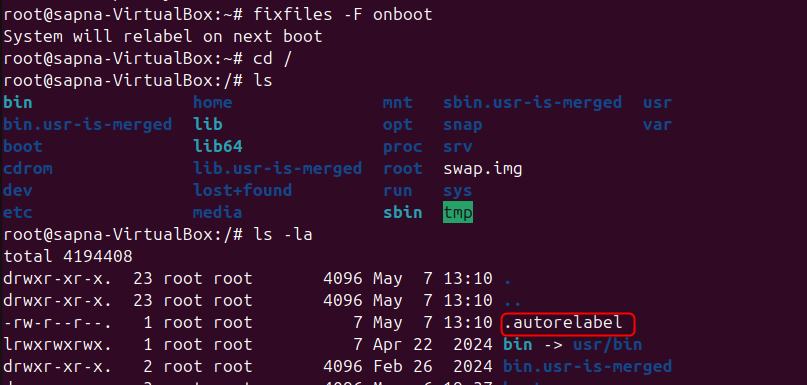
* if you want to change the type of mode to enforcing or disabled you can directly do it in this file – vi /etc/selinux/config file
* now say I want to move from permissive mode to enforcing mode, these are the steps I need to follow –
  + setenforce 1(after this command my system crashed)



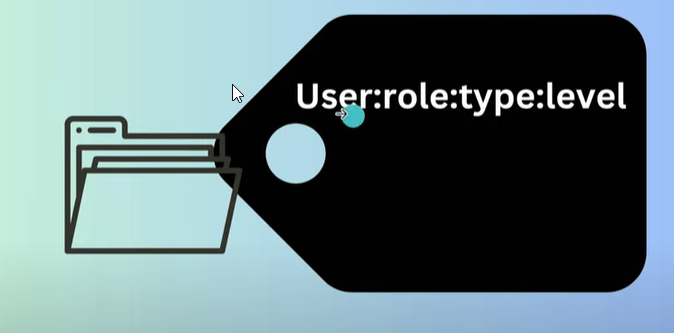
* + command - nano /etc/selinux/config
  + Change config file **SELINUX=enforcing**



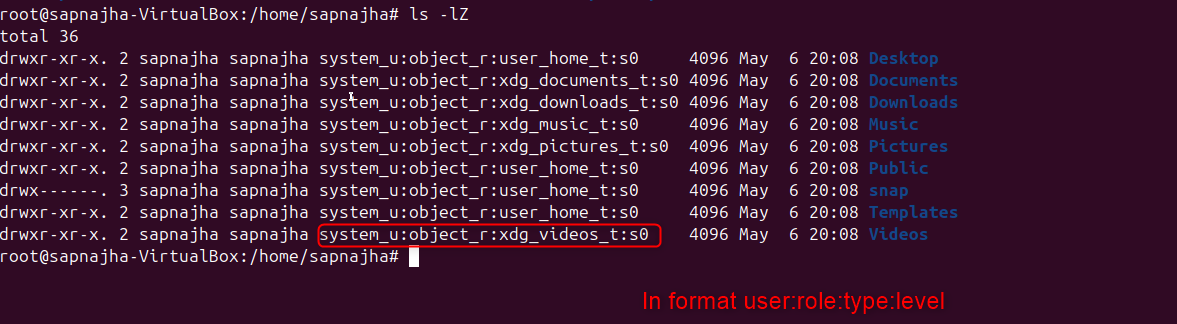
* + Ask the system to relabel files on next boot - touch /.autorelabel
  + Then reboot the system – reboot
  + Again system crashed
  + **Another method** – after the mode is permissive run command – **fixfiles -F onboot**
  + To prevent incorrectly labeled and unlabeled files from causing problems, SELinux automatically relabels file systems when changing from disabled state to permissive or enforcing mode
  + Before booting the system for relabeling, make sure the system will boot in permissive mode for example using the enforcing=0 kernel option
  + Didn’t try this since it could crash the system
  + Steps are first run fixfiles -F onboot
  + This will create a ./autorelabel file in the root directory
  + Then reboot the machine
  + Right in mode would be permissive and selinux will do the relabeling
  + Then change the mode to enforcing and again reboot
  + Lets try one last time –
  + ran fixfiles -F onboot

****

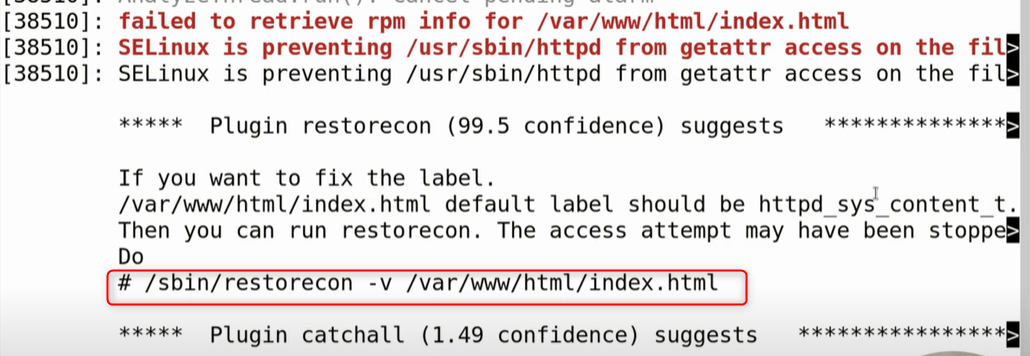
* + then reboot the machine
  + ran command - setenforce 1
  + and again system crashed
* Two main concepts of SELinux –
  + Labeling – In the hacker example we saw that the hacker was not allowed to enter the club because he didn’t have a tag. When we use SELinux, all the files folders and directories inside our machine will be labeled, and this label will define what all access this file/folder or process will have.
    - Label is usually in format – User:role:type:level



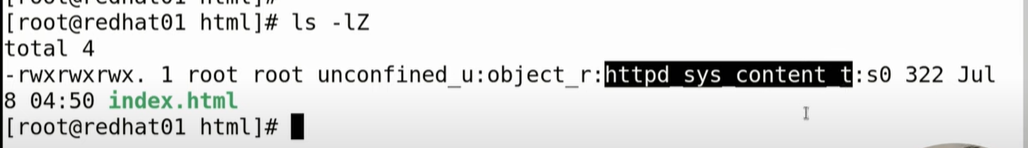
* + - To check the label of a file/directory – ls -Lz



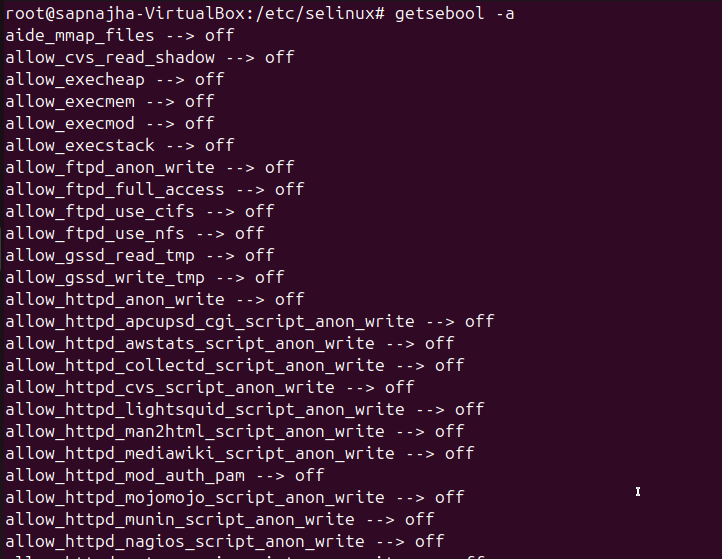
* + - To check the label of a process – ps axZ | grep httpd
    - To check label of a socket – netstat -taZ | grep httpd
  + Type enforcement
* Example – How a label works
* Lets go to location cd /var/www/html folder to find index.html(Content in Tutorial60)
* Just install apache2 service and then start apache2 service, you will see www folder then
* If apache is not installed www folder wont come
* Now we already have index.html, in the video he changed a few content and then in firefox ran <ip>:80
* And the new content that is present in index.html will be printed on the web page
* Now the root user creates another file in /test directory called index.html and changes a bit of content
* Now he moved the index.html from /test folder to /var/www/html folder basically overwrite the old file
* Now when he runs the ip again on firefox, he gets the previous index.html the one which my default we get on red hat or apache that one
* Now he tries to give all permissions using chmod 777 to index.html but still the file doesn’t change
* The issue comes because of SELinux
* To check errors related to SELinux run – journalctl -b 0
* In red we can see the SELinux errors



* Here we can see the error saying SELinux is preventing to access the file
* And below we also have the solution on how to fix it
* It says /var/www/html/index.html default label should be httpd\_sys\_content\_t
* It also gives us the command we should run - /sbin/restorecon -v /var/www/html/index.html
* Now when we run ls -Lz the label is now changed



* Now when we run the ip on the browser again we can see the new modified data also being printed
* To change type in a label – chcon -t <type> FILENAME
* chcon means change context
* type is nothing but context
* where is the context stored - /etc/selinux/targeted/contexts/files/file\_contexts
* Boolean – just by setting some pre-defined properties to either ON or OFF
* Example – Ftp server to access home dir
* To check the Boolean – getsebool -a or semanage boolean -l



* To set a Boolean – setsebool -P boolean\_name on/off(-P means Permanent)

