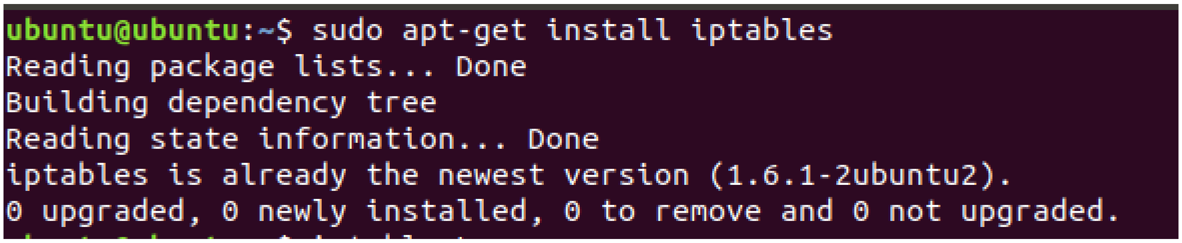
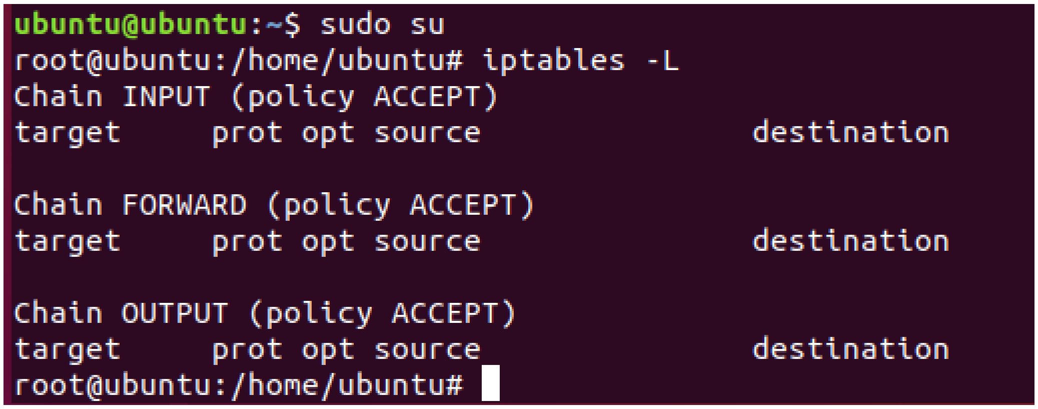
10 hardening tasks:

1. Firewall configuration

In this case we will be using the iptables tool to block and enable critical traffic with the help of how-to-configure-firewall-in-linux

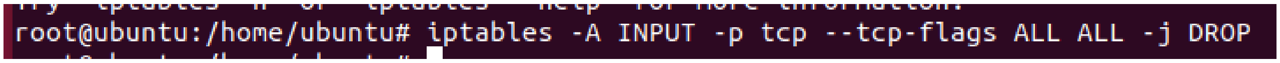


```
ubuntu@ubuntu:~$ sudo apt-get install iptables
Reading package lists... Done
Building dependency tree
Reading state information... Done
iptables is already the newest version (1.6.1-2ubuntu2).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```



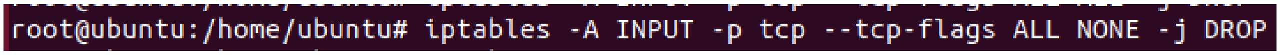
```
ubuntu@ubuntu:~$ sudo su
root@ubuntu:/home/ubuntu# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
root@ubuntu:/home/ubuntu#
```

^listing out all current rules.



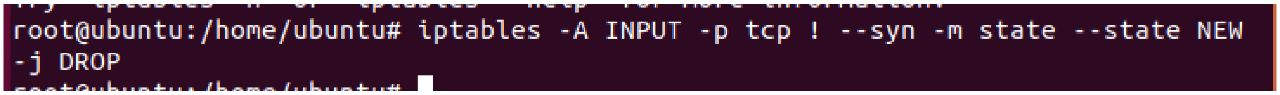
```
root@ubuntu:/home/ubuntu# iptables -A INPUT -p tcp --tcp-flags ALL ALL -j DROP
```

^XMAS are malformed packets commonly used by attackers. This should always be blocked as users commonly love to send crafted packets over the internet to cause havoc such as crashing the router, flooding the network, and denying actual services running on the network.



```
root@ubuntu:/home/ubuntu# iptables -A INPUT -p tcp --tcp-flags ALL NONE -j DROP
```

^blocking all empty flags sent to the host as attackers usually briefly send empty packets without much effort in order to find out more information on the network inside. As well as how attackers can send empty packets to look into how the OS responds which is an attack known as OS fingerprinting.

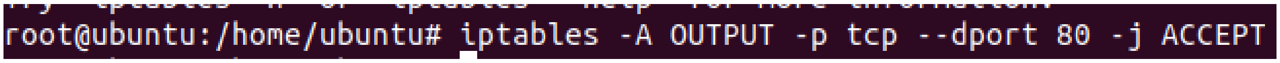


```
root@ubuntu:/home/ubuntu# iptables -A INPUT -p tcp ! --syn -m state --state NEW
-j DROP
```

^blocking a common distributed denial of service attack called a SYN flood where it over saturates the network with attempts to communicate to the host

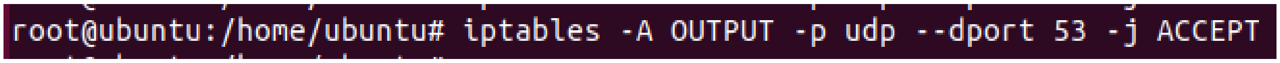
2. Network Surface Attack Area Reduction (Minimum Network Privileges through Unnecessary Port Blocking)

Only allow the ports required to do the task to minimize surface attack area.



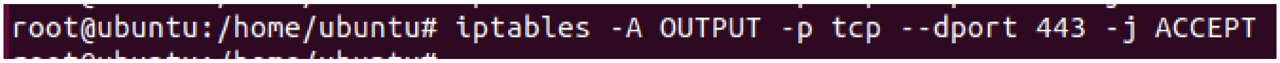
```
root@ubuntu:/home/ubuntu# iptables -A OUTPUT -p tcp --dport 80 -j ACCEPT
```

^allow outgoing http requests to be able to connect to the internet.



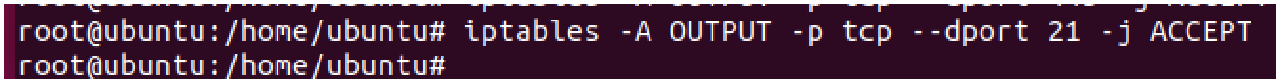
```
root@ubuntu:/home/ubuntu# iptables -A OUTPUT -p udp --dport 53 -j ACCEPT
```

^only allow outgoing requests to DNS servers to be able to assign IP addresses of the domain name.



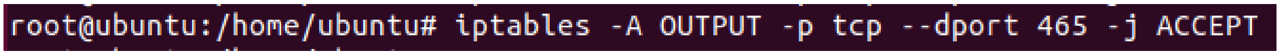
```
root@ubuntu:/home/ubuntu# iptables -A OUTPUT -p tcp --dport 443 -j ACCEPT
```

^ allow connections through a secure version of http.



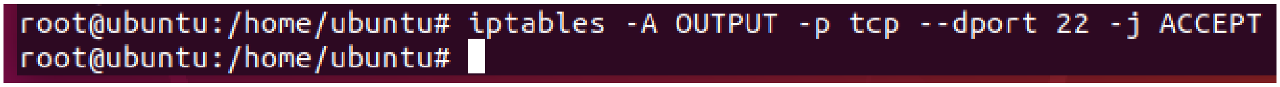
```
root@ubuntu:/home/ubuntu# iptables -A OUTPUT -p tcp --dport 21 -j ACCEPT
root@ubuntu:/home/ubuntu#
```

^allow connections to move files across for easier sharing. NOTE: block if you are not planning on sending files across the network.



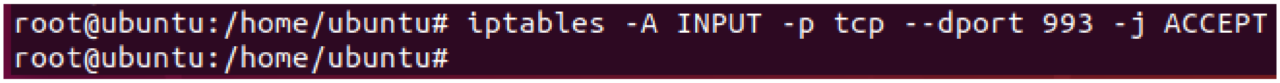
```
root@ubuntu:/home/ubuntu# iptables -A OUTPUT -p tcp --dport 465 -j ACCEPT
```

^to allow for sending emails for proper communication to coworkers.



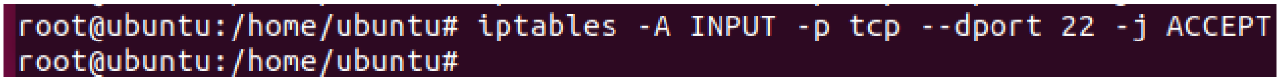
```
root@ubuntu:/home/ubuntu# iptables -A OUTPUT -p tcp --dport 22 -j ACCEPT
root@ubuntu:/home/ubuntu#
```

incoming connections:



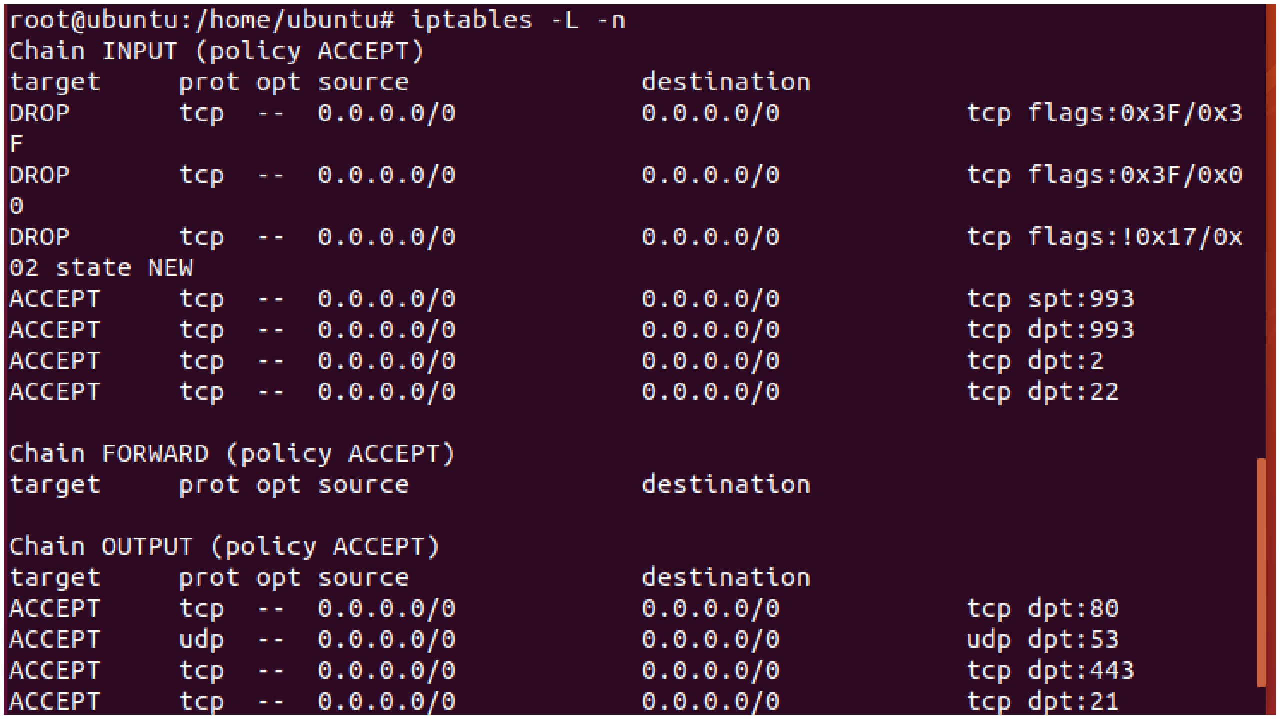
```
root@ubuntu:/home/ubuntu# iptables -A INPUT -p tcp --dport 993 -j ACCEPT
root@ubuntu:/home/ubuntu#
```

^to receive emails for communication purposes

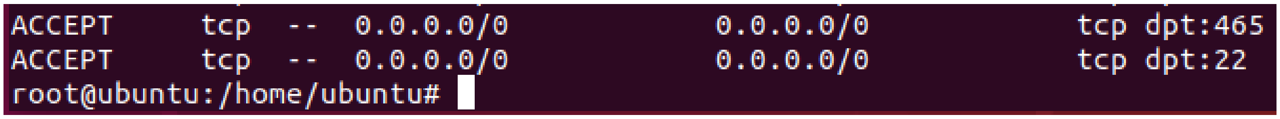


```
root@ubuntu:/home/ubuntu# iptables -A INPUT -p tcp --dport 22 -j ACCEPT
root@ubuntu:/home/ubuntu#
```

to allow secure connections to the pc itself.

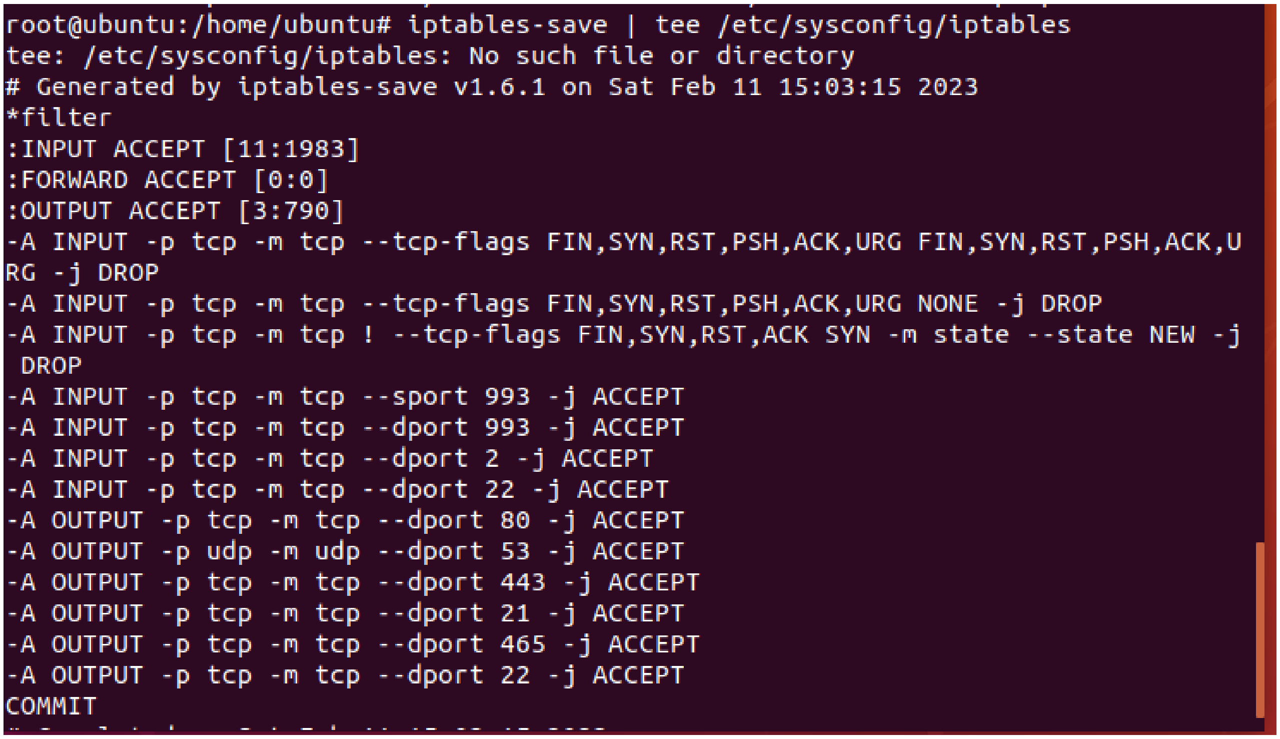


```
root@ubuntu:/home/ubuntu# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP       tcp  --  0.0.0.0/0            0.0.0.0/0            tcp flags:0x3F/0x3
F
DROP       tcp  --  0.0.0.0/0            0.0.0.0/0            tcp flags:0x3F/0x0
0
DROP       tcp  --  0.0.0.0/0            0.0.0.0/0            tcp flags:!0x17/0x
02 state NEW
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp spt:993
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:993
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:2
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:22

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:443
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:21
```



```
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:465
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:22
root@ubuntu:/home/ubuntu# 
```

^current list of the connections allowed in order to reduce network attack surface to not hinder work flow.

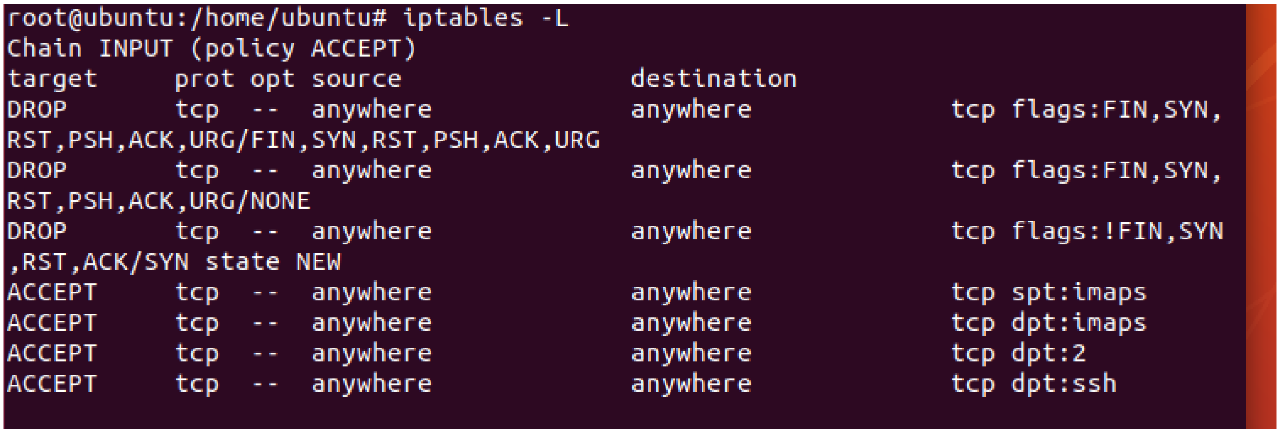


```
root@ubuntu:/home/ubuntu# iptables-save | tee /etc/sysconfig/iptables
tee: /etc/sysconfig/iptables: No such file or directory
# Generated by iptables-save v1.6.1 on Sat Feb 11 15:03:15 2023
*filter
:INPUT ACCEPT [11:1983]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [3:790]
-A INPUT -p tcp -m tcp --tcp-flags FIN,SYN,RST,PSH,ACK,URG FIN,SYN,RST,PSH,ACK,U
RG -j DROP
-A INPUT -p tcp -m tcp --tcp-flags FIN,SYN,RST,PSH,ACK,URG NONE -j DROP
-A INPUT -p tcp -m tcp ! --tcp-flags FIN,SYN,RST,ACK SYN -m state --state NEW -j
 DROP
-A INPUT -p tcp -m tcp --sport 993 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 993 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 2 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 22 -j ACCEPT
-A OUTPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A OUTPUT -p udp -m udp --dport 53 -j ACCEPT
-A OUTPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A OUTPUT -p tcp -m tcp --dport 21 -j ACCEPT
-A OUTPUT -p tcp -m tcp --dport 465 -j ACCEPT
-A OUTPUT -p tcp -m tcp --dport 22 -j ACCEPT
COMMIT
```

…

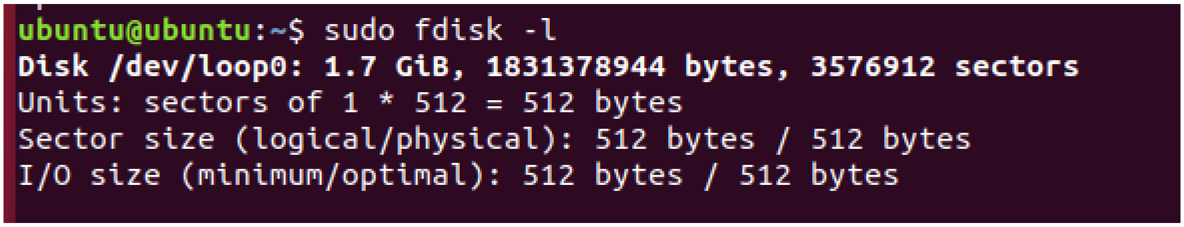
^saving the configuration

NOTE: Since rules are up and that there are no definitive way of testing it without doing anything malicious on a suspicious port to another computer, with the fact that all the malicious ports and activities associated to accomplished the tasks for testing are blocked, we can assume that the firewall is up and running as intended when the rules were shown such as below.



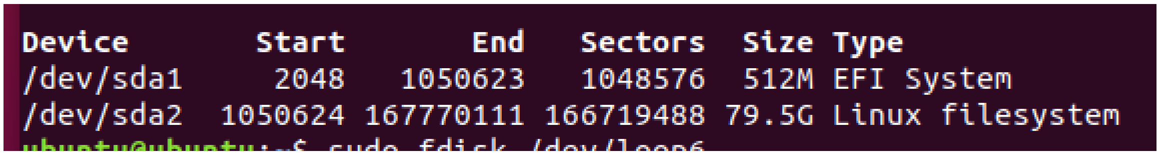
```
root@ubuntu:/home/ubuntu# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP       tcp  --  anywhere             anywhere             tcp flags:FIN,SYN,
RST,PSH,ACK,URG/FIN,SYN,RST,PSH,ACK,URG
DROP       tcp  --  anywhere             anywhere             tcp flags:FIN,SYN,
RST,PSH,ACK,URG/NONE
DROP       tcp  --  anywhere             anywhere             tcp flags:!FIN,SYN
,RST,ACK/SYN state NEW
ACCEPT     tcp  --  anywhere             anywhere             tcp spt:imaps
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:imaps
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:2
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:ssh
```

3. Hard Disk Encryption,



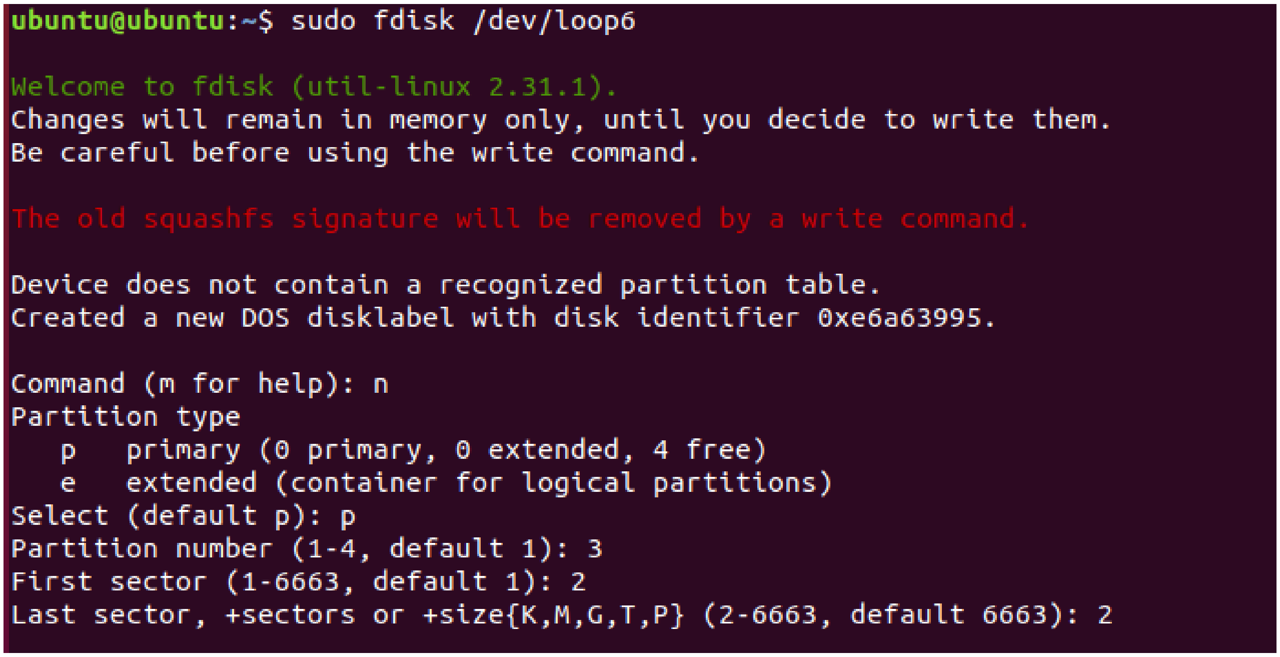
```
ubuntu@ubuntu:~$ sudo fdisk -l
Disk /dev/loop0: 1.7 GiB, 1831378944 bytes, 3576912 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

^creating a disk partition that will be encrypted so its contents will not be seen by any unauthorized users



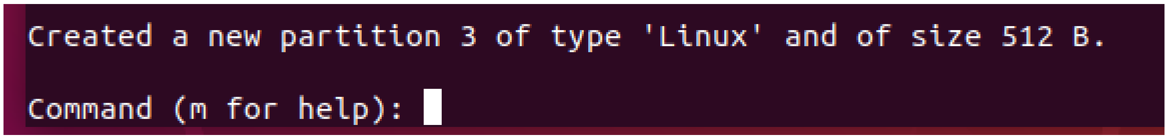
```
Device         Start       End   Sectors  Size Type
/dev/sda1       2048   1050623   1048576  512M EFI System
/dev/sda2    1050624 167770111 166719488 79.5G Linux filesystem
ubuntu@ubuntu:~$ sudo fdisk /dev/loop6
```

^showcases the list of current partitions.



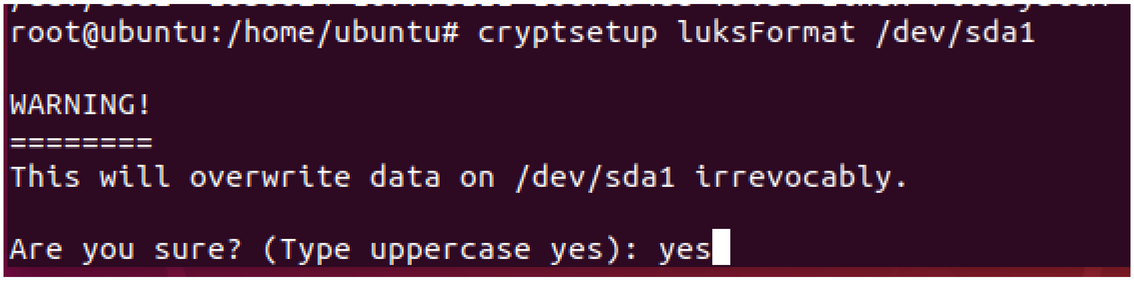
```
ubuntu@ubuntu:~$ sudo fdisk /dev/loop6

Welcome to fdisk (util-linux 2.31.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

The old squashfs signature will be removed by a write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0xe6a63995.

Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 3
First sector (1-6663, default 1): 2
Last sector, +sectors or +size{K,M,G,T,P} (2-6663, default 6663): 2
```

^creating a partition on an extra loop to not corrupt data.

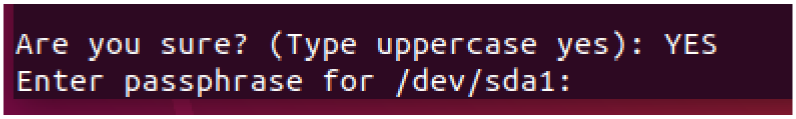


```
Created a new partition 3 of type 'Linux' and of size 512 B.

Command (m for help):
```

^successfully created a partition to encrypt.

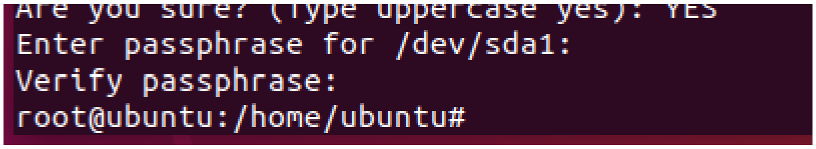


```
root@ubuntu:/home/ubuntu# cryptsetup luksFormat /dev/sda1

WARNING!
========
This will overwrite data on /dev/sda1 irrevocably.

Are you sure? (Type uppercase yes): yes
```

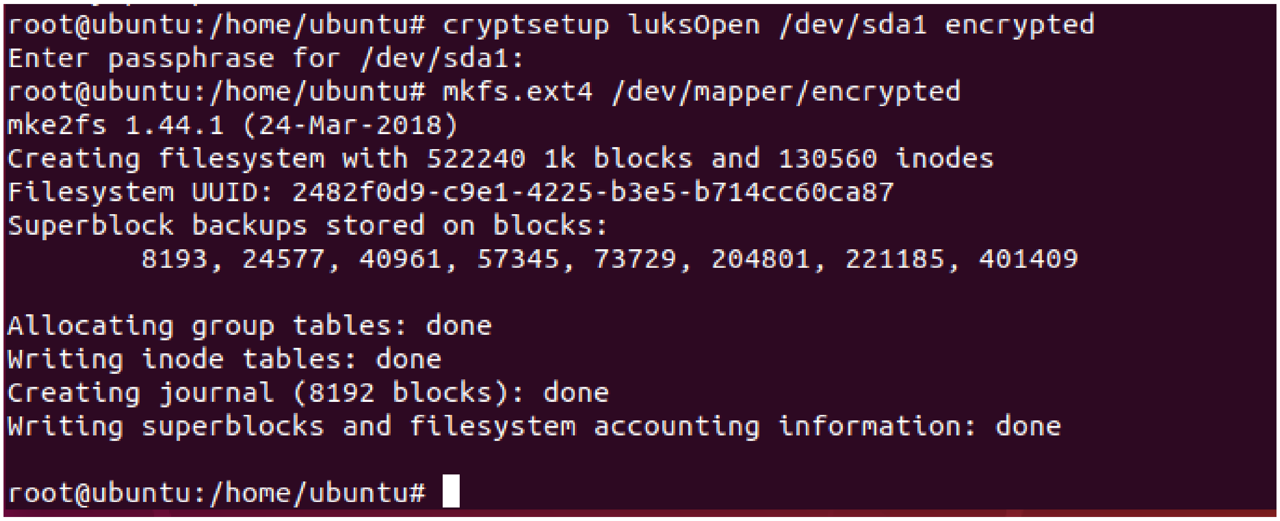


```
Are you sure? (Type uppercase yes): YES
Enter passphrase for /dev/sda1:
```

NOTE: due to loop6 execution error that always corrupts the VM i have to assume that the partitioning works and that I will be encrypting sda1 instead.

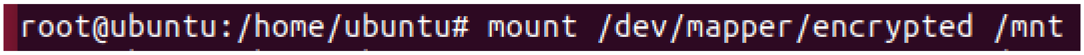


```
Are you sure? (Type uppercase yes): YES
Enter passphrase for /dev/sda1:
Verify passphrase:
root@ubuntu:/home/ubuntu#
```



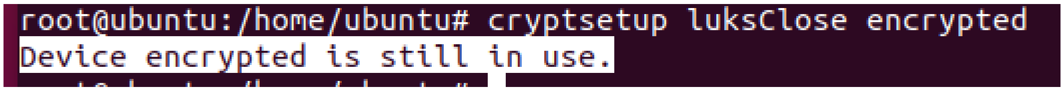
```
root@ubuntu:/home/ubuntu# cryptsetup luksOpen /dev/sda1 encrypted
Enter passphrase for /dev/sda1:
root@ubuntu:/home/ubuntu# mkfs.ext4 /dev/mapper/encrypted
mke2fs 1.44.1 (24-Mar-2018)
Creating filesystem with 522240 1k blocks and 130560 inodes
Filesystem UUID: 2482f0d9-c9e1-4225-b3e5-b714cc60ca87
Superblock backups stored on blocks:
        8193, 24577, 40961, 57345, 73729, 204801, 221185, 401409

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

root@ubuntu:/home/ubuntu#
```

^opening up the encrypted drive and creating a file for the mapper.



```
root@ubuntu:/home/ubuntu# mount /dev/mapper/encrypted /mnt
```

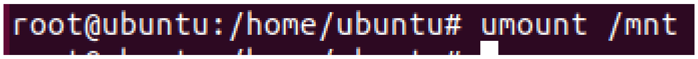
^Mounting the encrypted drive



```
root@ubuntu:/home/ubuntu# cryptsetup luksClose encrypted
Device encrypted is still in use.
```

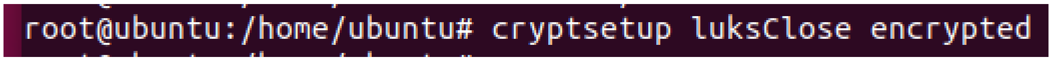
^showcases how the encryption is in place.

testing whether or not the drive is encrypted and that it requires a password to operate/open.



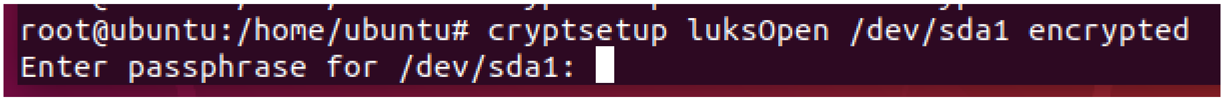
```
root@ubuntu:/home/ubuntu# umount /mnt
```

^unmounting the encrypted drive to stop it from being used



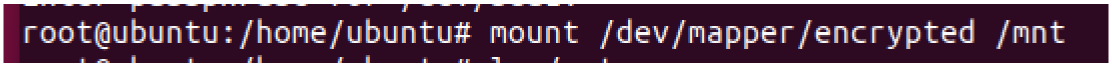
```
root@ubuntu:/home/ubuntu# cryptsetup luksClose encrypted
```

^closing the encrypted contents so that no one can access the mapped storage offline.



```
root@ubuntu:/home/ubuntu# cryptsetup luksOpen /dev/sda1 encrypted
Enter passphrase for /dev/sda1:
```

^ In order to access the encrypted hard disk contents it requires a password so that only authenticated users can read it. NOTE: This proves that the drive was successfully encrypted

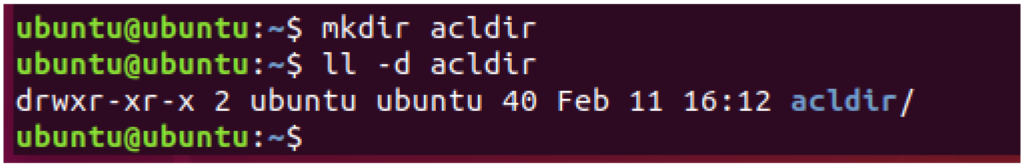


```
root@ubuntu:/home/ubuntu# mount /dev/mapper/encrypted /mnt
```

^successfully remounted the encrypted drive

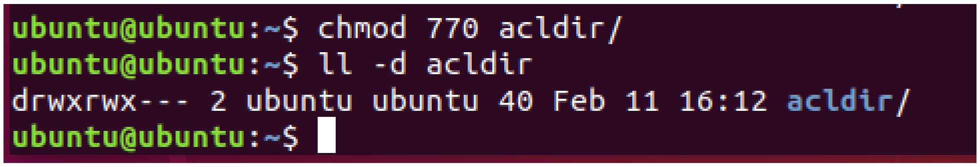
4. Access Control List

In this scenario I will be using the lecture notes on Access control lists to guide my activities.

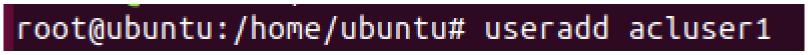


```
ubuntu@ubuntu:~$ mkdir acldir
ubuntu@ubuntu:~$ ll -d acldir
drwxr-xr-x 2 ubuntu ubuntu 40 Feb 11 16:12 acldir/
ubuntu@ubuntu:~$
```

^creating the directory to control access to and from.

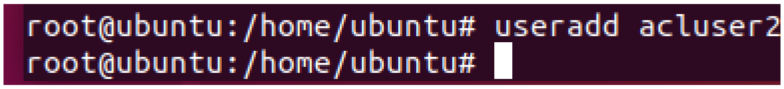


```
ubuntu@ubuntu:~$ chmod 770 acldir/
ubuntu@ubuntu:~$ ll -d acldir
drwxrwx--- 2 ubuntu ubuntu 40 Feb 11 16:12 acldir/
ubuntu@ubuntu:~$
```



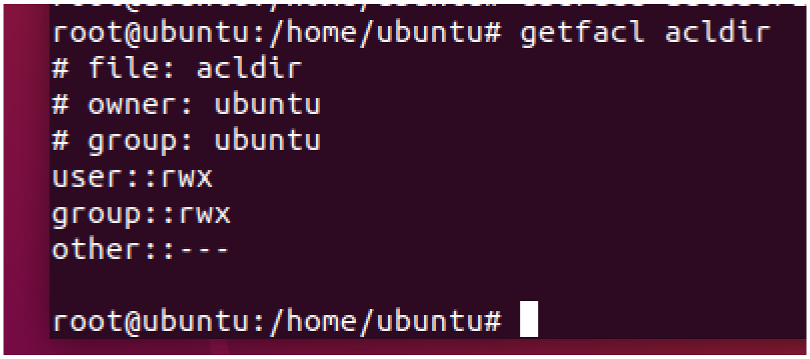
```
root@ubuntu:/home/ubuntu# useradd acluser1
```

^creating a new acluser1 from root that would be able to access the acldir.



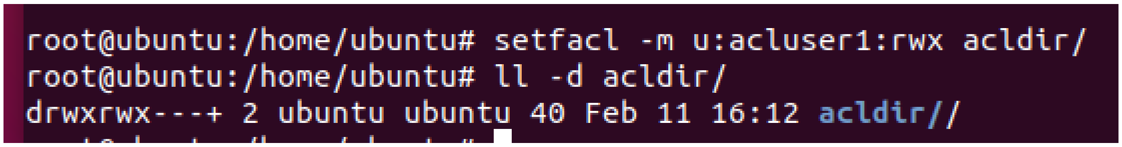
```
root@ubuntu:/home/ubuntu# useradd acluser2
root@ubuntu:/home/ubuntu#
```

^creating a user that wouldn't be able to access the acldir



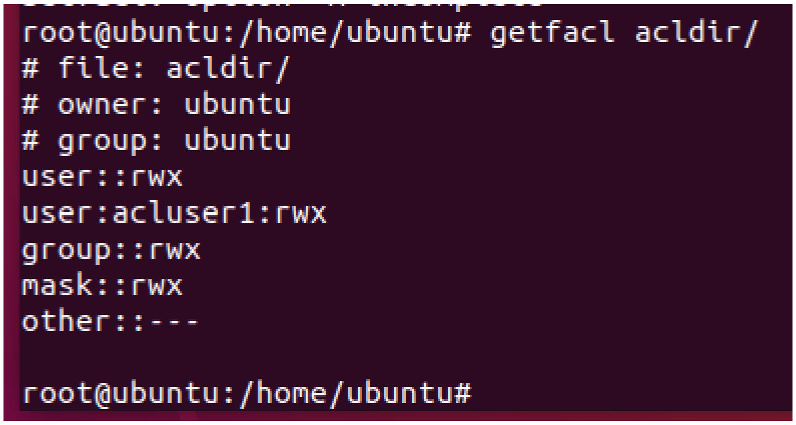
```
root@ubuntu:/home/ubuntu# getfacl acldir
# file: acldir
# owner: ubuntu
# group: ubuntu
user::rwx
group::rwx
other::---

root@ubuntu:/home/ubuntu#
```

^showcasing who is part of the directory in terms of access based on roles.



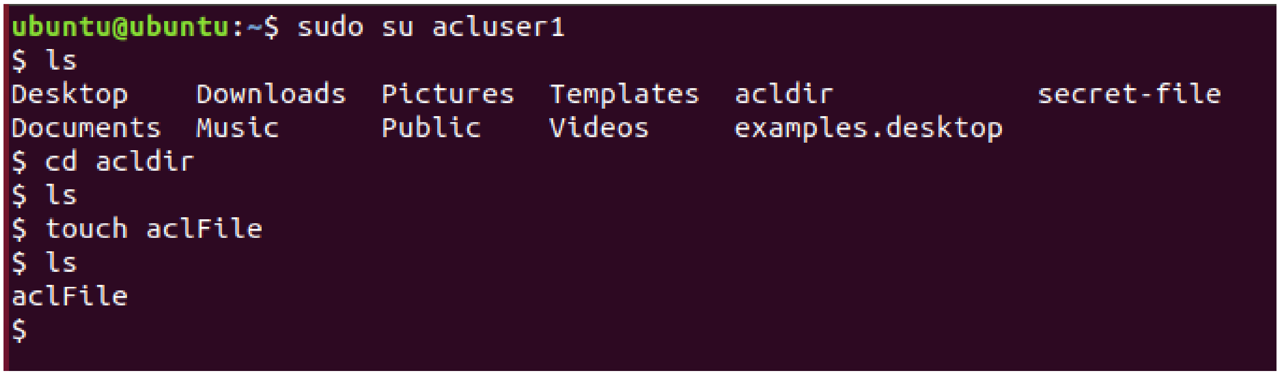
```
root@ubuntu:/home/ubuntu# setfacl -m u:acluser1:rwx acldir/
root@ubuntu:/home/ubuntu# ll -d acldir/
drwxrwx---+ 2 ubuntu ubuntu 40 Feb 11 16:12 acldir//
```

^assigns an acluser1 to acldir to be able to access it.



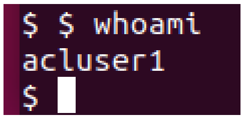
```
root@ubuntu:/home/ubuntu# getfacl acldir/
# file: acldir/
# owner: ubuntu
# group: ubuntu
user::rwx
user:acluser1:rwx
group::rwx
mask::rwx
other::---

root@ubuntu:/home/ubuntu#
```

^showcasing how acluser1 gets access to the directory while acleuser2 doesn't.



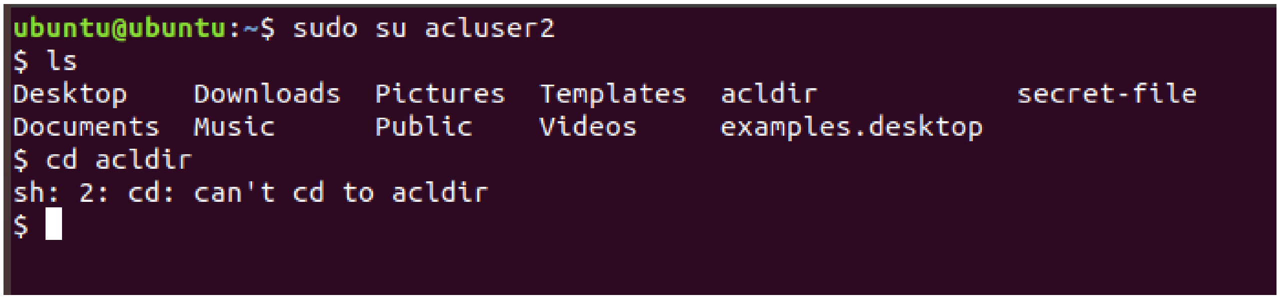
```
ubuntu@ubuntu:~$ sudo su acluser1
$ ls
Desktop    Downloads  Pictures  Templates  acldir            secret-file
Documents  Music      Public    Videos     examples.desktop
$ cd acldir
$ ls
$ touch aclFile
$ ls
aclFile
$
```

^providing proof of access to the directory.



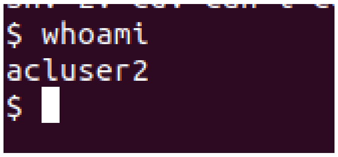
```
$ $ whoami
acluser1
$
```

^current user is acluser1 that has access the directory



```
ubuntu@ubuntu:~$ sudo su acluser2
$ ls
Desktop    Downloads  Pictures  Templates  acldir            secret-file
Documents  Music      Public    Videos     examples.desktop
$ cd acldir
sh: 2: cd: can't cd to acldir
$
```

^logged in as acluser2 in which users that are not registered on the acldir should not have access to it.

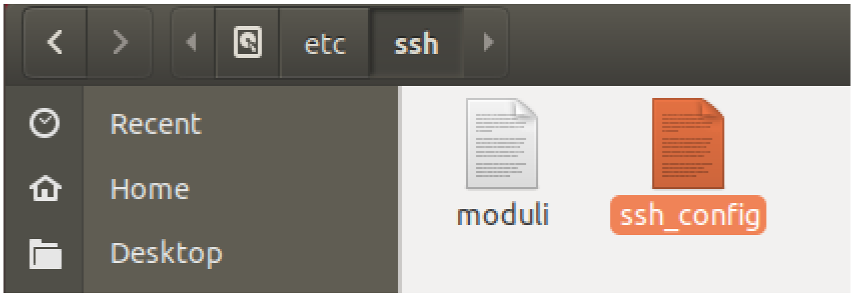


```
$ whoami
acluser2
$
```

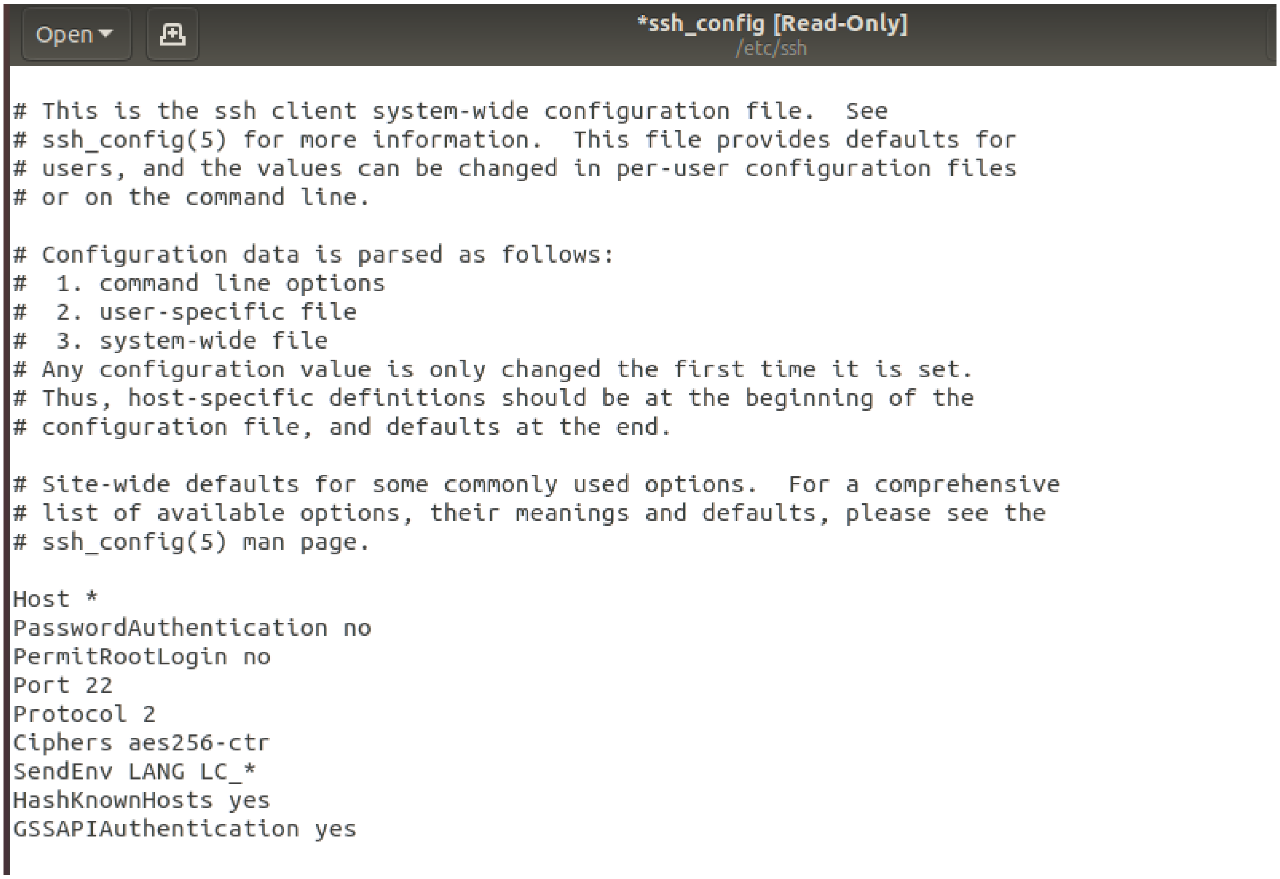
^this showcased how the OS system was successfully hardened as unauthorized users that were part of the others group that was not registered to have direct access was not able to go into the directory in the first place to read possible sensitive data.

5. Unencrypted Sharing Disabled

To only allow encrypted sharing, must configure all the services involved in sharing to use only the encrypted counter parts.



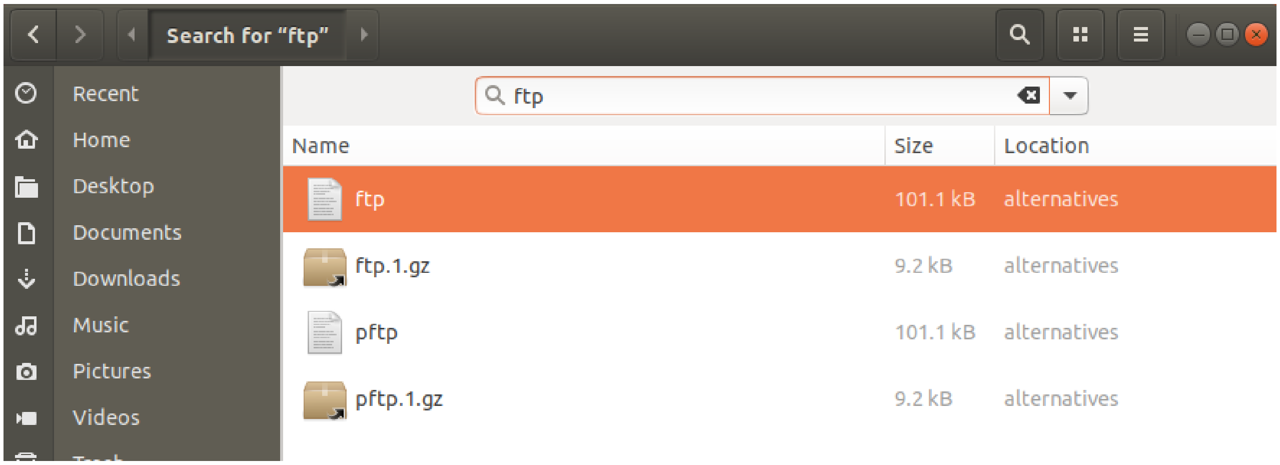
^go to ssh configuration for editing



```
# This is the ssh client system-wide configuration file.  See
# ssh_config(5) for more information.  This file provides defaults for
# users, and the values can be changed in per-user configuration files
# or on the command line.

# Configuration data is parsed as follows:
#  1. command line options
#  2. user-specific file
#  3. system-wide file
# Any configuration value is only changed the first time it is set.
# Thus, host-specific definitions should be at the beginning of the
# configuration file, and defaults at the end.

# Site-wide defaults for some commonly used options.  For a comprehensive
# list of available options, their meanings and defaults, please see the
# ssh_config(5) man page.

Host *
PasswordAuthentication no
PermitRootLogin no
Port 22
Protocol 2
Ciphers aes256-ctr
SendEnv LANG LC_*
HashKnownHosts yes
GSSAPIAuthentication yes
```

^Utilize key authentication instead of regular password authentication, as well as use ssh on the assumed port 22 with the latest protocol version.

NOTE: because of how other insecure protocols and services were blocked previously through firewall configurations and network attack surface reduction, I am assuming that there is no need to manually go in depth into the other possible file sharing services that could help attackers exfiltrate sensitive data across the network as it wouldn't be possible with how the other ports are already covered and that there are no more or very unnecessary configuration files to change for common unsecure services such as for ftp shown below; this is because of how as secure one such as the ssh configuration file allowed for a hardening of a more secure ssh service was already previously done to further harden the OS Client for preventing the possibility unencrypted file sharing even more.



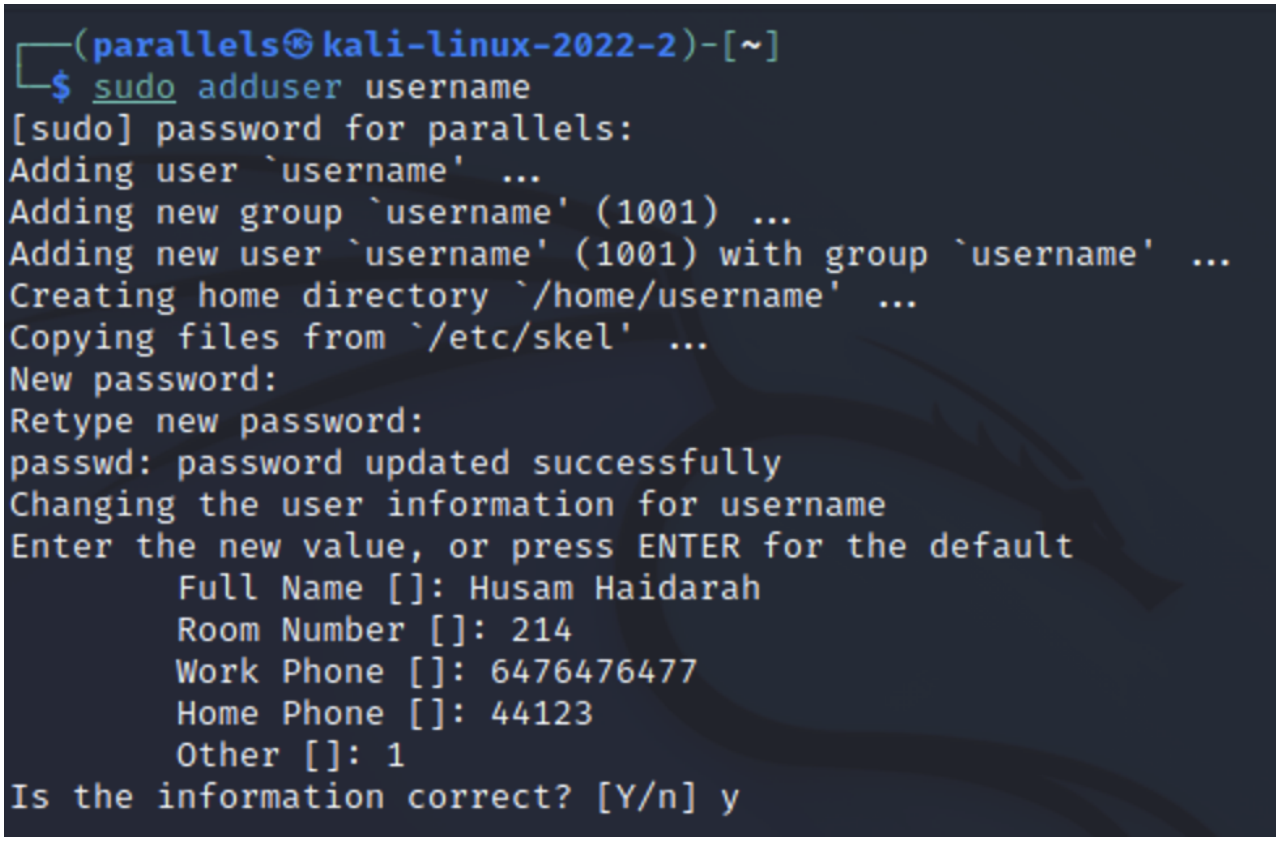
ftp

Name	Size	Location
ftp	101.1 kB	alternatives
ftp.1.gz	9.2 kB	alternatives
pftp	101.1 kB	alternatives
pftp.1.gz	9.2 kB	alternatives

NOTE: Because it is impossible to disable unencrypted file sharing as a whole in terms of one service, you can only whitelist certain sharing services through secure means. This means that it extends the firewall configuration and into the secure services themselves to ensure that only the latest versions of encryption through secure means is used.

6. Account Management,

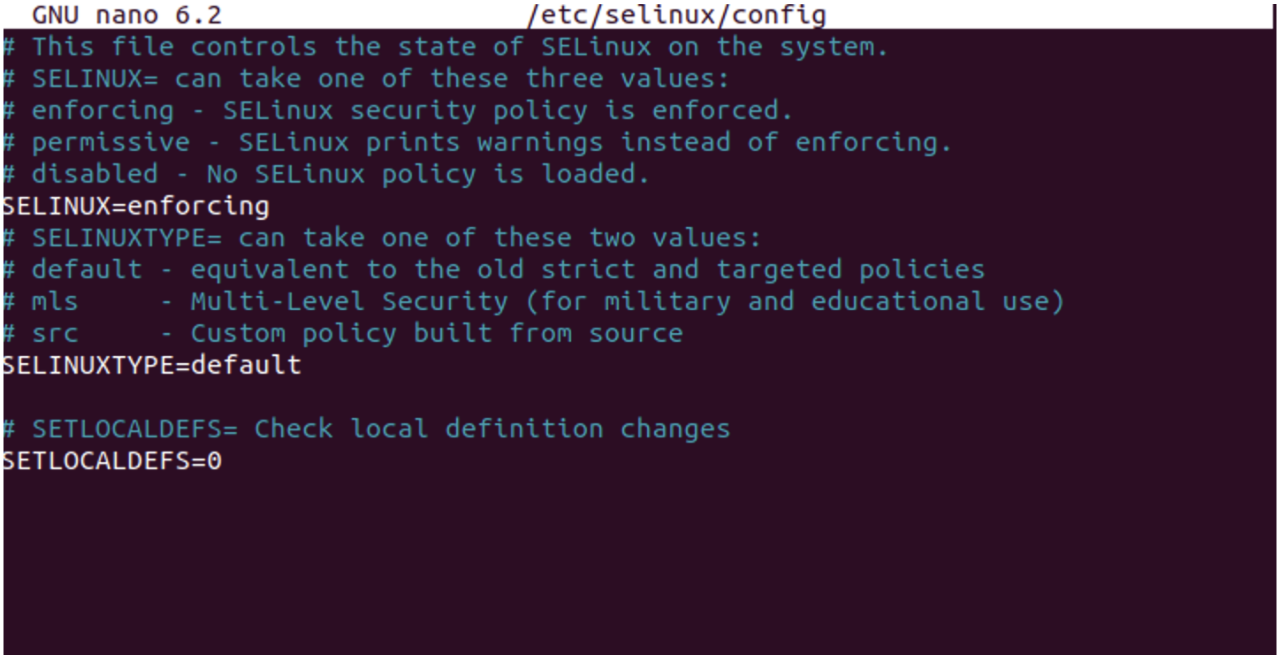
By running the command sudo adduser username in Kali this enable me to modify and add user into my account



```
┌──(parallels㉿kali-linux-2022-2)-[~]
└─$ sudo adduser username
[sudo] password for parallels:
Adding user `username' ...
Adding new group `username' (1001) ...
Adding new user `username' (1001) with group `username' ...
Creating home directory `/home/username' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for username
Enter the new value, or press ENTER for the default
        Full Name []: Husam Haidarah
        Room Number []: 214
        Work Phone []: 6476476477
        Home Phone []: 44123
        Other []: 1
Is the information correct? [Y/n] y
```

7. enable SELinux,

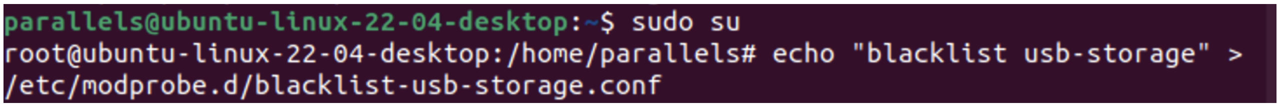
By using enforcing this will enable the SELinux



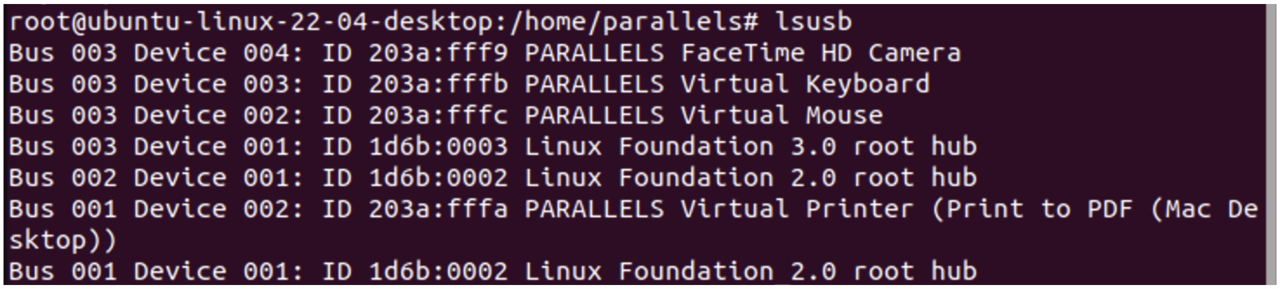
```
  GNU nano 6.2                    /etc/selinux/config
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
# enforcing - SELinux security policy is enforced.
# permissive - SELinux prints warnings instead of enforcing.
# disabled - No SELinux policy is loaded.
SELINUX=enforcing
# SELINUXTYPE= can take one of these two values:
# default - equivalent to the old strict and targeted policies
# mls     - Multi-Level Security (for military and educational use)
# src     - Custom policy built from source
SELINUXTYPE=default

# SETLOCALDEFS= Check local definition changes
SETLOCALDEFS=0
```

8. disable USB usage (prevent physically copying over the data),

The system won't identify any USB storage devices and the USB storage driver module won't be able to load. after delete the "blacklist-usb-storage.conf"



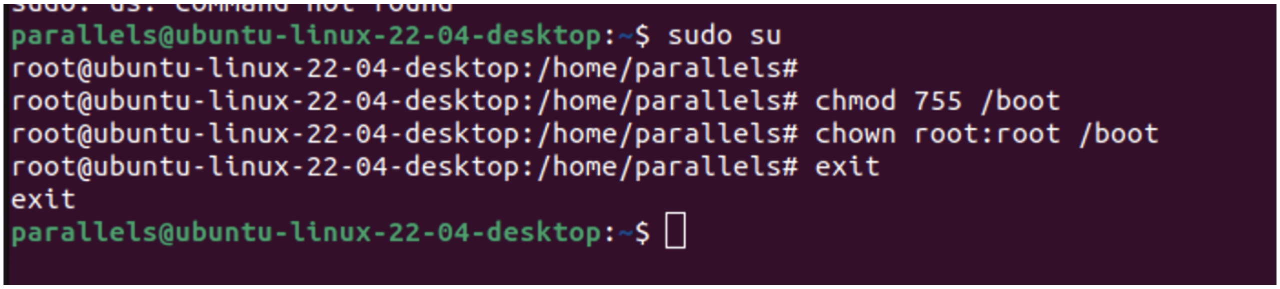
```
parallels@ubuntu-linux-22-04-desktop:~$ sudo su
root@ubuntu-linux-22-04-desktop:/home/parallels# echo "blacklist usb-storage" >
/etc/modprobe.d/blacklist-usb-storage.conf
```



```
root@ubuntu-linux-22-04-desktop:/home/parallels# lsusb
Bus 003 Device 004: ID 203a:fff9 PARALLELS FaceTime HD Camera
Bus 003 Device 003: ID 203a:fffb PARALLELS Virtual Keyboard
Bus 003 Device 002: ID 203a:fffc PARALLELS Virtual Mouse
Bus 003 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 002: ID 203a:fffa PARALLELS Virtual Printer (Print to PDF (Mac De
sktop))
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

9. lock boot directory,

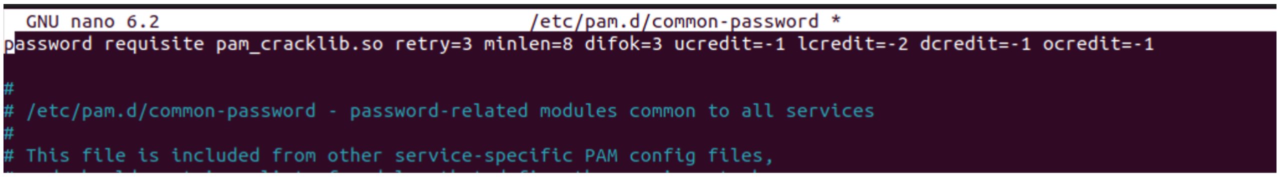
The /boot directory is only fully accessible to the root user, while other users can only read and execute files in the directory.



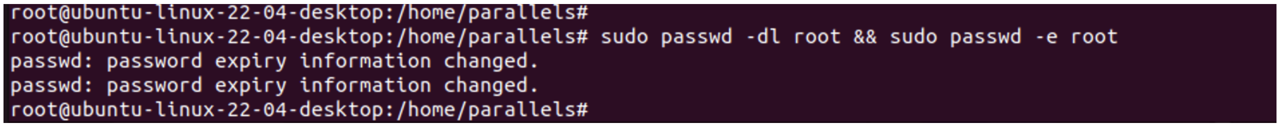
```
sudo: us: command not found
parallels@ubuntu-linux-22-04-desktop:~$ sudo su
root@ubuntu-linux-22-04-desktop:/home/parallels#
root@ubuntu-linux-22-04-desktop:/home/parallels# chmod 755 /boot
root@ubuntu-linux-22-04-desktop:/home/parallels# chown root:root /boot
root@ubuntu-linux-22-04-desktop:/home/parallels# exit
exit
parallels@ubuntu-linux-22-04-desktop:~$ 
```

10. Enforce strong password policies.

This command compels the passcode policy to be applied to the root account by first disabling the root login information and afterwards instantly enabling it again.

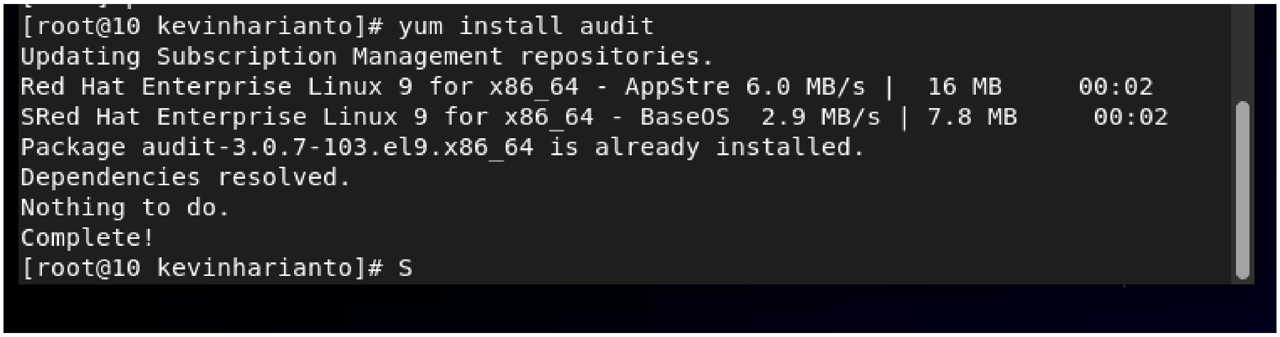


```
  GNU nano 6.2                              /etc/pam.d/common-password *
password requisite pam_cracklib.so retry=3 minlen=8 difok=3 ucredit=-1 lcredit=-2 dcredit=-1 ocredit=-1

#
# /etc/pam.d/common-password - password-related modules common to all services
#
# This file is included from other service-specific PAM config files,
```



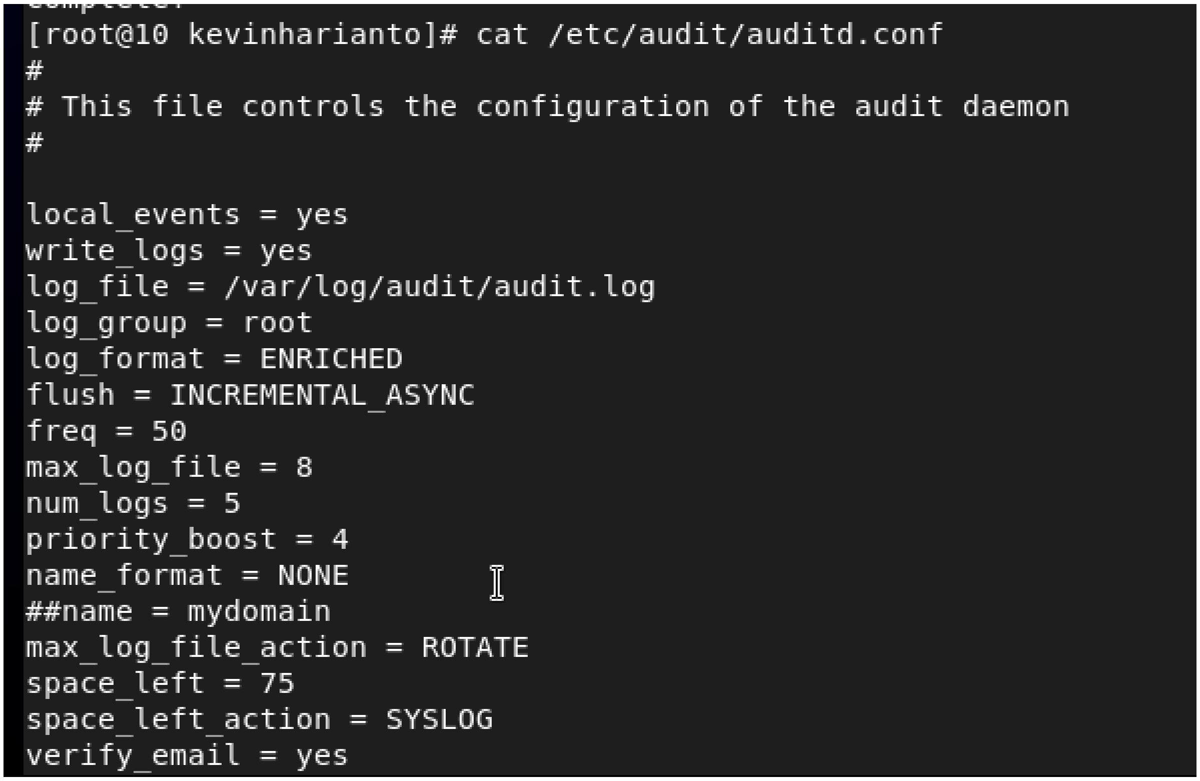
```
root@ubuntu-linux-22-04-desktop:/home/parallels#
root@ubuntu-linux-22-04-desktop:/home/parallels# sudo passwd -dl root && sudo passwd -e root
passwd: password expiry information changed.
passwd: password expiry information changed.
root@ubuntu-linux-22-04-desktop:/home/parallels#
```

Enabling the use of IT Security auditing services to monitor for any anomalies in relation to file access activities



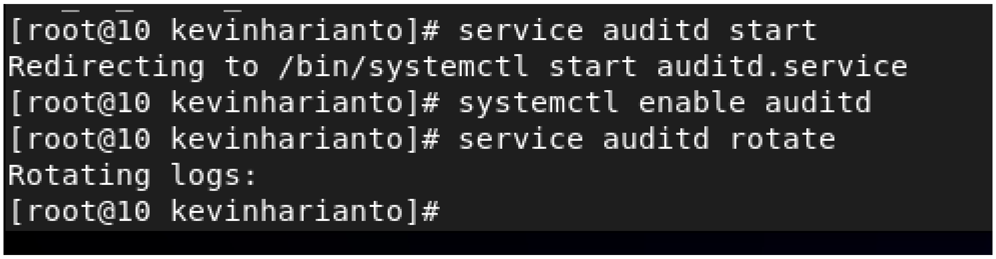
```
[root@10 kevinharianto]# yum install audit
Updating Subscription Management repositories.
Red Hat Enterprise Linux 9 for x86_64 - AppStre 6.0 MB/s |  16 MB     00:02
SRed Hat Enterprise Linux 9 for x86_64 - BaseOS  2.9 MB/s | 7.8 MB     00:02
Package audit-3.0.7-103.el9.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@10 kevinharianto]# S
```

^installing audit utilities



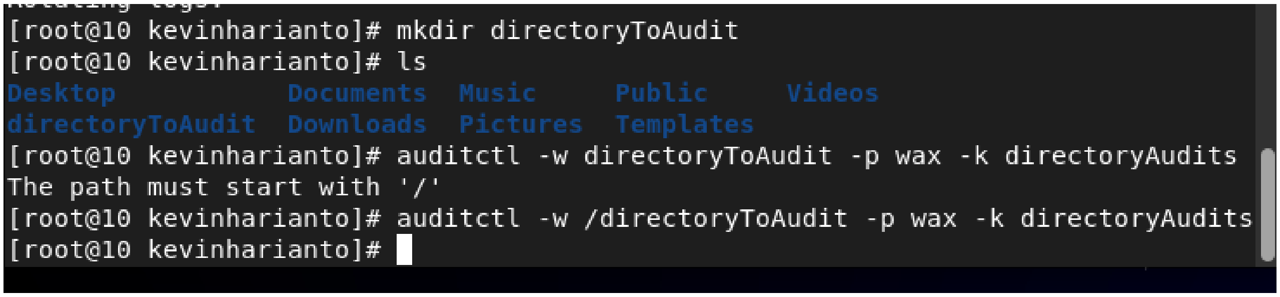
```
[root@10 kevinharianto]# cat /etc/audit/auditd.conf
#
# This file controls the configuration of the audit daemon
#

local_events = yes
write_logs = yes
log_file = /var/log/audit/audit.log
log_group = root
log_format = ENRICHED
flush = INCREMENTAL_ASYNC
freq = 50
max_log_file = 8
num_logs = 5
priority_boost = 4
name_format = NONE
##name = mydomain
max_log_file_action = ROTATE
space_left = 75
space_left_action = SYSLOG
verify_email = yes
```

^showcasing the establishment of the current configuration of the audit policy.



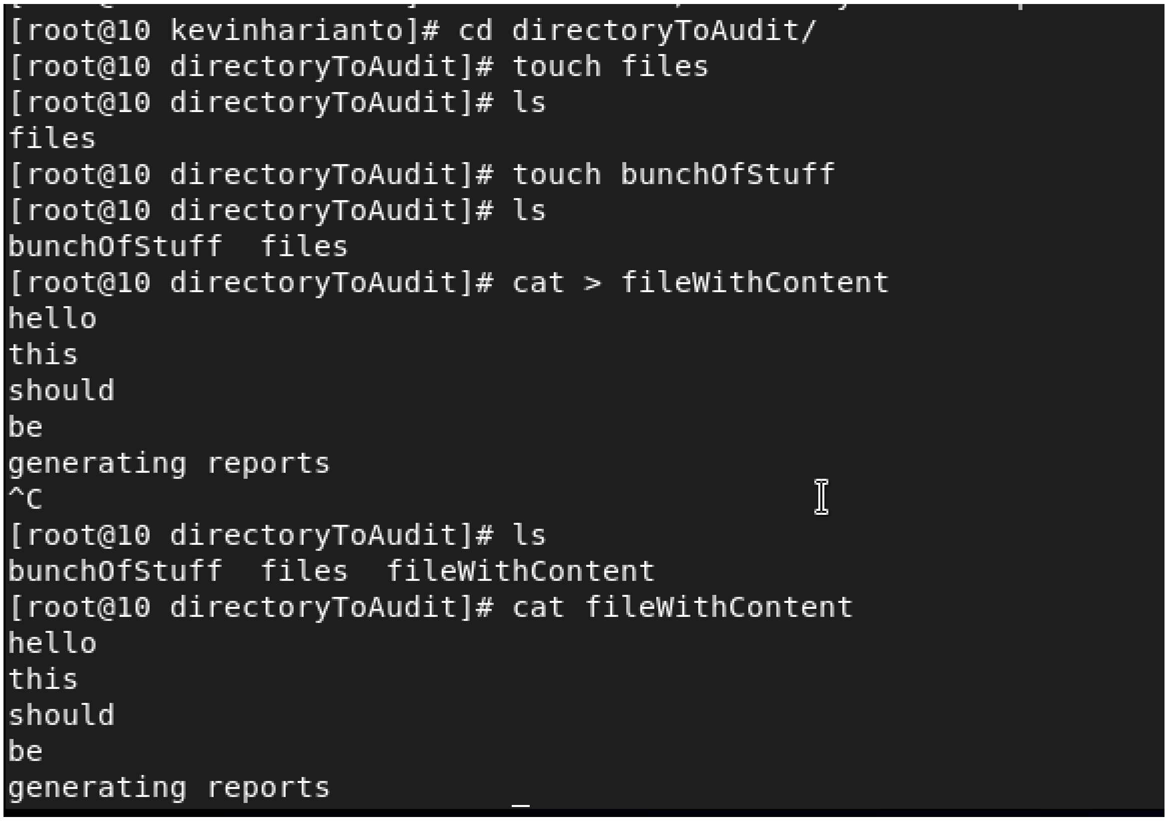
```
[root@10 kevinharianto]# service auditd start
Redirecting to /bin/systemctl start auditd.service
[root@10 kevinharianto]# systemctl enable auditd
[root@10 kevinharianto]# service auditd rotate
Rotating logs:
[root@10 kevinharianto]#
```

^enabling auditing services on the RedHat linux system.



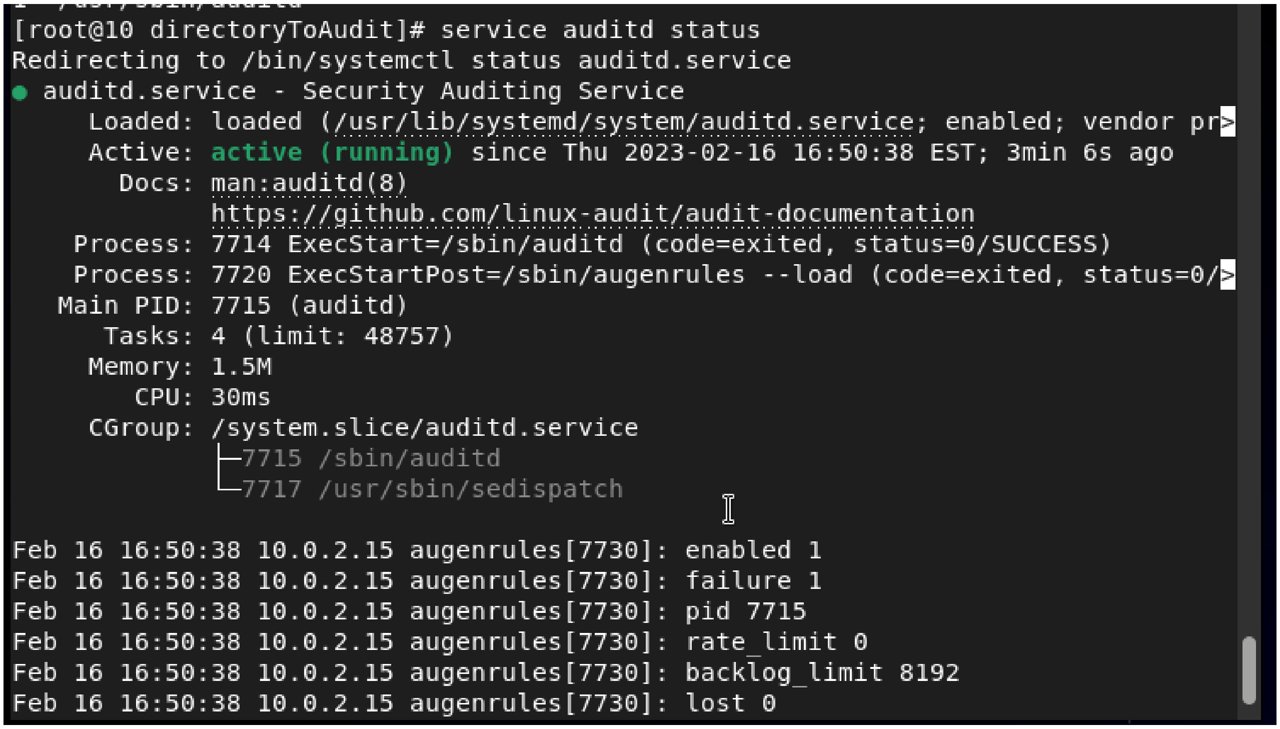
```
[root@10 kevinharianto]# mkdir directoryToAudit
[root@10 kevinharianto]# ls
Desktop           Documents  Music     Public     Videos
directoryToAudit  Downloads  Pictures  Templates
[root@10 kevinharianto]# auditctl -w directoryToAudit -p wax -k directoryAudits
The path must start with '/'
[root@10 kevinharianto]# auditctl -w /directoryToAudit -p wax -k directoryAudits
[root@10 kevinharianto]# 
```

^choosing a directory to audit activities in. This would allow for the generation of audit events and produce a report from the OS' auditing system to show some of the actions captured.
Planning on capturing writes, accesses, and executions.



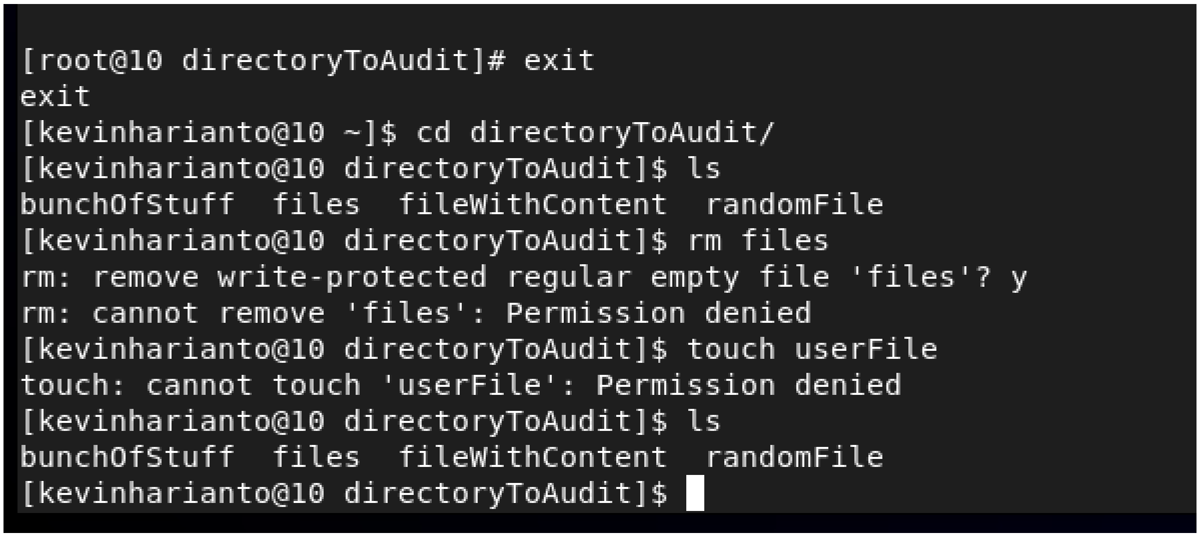
```
[root@10 kevinharianto]# cd directoryToAudit/
[root@10 directoryToAudit]# touch files
[root@10 directoryToAudit]# ls
files
[root@10 directoryToAudit]# touch bunchOfStuff
[root@10 directoryToAudit]# ls
bunchOfStuff  files
[root@10 directoryToAudit]# cat > fileWithContent
hello
this
should
be
generating reports
^C
[root@10 directoryToAudit]# ls
bunchOfStuff  files  fileWithContent
[root@10 directoryToAudit]# cat fileWithContent
hello
this
should
be
generating reports
```

^doing activities within the target directory that will be logged by the audit utilities.



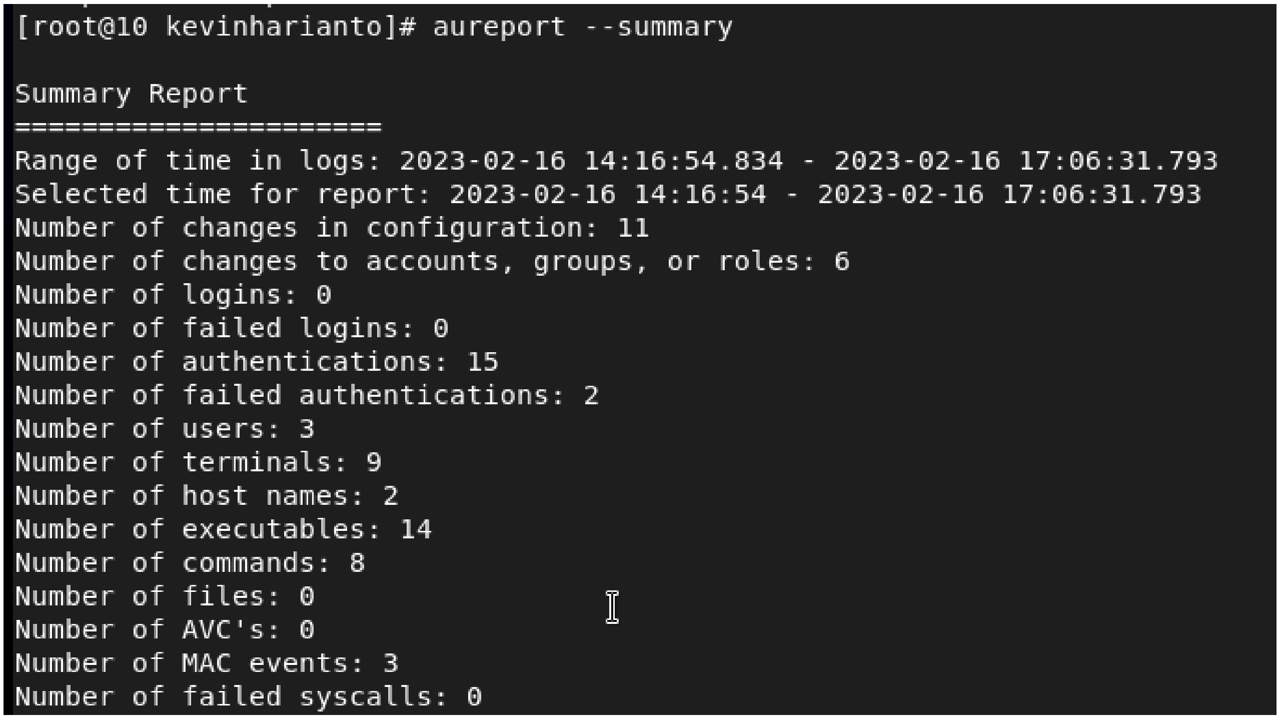
```
[root@10 directoryToAudit]# service auditd status
Redirecting to /bin/systemctl status auditd.service
● auditd.service - Security Auditing Service
     Loaded: loaded (/usr/lib/systemd/system/auditd.service; enabled; vendor pr>
     Active: active (running) since Thu 2023-02-16 16:50:38 EST; 3min 6s ago
       Docs: man:auditd(8)
             https://github.com/linux-audit/audit-documentation
    Process: 7714 ExecStart=/sbin/auditd (code=exited, status=0/SUCCESS)
    Process: 7720 ExecStartPost=/sbin/augenrules --load (code=exited, status=0/>
   Main PID: 7715 (auditd)
      Tasks: 4 (limit: 48757)
     Memory: 1.5M
        CPU: 30ms
     CGroup: /system.slice/auditd.service
             ├─7715 /sbin/auditd
             └─7717 /usr/sbin/sedispatch

Feb 16 16:50:38 10.0.2.15 augenrules[7730]: enabled 1
Feb 16 16:50:38 10.0.2.15 augenrules[7730]: failure 1
Feb 16 16:50:38 10.0.2.15 augenrules[7730]: pid 7715
Feb 16 16:50:38 10.0.2.15 augenrules[7730]: rate_limit 0
Feb 16 16:50:38 10.0.2.15 augenrules[7730]: backlog_limit 8192
Feb 16 16:50:38 10.0.2.15 augenrules[7730]: lost 0
```

^ensure auditing utility is running



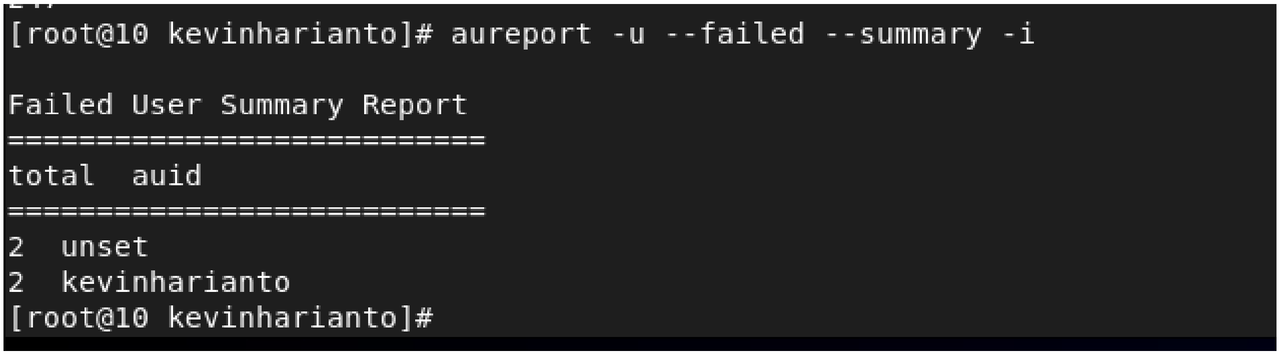
```
[root@10 directoryToAudit]# exit
exit
[kevinharianto@10 ~]$ cd directoryToAudit/
[kevinharianto@10 directoryToAudit]$ ls
bunchOfStuff  files  fileWithContent  randomFile
[kevinharianto@10 directoryToAudit]$ rm files
rm: remove write-protected regular empty file 'files'? y
rm: cannot remove 'files': Permission denied
[kevinharianto@10 directoryToAudit]$ touch userFile
touch: cannot touch 'userFile': Permission denied
[kevinharianto@10 directoryToAudit]$ ls
bunchOfStuff  files  fileWithContent  randomFile
[kevinharianto@10 directoryToAudit]$ 
```

^generating events from a lower privileged account.



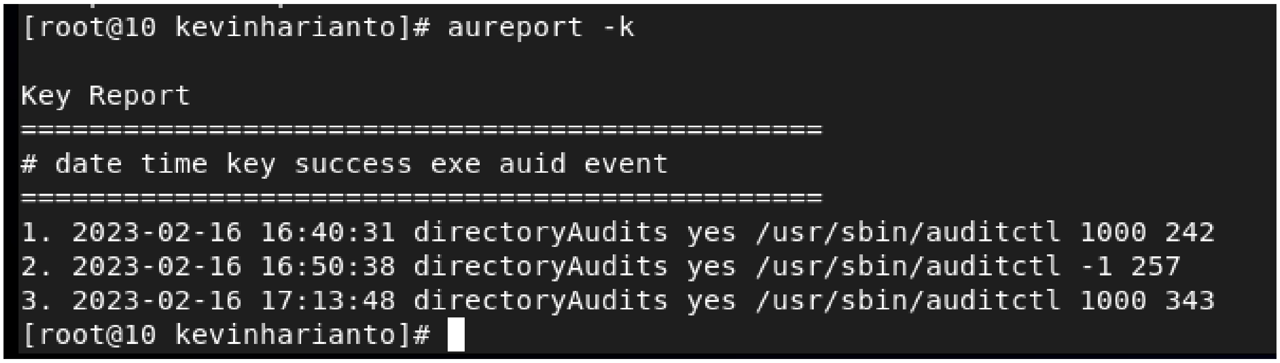
```
[root@10 kevinharianto]# aureport --summary

Summary Report
======================
Range of time in logs: 2023-02-16 14:16:54.834 - 2023-02-16 17:06:31.793
Selected time for report: 2023-02-16 14:16:54 - 2023-02-16 17:06:31.793
Number of changes in configuration: 11
Number of changes to accounts, groups, or roles: 6
Number of logins: 0
Number of failed logins: 0
Number of authentications: 15
Number of failed authentications: 2
Number of users: 3
Number of terminals: 9
Number of host names: 2
Number of executables: 14
Number of commands: 8
Number of files: 0
Number of AVC's: 0
Number of MAC events: 3
Number of failed syscalls: 0
```

^audit report



```
[root@10 kevinharianto]# aureport -u --failed --summary -i

Failed User Summary Report
============================
total  auid
============================
2  unset
2  kevinharianto
[root@10 kevinharianto]#
```

^this report showcases the direct audit report on the failed activities generated purposely by the user. This proves how the directory was successfully audited.



```
[root@10 kevinharianto]# aureport -k

Key Report
===============================================
# date time key success exe auid event
===============================================
1. 2023-02-16 16:40:31 directoryAudits yes /usr/sbin/auditctl 1000 242
2. 2023-02-16 16:50:38 directoryAudits yes /usr/sbin/auditctl -1 257
3. 2023-02-16 17:13:48 directoryAudits yes /usr/sbin/auditctl 1000 343
[root@10 kevinharianto]# █
```

^This audit report also showcased how the audit policy which was defined as directoryAudits was successful in capturing events on the directory specified to audit previously shown.