Sri Lanka Institute of Information Technology

Specialized in Cyber Security

Year 3, Semester 1

IE3062: Data and Operating System Security

Group 35

Student ID No.	Name
IT23136106	D.M.M. Pasindu Supushmika
IT23227286	Peiris W.S.S.N
IT23228962	Jayasinghe J.M.S.S
IT23318120	Udith Kaushalya

Contents

1. OS Security 2
1.1 Comparison of the OSs 2
1.2 Installation Process 3
1.2.1 Download the OS 3
1.2.2 Install the OS 4
1.3 OS Security Analysis & Hardening 4
1.3.1 Threat Analysis 4
1.3.2 Hardening Configurations 4
2. Data base Security 17
2.1 Installation Process 17
2.1.1 Download Oracle DBMS 17
2.1.2 Install Oracle DBMS 18
2.1.3 Setup SQLPlus: 20
2.2 Identification and implementation of configurations to harden Oracle DBMS 20
2.2.1 Secure default Accounts & passwords 20
2.2.2 Enforce Principle of Least Privilege 21
2.2.3 Implement Strong Password Policies 22
2.2.4 Harden Network Configuration & Listener 23
2.2.5 Enable and Configure Auditing 24
2.3 Entity Relational Diagram for Customer Management System 26
2.3.1 ER Diagram 26
2.3.2 Created tables and records of sample data 26
2.4 User and Role Management 30
2.5 View Implementation 32
2.6 Virtual Private Database Implementation 33
2.6.1 VPD Policy Function 33
2.6.2 VPD Policy Application 34
2.7 Data Encryption Implementation 35
2.8 Data Masking 35
2.9 FGA Policies Configured 36
3. Big Data Security 37
3.1 Description 37
3.2 The main security requirements of big data 37
3.3 Common Attacks on Big Data Systems 38
3.4 Security Controls to Mitigate Attacks 39
3.5 Threat to Control Mapping 40

1. OS Security

1.1 Comparison of the OSs

Feature	**Ubuntu Server**	**CentOS/Alma Linux**	**Debian**	**Fedora Server**	**Arch Linux**
Ease of Use	Very User-friendly, simple installation, strong GUI/CLI balance, extensive docs.	Moderate ease; More suited for sysadmins; Requires more manual Setup.	Moderate; Requires CLI skills and deeper Linux knowledge.	Fairly user friendly but less stable; Focuses on new features.	Difficult for beginners; Steep learning curve; Requires manual configurations.
Community Support	Extremely strong; Forums, AskUbuntu, StackOverFlow, and Canonical Support	Strong Community; active support continues under AlmaLinux/Rocky after CentOS changes.	Very strong; one of the oldest communities with mailing lists and extensive bug tracking.	Strong; backed by Red Hat and open-source community with focus on innovation.	Niche but dedicated; wiki and forums are highly detailed but not beginner-friendly.
Long-Term Support	LTS versions supported for 5 years(optionally extendable to 10 years.)	Previously up to 10 years; CentOS Stream now rolling updates; AlmaLinux provides LTS.	Stable releases with ~3 years support; famous for long-lasting security updates.	Short cycles (~13 months); requires frequent upgrades.	Rolling release; always up-to-date but no official long-term support.
Security Updates	Regular, timely updates; Canonical security team ensures patched vulnerabilities.	Reliable, but slower than Fedora or Ubuntu; focuses on stability first.	Strong; security updates reliable but can lag behind bleeding-edge distros.	Very fast updates; security patches deployed quickly due to short cycle.	Security depends on admin; rolling updates may introduce instability.
Licensing Costs	Free and open source; optional paid support from Canonical available.	Free and open source; AlmaLinux and Rocky are completely free.	Free and open source; no licensing fees at all.	Free and open source; commercial RHEL support option available separately.	Free and open source; no licensing, but full responsibility falls on the admin.
Performance & Stability	Balanced between performance and usability; optimized for	Extremely stable; widely used for hosting, servers, and data centers.	Rock-solid stability; widely used for backend systems and	Focuses on innovation; less stable due to rapid release cycle.	Lightweight and fast; stability depends on frequent

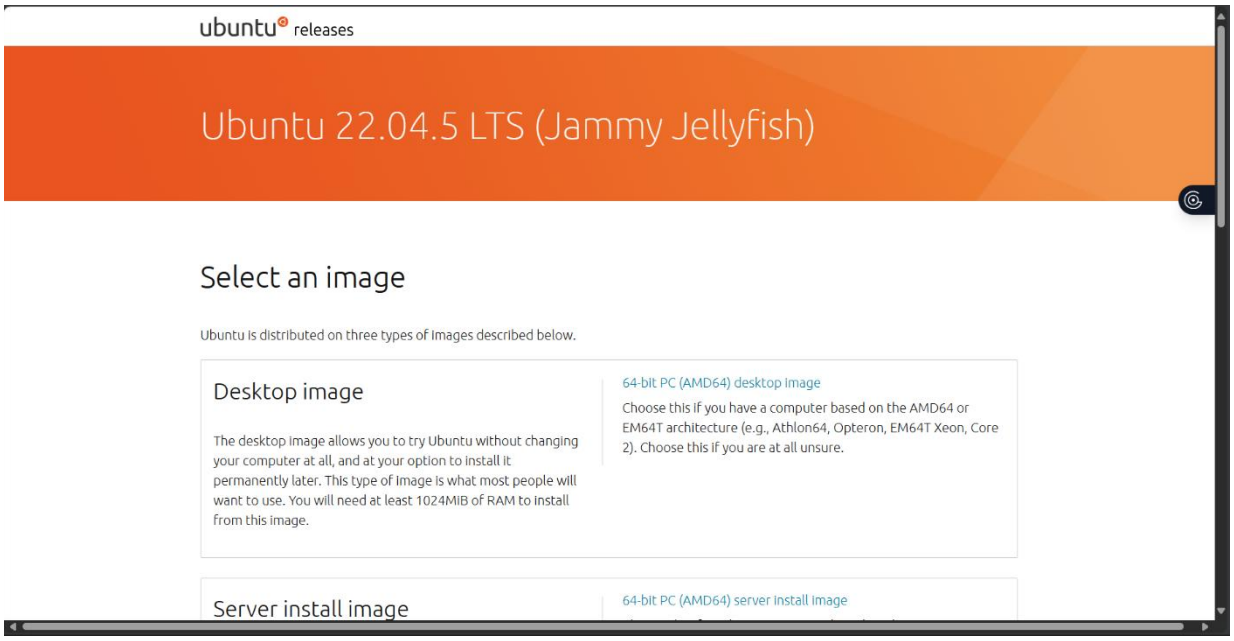
	cloud & containers.		research labs.		manual maintenance.
Target Audience	Beginners, students, developers, SMEs, cloud hosting, general-purpose servers.	Sysadmins, enterprise data centers, hosting providers, production servers.	Universities, researchers, advanced Linux users, developers who value stability.	Developers who need cutting-edge features, testers, security researchers.	Power users, enthusiasts, sysadmins who want full control and customization.
Pros	Easy to use, large support base, regular updates, extensive documentation.	Enterprise-grade stability, compatible with RHEL, long-term support.	Highly stable, secure, massive software repositories.	Cutting-edge, strong security features, early adoption of new tech.	Highly customizable, minimalistic, always up-to-date.
Cons	May include bloatware compared to minimal distros.	Less frequent updates; CentOS Stream shift caused enterprise uncertainty.	Setup can be complex; slower to adopt new features.	Shorter support cycle; requires frequent upgrades.	Steep learning curve; requires lots of manual setup and maintenance.

Table 1: Compression between 5 OSs

1.2 Installation Process

1.2.1 Download the OS

Download the Ubuntu OS (Desktop Image) via this Link:
[Ubuntu 22.04.5 LTS (Jammy Jellyfish)]



1.2.2 Install the OS

1.3 OS Security Analysis & Hardening

For the company mentioned, Ubuntu will run a database for an HR system. This section analyze the server's security risks and lists practical hardening actions (with commands and verification steps) Apply these on the Ubuntu Server you installed and include screenshots of each step in the report.

1.3.1 Threat Analysis

1. Brute-force / Credential Guessing: Attackers repeatedly try SSH or DB accounts.
2. Exploitable unpatched software known as CVEs on OS or services.
3. Unauthorized access / privilege escalation due to poor user account control or exploitable daemons.
4. Service / Port scanning and remote code execution: exposed services may be attacked.
5. Malware / Backdoors / Rootkits: Attackers install persistent malware.
6. Network attacks (MITM, IP spoofing, ARP poisoning): Especially on poorly segmented networks.
7. Data theft / Leakage: Sensitive HR data copies or exfiltrated.
8. Denial of Service (DoS): Resource exhaustion or network flood.
9. Log tempering / Lack of audit trail: Attackers remove traces.
10. Physical / Disk compromise: Stolen or misused drives without encryption.

1.3.2 Hardening Configurations

Imp: The following codes and config snippets should run on the server. Do not disable password authentication before installing the SSH key, and ensure that you have an active admin account with key access before locking yourself out.

1. **Update and Patch Regularly**

Fixes vulnerabilities before attackers exploit them.

Attacks prevented:

- Exploits of outdated software
- Kernel CVEs.

Commands:
sudo apt update
sudo apt upgrade

Verify:
grep "Packages were upgraded" /var/log/unattended-upgrades/unattended-upgrades.log

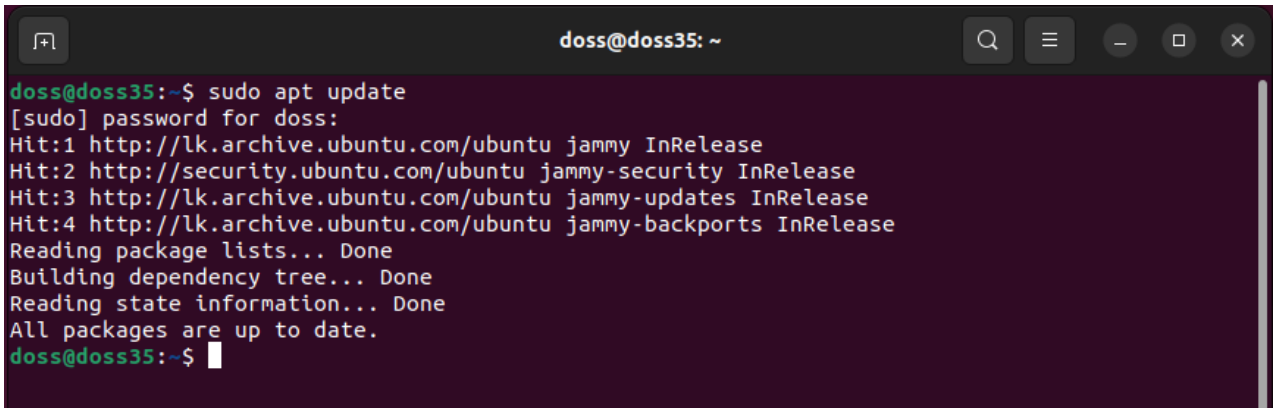


Figure 1: Update Ubuntu

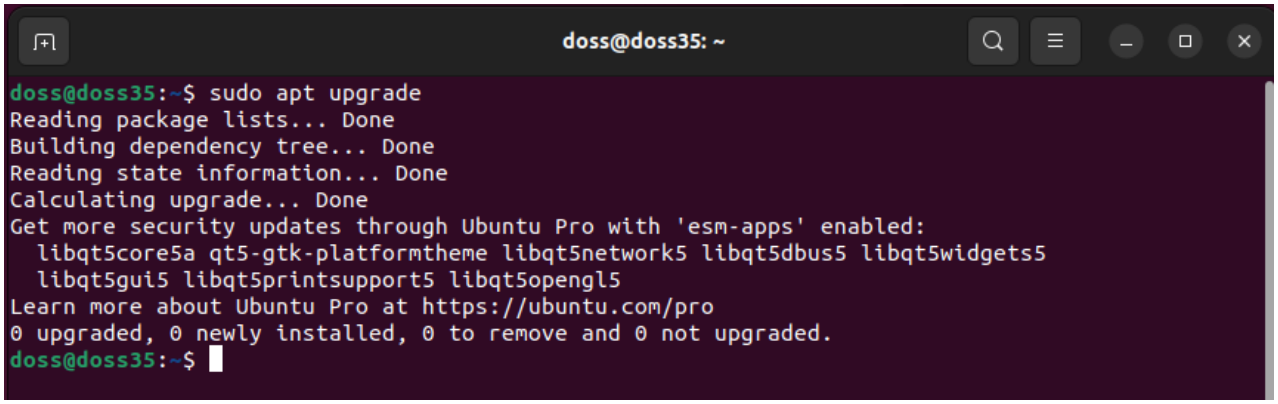


Figure 2: Upgrade Ubuntu

2. Firewall Configuration (UFW)

Only required ports are allowed.

Attack prevented:

- Port scanning
- Unauthorized access

Commands:

sudo apt install ufw
sudo ufw default deny incoming
sudo ufw default allow outgoing
sudo ufw allow ssh
sudo ufw enable

Verify:

sudo ufw status verbose

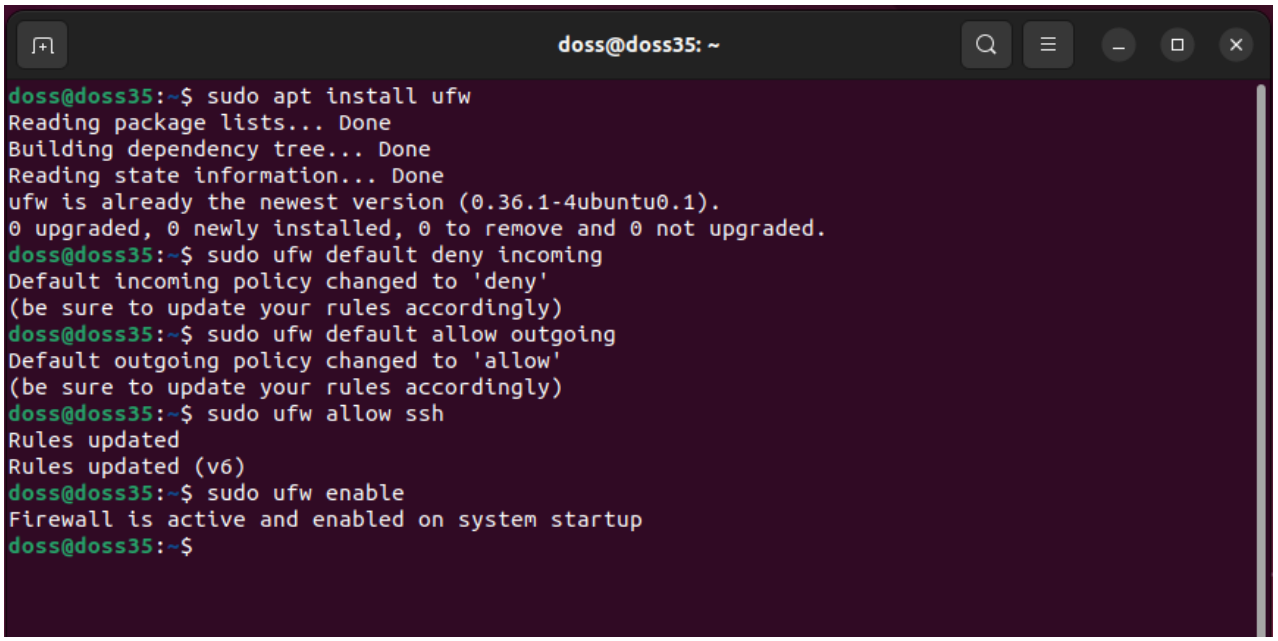


Figure 3: Firewall Configuration - UFW

3. Disable Root Login via SSH

Forces attackers to guess both username + password.

Attack prevented:

- Direct brute force on root

Commands:

sudo nano /etc/ssh/sshd_config

Added: PermitRootLogin no

sudo systemctl restart ssh

sudo systemctl status ssh

Verify:

sudo ufw status verbose shows rules.

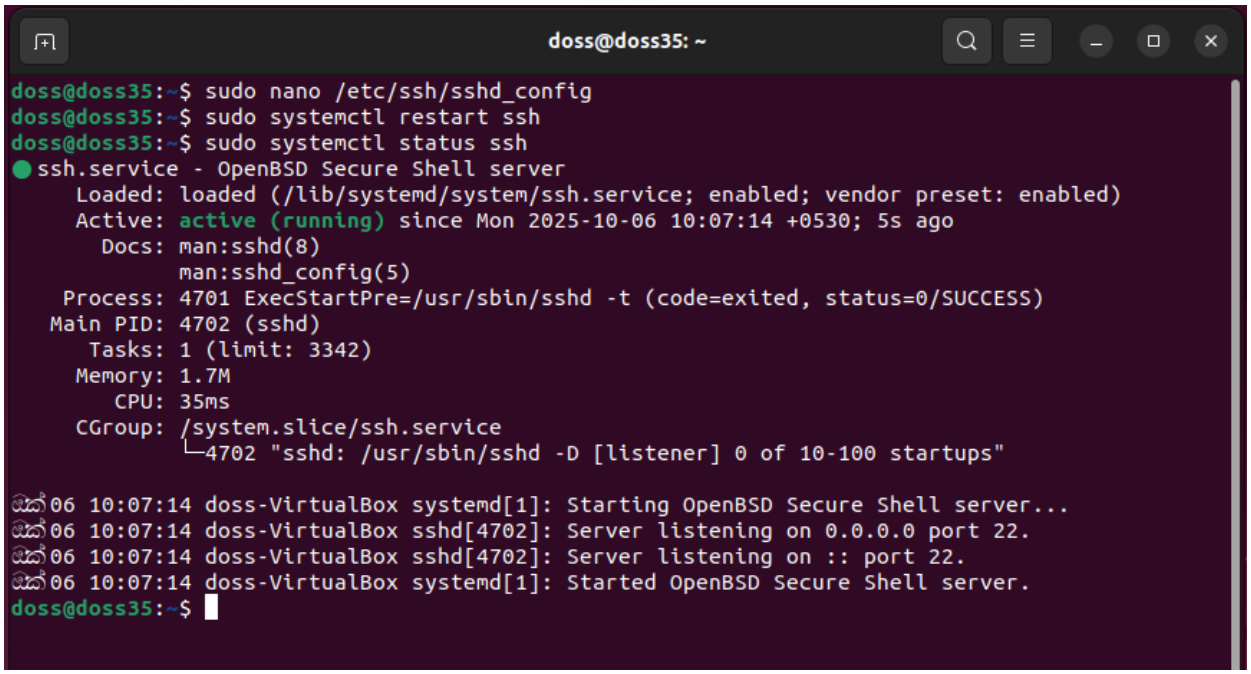


Figure 4: Disable Root Login via SSH

Figure 5: Disable Root Login via SSH

4. Enforce Strong SSH Authentication

SSH keys are far stronger than passwords.

Attacks prevented:

- Password brute force.

Commands (& simple jail):
sudo nano /etc/ssh/sshd_config
Added:
Password Authentication no
Allow Users your username
sudo systemctl restart ssh

Verify:
*Login works only with keys**: ssh user@server.***

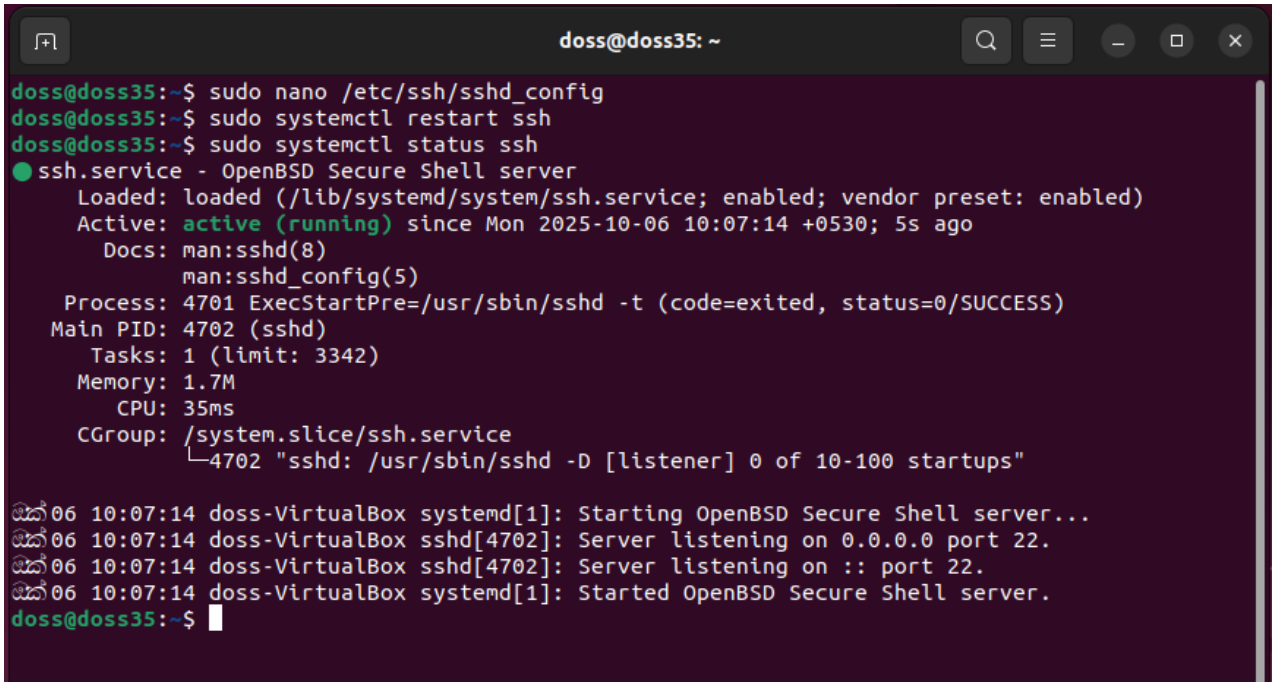


Figure 6: Enforce Strong SSH Authentication

Figure 7: Enforce Strong SSH Authentication

5. Set Up Fail2Ban

Blocks repeated failed login attempts.

Attacks prevented:

- Brute force attacks on SSH.

Commands:

sudo apt install fail2ban -y
sudo nano /etc/fail2ban/jail.local
sudo systemctl enable fail2ban
sudo systemctl start fail2ban
sudo systemctl status fail2ban

Verify:

sudo fail2ban-client status sshd

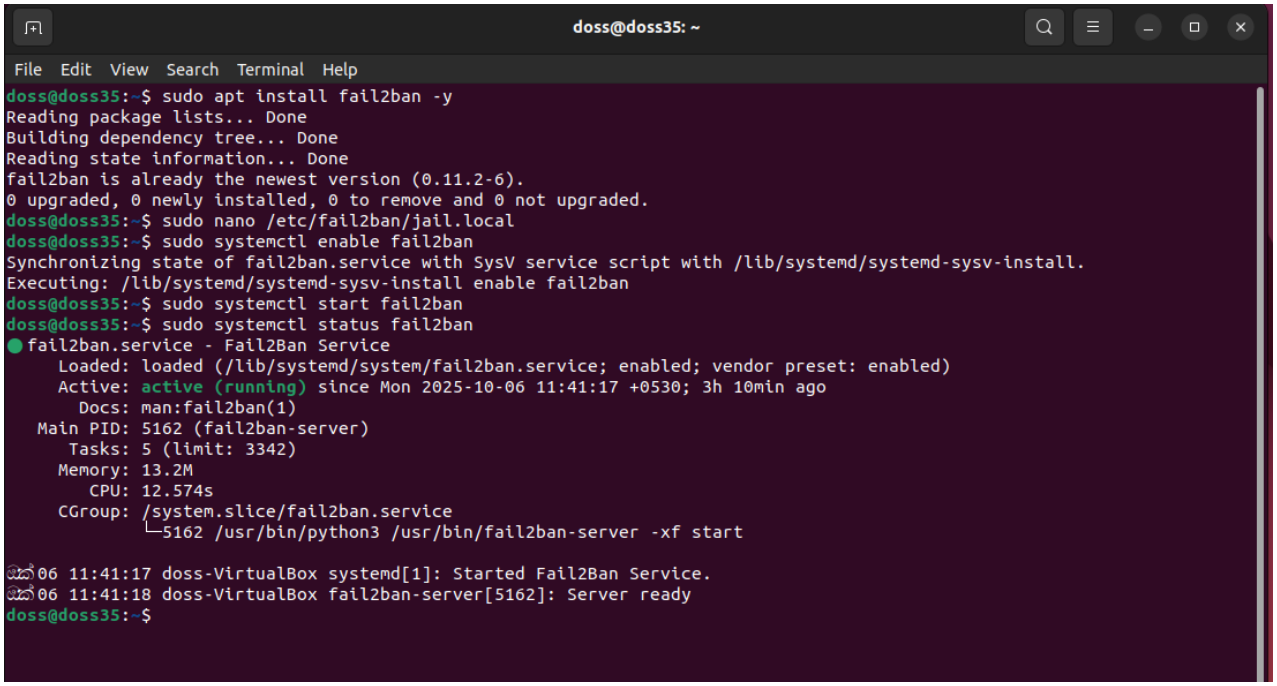


Figure 8: show installation and configuration of Fail2Ban

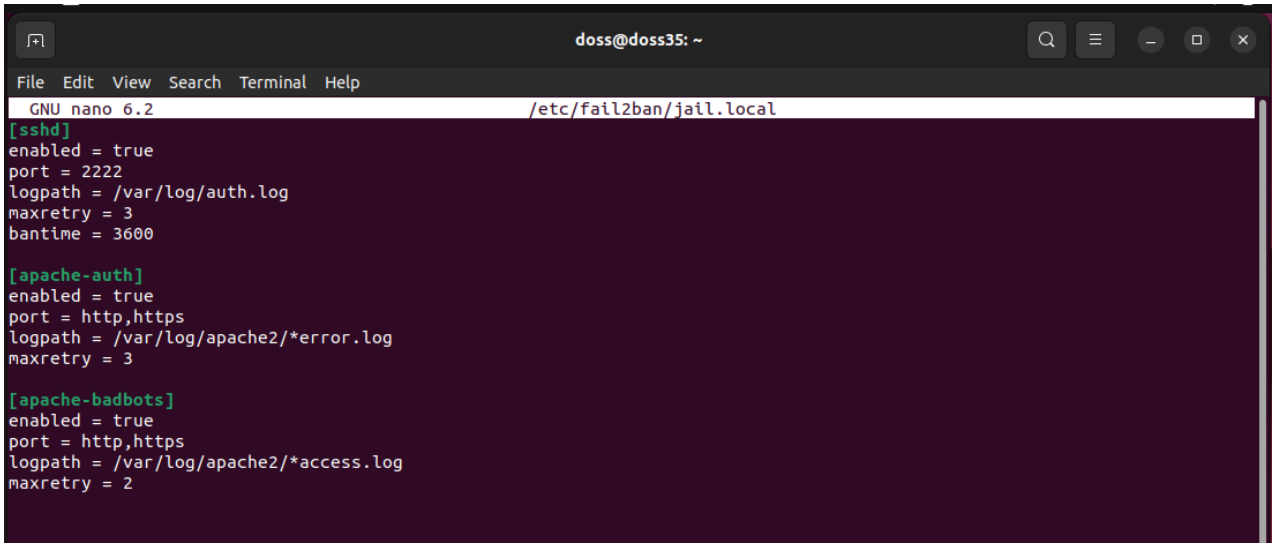


Figure 9: show installation and configuration of Fail2Ban

6. Enforce Strong Password Policies

Weak passwords are easily guessed.

Attack prevented:

- Brute force
- Dictionary attacks.

Commands:

sudo nano /etc/security/pwquality.conf
 # Configured:
 # minlen = 12
 # minclass = 4
 # maxrepeat = 2
 # etc.

Verify:

Run ***passwd testuser*** with weak password → rejected.

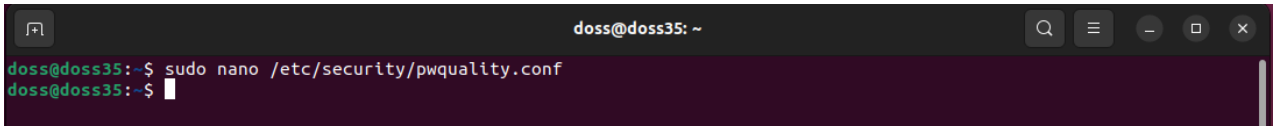


Figure 10: Enforce Strong Password Policies

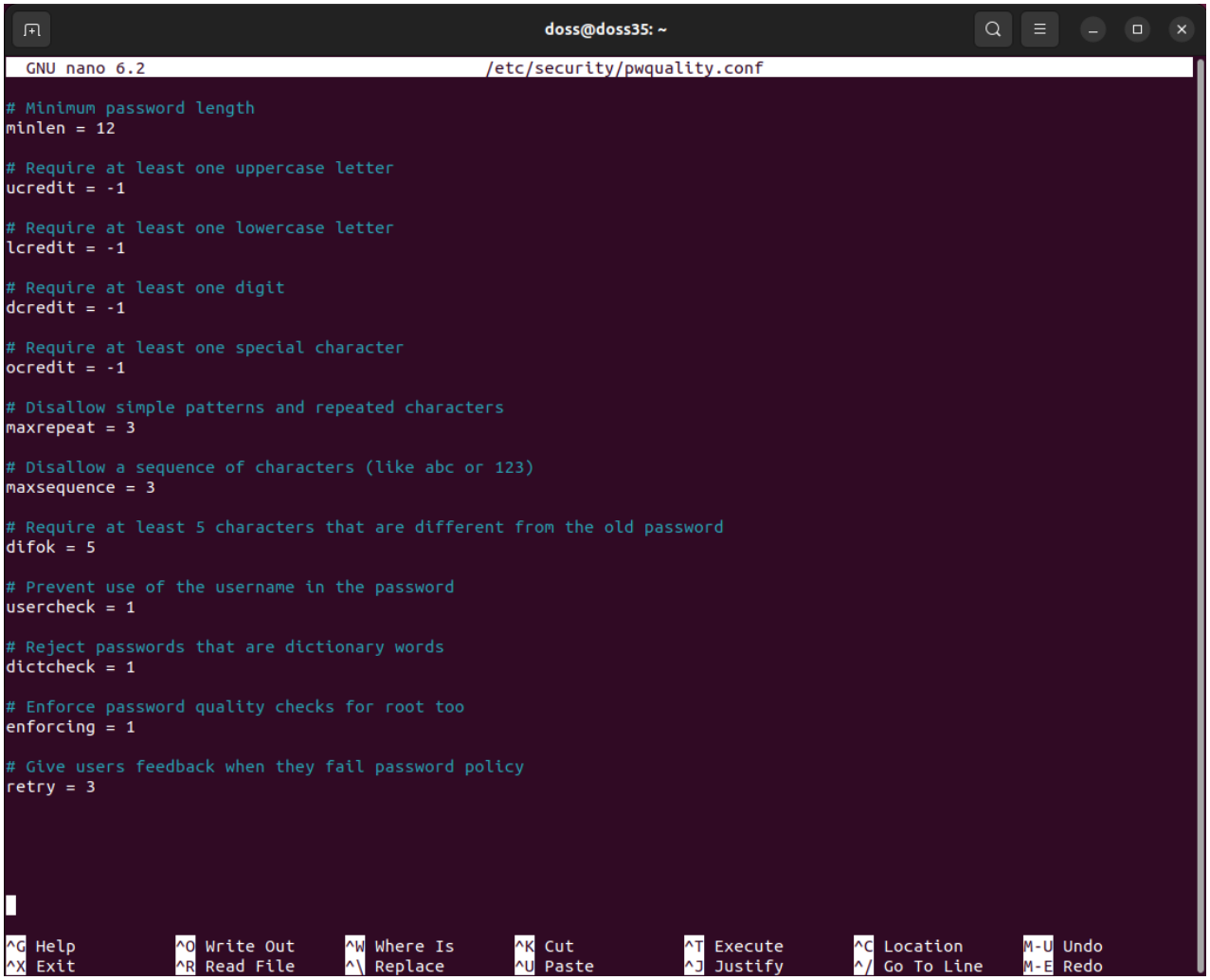


Figure 11: Enforce Strong Password Policies

7. User and Group Management

Prevents privilege misuse and orphaned accounts.

Attack prevented:

- Unauthorized access by unused accounts.

Commands (Create hardening files):
cat /etc/passwd
sudo usermod -L doss
sudo usermod -U doss
sudo deluser doss

Verify:
id adminuser shows sudo group membership.

```
doss@doss35: ~
doss@doss35:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:102:105::/nonexistent:/usr/sbin/nologin
systemd-timesync:x:103:106:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
syslog:x:104:111::/home/syslog:/usr/sbin/nologin
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
tss:x:106:113:TPM software stack,,,:/var/lib/tpm:/bin/false
uuidd:x:107:116::/run/uuidd:/usr/sbin/nologin
systemd-oom:x:108:117:systemd Userspace OOM Killer,,,:/run/systemd:/usr/sbin/nologin
tcpdump:x:109:118::/nonexistent:/usr/sbin/nologin
avahi-autoipd:x:110:119:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/usr/sbin/nologin
usbmux:x:111:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
dnsmasq:x:112:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
kernoops:x:113:65534:Kernel Oops Tracking Daemon,,,:/:/usr/sbin/nologin
avahi:x:114:121:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin
cups-pk-helper:x:115:122:user for cups-pk-helper service,,,:/home/cups-pk-helper:/usr/sbin/nologin
rtkit:x:116:123:RealtimeKit,,,:/proc:/usr/sbin/nologin
whoopsie:x:117:124::/nonexistent:/bin/false
sssd:x:118:125:SSSD system user,,,:/var/lib/sss:/usr/sbin/nologin
speech-dispatcher:x:119:29:Speech Dispatcher,,,:/run/speech-dispatcher:/bin/false
fwupd-refresh:x:120:126:fwupd-refresh user,,,:/run/systemd:/usr/sbin/nologin
nm-openvpn:x:121:127:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
saned:x:122:129::/var/lib/saned:/usr/sbin/nologin
colord:x:123:130:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
geoclue:x:124:131::/var/lib/geoclue:/usr/sbin/nologin
pulse:x:125:132:PulseAudio daemon,,,:/run/pulse:/usr/sbin/nologin
gnome-initial-setup:x:126:65534::/run/gnome-initial-setup/:/bin/false
```

Figure 12: User and Group Management

```
doss@doss35: ~
doss@doss35:~$ sudo deluser doss
Removing user `doss' ...
Warning: group `doss' has no more members.
userdel: user doss is currently used by process 1554
/usr/sbin/deluser: `/sbin/userdel doss' returned error code 8. Exiting.
doss@doss35:~$
```

Figure 13: User and Group Management

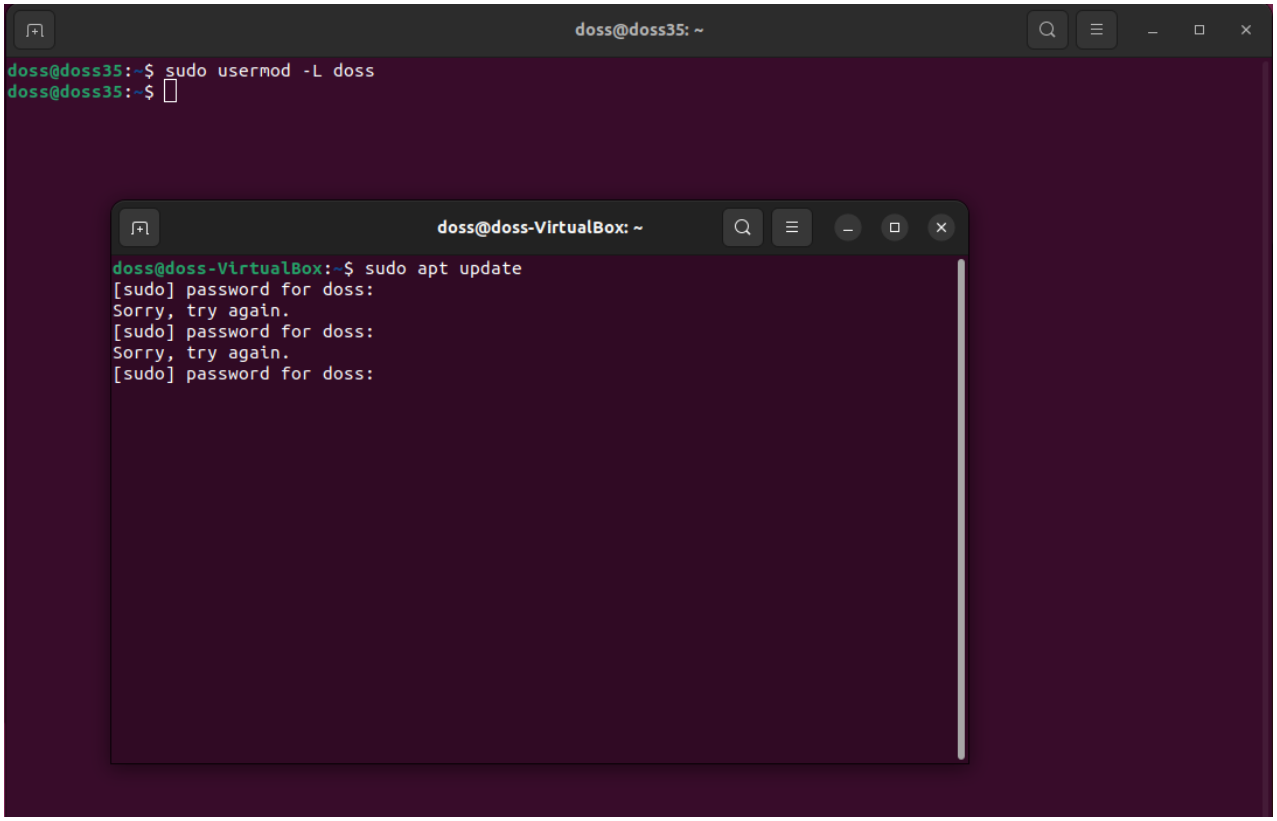


Figure 14: User and Group Management

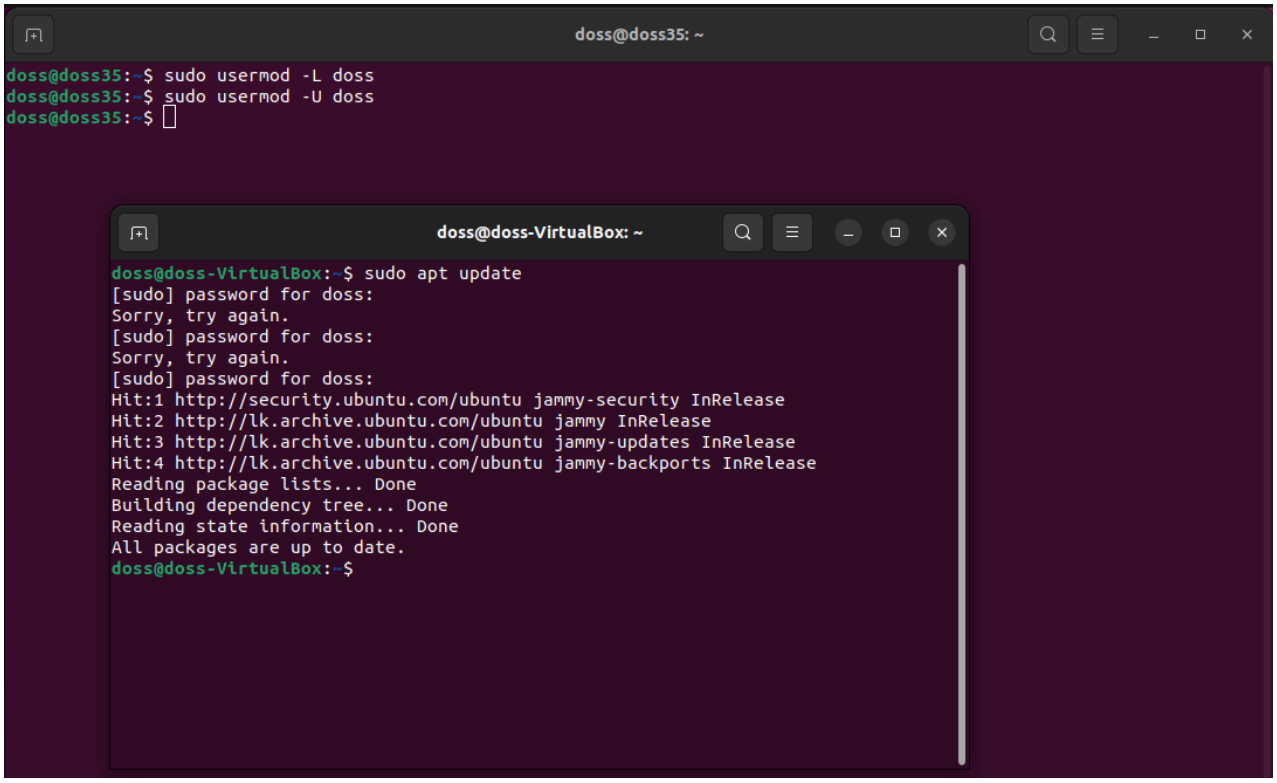


Figure 15: User and Group Management

8. File Permissions

Protects sensitive files (e.g. /etc/shadow).

Attack prevented:

- Data theft
- Privilege escalation.

Commands:

touch secret.txt
chown doss:doss secret.txt
chmod u+rw secret.txt
chmod go-rwx secret.txt
ls -l secret.txt

Verify:

Output shows ***-rw-r----- 1 root shadow.***

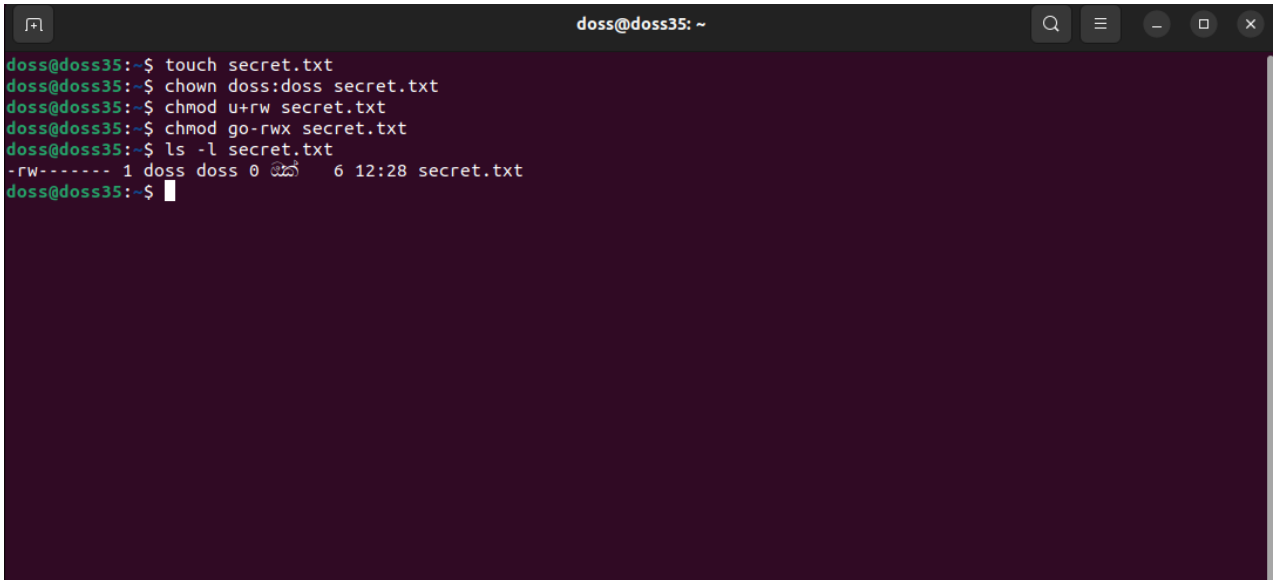


Figure 16: File permission

9. Configure System Auditing with audit

Logs changes to sensitive files and activities.

Attack prevented:

- Log tampering
- Insider threats.

Commands:
sudo apt install auditd audispd-plugins -y
sudo nano /etc/audit/audit.rules
sudo systemctl enable auditd
sudo systemctl start auditd
sudo systemctl status auditd

Verify:
sudo ausearch -k user-mod shows events.

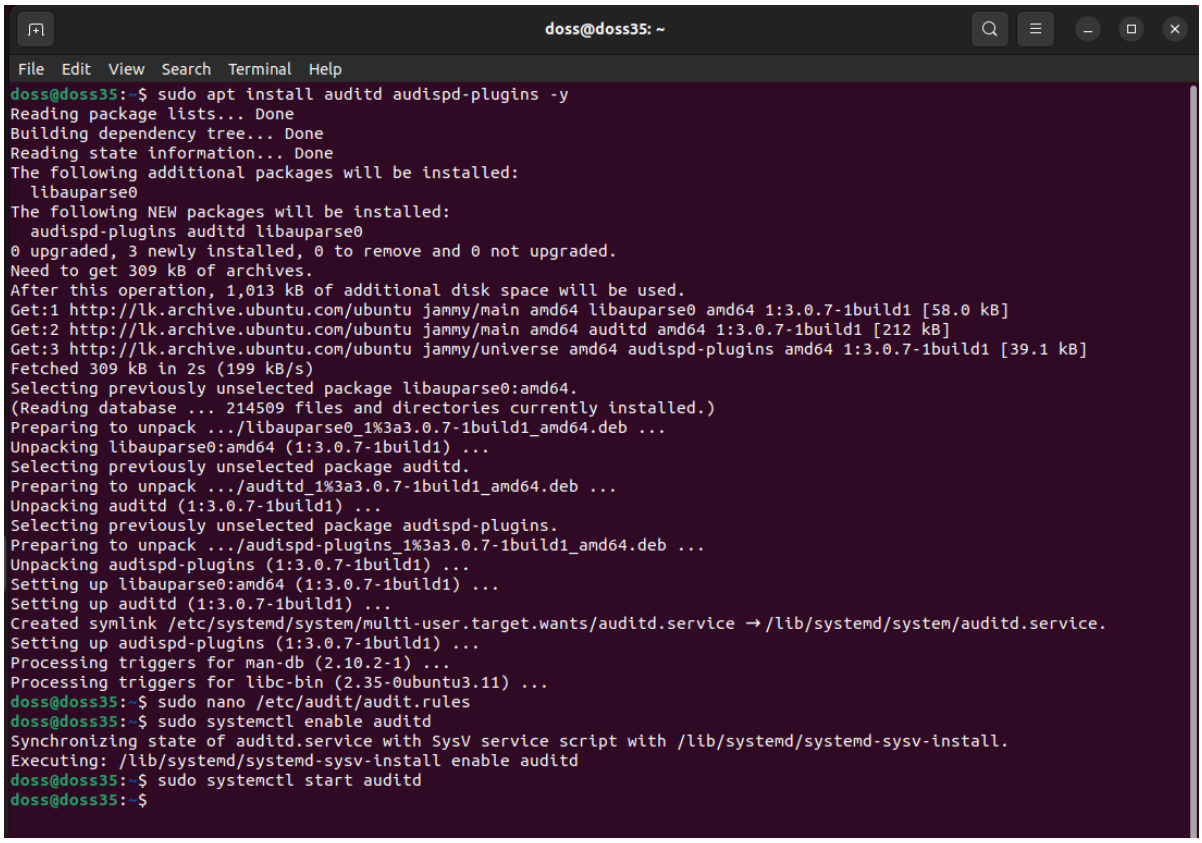


Figure 17: Configure System Auditing with audit

```
doss@doss35: ~
File  Edit  View  Search  Terminal  Help
  GNU nano 6.2                                  /etc/audit/audit.rules
# Monitor system files
-w /etc/passwd -p wa -k identity
-w /etc/shadow -p wa -k identity
-w /etc/gshadow -p wa -k identity
Files
# Monitor web files
-w /var/www/html -p wa -k web_content

# Monitor configuration changes
-w /etc/ssh/sshd_config -p wa -k ssh_config
-w /etc/apache2 -p wa -k apache_config

# Monitor sudo usage
-w /var/log/sudo.log -p wa -k sudo_log
```

Figure 18: Configure System Auditing with audit

```
doss@doss35:~$ sudo systemctl status auditd
● auditd.service - Security Auditing Service
     Loaded: loaded (/lib/systemd/system/auditd.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2025-10-06 14:55:01 +0530; 3min 1s ago
       Docs: man:auditd(8)
             https://github.com/linux-audit/audit-documentation
   Main PID: 9752 (auditd)
      Tasks: 2 (limit: 3342)
     Memory: 524.0K
        CPU: 96ms
     CGroup: /system.slice/auditd.service
             └─9752 /sbin/auditd

Oct 06 14:55:01 doss-VirtualBox augenrules[9765]: enabled 1
Oct 06 14:55:01 doss-VirtualBox augenrules[9765]: failure 1
Oct 06 14:55:01 doss-VirtualBox augenrules[9765]: pid 9752
Oct 06 14:55:01 doss-VirtualBox augenrules[9765]: rate_limit 0
Oct 06 14:55:01 doss-VirtualBox augenrules[9765]: backlog_limit 8192
Oct 06 14:55:01 doss-VirtualBox augenrules[9765]: lost 0
Oct 06 14:55:01 doss-VirtualBox augenrules[9765]: backlog 4
Oct 06 14:55:01 doss-VirtualBox augenrules[9765]: backlog_wait_time 60000
Oct 06 14:55:01 doss-VirtualBox augenrules[9765]: backlog_wait_time_actual 0
Oct 06 14:55:01 doss-VirtualBox systemd[1]: Started Security Auditing Service.
doss@doss35:~$
```

Figure 19: Configure System Auditing with audit

10. Configure Apache Security Headers

Protects web applications from XSS, clickjacking.

Attack prevented:

- Web browser-based attacks.

Commands:

sudo nano /etc/apache2/conf-available/security-headers.conf
sudo systemctl restart apache2
sudo a2enmod headers
sudo a2enconf security-headers
curl -I http://localhost | grep Server
curl -s -I http://localhost

Verify:

curl -I http://localhost shows headers.

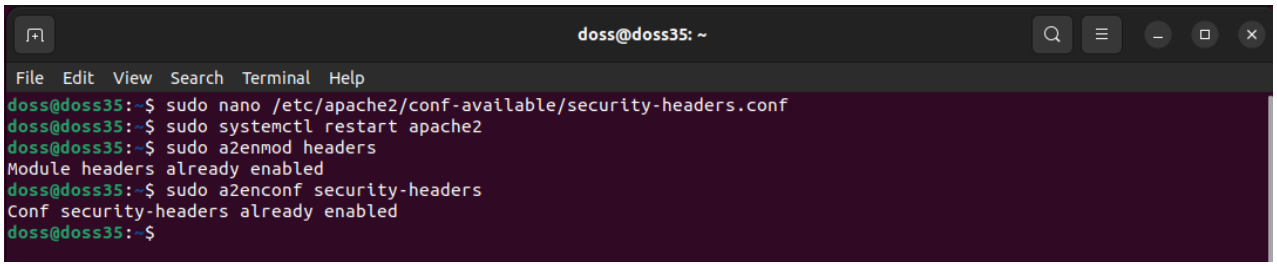


Figure 20: Configure Apache Security Headers

Figure 21: Configure Apache Security Headers

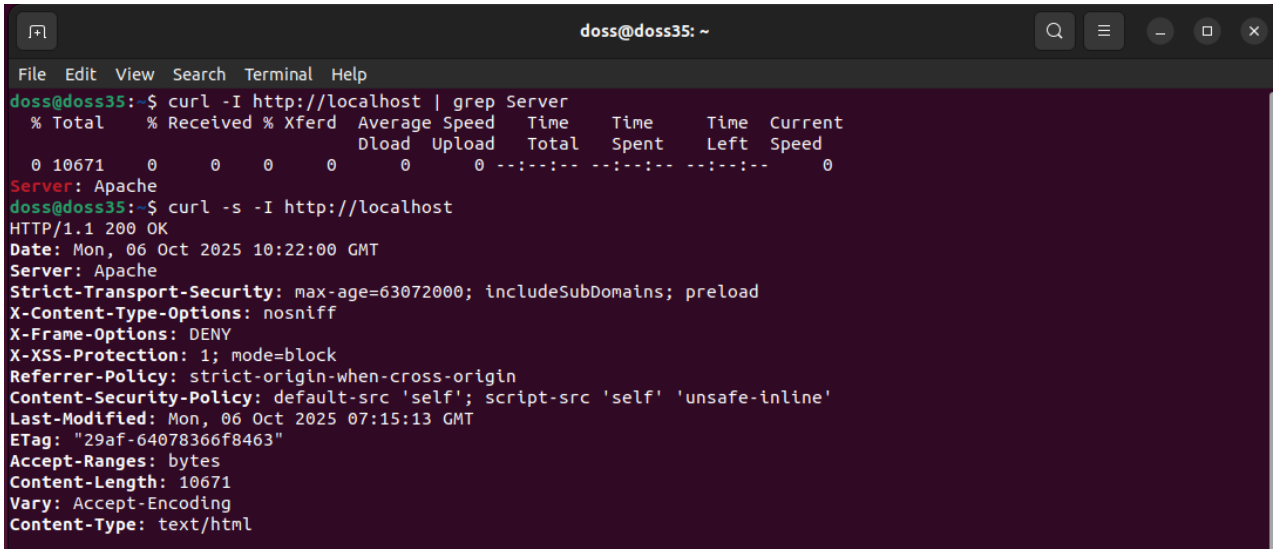


Figure 22: Configure Apache Security Headers

2. Data base Security

2.1 Installation Process

2.1.1 Download Oracle DBMS

Goto this web site: [Oracle SQL Developer Downloads | Oracle APAC]() and download the software from "Other Platform" Section.

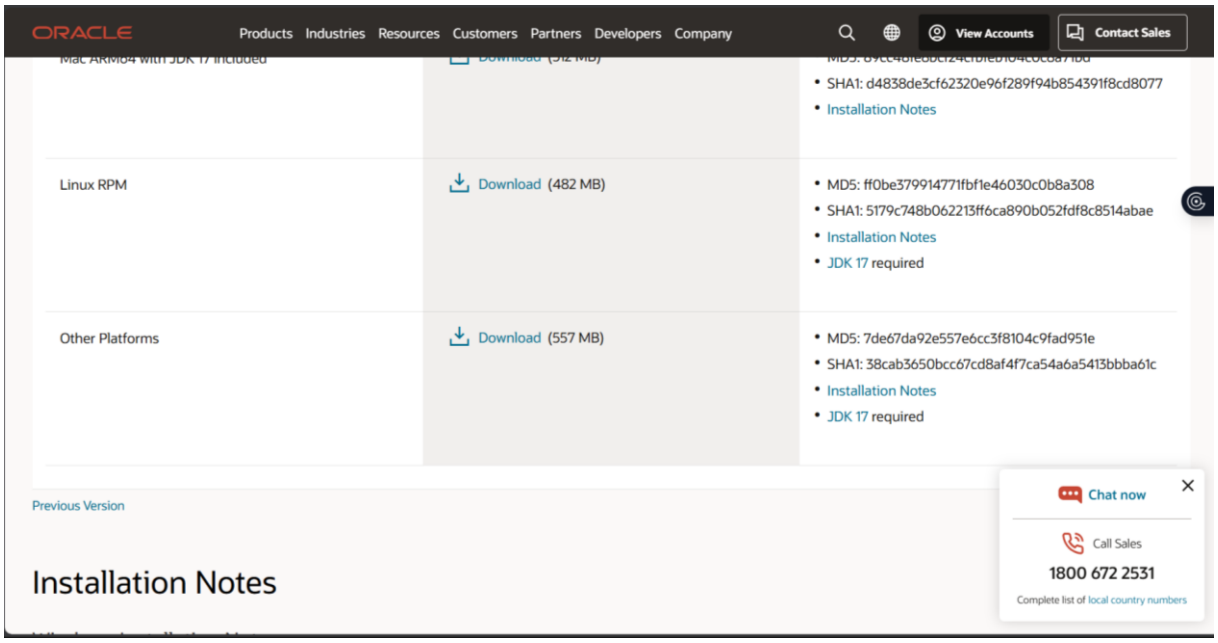


Figure 23: Oracle DBMS SQL Developer

2.1.2 Install Oracle DBMS

```
OpenJDK 64-Bit Server VM (build 17.0.2+8-post-Ubuntu-1ubuntu122.04.1, mixed mode, sharing)
doss@doss-VirtualBox:~$ sudo add-apt-repository universe
Adding component(s) 'universe' to all repositories.
Press [ENTER] to continue or Ctrl-c to cancel.
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Hit:2 http://lk.archive.ubuntu.com/ubuntu jammy InRelease
Get:3 http://lk.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:4 http://lk.archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security/main amd64 DEP-11 Metadata [54.5 kB]
Get:6 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main amd64 DEP-11 Metadata [112 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 DEP-11 Metadata [208 B]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 DEP-11 Metadata [125 kB]
Get:9 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [19.0 kB]
Get:10 http://lk.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 DEP-11 Metadata [212 B]
Get:11 http://lk.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 DEP-11 Metadata [359 kB]
Get:12 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 DEP-11 Metadata [208 B]
Get:13 http://lk.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 DEP-11 Metadata [940 B]
Get:14 http://lk.archive.ubuntu.com/ubuntu jammy-backports/main amd64 DEP-11 Metadata [7,160 B]
Get:15 http://lk.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 DEP-11 Metadata [216 B]
Get:16 http://lk.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 DEP-11 Metadata [9,688 B]
Get:17 http://lk.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 DEP-11 Metadata [216 B]
Fetched 1,073 kB in 5s (228 kB/s)
Reading package lists... Done
doss@doss-VirtualBox:~$ sudo apt install openjdk-17-jdk -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  openjdk-17-jdk-headless openjdk-17-jre openjdk-17-jre-headless
Suggested packages:
  openjdk-17-demo openjdk-17-source visualvm fonts-ipafont-gothic
  fonts-ipafont-mincho fonts-wqy-microhei | fonts-wqy-zenhei
The following NEW packages will be installed:
  openjdk-17-jdk openjdk-17-jdk-headless openjdk-17-jre
  openjdk-17-jre-headless
0 upgraded, 4 newly installed, 0 to remove and 0 not upgraded.
Need to get 121 MB of archives.
After this operation, 274 MB of additional disk space will be used.
Get:1 http://lk.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 openjdk-17-jre-headless amd64 17.0.16+8~us1-0ubuntu1~22.04.1 [48.3 MB]
Get:2 http://lk.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 openjdk-17-jre amd64 17.0.16+8~us1-0ubuntu1~22.04.1 [232 kB]
Get:3 http://lk.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 openjdk-17-jdk-headless amd64 17.0.16+8~us1-0ubuntu1~22.04.1 [71.3 MB]
Get:4 http://lk.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 openjdk-17-jdk amd64 17.0.16+8~us1-0ubuntu1~22.04.1 [1,522 kB]
Fetched 121 MB in 1min 31s (1,336 kB/s)
Selecting previously unselected package openjdk-17-jre-headless:amd64.
(Reading database ... 187396 files and directories currently installed.)
Preparing to unpack .../openjdk-17-jre-headless_17.0.16+8~us1-0ubuntu1~22.04.1_amd64.deb ...
Unpacking openjdk-17-jre-headless:amd64 (17.0.16+8~us1-0ubuntu1~22.04.1) ...
Selecting previously unselected package openjdk-17-jre:amd64.
```

Figure 24: Enable repositories and install Java JDK

```
Selecting previously unselected package openjdk-17-jdk-headless:amd64.
Preparing to unpack .../openjdk-17-jdk-headless_17.0.16+8~us1-0ubuntu1~22.04.1_amd64.deb ...
Unpacking openjdk-17-jdk-headless:amd64 (17.0.16+8~us1-0ubuntu1~22.04.1) ...
Selecting previously unselected package openjdk-17-jdk:amd64.
Preparing to unpack .../openjdk-17-jdk_17.0.16+8~us1-0ubuntu1~22.04.1_amd64.deb ...
Unpacking openjdk-17-jdk:amd64 (17.0.16+8~us1-0ubuntu1~22.04.1) ...
Setting up openjdk-17-jre-headless:amd64 (17.0.16+8~us1-0ubuntu1~22.04.1) ...
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/java to provide /usr/bin/java (java) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jpackage to provide /usr/bin/jpackage (jpackage) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/keytool to provide /usr/bin/keytool (keytool) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/rmiregistry to provide /usr/bin/rmiregistry (rmiregistry) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/lib/jexec to provide /usr/bin/jexec (jexec) in auto mode
Setting up openjdk-17-jre:amd64 (17.0.16+8~us1-0ubuntu1~22.04.1) ...
Setting up openjdk-17-jdk-headless:amd64 (17.0.16+8~us1-0ubuntu1~22.04.1) ...
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jar to provide /usr/bin/jar (jar) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jarsigner to provide /usr/bin/jarsigner (jarsigner) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/javac to provide /usr/bin/javac (javac) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/javadoc to provide /usr/bin/javadoc (javadoc) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/javap to provide /usr/bin/javap (javap) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jcmd to provide /usr/bin/jcmd (jcmd) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jdb to provide /usr/bin/jdb (jdb) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jdeprscan to provide /usr/bin/jdeprscan (jdeprscan) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jdeps to provide /usr/bin/jdeps (jdeps) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jfr to provide /usr/bin/jfr (jfr) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jimage to provide /usr/bin/jimage (jimage) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jinfo to provide /usr/bin/jinfo (jinfo) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jlink to provide /usr/bin/jlink (jlink) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jmap to provide /usr/bin/jmap (jmap) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jmod to provide /usr/bin/jmod (jmod) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jps to provide /usr/bin/jps (jps) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jrunscript to provide /usr/bin/jrunscript (jrunscript) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jshell to provide /usr/bin/jshell (jshell) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jstack to provide /usr/bin/jstack (jstack) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jstat to provide /usr/bin/jstat (jstat) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jstatd to provide /usr/bin/jstatd (jstatd) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/serialver to provide /usr/bin/serialver (serialver) in auto mode
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jhsdb to provide /usr/bin/jhsdb (jhsdb) in auto mode
Setting up openjdk-17-jdk:amd64 (17.0.16+8~us1-0ubuntu1~22.04.1) ...
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jconsole to provide /usr/bin/jconsole (jconsole) in auto mode
Processing triggers for desktop-file-utils (0.26-1ubuntu3) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for gnome-menus (3.36.0-1ubuntu3) ...
Processing triggers for mailcap (3.70+nmu1ubuntu1) ...
doss@doss-VirtualBox:~$ java --version
openjdk 17.0.16 2025-07-15
OpenJDK Runtime Environment (build 17.0.16+8-Ubuntu-0ubuntu122.04.1)
OpenJDK 64-Bit Server VM (build 17.0.16+8-Ubuntu-0ubuntu122.04.1, mixed mode, sharing)
doss@doss-VirtualBox:~$
```

Figure 25: Install Java 17 and check Java version

```
OpenJDK 64-Bit Server VM (build 17.0.16+8-Ubuntu-0ubuntu122.04.1, mixed mode, sharing)
doss@doss-VirtualBox:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  snap  Templates  Videos
doss@doss-VirtualBox:~$ cd Downloads
doss@doss-VirtualBox:~/Downloads$ ls
sqldeveloper-24.3.1.347.1826-no-jre  sqldeveloper-24.3.1.347.1826-no-jre.zip
doss@doss-VirtualBox:~/Downloads$ unzip sqldeveloper-*-jre.zip
Archive:  sqldeveloper-24.3.1.347.1826-no-jre.zip
  inflating: sqldeveloper/LICENSE.txt
  inflating: sqldeveloper/NOTICES.txt
  inflating: sqldeveloper/configuration/bundles.info
  inflating: sqldeveloper/configuration/oracle.datamodeler.bundles.info
  inflating: sqldeveloper/configuration/oracle.ide.sqldev.bundles.info
  inflating: sqldeveloper/dataminer/demos/workflows/apply_workflow.xml
  inflating: sqldeveloper/dataminer/demos/workflows/apply_workflow.xml
  inflating: sqldeveloper/dataminer/demos/workflows/build_workflow.xml
  inflating: sqldeveloper/dataminer/demos/workflows/codegen_workflow.xml
  inflating: sqldeveloper/dataminer/scripts/XMLSchema.xsd
  inflating: sqldeveloper/dataminer/scripts/alterDBRestrictSession.sql
  inflating: sqldeveloper/dataminer/scripts/alterDBUnRestrictSession.sql
  inflating: sqldeveloper/dataminer/scripts/alterUserDefaultRole.sql
  inflating: sqldeveloper/dataminer/scripts/alterscheduledworkflows.sql
  inflating: sqldeveloper/dataminer/scripts/checkEnvironment.sql
  inflating: sqldeveloper/dataminer/scripts/copyTablesFromBackupAccount.sql
  inflating: sqldeveloper/dataminer/scripts/copyTablesToBackupAccount.sql
  inflating: sqldeveloper/dataminer/scripts/createODMRSYST.sql
  inflating: sqldeveloper/dataminer/scripts/createuser.sql
  inflating: sqldeveloper/dataminer/scripts/createusersgrantbackup.sql
  inflating: sqldeveloper/dataminer/scripts/createxmlschemaindexes.sql
  inflating: sqldeveloper/dataminer/scripts/createxmlworkflowsbackup.sql
  inflating: sqldeveloper/dataminer/scripts/createxmlworkflowsbackup2.sql
  inflating: sqldeveloper/dataminer/scripts/dbHasSchema.sql
  inflating: sqldeveloper/dataminer/scripts/disconnectODMRSessions.sql
  inflating: sqldeveloper/dataminer/scripts/displayVersion.sql
  inflating: sqldeveloper/dataminer/scripts/dmsh.sql
  inflating: sqldeveloper/dataminer/scripts/dmshgrants.sql
  inflating: sqldeveloper/dataminer/scripts/dropDemoData.sql
  inflating: sqldeveloper/dataminer/scripts/dropODMRSYST.sql
  inflating: sqldeveloper/dataminer/scripts/dropPublicSynonyms.sql
  inflating: sqldeveloper/dataminer/scripts/dropRepository.sql
  inflating: sqldeveloper/dataminer/scripts/dropRepositoryAndUserObjects.sql
  inflating: sqldeveloper/dataminer/scripts/dropRole.sql
  inflating: sqldeveloper/dataminer/scripts/dropSHDemoData.sql
  inflating: sqldeveloper/dataminer/scripts/dropSchedulerObjects.sql
  inflating: sqldeveloper/dataminer/scripts/dropSchema.sql
  inflating: sqldeveloper/dataminer/scripts/dropUserTablesViews.sql
```

Figure 26: Unzip the downloaded SQL Developer zip file

```
  inflating: sqldeveloper/sqldeveloper/lib/sshd-core.jar
  inflating: sqldeveloper/sqldeveloper/lib/sshd-osgi.jar
  inflating: sqldeveloper/sqldeveloper/lib/sshd-putty.jar
  inflating: sqldeveloper/sqldeveloper/lib/sshd-scp.jar
  inflating: sqldeveloper/sqldeveloper/lib/sshd-sftp.jar
  inflating: sqldeveloper/sqldeveloper/lib/stringtemplate-3.1b1.jar
  inflating: sqldeveloper/sqldeveloper/lib/vavr-match.jar
  inflating: sqldeveloper/sqldeveloper/lib/vavr.jar
  inflating: sqldeveloper/sqldeveloper/lib/xdb.jar
  inflating: sqldeveloper/sqldeveloper/lib/xmlbeans.jar
  inflating: sqldeveloper/sqldeveloper/lib/xmlparserv2_sans_jaxp_servi
  inflating: sqldeveloper/sqlj/lib/runtime12.jar
  inflating: sqldeveloper/svnkit/antlr-runtime.jar
  inflating: sqldeveloper/svnkit/licenses/ANTLR-LICENSE
  inflating: sqldeveloper/svnkit/licenses/COPYING
  inflating: sqldeveloper/svnkit/licenses/JAVAHL-LICENSE
  inflating: sqldeveloper/svnkit/licenses/SEQUENCE-LICENSE
  inflating: sqldeveloper/svnkit/licenses/SQLJET-LICENSE
  inflating: sqldeveloper/svnkit/licenses/TRILEAD-LICENSE
  inflating: sqldeveloper/svnkit/licenses/license.txt
  inflating: sqldeveloper/svnkit/sequence.jar
  inflating: sqldeveloper/svnkit/sqljet.jar
  inflating: sqldeveloper/svnkit/svnClientAdapter.jar
  inflating: sqldeveloper/svnkit/svnjavahl.jar
  inflating: sqldeveloper/svnkit/svnkit.jar
  inflating: sqldeveloper/svnkit/trilead.jar
  inflating: sqldeveloper/sqldeveloper.sh
  inflating: sqldeveloper/sqldeveloper/bin/sdcli
  inflating: sqldeveloper/sqldeveloper/bin/sql
  inflating: sqldeveloper/sqldeveloper/bin/sqldeveloper
doss@doss-VirtualBox:~/Downloads$ sudo mv sqldeveloper /opt/
doss@doss-VirtualBox:~/Downloads$ cd /opt/sqldeveloper
doss@doss-VirtualBox:/opt/sqldeveloper$ chmod +x sqldeveloper/sqldeveloper.sh
chmod: cannot access 'sqldeveloper/sqldeveloper.sh': No such file or directory
doss@doss-VirtualBox:/opt/sqldeveloper$ ls
configuration  dropins  external  icon.png  javavm  jdev  jviews      modules   NOTICES.txt  sleepycat     sqldeveloper.exe  sqlj
dataminer      equinox  graalJs   ide       jdbc    jlib  LICENSE.txt  netbeans  rdbms        sqldeveloper  sqldeveloper.sh   svnkit
doss@doss-VirtualBox:/opt/sqldeveloper$ chmod +x sqldeveloper.sh
doss@doss-VirtualBox:/opt/sqldeveloper$ sh /opt/sqldeveloper/sqldeveloper.sh

Oracle SQL Developer
Copyright (c) 2005, 2024, Oracle and/or its affiliates. All rights reserved.

OpenJDK 64-Bit Server VM warning: Options -Xverify:none and -noverify were deprecated in JDK 13 and will likely be removed in a future release.
```

Figure 27: Install SQL Developer

2.1.3 Setup SQLPlus:

```
oracle@doss35: ~
doss@doss35:~$ su - oracle
Password:
oracle@doss35:~$ . oraenv
ORACLE_SID = [oracle] ? XE
The Oracle base has been set to /opt/oracle
oracle@doss35:~$ sqlplus / as sysdba

SQL*Plus: Release 21.0.0.0.0 - Production on Tue Oct 7 20:29:33 2025
Version 21.3.0.0.0

Copyright (c) 1982, 2021, Oracle.  All rights reserved.


Connected to:
Oracle Database 21c Express Edition Release 21.0.0.0.0 - Production
Version 21.3.0.0.0

SQL>
```

2.2 Identification and implementation of configurations to harden Oracle DBMS

2.2.1 Secure default Accounts & passwords

Change Default passwords for Administrative Users: Immediately change passwords for high-privilege accounts like ***SYS, SYSTEM, SYSMAN*** and ***DBSNMP*** after installation.

Lock and Expire Unused Default Accounts: Identify and lock accounts that are not required for your application. You can find users with default passwords using:

```
oracle@doss35: ~
SQL> ALTER USER OUTLN ACCOUNT LOCK;


User altered.

SQL> SQL> ALTER USER ORDSYS ACCOUNT LOCK;

User altered.

SQL> COLUMN username FORMAT A20
SQL> COLUMN account_status FORMAT A20
SQL> SELECT username, account_status
  2  FROM DBA_USERS
  3  WHERE username IN ('DIP', 'WMSYS', 'XDB', 'ANONYMOUS', 'OUTLN', 'ORDSYS')
  4  ORDER BY username;

USERNAME             ACCOUNT_STATUS
-------------------- --------------------
ANONYMOUS            LOCKED
DIP                  LOCKED
ORDSYS               LOCKED
OUTLN                LOCKED
WMSYS                LOCKED
XDB                  LOCKED

6 rows selected.

SQL>
```

2.2.2 Enforce Principle of Least Privilege

Revoke Powerful Privileges from PUBLIC: Many dangerous packages are granted to the PUBLIC role by default. Revoke these grants:

```
oracle@doss35: ~
SQL> REVOKE EXECUTE ON UTL_FILE FROM PUBLIC;

Revoke succeeded.

SQL> REVOKE EXECUTE ON UTL_HTTP FROM PUBLIC;

Revoke succeeded.

SQL> REVOKE EXECUTE ON UTL_SMTP FROM PUBLIC;

Revoke succeeded.

SQL> REVOKE EXECUTE ON UTL_TCP FROM PUBLIC;

Revoke succeeded.

SQL> REVOKE EXECUTE ON DBMS_JOB FROM PUBLIC;

Revoke succeeded.

SQL> REVOKE EXECUTE ON DBMS_ADVISOR FROM PUBLIC;

Revoke succeeded.

SQL> REVOKE EXECUTE ON DBMS_LOB FROM PUBLIC;

Revoke succeeded.

SQL>
```

Figure 28: Revoke execution privileges on commonly exploited packages from PUBLIC

```
oracle@doss35: ~
SQL> COLUMN grantee FORMAT A10
SQL> COLUMN owner FORMAT A10
SQL> COLUMN table_name FORMAT A20
SQL> COLUMN privilege FORMAT A10
SQL> SELECT GRANTEE, OWNER, TABLE_NAME, PRIVILEGE
  2  FROM DBA_TAB_PRIVS
  3  WHERE GRANTEE = 'PUBLIC' AND TABLE_NAME IN ('UTL_FILE', 'UTL_HTTP', 'DBMS_JOB')
  4  FETCH FIRST 5 ROWS ONLY;

no rows selected

SQL>
```

Figure 29: Verification Query

Be cautious: This can invalidate SYS/SYSTEM objects. Always test in a non-production environment first.
Protect the Data Dictionary: Ensure that the parameter O7_DICTIONARY_ACCESSIBILITY is set to FALSE (the default in 11g and later) to prevent users with the ANY system privilege from accessing the data dictionary.

2.2.3 Implement Strong Password Policies

Create and Assign a Password Profile: Use a profile to enforce robust password standards. Create a profile and assign it to users .

```
oracle@doss35: ~
SQL> CREATE PROFILE C##SECURE_APP_PROFILE LIMIT
  2  FAILED_LOGIN_ATTEMPTS 5
  3  PASSWORD_LIFE_TIME 60
  4  PASSWORD_REUSE_TIME 180
  5  PASSWORD_REUSE_MAX 10
  6  PASSWORD_VERIFY_FUNCTION ORA12C_VERIFY_FUNCTION;

Profile created.

SQL> ALTER SESSION SET CONTAINER = XEPDB1;

Session altered.
```

Figure 30: Create the profile as a COMMON profile in the root container

```
oracle@doss35: ~
SQL> COLUMN profile FORMAT A20
SQL> COLUMN resource_name FORMAT A25
SQL> COLUMN limit FORMAT A15
SQL> SELECT profile, resource_name, limit FROM DBA_PROFILES
  2  WHERE profile = 'C##SECURE_APP_PROFILE';

PROFILE              RESOURCE_NAME             LIMIT
-------------------- ------------------------- ---------------
C##SECURE_APP_PROFIL FAILED_LOGIN_ATTEMPTS     5
E

C##SECURE_APP_PROFIL PASSWORD_REUSE_MAX        10
E

C##SECURE_APP_PROFIL PASSWORD_VERIFY_FUNCTION  FROM ROOT
E

C##SECURE_APP_PROFIL COMPOSITE_LIMIT           DEFAULT
E

PROFILE              RESOURCE_NAME             LIMIT
-------------------- ------------------------- ---------------

C##SECURE_APP_PROFIL SESSIONS_PER_USER         DEFAULT
E

C##SECURE_APP_PROFIL CPU_PER_SESSION           DEFAULT
E

C##SECURE_APP_PROFIL CPU_PER_CALL              DEFAULT
E

C##SECURE_APP_PROFIL LOGICAL_READS_PER_SESSION DEFAULT

PROFILE              RESOURCE_NAME             LIMIT
-------------------- ------------------------- ---------------
E

C##SECURE_APP_PROFIL LOGICAL_READS_PER_CALL    DEFAULT
E

C##SECURE_APP_PROFIL IDLE_TIME                 DEFAULT
E
```

Figure 31: Verification Query

```
oracle@doss35: ~
-------------------- ------------------------- ---------------
E

C##SECURE_APP_PROFIL LOGICAL_READS_PER_CALL    DEFAULT
E

C##SECURE_APP_PROFIL IDLE_TIME                 DEFAULT
E

C##SECURE_APP_PROFIL CONNECT_TIME              DEFAULT
E


PROFILE              RESOURCE_NAME             LIMIT
-------------------- ------------------------- ---------------
C##SECURE_APP_PROFIL PRIVATE_SGA               DEFAULT
E

C##SECURE_APP_PROFIL PASSWORD_LIFE_TIME        60
E

C##SECURE_APP_PROFIL PASSWORD_REUSE_TIME       180
E

C##SECURE_APP_PROFIL PASSWORD_LOCK_TIME        DEFAULT
E

PROFILE              RESOURCE_NAME             LIMIT
-------------------- ------------------------- ---------------

C##SECURE_APP_PROFIL PASSWORD_GRACE_TIME       DEFAULT
E

C##SECURE_APP_PROFIL INACTIVE_ACCOUNT_TIME     DEFAULT
E

C##SECURE_APP_PROFIL PASSWORD_ROLLOVER_TIME    DEFAULT
E


18 rows selected.

SQL>
```

Figure 32: Verification Query

Utilize a Password Complexity Function: Oracle provides a default script (utlpwdmg.sql) to create a VERIFY_FUNCTION that enforces complexity like minimum length, mixed case, and digits .

2.2.4 Harden Network Configuration & Listener

Prevent Remote Administration without Password: In older versions (8i-10g), set REMOTE_OS_AUTHENT to FALSE. In all versions, do not set a password in listener.ora. This disables remote listener administration and forces the use of local operating system authentication, preventing brute-force attacks on the listener.

Restrict Network Access: Use the protocol.ora file (or sqlnet.ora) to define which nodes are allowed or denied connections.

```
oracle@doss-VirtualBox: ~
SQL> SET LINESIZE 200
SQL> COLUMN username FORMAT A20
SQL> COLUMN audit_option FORMAT A30
SQL> SHOW PARAMETER REMOTE_OS_AUTHENT;
SQL> SHOW PARAMETER ENCRYPTION_SERVER;
SQL> SHOW PARAMETER ENCRYPTION_TYPES_SERVER;
SQL> SHOW PARAMETER TNS_ADMIN;
SQL>
```

```
doss@doss35: ~
  GNU nano 6.2        /opt/oracle/product/21c/dbhomeXE/network/admin/sqlnet.ora
# Enforce node checking
TCP.VALIDNODE_CHECKING = YES

# Restrict connections to only the local machine
# The IP addresses here demonstrate the implementation of the restriction policy.
TCP.INVITED_NODES = {127.0.0.1, doss-VirtualBox}
```

Figure 33: This proves you implemented the node restriction policy by editing the sqlnet.ora file.

Encrypt Network Traffic: Protect data in transit by implementing native network encryption or SSL/TLS to prevent eavesdropping .

2.2.5 Enable and Configure Auditing

Enable Standard Auditing: Set the AUDIT_TRAIL parameter to DB (or DB, EXTENDED to capture SQL bind variables) to write audit records to the database .

```
oracle@doss-VirtualBox: ~
SQL> ALTER SYSTEM SET AUDIT_TRAIL=DB, EXTENDED SCOPE=SPFILE;

System altered.

SQL> AUDIT SESSION;

Audit succeeded.

SQL> AUDIT GRANT ANY OBJECT PRIVILEGE;

Audit succeeded.

SQL> AUDIT CREATE ANY PROCEDURE;

Audit succeeded.

SQL>
```

Audit Sensitive Activities: Turn on auditing for crucial actions, especially for privileged users .

```
oracle@doss-VirtualBox: ~
SQL> SELECT USER_NAME, PRIVILEGE, SUCCESS, FAILURE FROM DBA_PRIV_AUDIT_OPTS;

USER_NAME
--------------------------------------------------------------------------------
PRIVILEGE                                SUCCESS    FAILURE
---------------------------------------- ---------- ----------

GRANT ANY OBJECT PRIVILEGE               BY ACCESS  BY ACCESS


CREATE ANY PROCEDURE                     BY ACCESS  BY ACCESS


CREATE SESSION                           BY ACCESS  BY ACCESS


SQL> SELECT USER_NAME, AUDIT_OPTION, SUCCESS, FAILURE FROM DBA_STMT_AUDIT_OPTS;

USER_NAME
--------------------------------------------------------------------------------
AUDIT_OPTION                             SUCCESS    FAILURE
---------------------------------------- ---------- ----------

CREATE SESSION                           BY ACCESS  BY ACCESS


CREATE ANY PROCEDURE                     BY ACCESS  BY ACCESS


GRANT ANY OBJECT PRIVILEGE               BY ACCESS  BY ACCESS


SQL>
```

Figure 34: Verification Queries

Consider Fine-Grained Auditing (FGA): For more targeted monitoring, use FGA to audit access to specific sensitive columns or tables. This allows you to define conditions and even trigger alerts

2.3 Entity Relational Diagram for Customer Management System

2.3.1 ER Diagram

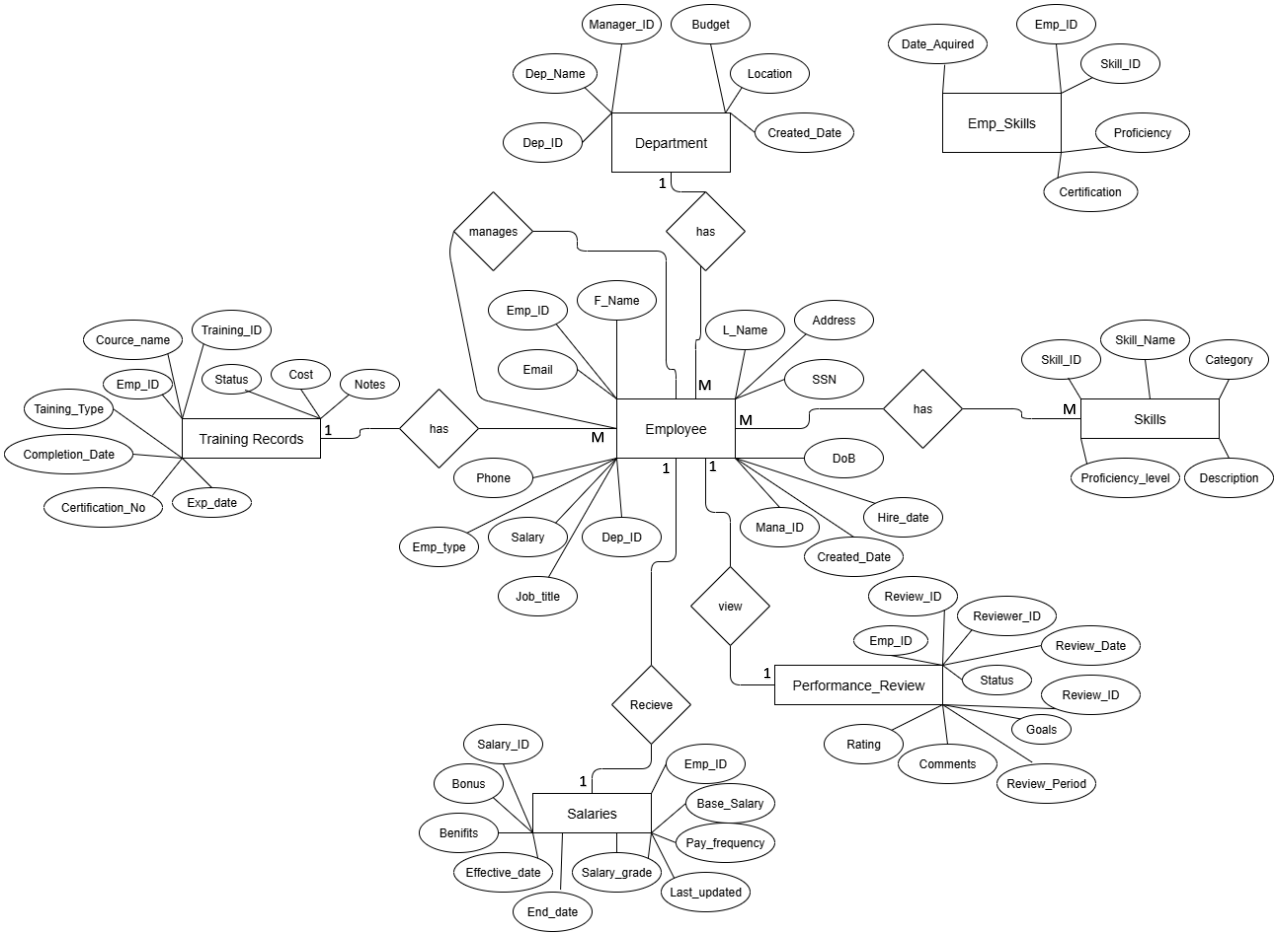


Figure 35: ER Diagram

- 1:M - Department to Employees, Training records to Employee
- 1:1 - Employee to Performance Review, Employee to Salaries
- M:N - Employee to Skills

2.3.2 Created tables and records of sample data

DEPARTMENTS (department_id, department_name, manager_id, budget, location, created_date)

EMPLOYEES (employee_id, first_name, last_name, email, phone, address, ssn, date_of_birth, hire_date, salary, department_id, manager_id, job_title, employment_type, created_date)

SALARIES (salary_id, employee_id, base_salary, bonus, benefits, effective_date, end_date, salary_grade, pay_frequency, last_updated)

PERFORMANCE_REVIEWS (review_id, employee_id, reviewer_id, review_date, rating, comments, goals, review_period, status, created_date)

SKILLS (skill_id, skill_name, category, description, proficiency_level)

EMPLOYEE_SKILLS (employee_id, skill_id, proficiency, date_acquired, certification)

TRAINING_RECORDS (training_id, employee_id, course_name, training_type, completion_date, status, cost, notes, certification_number, expiry_date)

```
oracle@doss-VirtualBox: ~

SQL> CREATE TABLE APP_OWNER.departments (department_id NUMBER PRIMARY KEY, department_name VARC
HAR2(100) NOT NULL, manager_id NUMBER, budget NUMBER(15,2), location VARCHAR2(100), created_dat
e TIMESTAMP DEFAULT CURRENT_TIMESTAMP);

Table created.

SQL> CREATE TABLE APP_OWNER.employees (employee_id NUMBER PRIMARY KEY, first_name VARCHAR2(50)
NOT NULL, last_name VARCHAR2(50) NOT NULL, email VARCHAR2(100), phone VARCHAR2(20), address VAR
CHAR2(200), ssn VARCHAR2(11), date_of_birth DATE, hire_date DATE, salary NUMBER(10,2), departme
nt_id NUMBER, manager_id NUMBER, job_title VARCHAR2(100), employment_type VARCHAR2(20), created
_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP);

Table created.

SQL> CREATE TABLE APP_OWNER.salaries (salary_id NUMBER PRIMARY KEY, employee_id NUMBER, base_sa
lary NUMBER(10,2), bonus NUMBER(10,2), benefits NUMBER(10,2), effective_date DATE, end_date DAT
E, salary_grade VARCHAR2(20), pay_frequency VARCHAR2(20), last_updated TIMESTAMP DEFAULT CURREN
T_TIMESTAMP);

Table created.

SQL> CREATE TABLE APP_OWNER.performance_reviews (review_id NUMBER PRIMARY KEY, employee_id NUMB
ER, reviewer_id NUMBER, review_date DATE, rating NUMBER(2,1), comments CLOB, goals CLOB, review
_period VARCHAR2(20), status VARCHAR2(20), created_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP);

Table created.

SQL> ALTER TABLE APP_OWNER.performance_reviews ADD CONSTRAINT review_reviewer_fk FOREIGN KEY (r
eviewer_id) REFERENCES APP_OWNER.employees(employee_id);

Table altered.
```

```
SQL> ALTER TABLE APP_OWNER.employees ADD CONSTRAINT emp_department_fk FOREIGN KEY (department_i
d) REFERENCES APP_OWNER.departments(department_id);

Table altered.

SQL> ALTER TABLE APP_OWNER.employees ADD CONSTRAINT emp_manager_fk FOREIGN KEY (manager_id) REF
ERENCES APP_OWNER.employees(employee_id);

Table altered.

SQL> ALTER TABLE APP_OWNER.salaries ADD CONSTRAINT salary_employee_fk FOREIGN KEY (employee_id)
 REFERENCES APP_OWNER.employees(employee_id);

Table altered.

SQL> ALTER TABLE APP_OWNER.performance_reviews ADD CONSTRAINT review_employee_fk FOREIGN KEY (e
mployee_id) REFERENCES APP_OWNER.employees(employee_id);

Table altered.
```

```
SQL> COLUMN TABLE_NAME FORMAT A30
SQL> SELECT TABLE_NAME
  2  FROM DBA_TABLES
  3  WHERE OWNER = 'APP_OWNER';

TABLE_NAME
------------------------------
DEPARTMENTS
EMPLOYEES
SALARIES
SKILLS
EMPLOYEE_SKILLS
PERFORMANCE_REVIEWS
TRAINING_RECORDS

7 rows selected.

SQL>
```

Figure 35,36,37 : created tables

```
oracle@doss-VirtualBox: ~
SQL> INSERT INTO APP_OWNER.departments VALUES (1, 'Human Resources', 1, 10000000, 'Colombo', CURRENT_TIMESTAMP);

1 row created.

SQL> INSERT INTO APP_OWNER.departments VALUES (2, 'Information Technology', 2, 15000000, 'Malabe', CURRENT_TIMESTAMP);

1 row created.

SQL> INSERT INTO APP_OWNER.departments VALUES (3, 'Finance', 3, 12000000, 'Kandy', CURRENT_TIMESTAMP);

1 row created.

SQL> INSERT INTO APP_OWNER.departments VALUES (4, 'Marketing', 4, 9000000, 'Galle', CURRENT_TIMESTAMP);

1 row created.

SQL> INSERT INTO APP_OWNER.departments VALUES (5, 'Operations', 5, 11000000, 'Kurunegala', CURRENT_TIMESTAMP);

1 row created.

SQL>
```

Figure 38 : sample data for departments

```
oracle@doss-VirtualBox: ~
SQL> INSERT INTO APP_OWNER.employees VALUES (1, 'Nimal', 'Perera', 'nimal.perera@lanka.lk', '077-1234567', '45 Galle Rd, Colombo 03', '750314
123V', DATE '1975-03-15', DATE '2015-01-15', 4800000, 1, NULL, 'HR Director', 'Full-time', CURRENT_TIMESTAMP);

1 row created.

SQL> INSERT INTO APP_OWNER.employees VALUES (2, 'Tharushi', 'Silva', 'tharushi.silva@lanka.lk', '071-2345678', '12 IT Park, Malabe', '8020242
13V', DATE '1980-07-22', DATE '2018-03-20', 4400000, 2, NULL, 'IT Manager', 'Full-time', CURRENT_TIMESTAMP);

1 row created.

SQL> INSERT INTO APP_OWNER.employees VALUES (3, 'Saman', 'Bandara', 'saman.bandara@lanka.lk', '076-3456789', '78 Central Ave, Kandy', '783124
567V', DATE '1978-11-08', DATE '2016-06-10', 4600000, 3, NULL, 'Finance Director', 'Full-time', CURRENT_TIMESTAMP);

1 row created.

SQL> INSERT INTO APP_OWNER.employees VALUES (4, 'Priya', 'Fernando', 'priya.fernando@lanka.lk', '070-4567890', '32 Sea St, Galle', '821487654
V', DATE '1982-05-28', DATE '2019-01-08', 3800000, 4, NULL, 'Marketing Manager', 'Full-time', CURRENT_TIMESTAMP);

1 row created.

SQL> INSERT INTO APP_OWNER.employees VALUES (5, 'Ranjan', 'Gamage', 'ranjan.gamage@lanka.lk', '078-5678901', '65 Lake Rd, Kurunegala', '72337
8912V', DATE '1972-12-03', DATE '2014-03-15', 4200000, 5, NULL, 'Operations Director', 'Full-time', CURRENT_TIMESTAMP);

1 row created.

SQL> INSERT INTO APP_OWNER.employees VALUES (7, 'Dinesh', 'Ranaweera', 'dinesh.ranaweera@lanka.lk', '071-7890123', '14 HR Tower, Colombo 03',
 '900251234V', DATE '1990-01-25', DATE '2021-09-30', 3000000, 1, 1, 'HR Business Partner', 'Full-time', CURRENT_TIMESTAMP);

1 row created.

SQL> INSERT INTO APP_OWNER.employees VALUES (8, 'Ayesha', 'Wickramasinghe', 'ayesha.wicks@lanka.lk', '076-8901234', '25 Tech Hub, Malabe', '8
81023456V', DATE '1988-04-12', DATE '2022-01-10', 3400000, 2, 2, 'Senior Developer', 'Full-time', CURRENT_TIMESTAMP);

1 row created.

SQL> INSERT INTO APP_OWNER.employees VALUES (9, 'Isuru', 'Jayasinghe', 'isuru.jaya@lanka.lk', '070-9012345', '36 Finance Rd, Kandy', '9222067
89V', DATE '1992-08-08', DATE '2023-03-15', 2800000, 3, 3, 'Financial Analyst', 'Full-time', CURRENT_TIMESTAMP);

1 row created.

SQL> INSERT INTO APP_OWNER.employees VALUES (10, 'Shanika', 'Priyadarshana', 'shanika.priya@lanka.lk', '078-0123456', '74 Marketing Rd, Galle
', '911649876V', DATE '1991-06-14', DATE '2022-11-05', 2600000, 4, 4, 'Marketing Specialist', 'Full-time', CURRENT_TIMESTAMP);
```

Figure 39 : sample data for employees

```
oracle@doss-VirtualBox: ~
SQL> INSERT INTO APP_OWNER.skills VALUES (1, 'Java Programming', 'Technical', 'Object-oriented programming in Java', 'Advanced');

1 row created.

SQL> INSERT INTO APP_OWNER.skills VALUES (2, 'Python', 'Technical', 'Python development and scripting', 'Advanced');

1 row created.

SQL> INSERT INTO APP_OWNER.skills VALUES (3, 'Project Management', 'Management', 'Agile and Waterfall methodologies', 'Expert');

1 row created.

SQL> INSERT INTO APP_OWNER.skills VALUES (4, 'Financial Analysis', 'Finance', 'Financial modeling and analysis', 'Advanced');

1 row created.

SQL> INSERT INTO APP_OWNER.skills VALUES (5, 'Digital Marketing', 'Marketing', 'SEO, SEM, and social media marketing', 'Advanced');

1 row created.

SQL> INSERT INTO APP_OWNER.skills VALUES (6, 'HR Policies', 'HR', 'Employment law and policy development', 'Expert');

1 row created.

SQL> INSERT INTO APP_OWNER.skills VALUES (7, 'SQL Database', 'Technical', 'Database design and query optimization', 'Advanced');

1 row created.

SQL> INSERT INTO APP_OWNER.skills VALUES (8, 'Leadership', 'Soft Skills', 'Team leadership and people management', 'Expert');

1 row created.

SQL> INSERT INTO APP_OWNER.skills VALUES (9, 'Communication', 'Soft Skills', 'Verbal and written communication', 'Advanced');

1 row created.

SQL> INSERT INTO APP_OWNER.skills VALUES (10, 'Data Analysis', 'Analytical', 'Statistical analysis and data visualization', 'Intermediate');

1 row created.

SQL>
```

Figure 40 : sample data for skills

```
oracle@doss-VirtualBox: ~
SQL> INSERT INTO APP_OWNER.employee_skills VALUES (1, 6, 'Expert', DATE '2015-01-15', 'SHRM-CP');

1 row created.

SQL> INSERT INTO APP_OWNER.employee_skills VALUES (2, 1, 'Advanced', DATE '2018-03-20', 'Java Professional');

1 row created.

SQL> INSERT INTO APP_OWNER.employee_skills VALUES (3, 4, 'Advanced', DATE '2016-06-10', 'CFA Level II');

1 row created.

SQL> INSERT INTO APP_OWNER.employee_skills VALUES (4, 5, 'Advanced', DATE '2019-01-08', 'Digital Marketing Cert');

1 row created.

SQL> INSERT INTO APP_OWNER.employee_skills VALUES (5, 3, 'Advanced', DATE '2014-03-15', 'PMP');

1 row created.

SQL> INSERT INTO APP_OWNER.employee_skills VALUES (7, 6, 'Advanced', DATE '2021-09-30', 'PHR');

1 row created.

SQL> INSERT INTO APP_OWNER.employee_skills VALUES (8, 1, 'Advanced', DATE '2022-01-10', 'Java Architect');

1 row created.

SQL> INSERT INTO APP_OWNER.employee_skills VALUES (9, 4, 'Intermediate', DATE '2023-03-15', null);

1 row created.

SQL> INSERT INTO APP_OWNER.employee_skills VALUES (10, 5, 'Intermediate', DATE '2022-11-05', null);

1 row created.

SQL> INSERT INTO APP_OWNER.employee_skills VALUES (11, 3, 'Intermediate', DATE '2021-08-19', null);

1 row created.
```

Figure 41 : sample data for employee_skills

2.4 User and Role Management

Roles Created and Configured

1. SYSADMIN_ROLE - Full system permissions

2. MANAGER_ROLE - Database read/write permissions:

- *SELECT, INSERT, UPDATE on employees, salaries, performance_reviews*

- *SELECT, INSERT, UPDATE, DELETE on training_records, skills, employee_skills*

- *SELECT on departments*

3. EXECUTIVE_ROLE - Read-only permissions on all tables

```
oracle@doss-VirtualBox: ~
SQL> CREATE ROLE SYSADMIN_ROLE;

Role created.

SQL> CREATE ROLE MANAGER_ROLE;

Role created.

SQL> CREATE ROLE EXECUTIVE_ROLE;

Role created.
```

Figure 42 : created Roles

```
oracle@doss-VirtualBox: ~
SQL> GRANT CREATE SESSION TO SYSADMIN_ROLE;

Grant succeeded.

SQL> GRANT SELECT, INSERT, UPDATE, DELETE ON APP_OWNER.departments TO SYSADMIN_ROLE;

Grant succeeded.

SQL> GRANT SELECT, INSERT, UPDATE, DELETE ON APP_OWNER.employees TO SYSADMIN_ROLE;

Grant succeeded.

SQL> GRANT SELECT, INSERT, UPDATE, DELETE ON APP_OWNER.salaries TO SYSADMIN_ROLE;

Grant succeeded.

SQL> GRANT SELECT, INSERT, UPDATE, DELETE ON APP_OWNER.performance_reviews TO SYSADMIN_ROLE;

Grant succeeded.

SQL> GRANT SELECT, INSERT, UPDATE, DELETE ON APP_OWNER.training_records TO SYSADMIN_ROLE;

Grant succeeded.

SQL> GRANT SELECT, INSERT, UPDATE, DELETE ON APP_OWNER.skills TO SYSADMIN_ROLE;

Grant succeeded.

SQL> GRANT SELECT, INSERT, UPDATE, DELETE ON APP_OWNER.employee_skills TO SYSADMIN_ROLE;

Grant succeeded.
```

Figure 43 : SYSADMIN_ROLE pemissions

```
oracle@doss-VirtualBox: ~
SQL> GRANT CREATE SESSION TO MANAGER_ROLE;

Grant succeeded.

SQL> GRANT SELECT ON APP_OWNER.departments TO MANAGER_ROLE;

Grant succeeded.

SQL> GRANT SELECT, INSERT, UPDATE ON APP_OWNER.employees TO MANAGER_ROLE;

Grant succeeded.

SQL> GRANT SELECT, INSERT, UPDATE ON APP_OWNER.salaries TO MANAGER_ROLE;

Grant succeeded.

SQL> GRANT SELECT, INSERT, UPDATE ON APP_OWNER.performance_reviews TO MANAGER_ROLE;

Grant succeeded.

SQL> GRANT SELECT, INSERT, UPDATE, DELETE ON APP_OWNER.training_records TO MANAGER_ROLE;

Grant succeeded.

SQL> GRANT SELECT, INSERT, UPDATE, DELETE ON APP_OWNER.skills TO MANAGER_ROLE;

Grant succeeded.

SQL> GRANT SELECT, INSERT, UPDATE, DELETE ON APP_OWNER.employee_skills TO MANAGER_ROLE;

Grant succeeded.
```

Figure 44 : MANAGER_ROLE pemissions

```
oracle@doss-VirtualBox: ~
SQL> GRANT CREATE SESSION TO EXECUTIVE_ROLE;

Grant succeeded.

SQL> GRANT SELECT ON APP_OWNER.departments TO EXECUTIVE_ROLE;

Grant succeeded.

SQL> GRANT SELECT ON APP_OWNER.employees TO EXECUTIVE_ROLE;

Grant succeeded.

SQL> GRANT SELECT ON APP_OWNER.salaries TO EXECUTIVE_ROLE;

Grant succeeded.

SQL> GRANT SELECT ON APP_OWNER.performance_reviews TO EXECUTIVE_ROLE;

Grant succeeded.

SQL> GRANT SELECT ON APP_OWNER.training_records TO EXECUTIVE_ROLE;

Grant succeeded.

SQL> GRANT SELECT ON APP_OWNER.skills TO EXECUTIVE_ROLE;

Grant succeeded.

SQL> GRANT SELECT ON APP_OWNER.employee_skills TO EXECUTIVE_ROLE;

Grant succeeded.
```

Figure 45 : EXECUTIVE_ROLE pemissions

```
oracle@doss-VirtualBox: ~
SQL> COLUMN ROLE FORMAT A30
SQL> SELECT ROLE FROM DBA_ROLES WHERE ROLE IN ('SYSADMIN_ROLE', 'MANAGER_ROLE', 'EXECUTIVE_ROLE');

ROLE
------------------------------
EXECUTIVE_ROLE
MANAGER_ROLE
SYSADMIN_ROLE
```

Figure 46 : proof of the Roles

2.5 View Implementation

Manager Employee Salary View

- View Name: manager_employee_salary_v
- Purpose: Allows managers to view employee personal and salary details
- Columns: employee_id, first_name, last_name, email, phone, department_id, job_title, base_salary, bonus, benefits, effective_date, salary_grade

```
oracle@doss-VirtualBox: ~
SQL> CREATE OR REPLACE VIEW APP_OWNER.manager_employee_salary_v AS
  2  SELECT
  3      e.employee_id,
  4      e.first_name,
  5      e.last_name,
  6      e.email,
  7      e.phone,
  8      e.department_id,
  9      e.job_title,
 10      s.base_salary,
 11      s.bonus,
 12      s.benefits,
 13      s.effective_date,
 14      s.salary_grade
 15  FROM APP_OWNER.employees e
 16  JOIN APP_OWNER.salaries s ON e.employee_id = s.employee_id;

View created.

SQL> GRANT SELECT ON APP_OWNER.manager_employee_salary_v TO MANAGER_ROLE;

Grant succeeded.
```

Figure 47 : view creation

```
Grant succeeded.

SQL> COLUMN VIEW_NAME FORMAT A30
SQL> SELECT VIEW_NAME, OWNER
  2  FROM DBA_VIEWS
  3  WHERE VIEW_NAME = 'MANAGER_EMPLOYEE_SALARY_V' AND OWNER = 'APP_OWNER';

VIEW_NAME
------------------------------
OWNER
--------------------------------------------------------------------------------
MANAGER_EMPLOYEE_SALARY_V
APP_OWNER


SQL>
```

Figure 48 : proof of the view

2.6 Virtual Private Database Implementation

2.6.1 VPD Policy Function

- Function Name: manager_vpd_policy
- Logic: Restricts data access based on user's department affiliation
- Implementation: Returns predicate department_id = [user_department_id]

```
oracle@doss-VirtualBox: ~
SQL> CREATE OR REPLACE FUNCTION APP_OWNER.manager_vpd_policy
  2  (schema_name IN VARCHAR2, table_name IN VARCHAR2)
  3  RETURN VARCHAR2
  4  AS
  5      v_dept_id APP_OWNER.employees.department_id%TYPE;
  6      v_predicate VARCHAR2(200);
  7  BEGIN
  8      IF SYS_CONTEXT('USERENV', 'SESSION_USER') IN ('SYS', 'SYSTEM') THEN
  9          RETURN '1=1';
 10      END IF;
 11      BEGIN
 12          SELECT department_id INTO v_dept_id
 13          FROM APP_OWNER.employees e
 14          WHERE e.employee_id = (
 15               SELECT manager_id FROM APP_OWNER.departments WHERE ROWNUM = 1
 16          );
 17      EXCEPTION
 18          WHEN NO_DATA_FOUND THEN
 19              RETURN '1=0';
 20      END;
 21      v_predicate := 'department_id = ' || v_dept_id;
 22      RETURN v_predicate;
 23  END;
 24  /

Function created.

SQL> SHOW ERRORS FUNCTION APP_OWNER.manager_vpd_policy;
No errors.
```

Figure 49 : VPD Policy Function

2.6.2 VPD Policy Application

- Policy Name: manager_dept_vpd_policy
- Security Relevant Columns: DEPARTMENT_ID
- Statement Types: SELECT

```
SQL> SHOW ERRORS FUNCTION APP_OWNER.manager_vpd_policy;
No errors.
SQL>
SQL> BEGIN
  2    DBMS_RLS.ADD_POLICY (
  3      object_schema    => 'APP_OWNER',
  4      object_name      => 'EMPLOYEES',
  5      policy_name      => 'manager_dept_vpd_policy',
  6      function_schema  => 'APP_OWNER',
  7      policy_function  => 'manager_vpd_policy',
  8      statement_types  => 'SELECT', -- Only restrict reading data
  9      sec_relevant_cols => 'DEPARTMENT_ID',
 10      policy_type      => DBMS_RLS.DYNAMIC
 11    );
 12  END;
 13  /

PL/SQL procedure successfully completed.

SQL>
```

Figure 50 : VPD Policy Application

2.7 Data Encryption Implementation

Encrypted Columns

1. EMPLOYEES.SSN - AES256 encryption with NO SALT
2. SALARIES.BASE_SALARY - AES256 encryption

```
oracle@doss-VirtualBox: ~
SQL> ALTER TABLE APP_OWNER.employees
  2  MODIFY (ssn ENCRYPT USING 'AES256' NO SALT);

Table altered.

SQL> ALTER TABLE APP_OWNER.salaries
  2  MODIFY (base_salary ENCRYPT USING 'AES256');

Table altered.

SQL>
SQL> COLUMN TABLE_NAME FORMAT A20
SQL> COLUMN COLUMN_NAME FORMAT A20
SQL> COLUMN COLUMN_NAME FORMAT A20
SQL> COLUMN COLUMN_NAME FORMAT A20
SQL> COLUMN COLUMN_NAME FORMAT A20
SQL> COLUMN COLUMN_NAME FORMAT A20
SQL> COLUMN COLUMN_NAME FORMAT A20
SQL> COLUMN COLUMN_NAME FORMAT A20
SQL> COLUMN COLUMN_NAME FORMAT A20
SQL> COLUMN COLUMN_NAME FORMAT A20
SQL> SELECT TABLE_NAME, COLUMN_NAME, ENCRYPTION_ALG
  2  FROM DBA_ENCRYPTED_COLUMNS
  3  WHERE OWNER = 'APP_OWNER'
  4  AND COLUMN_NAME IN ('SSN', 'BASE_SALARY');

TABLE_NAME           COLUMN_NAME          ENCRYPTION_ALG
-------------------- -------------------- ------------------------------
EMPLOYEES            SSN                  AES 256 bits key
SALARIES             BASE_SALARY          AES 256 bits key

SQL>
```

Figure 51 : Encrypted Columns

2.8 Data Masking

Data Redaction Policy

- Policy Name: redact_employee_email
- Applied To: EMPLOYEES.EMAIL column
- Redaction Type: FULL
- Expression: Always applied (1=1)
- Purpose: Protects employee email addresses from unauthorized viewing

```
oracle@doss-VirtualBox: ~

PL/SQL procedure successfully completed.

SQL> BEGIN
  2      DBMS_REDACT.ADD_POLICY (
  3          object_schema       => 'APP_OWNER',
  4          object_name         => 'EMPLOYEES',
  5          policy_name         => 'redact_employee_email',
  6          expression          => '1=1',
  7          column_name         => 'EMAIL',
  8          function_type       => DBMS_REDACT.FULL,
  9          function_parameters => NULL
 10      );
 11  END;
 12  /

PL/SQL procedure successfully completed.
```

Figure 52 : Data Encryption

2.9 FGA Policies Configured

audit_ssn_access - Monitors access to SSN column in EMPLOYEES table

```
oracle@doss-VirtualBox: ~

SQL> BEGIN
  2      DBMS_FGA.ADD_POLICY (
  3          object_schema     => 'APP_OWNER',
  4          object_name       => 'EMPLOYEES',
  5          policy_name       => 'audit_ssn_access',
  6          audit_condition   => '1=1',
  7          audit_column      => 'SSN',
  8          enable            => TRUE
  9      );
 10  END;
 11  /

PL/SQL procedure successfully completed.
```

Figure 53 : audit_ssn_access

audit_salary_access - Monitors access to BASE_SALARY column in SALARIES table

```
oracle@doss-VirtualBox: ~

SQL> BEGIN
  2      DBMS_FGA.ADD_POLICY (
  3          object_schema     => 'APP_OWNER',
  4          object_name       => 'SALARIES',
  5          policy_name       => 'audit_salary_access',
  6          audit_condition   => '1=1',
  7          audit_column      => 'BASE_SALARY',
  8          enable            => TRUE
  9      );
 10  END;
 11  /

PL/SQL procedure successfully completed.

SQL>
```

Figure 54 : audit_salary_access

Audit Condition: Always active (1=1)
Status: Enabled

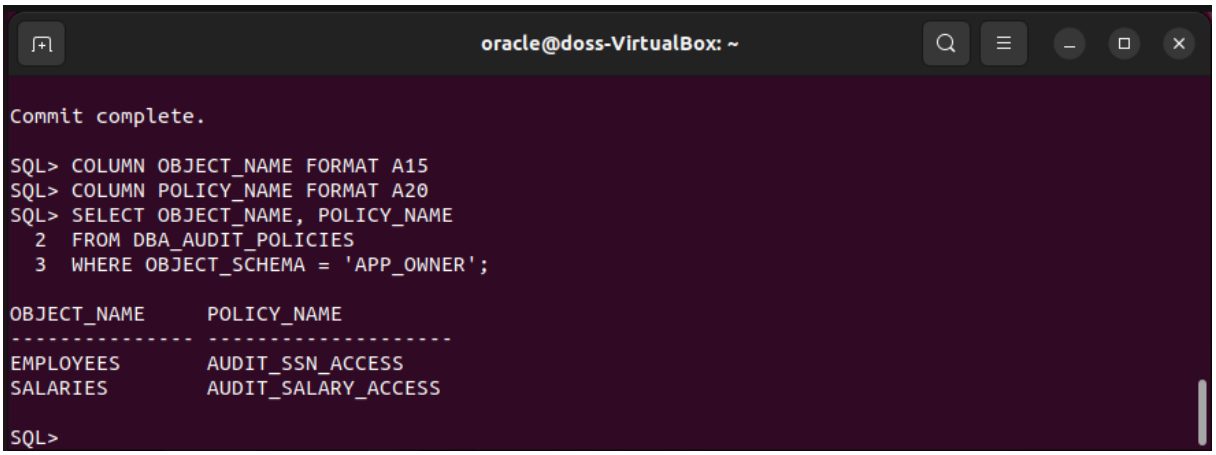


Figure 55 : Policies verified through DBA_AUDIT_POLICIES view

3. Big Data Security

3.1 Description

Big Data security involves protecting massive, complex datasets that are characterized by the three features Volume, Velocity, and Variety. Unlike traditional databases, Big Data systems face unique security challenges due to their distributed nature, real time processing requirements, and diverse data types including structured, semi-structured, and unstructured data.

Key Security Challenges in Big Data:

- **Distributed Architecture Vulnerabilities**: Multiple nodes increase attack surface
- **Real-time Processing Risks**: Streaming data requires continuous security monitoring
- **Diverse Data Types**: Different security requirements for various data formats
- **Scalability Issues**: Security measures must scale with data growth
- **Privacy Concerns**: Massive datasets increase privacy violation risks

3.2 The main security requirements of big data

Security Requirement	Big Data Systems	Traditional Databases
Data Confidentiality	Advanced encryption across distributed nodes, data anonymization techniques	Basic column-level encryption in centralized systems
Data Integrity	Distributed integrity checks, blockchain verification, stream validation	ACID transactions, basic checksums, manual verification
Data Availability	High availability clusters, geo-redundancy, fault-tolerant architectures	Basic backup systems, failover mechanisms
Authentication & Access Control	Multi-factor authentication, API-based access, cloud identity	Simple username/password, role-based access control

	services	
Accountability & Auditing	Distributed logging, real-time monitoring, cross-platform audit trails	Centralized logging, periodic audit reports
Privacy Preservation	Differential privacy, advanced anonymization, data minimization	Basic data masking, simple anonymization
Secure Data Sharing	Federated learning, secure multi-party computation, encrypted analytics	Limited data sharing, basic export controls

3.3 Common Attacks on Big Data Systems

1. **Data Breaches and Unauthorized Access**

- **Explanation**: Attackers exploit weak authentication or misconfigured access controls in distributed Big Data platforms like Hadoop or NoSQL databases to gain unauthorized access to massive datasets containing sensitive information.

- **Impact**: Massive exposure of personal data, regulatory fines (GDPR, CCPA), reputational damage, and financial losses due to stolen intellectual property.

2. **Distributed Denial of Service (DDoS) Attacks**

- **Explanation:** Attackers overwhelm Big Data infrastructure with massive traffic volumes, targeting data processing nodes, storage systems, or analytics services, causing service disruption and resource exhaustion.

- **Impact**: Service unavailability, disrupted business operations, financial losses from downtime, and compromised data processing capabilities

3. **Data Tampering and Integrity Attacks**

- **Explanation**: Malicious actors intercept and modify data during transmission between distributed nodes or alter stored data in data lakes, compromising data accuracy and reliability for analytics and decision-making.

- **Impact**: Inaccurate business intelligence, flawed decision-making, regulatory compliance violations, and loss of data trustworthiness.

4. **Inference Attacks**

- **Explanation**: Attackers use sophisticated query patterns and statistical analysis to deduce sensitive information from aggregated data results, even without direct access to raw data, exploiting privacy vulnerabilities in analytics outputs.

- **Impact**: Privacy violations, intellectual property theft, regulatory non-compliance, and reputational harm from inferred sensitive information disclosure.

5. **Insider Threats**

- **Explanation**: Authorized users with extensive data access privileges misuse their position to steal, manipulate, or expose sensitive Big Data, often going undetected due to the volume of data and

access patterns.

- **Impact**: Data theft, sabotage of analytics systems, unauthorized data modification, and significant business disruption from trusted personnel actions.

3.4 Security Controls to Mitigate Attacks

1. **For Data Breaches and Unauthorized Access:**

Control 1: Zero-Trust Architecture with Micro-segmentation

- Implement strict identity verification for every access request regardless of source
- Segment Big Data environments into secure zones with separate access controls
- Continuously monitor and validate access patterns across all data nodes

Control 2: Advanced Encryption with Key Management

- Deploy end-to-end encryption for data at rest, in transit, and during processing
- Implement robust key management systems with automatic key rotation
- Use format-preserving encryption for structured data and tokenization for sensitive fields

2. **For DDoS Attacks:**

Control 1: Multi-layered DDoS Protection

- Deploy cloud-based DDoS protection services with traffic scrubbing capabilities
- Implement rate limiting and traffic shaping at network and application layers
- Use behavioral analysis to distinguish between legitimate data processing and attack traffic

Control 2: Elastic Resource Management

- Implement auto-scaling mechanisms to handle traffic spikes
- Deploy load balancers with health monitoring and traffic distribution
- Maintain redundant data processing capacity across multiple availability zones

3. **For Data Tampering and Integrity Attacks:**

Control 1: Blockchain-based Integrity Verification

- Implement distributed ledger technology to create immutable audit trails
- Use cryptographic hashing to verify data integrity across nodes
- Deploy real-time integrity monitoring with automated alerting

Control 2: Digital Signatures and Checksums

- Apply digital signatures to critical data transactions and processing results
- Implement regular checksum verification for stored data
- Use secure hash algorithms (SHA-256, SHA-3) for data validation

4. **For Inference Attacks:**

Control 1: Differential Privacy Implementation

- Add calibrated noise to query results to prevent individual record identification
- Implement privacy budgets to limit the amount of information revealed through multiple queries
- Use k-anonymity and l-diversity techniques for data publication

Control 2: Query Restriction and Monitoring

- Enforce query complexity limits and result set size restrictions
- Monitor query patterns for suspicious analytical behavior

- Implement purpose-based access controls limiting analytical scope

5. **For Insider Threats:**

Control 1: User Behavior Analytics (UBA)

- Deploy machine learning algorithms to detect anomalous user behavior patterns
- Implement real-time monitoring of data access and modification activities
- Establish baseline behavior profiles and flag significant deviations

Control 2: Principle of Least Privilege with Just-in-Time Access

- Enforce strict role-based access controls with minimal necessary permissions
- Implement just-in-time privilege elevation with time-bound access
- Conduct regular access reviews and privilege audits

3.5 Threat to Control Mapping

Threat	OS Security Controls	Database Security Controls
Brute-force Attacks	Fail2Ban, SSH key enforcement, Strong passwords	Account locking, Password policies
Unauthorized Access	UFW, Disabled root SSH, File permissions	VPD, Role-based access, Encryption
Data Theft	File encryption, Secure permissions	TDE, Data redaction, Column encryption
Privilege Escalation	User management, Sudo controls	Least privilege, PUBLIC role revocation
Log Tampering	Auditd configuration	FGA policies, Database auditing
Service Exploitation	Regular updates, Service hardening	Network restrictions, Listener security