📝 Cybersecurity Investigation Report – Incident #1

Title: Securing Sensitive File Access on Linux & Monitoring via auditd
Analyst: Muhaned Alhashimy
Date: Aug 1, 2025
Environment: Kali Linux (VM)

🎯 Objective

To simulate improper file access control on Linux using overly permissive permissions, exploit them as unauthorized users, and then fix the configuration using ACLs and monitor file activity using auditd.

🛠️ Tools & Commands Used

📁 Folder and File Creation

```
sudo mkdir -p /research_data
echo "Sensitive: Salary data Q3" | sudo tee
/research_data/report.txt
sudo chmod 777 /research_data/report.txt
```

Created a sensitive file with insecure permissions (777), making it fully accessible by any user.

👥 User Simulation

```
sudo useradd -m admin_user
sudo useradd -m edit_user
sudo useradd -m read_user
echo "edit_user:123" | sudo chpasswd
echo "read_user:123" | sudo chpasswd
```

Simulated three users: administrator, editor, and reader.

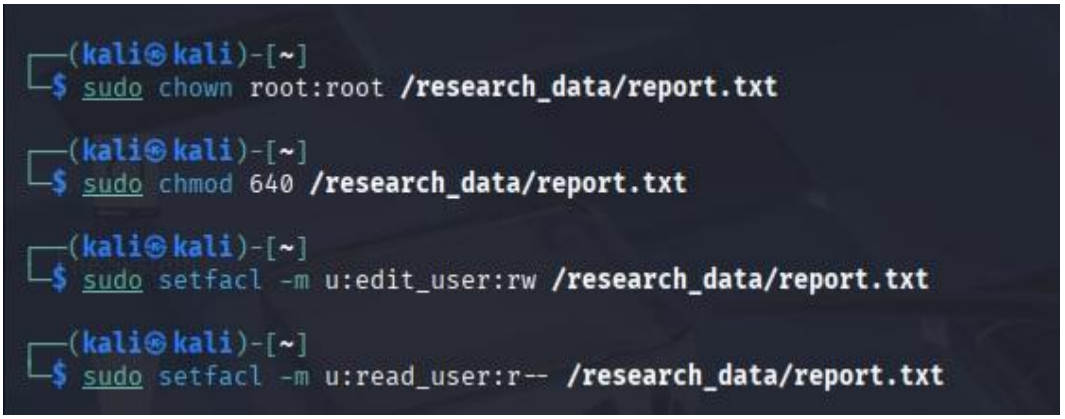
🔓 Exploiting Insecure Access

```
sudo su - read_user
cd /research_data
rm report.txt
```

The `read_user` was able to delete the file due to incorrect permissions, demonstrating a security risk.

🔐 Applying Least Privilege (ACLs)

```
sudo chown root:root /research_data/report.txt
sudo chmod 640 /research_data/report.txt
sudo setfacl -m u:edit_user:rw /research_data/report.txt
sudo setfacl -m u:read_user:r-- /research_data/report.txt
```



Applied proper file ownership and ACLs to ensure specific users have only the minimum required access.

✅ Testing Access

```
# As read_user
cat /research_data/report.txt        # ✅
echo "test" >> report.txt            # ❌
# As edit_user
nano /research_data/report.txt       # ✅
```

```
┌──(kali㉿kali)-[~]
└─$ getfacl /research_data/report.txt
getfacl: Removing leading '/' from absolute path names
# file: research_data/report.txt
# owner: root
# group: root
user::rw-
user:1002:rwx
user:admin_user:r--
user:edit_user:rw-
user:read_user:r--
group::r--
mask::rwx
other::---

┌──(kali㉿kali)-[~]
└─$ sudo su - read_user
$ cat /research_data/report.txt
Sensitive: Salary data Q3
$ echo "malicious" >> /research_data/report.txt
-sh: 2: cannot create /research_data/report.txt: Permission denied
$ exit

┌──(kali㉿kali)-[~]
└─$ sudo su - edit_user
$ nano /research_data/report.txt
$ exit
```

```
File  Actions  Edit  View  Help
  GNU nano 8.4                                          /research_data/report.txt *
Sensitive: Salary data Q3
by Muhaned Alhashimy
```

Readers can view the file. Editors can modify it. No one except root can delete it.

🛡️ Enabling File Auditing

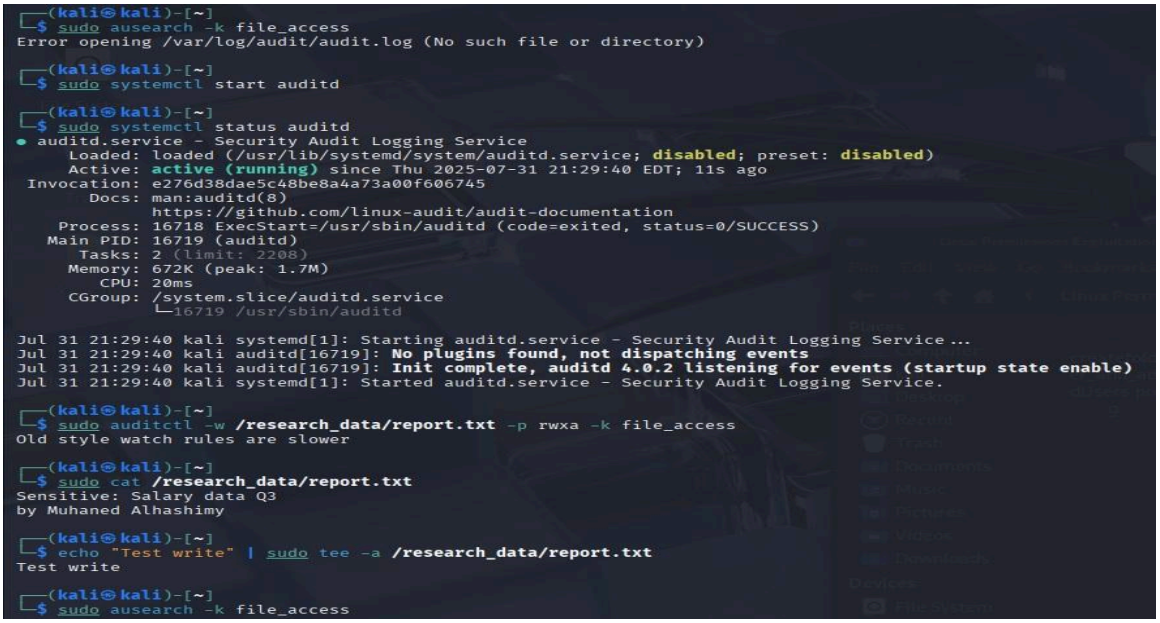
```
sudo apt install auditd -y
sudo systemctl start auditd
sudo auditctl -w /research_data/report.txt -p rwxa -k file_access
```

```
┌──(kali㉿kali)-[~]
└─$ sudo apt install auditd -y
The following packages were automatically installed and are no longer required:
  python3-packaging-whl  python3-pyinstaller-hooks-contrib  python3-wheel-whl
Use 'sudo apt autoremove' to remove them.

Installing:
  auditd

Installing dependencies:
  libauparse0t64

Suggested packages:
  audispd-plugins

Summary:
  Upgrading: 0, Installing: 2, Removing: 0, Not Upgrading: 0
  Download size: 286 kB
  Space needed: 954 kB / 63.1 GB available

Get:1 http://http.kali.org/kali kali-rolling/main amd64 libauparse0t64 amd64 1:4.0.2-2+b2 [68.6 kB]
Get:2 http://http.kali.org/kali kali-rolling/main amd64 auditd amd64 1:4.0.2-2+b2 [217 kB]
Fetched 286 kB in 2s (178 kB/s)
Selecting previously unselected package libauparse0t64:amd64.
(Reading database ... 418499 files and directories currently installed.)
Preparing to unpack .../libauparse0t64_1%3a4.0.2-2+b2_amd64.deb ...
Adding 'diversion of /lib/x86_64-linux-gnu/libauparse.so.0 to /lib/x86_64-linux-gnu/libauparse.so.0.usr-is-merged by libauparse0t64'
Adding 'diversion of /lib/x86_64-linux-gnu/libauparse.so.0.0.0 to /lib/x86_64-linux-gnu/libauparse.so.0.0.0.usr-is-merged by libauparse0t64'
Unpacking libauparse0t64:amd64 (1:4.0.2-2+b2) ...
Selecting previously unselected package auditd.
Preparing to unpack .../auditd_1%3a4.0.2-2+b2_amd64.deb ...
Unpacking auditd (1:4.0.2-2+b2) ...
Setting up libauparse0t64:amd64 (1:4.0.2-2+b2) ...
Setting up auditd (1:4.0.2-2+b2) ...
update-rc.d: We have no instructions for the auditd init script.
update-rc.d: It looks like a non-network service, we enable it.
audit-rules.service is a disabled or a static unit, not starting it.
auditd.service is a disabled or a static unit, not starting it.
Processing triggers for man-db (2.13.1-1) ...
Processing triggers for kali-menu (2025.3.0) ...
Processing triggers for libc-bin (2.41-9) ...

┌──(kali㉿kali)-[~]
└─$ sudo auditctl -w /research_data/report.txt -p rwxa -k file_access
Old style watch rules are slower
```

Installed `auditd` and set up a rule to monitor all read, write, execute, and attribute changes on the file.

🔍 Capturing File Access Logs

`sudo ausearch -k file_access`



```
┌──(kali㉿kali)-[~]
└─$ sudo ausearch -k file_access
Error opening /var/log/audit/audit.log (No such file or directory)

┌──(kali㉿kali)-[~]
└─$ sudo systemctl start auditd

┌──(kali㉿kali)-[~]
└─$ sudo systemctl status auditd
● auditd.service - Security Audit Logging Service
     Loaded: loaded (/usr/lib/systemd/system/auditd.service; disabled; preset: disabled)
     Active: active (running) since Thu 2025-07-31 21:29:40 EDT; 11s ago
 Invocation: e276d38dae5c48be8a4a73a00f606745
       Docs: man:auditd(8)
             https://github.com/linux-audit/audit-documentation
    Process: 16718 ExecStart=/usr/sbin/auditd (code=exited, status=0/SUCCESS)
   Main PID: 16719 (auditd)
      Tasks: 2 (limit: 2208)
     Memory: 672K (peak: 1.7M)
        CPU: 20ms
     CGroup: /system.slice/auditd.service
             └─16719 /usr/sbin/auditd

Jul 31 21:29:40 kali systemd[1]: Starting auditd.service - Security Audit Logging Service...
Jul 31 21:29:40 kali auditd[16719]: No plugins found, not dispatching events
Jul 31 21:29:40 kali auditd[16719]: Init complete, auditd 4.0.2 listening for events (startup state enable)
Jul 31 21:29:40 kali systemd[1]: Started auditd.service - Security Audit Logging Service.

┌──(kali㉿kali)-[~]
└─$ sudo auditctl -w /research_data/report.txt -p rwxa -k file_access
Old style watch rules are slower

┌──(kali㉿kali)-[~]
└─$ sudo cat /research_data/report.txt
Sensitive: Salary data Q3
by Muhaned Alhashimy

┌──(kali㉿kali)-[~]
└─$ echo "Test write" | sudo tee -a /research_data/report.txt
Test write

┌──(kali㉿kali)-[~]
└─$ sudo ausearch -k file_access
```

Displayed a full list of all actions performed on the file, including username, action type, and timestamp.

📸 Screenshots

- `getfacl` output (verifying access rules)
- `auditd` logs showing file events
- `Permission denied` error when unauthorized write attempted
- Successful edit with `nano` as `edit_user`

📝 *(Insert screenshots in the appropriate places in your report)*

✅ Conclusion

The sensitive file was originally configured with insecure, world-writable permissions.
Unauthorized users could delete or overwrite it.
ACLs were implemented to enforce the principle of least privilege.
The auditd tool was configured to monitor and log access attempts, providing visibility and traceability.

💡 Reflection

This investigation demonstrated the critical importance of properly managing file permissions in Linux environments. Overly permissive settings such as `777` can lead to data leaks or tampering. Using ACLs and enabling auditing are effective ways to strengthen system security.