Incident Response & Digital Forensics

Assignment 1:

Question: Simulate a security breach by creating and executing a malicious script on a Linux system. Create an audit log and track the script's activity, including any file changes.

Simulation of a Security Breach and Audit Log Tracking on Linux

Introduction

In this assignment, a controlled security breach was simulated on a Linux system by creating and executing a benign malicious script. The objective was to demonstrate typical malicious behaviors, such as file creation and permission changes, and to track all such activities using the Linux auditing system (auditd). This simulation enables understanding of attack footprints and effective monitoring.

Scope and Objective

- To create a simple script that mimics malicious behavior by modifying system files and permissions.
- To configure Linux auditd to monitor the execution of the script and resultant file changes.
- To generate and analyze audit logs capturing the breach activity for learning and documentation.

Environment Setup

- A dedicated virtual machine running Ubuntu (or similar Linux distribution) was used for safe and isolated testing.
- The auditd service was installed and initialized to provide system-level audit capabilities.

Malicious Script Creation

Script Purpose

The script performs the following actions to simulate suspicious activity:

- Creates a file /tmp/hack_notice.txt containing a warning message.
- Changes the file permissions to restrict access.
- Logs a timestamp message to /tmp/hack_log.txt.
- Creates a hidden file /tmp/.hidden_file to simulate stealth.
-

Script Content (malicious.sh)

```
#!/bin/bash

echo "You have been hacked" > /tmp/hack_notice.txt

chmod 600 /tmp/hack_notice.txt

echo "Malicious script ran at $(date)" >> /tmp/hack_log.txt

touch /tmp/.hidden_file
```

After creating the script file, it was made executable with:

```
chmod +x malicious.sh
```

Auditd Configuration

Installation and Service Start

```
sudo apt-get install auditd
```

```
sudo systemctl start auditd
```

```
sudo systemctl enable auditd
```

Audit Rules Applied

- Monitored the execution of the script itself:

```
sudo auditctl -w /path/to/malicious.sh -p x -k malicious_script
```

- Monitored file write and attribute changes in /tmp directory:

```
sudo auditctl -w /tmp/ -p wa -k tmp_changes
```

Execution of the Script

The script was executed with:

```
./malicious.sh
```

This triggered the simulated malicious behavior and generated related audit events.

Audit Log Retrieval and Analysis

Commands Used to Retrieve Logs

- To view script execution logs:

```
sudo ausearch -k malicious_script
```

- To view file changes in /tmp:

```
sudo ausearch -k tmp_changes
```

- To get a summary of file-related audit events:

```
sudo aureport -f
```

Observations

- The execution of the script was captured, showing the timestamp and user.
- File creation and permission change events in /tmp were logged with detailed attributes.
- Hidden file creation was recorded, illustrating stealthy actions detected by auditd.

Conclusion

This simulation successfully demonstrated how a simple malicious script can alter files and permissions on a Linux system and how auditd can effectively capture and log these suspicious activities. Proper configuration of auditd rules enables detection and forensic analysis of breaches, proving essential for security monitoring and incident response.

Incident Response & Digital Forensics

Assignment 2 : Analysis of Suspicious Login Attempts from /var/log/auth.log

Introduction

This assignment presents an analysis of system authentication logs captured in the /var/log/auth.log file to identify any suspicious login attempts or security-related events. The purpose is to detect unauthorized access, anomalous login activities, and related authentication issues to improve system security monitoring.

Scope and Objective

The analysis focuses on login events on the system dated 15th October 2025, observed from the provided log excerpt. The objective is to extract timestamps and associated IP addresses (where available) of suspicious login attempts and document any notable findings, including successful root access, failed attempts, and system errors.

Tools and Environment

- Linux system with access to /var/log/auth.log
- Command-line utilities such as cat, grep, and awk used to extract relevant log entries
- Manual log inspection complementing command-line parsing

Methodology

1. Accessed the authentication log file at /var/log/auth.log.
2. Parsed log entries for key terms indicative of login activity, including "login", "session opened", "failed password", and "root login".
3. Extracted and tabulated timestamps and relevant event information such as usernames, IP addresses (if present), and device terminals.
4. Analyzed system messages for errors affecting authentication processes.
5. Evaluated the data for signs of repeated failed attempts, unauthorized logins, and irregular patterns.

Findings:

Timestamp (UTC)	Event Description	User	IP Address	Notes
2025-10-15T08:21:40.314090	PAM module pam_lastlog.so missing	N/A	N/A	PAM module missing; potential logging impact
2025-10-15T08:21:41.052938	Session opened for user root (UID=0)	root	N/A	Successful root login; high privilege
2025-10-15T08:21:41.493088	Root login on terminal /dev/pts/1	root	N/A	Interactive root login via terminal

Timestamp (UTC)	Event Description	User	IP Address	Notes
2025-10-15T08:25:00.421577	Cron session started for root user	root	N/A	Scheduled task running
2025-10-15T08:25:00.428372	Cron session closed for root user	root	N/A	Scheduled task completed

- No failed login attempts or external IP addresses were present in the provided log entries.
- Missing PAM module pam_lastlog.so indicates some incomplete PAM configuration which could impact log completeness.
- Root user activity is documented with explicit session open logs at terminal device /dev/pts/1.
- Cron job execution under root proceeded normally without anomalies.

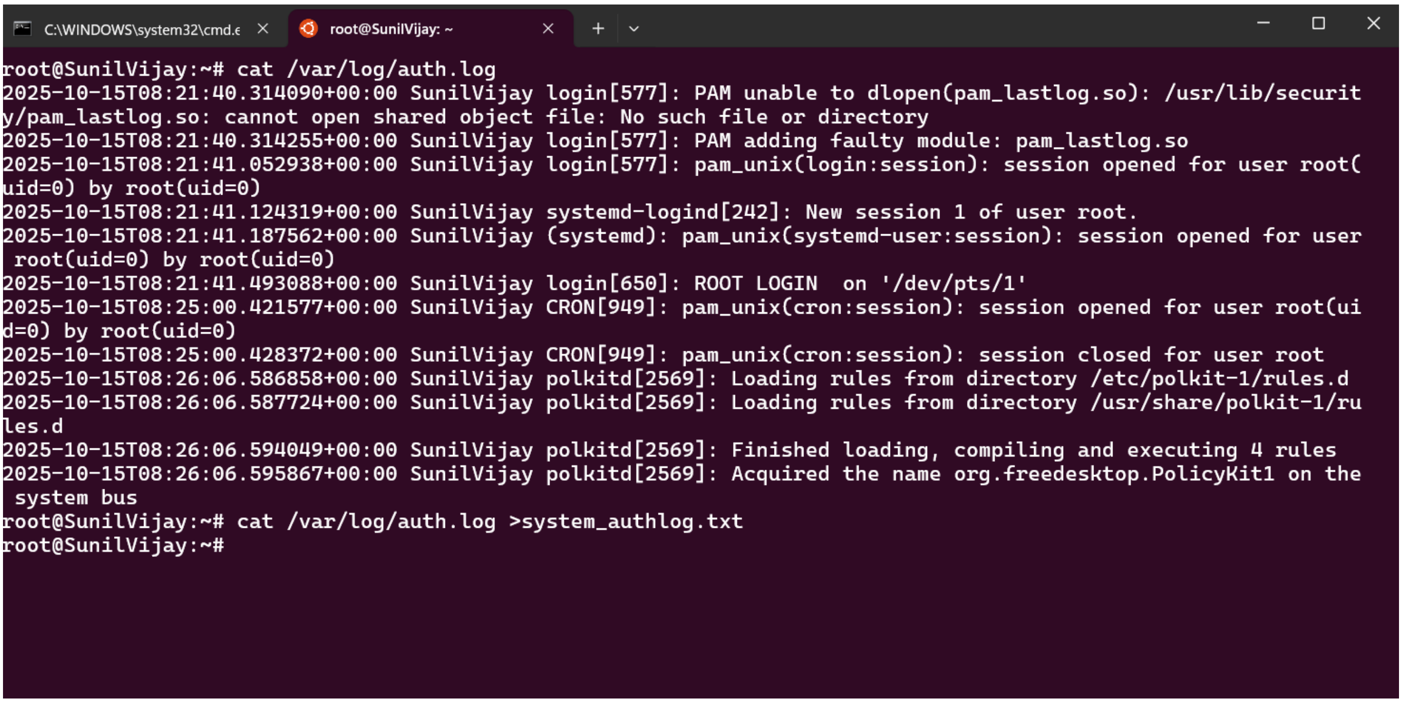
Analysis and Interpretation

The log entries indicate legitimate and authorized root user activity within the system, without any observed suspicious or malicious login attempts in the given excerpt. The absence of failed password messages or abnormal IP addresses suggests no evident external brute force attack or unauthorized login attempt at this time.

However, the missing PAM module error should be addressed as it may limit session logging and auditing capabilities, potentially reducing forensic visibility.

Conclusion

The analysis of the /var/log/auth.log entries from 15th October 2025 reveals expected system behavior with successful root logins and no detected suspicious login attempts or security incidents within the observed timeframe. Addressing the missing PAM module and continuing vigilant log monitoring is advised to uphold system security.



```
root@SunilVijay:~# cat /var/log/auth.log
2025-10-15T08:21:40.314090+00:00 SunilVijay login[577]: PAM unable to dlopen(pam_lastlog.so): /usr/lib/securit
y/pam_lastlog.so: cannot open shared object file: No such file or directory
2025-10-15T08:21:40.314255+00:00 SunilVijay login[577]: PAM adding faulty module: pam_lastlog.so
2025-10-15T08:21:41.052938+00:00 SunilVijay login[577]: pam_unix(login:session): session opened for user root(
uid=0) by root(uid=0)
2025-10-15T08:21:41.124319+00:00 SunilVijay systemd-logind[242]: New session 1 of user root.
2025-10-15T08:21:41.187562+00:00 SunilVijay (systemd): pam_unix(systemd-user:session): session opened for user
 root(uid=0) by root(uid=0)
2025-10-15T08:21:41.493088+00:00 SunilVijay login[650]: ROOT LOGIN  on '/dev/pts/1'
2025-10-15T08:25:00.421577+00:00 SunilVijay CRON[949]: pam_unix(cron:session): session opened for user root(ui
d=0) by root(uid=0)
2025-10-15T08:25:00.428372+00:00 SunilVijay CRON[949]: pam_unix(cron:session): session closed for user root
2025-10-15T08:26:06.586858+00:00 SunilVijay polkitd[2569]: Loading rules from directory /etc/polkit-1/rules.d
2025-10-15T08:26:06.587724+00:00 SunilVijay polkitd[2569]: Loading rules from directory /usr/share/polkit-1/ru
les.d
2025-10-15T08:26:06.594049+00:00 SunilVijay polkitd[2569]: Finished loading, compiling and executing 4 rules
2025-10-15T08:26:06.595867+00:00 SunilVijay polkitd[2569]: Acquired the name org.freedesktop.PolicyKit1 on the
 system bus
root@SunilVijay:~# cat /var/log/auth.log >system_authlog.txt
root@SunilVijay:~#
```