# **Linux Forensics**

**🐧 Linux Forensics – Notes**

**Task 1 – Introduction**

* Linux is widely used in servers, cloud, IoT, and even desktops.
* Forensics on Linux requires knowledge of its file system, logs, and configuration.
* Many tools/logs are text-based (easy to parse but volatile).

**Task 2 – Linux Forensics**

* Linux forensics involves analyzing:
  + System & user activity.
  + Configuration files.
  + Log files.
  + Persistence methods (cron jobs, services, etc.).
* Evidence is often in **plain text files** under /etc, /var/log, and user home directories.

**Task 3 – OS and Account Information**

* **Check OS details:**
  + /etc/os-release → distribution, version.
  + uname -a → kernel version, architecture.
* **Account info:**
  + /etc/passwd → user accounts.
  + /etc/shadow → password hashes.
  + /etc/group → group memberships.
* Look for suspicious users, privilege escalations.

**Task 4 – System Configuration**

* Important config files:
  + /etc/hosts → hostname ↔ IP mappings.
  + /etc/hostname → system hostname.
  + /etc/resolv.conf → DNS settings.
  + /etc/ssh/sshd\_config → SSH settings (check for weak configs, root login allowed).
* Check **authorized\_keys** in ~/.ssh/ for backdoor access.

**Task 5 – Persistence Mechanisms**

* **Cron jobs**:
  + /etc/crontab, /etc/cron.\*, and user crontabs.
* **Systemd services**:
  + /etc/systemd/system/ → custom or malicious services.
  + systemctl list-unit-files for enabled services.
* **Init scripts**:
  + /etc/init.d/
* Look for **suspicious startup scripts**.

**Task 6 – Evidence of Execution**

* **Bash history**:
  + ~/.bash\_history (check for deleted/hidden history).
* **Command logging**:
  + /var/log/auth.log (Debian/Ubuntu).
  + /var/log/secure (RHEL/CentOS).
* **Last executed commands**:
  + last, lastlog → user login history.
  + /var/log/wtmp, /var/log/btmp → successful/failed logins.

**Task 7 – Log Files**

* Key logs under /var/log/:
  + auth.log or secure → authentication events.
  + syslog or messages → system-wide events.
  + dmesg → kernel messages.
  + cron.log → cron jobs execution.
* Web servers:
  + /var/log/apache2/access.log or /var/log/nginx/access.log.

**Task 8 – Conclusion**

* Linux forensic artifacts are **mostly plain text files**.
* Core focus: accounts, persistence, execution evidence, and logs.
* Need to check:
  + System configs → unusual modifications.
  + Logs → brute force, suspicious commands.
  + Persistence → cron, systemd, SSH keys.

⚡ Pro tip: Always work on a **forensic copy** of Linux disk (not live system). Tools like Autopsy, Plaso/Log2Timeline, Sleuth Kit, or even simple grep/awk are widely used.

