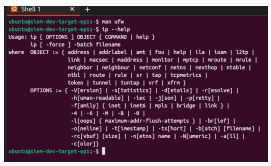
Linux Tutorial Part Two

In this article we will cover:

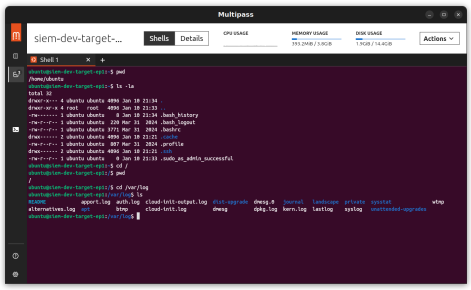
- **Directory Structure & Overview**
- **Basic File System Navigation**
- **File Manipulation Basics**
- **System Logs & Processes Overview**

> The ‘man’ and ‘help’ commands followed by another command will display the “man pages” “help” information for that command. This documentation is useful, especially without internet. Most commands can be followed by ‘-h’ or ‘—help’ to list usage and options. ✅ My biggest tip is learn how to research and find the answer. It’s okay if you don’t know or can’t remember everything, no one does. There’s a plethora of information on the internet, from vendor documentation and forums, to YouTube videos and blogs for almost anything. Finally, it’s also okay to ask for help if you’re struggling to find the answer.



Directory Structure & Overview

In linux, everything is a file. The base, or root, of the file system is represented as ‘ `/` ’ When you open a new terminal, you’re typically going to be in your current user’s home directory ‘ `/home/$USER` ', and have a ‘ `~` ’ next to your prompt. The same is true for ‘ `/` ’.

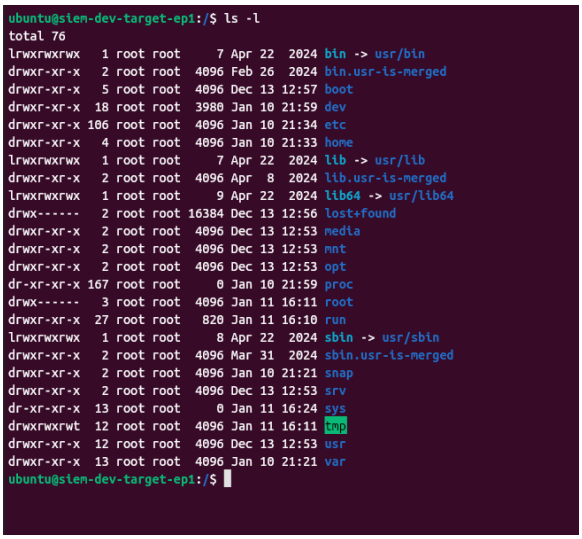


Use ‘ `pwd` ’ at any time to print the path of your current directory. Display the contents of the current directory with ‘ `ls` ’. You can also show “hidden” files by using ‘ `ls -a` ’, or get more info with ‘ `ls -la` ’ (for a long list), some distributions may slightly differ in aliases.

Basic File System Navigation

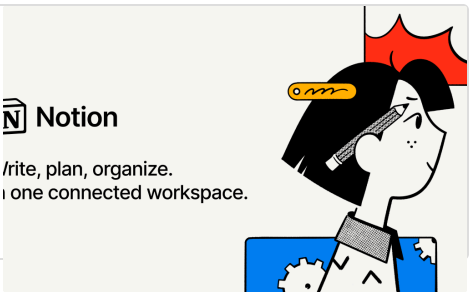
Changing directories can be done by using ‘ `cd` ’ followed by the directory name, or by the absolute path ’ `cd /var/log` ’. You can change to your home directory from anywhere with ‘ `cd ~` ’.

From the root of the file system, list the contents ‘ `ls -l` ’. You should see something similar to the output below.



Notice, along the left column, you’ll see the following data below. At a high level, these are permissions, ‘r’=read, ‘w’=write, ‘x’=execute. The first character tells you the “type”, ‘d’=directory(folder), ‘l’=link(symlink), and ‘-’=file. The next 3 are permissions for the “owner”, next 3 for the “group”, and the last 3 for “everyone else”. The example above shows 2 columns displaying “root” all the way down, the first space is the “owner”, second space is the “group”.

Find the hierarchy table here:

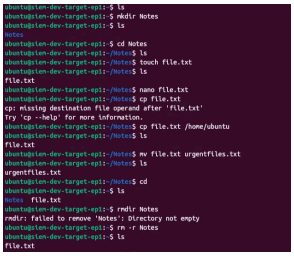


File Manipulation Basics

Creating, modifying, and deleting files is pretty straight forward. The commands we will need for now are `touch, mkdir, mv, rm, rmdir, nano, and cp.`

Start off by creating a new directory named “Notes” with ‘ `mkdir Notes` ’ then use ‘ `cd Notes` ’ to move into the directory. Obviously if we look, there’s nothing here, using ‘ `touch file.txt` ’ we can create a new file called file.txt. Nano is a simple text-based editor, use ‘ `nano`

file.txt ' to modify the new file, directions are in the photo below.



Now say you want to copy that file, using ' cp ' we can do just that. Also note for the sake of demonstration I got 2 error messages. When you want to copy a file, you'll need to specify where you want it to go.

Copy the file to your users home directory ' cp file.txt /home/$user '. You should have 2 copies now, 1 at /home/user/Notes and another at /home/user. Now rename the file in the Notes directory, the ' mv ' command can both move and rename files. Use ' mv file.txt urgentfiles.txt ' to rename the file, then assume it now contains sensitive data inside.

In the case you want to delete a directory, you can use ' rmdir ', however, the directory MUST already be empty for this command to work. If you ' mv

`urgentfiles.txt /home/user`', you may then delete the directory.

Alternatively, you could, with caution, use '`rm -r Notes`', this will recursively delete the directory and all of the contents. If you opted to move the urgentfiles.txt file it should be in your home directory with file.txt. Use '`rm file.txt`' to remove the old file.

Bonus: Shred

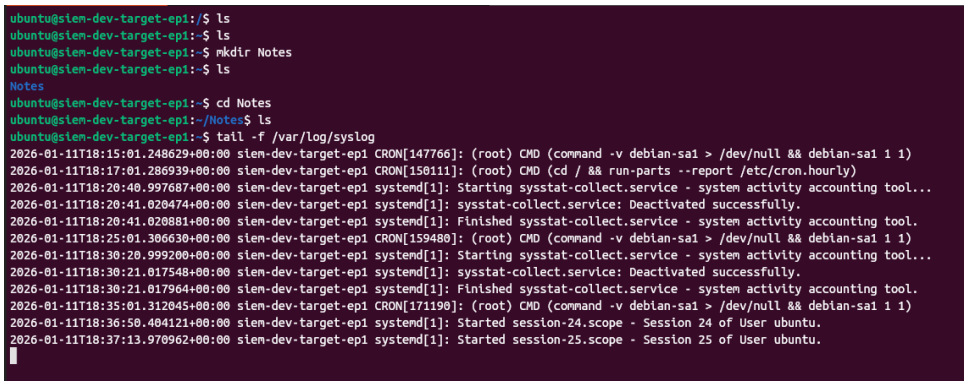
The '`shred`' command can be used to overwrite and zero out file contents prior to deletion to ensure sensitive data cannot be recovered. Use '`cat urgentfiles.txt`' to see the data beforehand. Use '`shred urgentfiles.txt`' and you'll see the data is no longer recognizable.

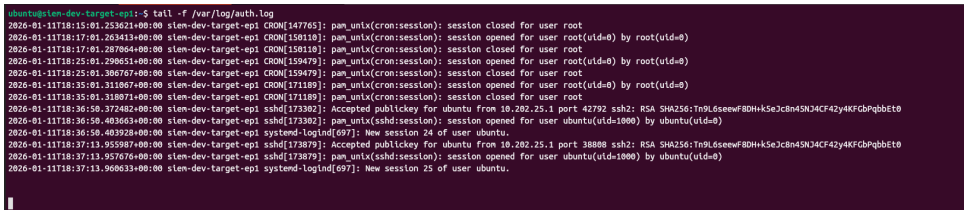
System Logs & Processes Overview

Modern Linux systems often use systemd's journald for centralized, structured logging. This also supports

real-time monitoring and advanced filtering via the ‘ `journalctl` ’ command.

Another option is to use ‘ `tail -f` ’ /var/log/syslog or /var/log/auth.log, which will display the most recent logs and follow any new events. This can be useful for diagnosing user login or connectivity issues, as you’ll see auth.log update in real time.





System logs collect information about the operating system, including boot messages, kernel activities, hardware events, and startup processes. These logs are essential for diagnosing issues and monitoring overall system health. When it comes to the location of specific logs, it can differ depending on the distribution used. In Ubuntu, you can find some logs in /var/log.

Some require a specific command to display the logs, as mentioned above, others you can open with '`nano`'. The /var/log/messages and /var/log/syslog files are central repositories for general system messages.

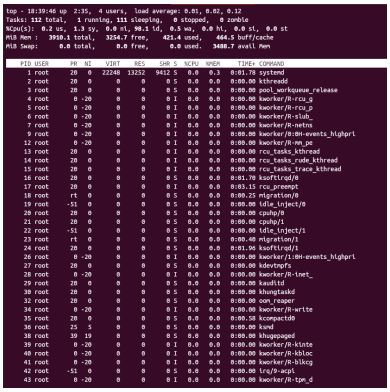
Application logs record events specific to individual software applications, such as errors, warnings, and user activities. These logs help in diagnosing application-specific problems and analyzing performance. (/var/log/httpd/access_log and /var/log/httpd/error_log for Apache)

Security logs monitor authentication attempts, access control, and authorization events. Files like /var/log/auth.log, /var/log/secure, and /var/log/faillog are used to track user logins, failed attempts, and privilege escalations, making them vital for detecting potential security breaches. The `last` command will display recent failed login attempts.

Processes: Overview

Processes are exactly what they sound like, stuff running on the machine. Management is out of scope for this, but here are some ways to view some process information.

The `top` command is a way to see running processes, consumption, health, etc and refreshes every 2* seconds.

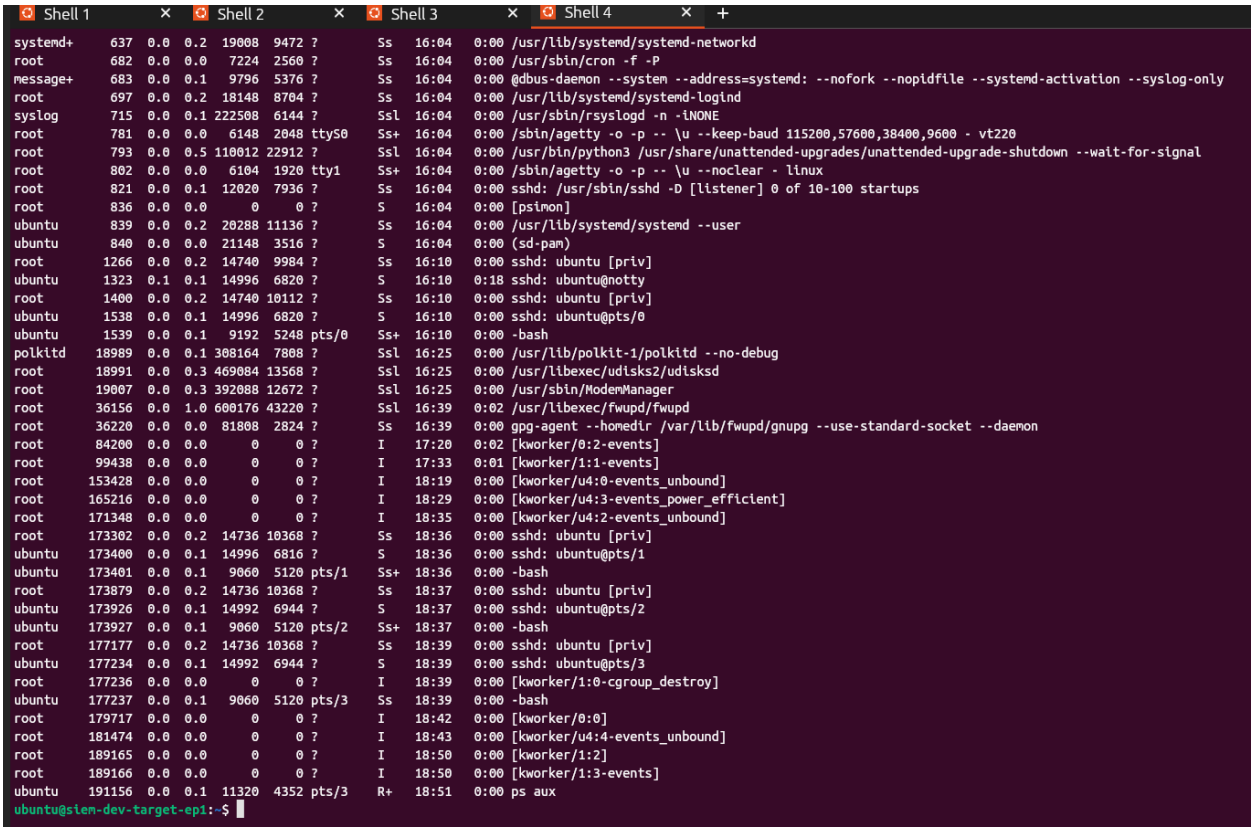


Using `pstree` can show the process hierarchy.

```
ubuntu@siem-dev-target-ep1:~$ pstree
systemd─┬─ModemManager───3*[{ModemManager}]
        ├─2*[agetty]
        ├─cron
        ├─dbus-daemon
        ├─fwupd───5*[{fwupd}]
        ├─gpg-agent
        ├─multipathd───6*[{multipathd}]
        ├─polkitd───3*[{polkitd}]
        ├─rsyslogd───3*[{rsyslogd}]
        ├─sshd─┬─sshd───sshd
        │      ├─2*[sshd───sshd───bash───tail]
        │      ├─sshd───sshd───bash───top
        │      └─sshd───sshd───bash───pstree
        ├─systemd───(sd-pam)
        ├─systemd-journal
        ├─systemd-logind
        ├─systemd-network
        ├─systemd-resolve
        ├─systemd-timesyn───{systemd-timesyn}
        ├─systemd-udevd
        ├─udisksd───5*[{udisksd}]
        └─unattended-upgr───{unattended-upgr}
```

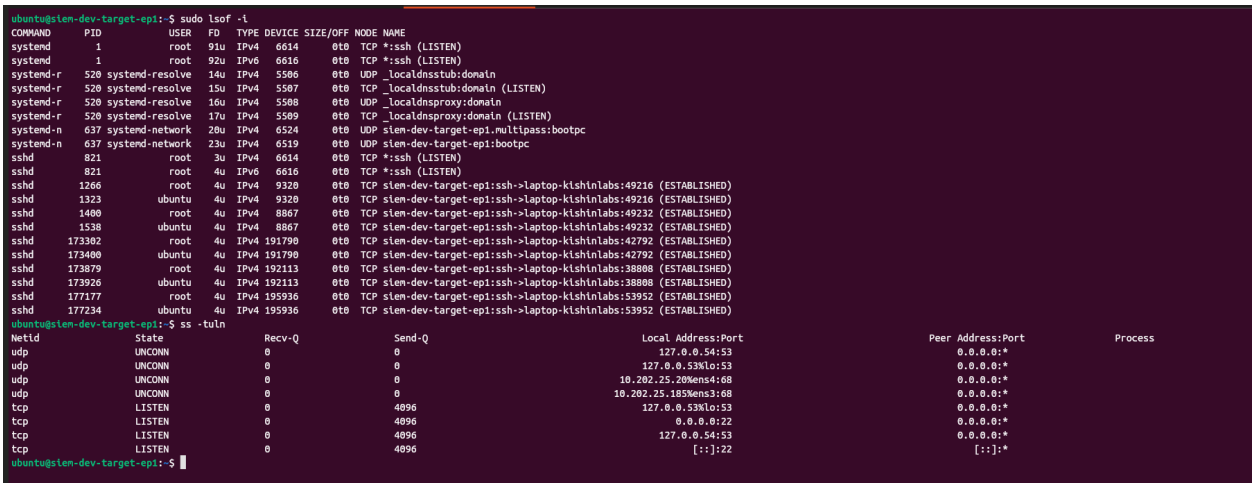
Using `ps aux` :

```
ubuntu@siem-dev-target-ep1:~$ ps aux
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  0.3  22248 13252 ?        Ss   16:03   0:01 /sbin/init
root           2  0.0  0.0      0     0 ?        S    16:03   0:00 [kthreadd]
root           3  0.0  0.0      0     0 ?        S    16:03   0:00 [pool_workqueue_release]
root           4  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-rcu_g]
root           5  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-rcu_p]
root           6  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-slub_]
root           7  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-netns]
root           9  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/0:0H-events_highpri]
root          12  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-mm_pe]
root          13  0.0  0.0      0     0 ?        I    16:03   0:00 [rcu_tasks_kthread]
root          14  0.0  0.0      0     0 ?        I    16:03   0:00 [rcu_tasks_rude_kthread]
root          15  0.0  0.0      0     0 ?        I    16:03   0:00 [rcu_tasks_trace_kthread]
root          16  0.0  0.0      0     0 ?        S    16:03   0:01 [ksoftirqd/0]
root          17  0.0  0.0      0     0 ?        I    16:03   0:03 [rcu_preempt]
root          18  0.0  0.0      0     0 ?        S    16:03   0:00 [migration/0]
root          19  0.0  0.0      0     0 ?        S    16:03   0:00 [idle_inject/0]
root          20  0.0  0.0      0     0 ?        S    16:03   0:00 [cpuhp/0]
root          21  0.0  0.0      0     0 ?        S    16:03   0:00 [cpuhp/1]
root          22  0.0  0.0      0     0 ?        S    16:03   0:00 [idle_inject/1]
root          23  0.0  0.0      0     0 ?        S    16:03   0:00 [migration/1]
root          24  0.0  0.0      0     0 ?        S    16:03   0:02 [ksoftirqd/1]
root          26  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/1:0H-events_highpri]
root          27  0.0  0.0      0     0 ?        S    16:03   0:00 [kdevtmpfs]
root          28  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-inet_]
root          29  0.0  0.0      0     0 ?        S    16:03   0:00 [kauditd]
root          30  0.0  0.0      0     0 ?        S    16:03   0:00 [khungtaskd]
root          32  0.0  0.0      0     0 ?        S    16:03   0:00 [oom_reaper]
root          34  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-write]
root          35  0.0  0.0      0     0 ?        S    16:03   0:00 [kcompactd0]
root          36  0.0  0.0      0     0 ?        SN   16:03   0:00 [ksmd]
root          38  0.0  0.0      0     0 ?        SN   16:03   0:00 [khugepaged]
root          39  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-kinte]
root          40  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-kbloc]
root          41  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-blkcg]
root          42  0.0  0.0      0     0 ?        S    16:03   0:00 [irq/9-acpi]
root          43  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-tpm_d]
root          44  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-ata_s]
root          45  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-md]
root          46  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-md_bi]
root          47  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-edac-]
root          48  0.0  0.0      0     0 ?        I<   16:03   0:00 [kworker/R-devfr]
```



Since everything in linux is a file, the ‘ `lsof` ’ command (list open files) is powerful. Using ‘ `sudo lsof -i` ' will show open “files” with IP, protocol information, and more.

Lastly, another command to consider getting familiar with is ‘ `ss` ' . Check out the options and flags and give them a shot. I have included an example with ‘ `ss -tuln` ’. Info can be piped or filtered via ‘ `grep` ’.



```
ubuntu@siem-dev-target-ep1:~$ sudo lsof -i
COMMAND      PID            USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
systemd        1            root   91u  IPv4   6614      0t0  TCP *:ssh (LISTEN)
systemd        1            root   92u  IPv6   6616      0t0  TCP *:ssh (LISTEN)
systemd-r    520 systemd-resolve   14u  IPv4   5506      0t0  UDP _localdnsstub:domain
systemd-r    520 systemd-resolve   15u  IPv4   5507      0t0  TCP _localdnsstub:domain (LISTEN)
systemd-r    520 systemd-resolve   16u  IPv4   5508      0t0  UDP _localdnsproxy:domain
systemd-r    520 systemd-resolve   17u  IPv4   5509      0t0  TCP _localdnsproxy:domain (LISTEN)
systemd-n    637 systemd-network   20u  IPv4   6524      0t0  UDP siem-dev-target-ep1.multipass:bootpc
systemd-n    637 systemd-network   23u  IPv4   6519      0t0  UDP siem-dev-target-ep1:bootpc
sshd         821            root    3u  IPv4   6614      0t0  TCP *:ssh (LISTEN)
sshd         821            root    4u  IPv6   6616      0t0  TCP *:ssh (LISTEN)
sshd        1266            root    4u  IPv4   9320      0t0  TCP siem-dev-target-ep1:ssh->laptop-kishinlabs:49216 (ESTABLISHED)
sshd        1323          ubuntu    4u  IPv4   9320      0t0  TCP siem-dev-target-ep1:ssh->laptop-kishinlabs:49216 (ESTABLISHED)
sshd        1400            root    4u  IPv4   8867      0t0  TCP siem-dev-target-ep1:ssh->laptop-kishinlabs:49232 (ESTABLISHED)
sshd        1538          ubuntu    4u  IPv4   8867      0t0  TCP siem-dev-target-ep1:ssh->laptop-kishinlabs:49232 (ESTABLISHED)
sshd      173302            root    4u  IPv4 191790      0t0  TCP siem-dev-target-ep1:ssh->laptop-kishinlabs:42792 (ESTABLISHED)
sshd      173400          ubuntu    4u  IPv4 191790      0t0  TCP siem-dev-target-ep1:ssh->laptop-kishinlabs:42792 (ESTABLISHED)
sshd      173879            root    4u  IPv4 192113      0t0  TCP siem-dev-target-ep1:ssh->laptop-kishinlabs:38808 (ESTABLISHED)
sshd      173926          ubuntu    4u  IPv4 192113      0t0  TCP siem-dev-target-ep1:ssh->laptop-kishinlabs:38808 (ESTABLISHED)
sshd      177177            root    4u  IPv4 195936      0t0  TCP siem-dev-target-ep1:ssh->laptop-kishinlabs:53952 (ESTABLISHED)
sshd      177234          ubuntu    4u  IPv4 195936      0t0  TCP siem-dev-target-ep1:ssh->laptop-kishinlabs:53952 (ESTABLISHED)
ubuntu@siem-dev-target-ep1:~$ ss -tuln
Netid               State                Recv-Q               Send-Q                              Local Address:Port                              Peer Address:Port              Process
udp                 UNCONN               0                    0                                      127.0.0.54:53                                   0.0.0.0:*
udp                 UNCONN               0                    0                                 127.0.0.53%lo:53                                     0.0.0.0:*
udp                 UNCONN               0                    0                              10.202.25.20%ens4:68                                    0.0.0.0:*
udp                 UNCONN               0                    0                             10.202.25.185%ens3:68                                    0.0.0.0:*
tcp                 LISTEN               0                    4096                               127.0.0.53%lo:53                                     0.0.0.0:*
tcp                 LISTEN               0                    4096                                    0.0.0.0:22                                     0.0.0.0:*
tcp                 LISTEN               0                    4096                                   127.0.0.54:53                                   0.0.0.0:*
tcp                 LISTEN               0                    4096                                       [::]:22                                        [::]:*
ubuntu@siem-dev-target-ep1:~$
```

This is a high level overview but we are only on part 2. (:

- ***Alex a.k.a KishinInfosec 2025***

Hierarchy