CTF-Lite [Challenge]

> **Thursday, July-3 2025**

📁 CTF-Lite — File Permission Challenge

▼ Task 1: Recursive listing

Command: `ls -alR ~/CTF-Lite`

▼ **OUTPUT:**

> ls -alR ~/CTF-Lite

> /home/jynx/CTF-Lite:
> total 12
> drwxr-xr-x 3 jynx jynx 4096 Jul 3 13:30 .
> drwx------ 20 jynx jynx 4096 Jul 3 13:30 ..
> drwxr-xr-x 4 jynx jynx 4096 Jul 3 13:30 staging

> /home/jynx/CTF-Lite/staging:
> total 20
> drwxr-xr-x 4 jynx jynx 4096 Jul 3 13:30 .
> drwxr-xr-x 3 jynx jynx 4096 Jul 3 13:30 ..
> -rw-r--r-- 1 jynx jynx   25 Jul 3 13:30 notes.txt
> -rw-r--r-- 1 jynx jynx   0 Jul 3 13:30 README.md
> drwxr-xr-x 2 jynx jynx 4096 Jul 3 13:30 secrets
> drwxr-xr-x 3 jynx jynx 4096 Jul 3 13:30 vault

> /home/jynx/CTF-Lite/staging/secrets:
> total 12

> drwxr-xr-x 2 jynx jynx 4096 Jul 3 13:30 .
> drwxr-xr-x 4 jynx jynx 4096 Jul 3 13:30 ..
> -rw-r--r-- 1 jynx jynx   22 Jul 3 13:30 .invisible.txt

> /home/jynx/CTF-Lite/staging/vault:
> total 12
> drwxr-xr-x 3 jynx jynx 4096 Jul 3 13:30 .
> drwxr-xr-x 4 jynx jynx 4096 Jul 3 13:30 ..
> ---------- 1 jynx jynx   0 Jul 3 13:30 passwd_shadow
> -rwsr-xr-x 1 root root   0 Jul 3 13:30 root_exploit.sh
> drwxr-xr-x 2 jynx jynx 4096 Jul 3 13:30 .ssh

> /home/jynx/CTF-Lite/staging/vault/.ssh:
> total 12
> drwxr-xr-x 2 jynx jynx 4096 Jul 3 13:30 .
> drwxr-xr-x 3 jynx jynx 4096 Jul 3 13:30 ..
> -rw------- 1 jynx jynx   12 Jul 3 13:30 id_rsa

▼ SUMMARY:

Problematic Files:

- root_exploit.sh - seems suspicious, exploitable of sorts, more so because executable by user, group and others [777 permission]. All the more as its set to SUID bit.
- passwd_shadow - no permission [000], weird and raises concerns due to its nature itself and more so because of permissions.
- .invisible.txt - a hidden file, why would a normal text file be hidden, although its empty, it has read permissions [644] for all the levels, which can raise flags.
- .ssh - Found .ssh directory inside “/vault/”, normally, “.ssh/” is part of a user’s home folder. But here, it exists in an isolated vault directory. Contains a private key with restrictive permissions (600). This might

indicate credential leakage or staged access. Worth investigating further if this was an exfil point or planted key.

▼ TAKEAWAYS:

Understanding file permissions, nature of file creation and how to develop an investigative approach towards suspicious and alarming files.

▼ Task 2: Find files with world-writable permissions (777 or similar)

Command: `ls -alR ~/CTF-Lite | grep "rwxrwxrwx"`

▼ OUTPUT:

> No file with worldwide permissions.

▼ SUMMARY:

Problematic Files:

- Unable to find any problematic files in the sense, that aren't mentioned and reported as escalations previously, since no files are granted '777' world- writable permissions I'd leave this section to itself.

▼ Task 3: Find all files with `SUID` bit set

Command: `$ find . -perm -4000 -ls`

▼ OUTPUT:

> 3019047 0 -rwsr-xr-x 1 root root 0 Jul 3 13:30 ./staging/vault/root_exploit.sh

▼ SUMMARY:

Problematic Files:

- root_exploit.sh - seems RED FLAG, exploitable of sorts, as mentioned in TASK-1 itself. It was also reported this file has SUID bit [SUID allows

users to execute files with the permissions of the file owner (often root)], which is alarming in itself.

▼ TAKEAWAYS:

Understanding security concerns surrounding SUID bits and how un-examined files can lead to security vulnerabilities and potent harm to system itself.

SUID Bits?

→ SUID (Set User ID) is a special permission that allows a program to run with the privileges of the file's owner rather than the user who executes it.

→ SUID is a security concern because it can allow regular users to execute programs with elevated privileges (often root), creating potential attack vectors if the SUID program has vulnerabilities that can be exploited to gain unauthorized system access or if malicious SUID binaries are installed.

▼ Task 4:

Find all files NOT owned by you (check owner)

Command: `find ~ -type f ! -user $(whoami) -ls 2>/dev/null`

▼ OUTPUT:

> No output since I created the whole CTF question myself. It does although show all the files created by root user or system.

▼ SUMMARY:

Problematic Files:

- None to mention

▼ TAKEAWAYS:

Understanding how to write multi layered command and understand how collaborative Linux command can dangerously be effective to use.

▼ Task 5:
Find any files with no permissions at all (`----------`)

Command: `ls -alhR | grep "^----------"`

▼ OUTPUT:

> ***--------- 1 jynx jynx 0 Jul 3 13:30 passwd_shadow***

▼ SUMMARY:

Problematic Files:

- ***passwd_shadow*** is an alarming file from the beginning as mentioned in Task-1, no permissions for any user, group or others sets raise flags and suspicion, and if for a stance we say it is not harmful, in that case what is the purpose served by this file in any shape or form, since its inaccessible even to the user [owner].

▼ TAKEAWAYS:

Understanding how file permissions can be used to decide conditions and assess security threats and vulnerabilities. Skepticism and monitoring of file permissions can aid in tackling serious security lapse.

▼ Task 6:
Try reading the fake private key

Command: `-[~/CTF-Lite/staging/vault/.ssh] -$ cat id_rsa`

▼ OUTPUT:

> Private Key

▼ SUMMARY:

Problematic Files:

- I assessed in Task 1 that maybe there was an actual genuine PRIVATE KEY of a RSA algo, but the PRIVATE KEY is empty and seems pretty fabricated and raises genuine concerns as what might be the intent towards creating the file, is it a - ***red herring*** of sort? Interesting to say the least.

▼ TAKEAWAYS:

Understanding how attackers and serious security invasion gate-way which could turn out to be worse if missed priorly.

▼ Task 7: Search for all `.txt` or `.log` files with `find`

Command: `find ~/CTF-Lite \( -name "*.txt" -o -name *".log" \) -ls`

▼ OUTPUT:

```
3019042      4 -rw-r--r--   1 jynx     jynx          22 Jul  3
13:30 /home/jynx/CTF-Lite/staging/secrets/.invisible.txt
3019040      4 -rw-r--r--   1 jynx     jynx          25 Jul  3
13:30 /home/jynx/CTF-Lite/staging/notes.txt
```

▼ SUMMARY:

Problematic Files:

- Nothing problematic as such, intrinsically with a .txt or .log file, because Linux doesn't anyway consider or understand .extensions of any kind, it treats all of them same as ‘‘text files”. However, its a vice and a virtue- log or exe (executable) files are often disguised as simple .doc or .txt or .pdf files and are executed as soon as accessed by ill-informed users often in addition SUID bits which leads to invasive attempts by hackers

into gaining root privileges which they are essentially more often than not they are trying to achieve.

▼ TAKEAWAYS:

Understanding how attackers and hackers can trick ill-informed or novice users into opening and executing batch/executable/log file(s) disguised under the face of txt, doc or pdf files which are generally considered to be non-harmful as such.