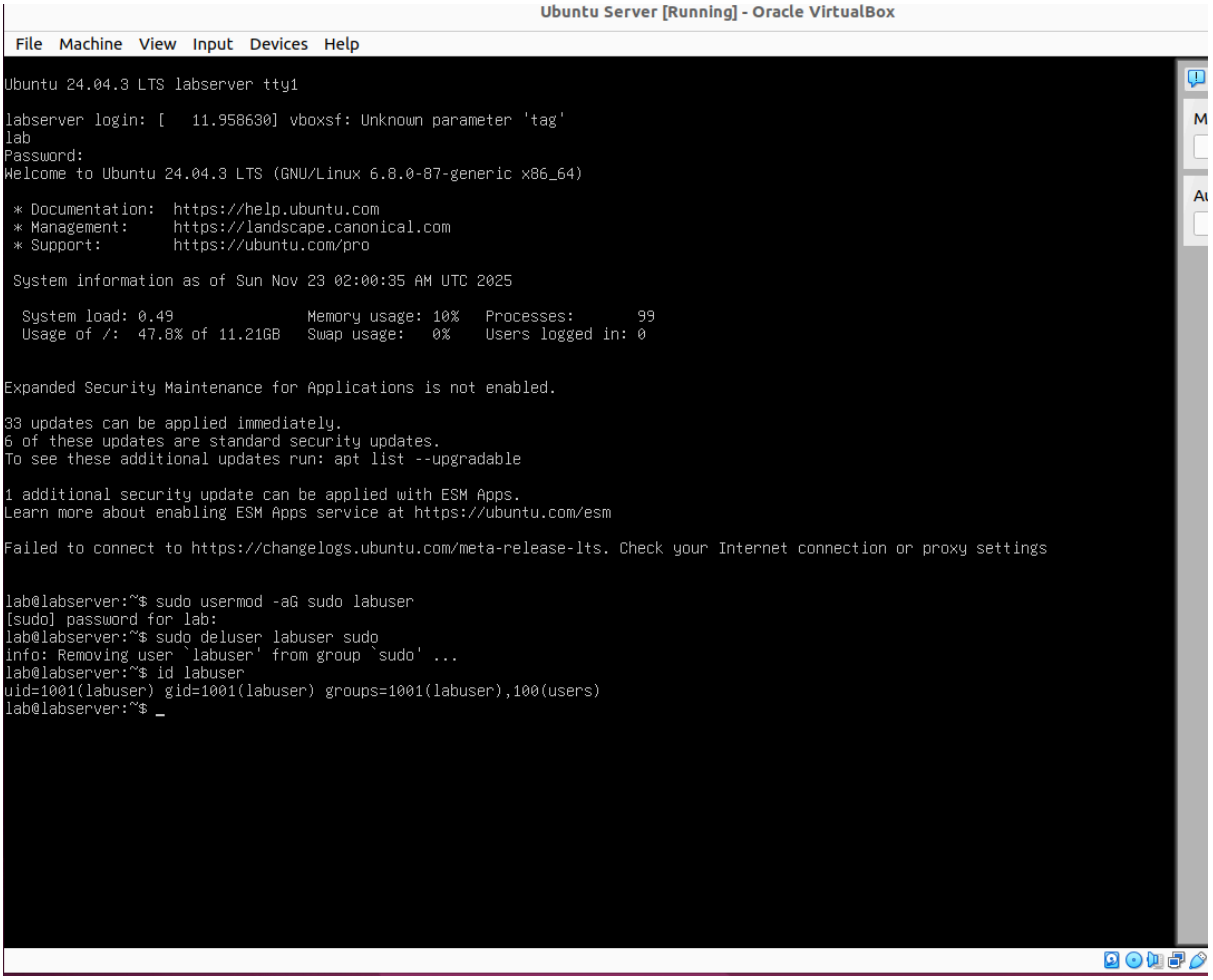
Linux Privilege Escalation

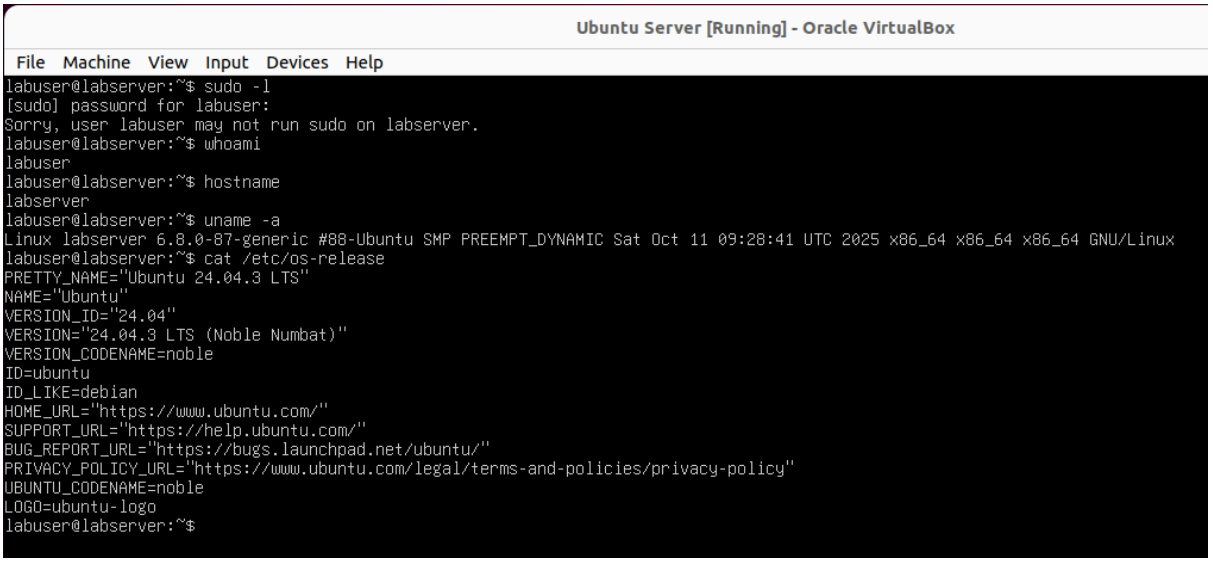
I made a user named labuser for the lab.
sudo adduser labuser
sudo usermod -aG sudo labuser
sudo deluser labuser sudo



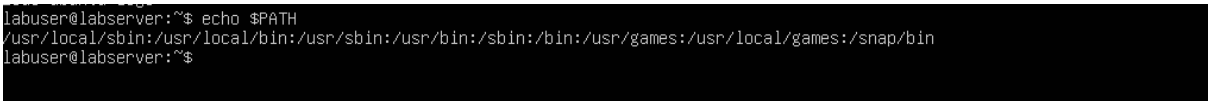
I checked sudo privileges using sudo -l

User labuser cannot run any sudo commands.

Then, I checked the OS, kernel, and environment.



This showed the OS version and kernel build. This is recent and unlikely vulnerable to known kernel exploits.
I checked the PATH.

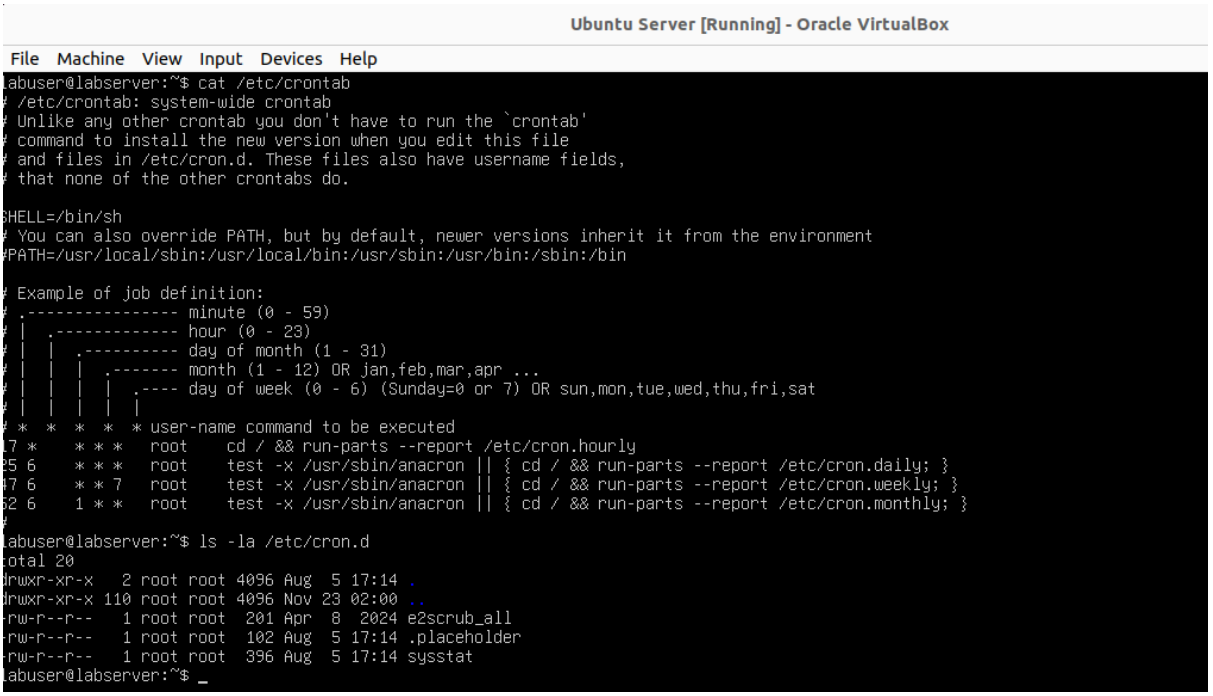


A misconfigured PATH which is writable directories before system paths can allow PATH hijacking. The path was correctly ordered and had no writable directories.

SUID binaries

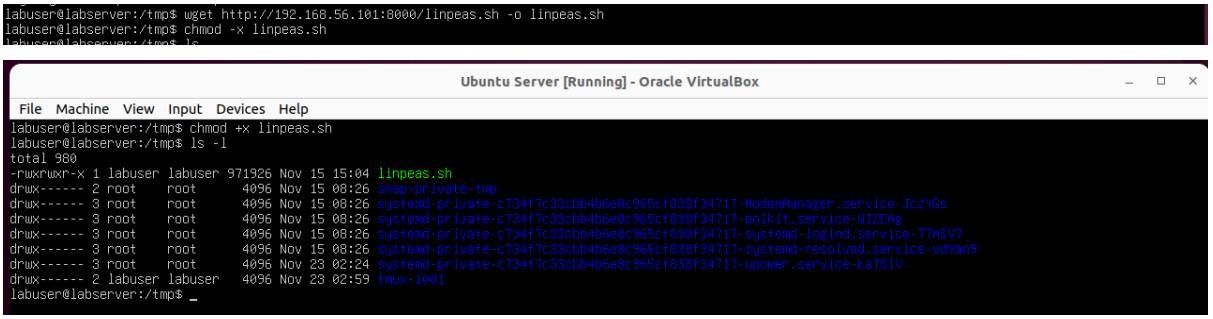
SUID binaries run as the file owner(root). No unusual or dangerous SUID binaries were found.

Cron jobs

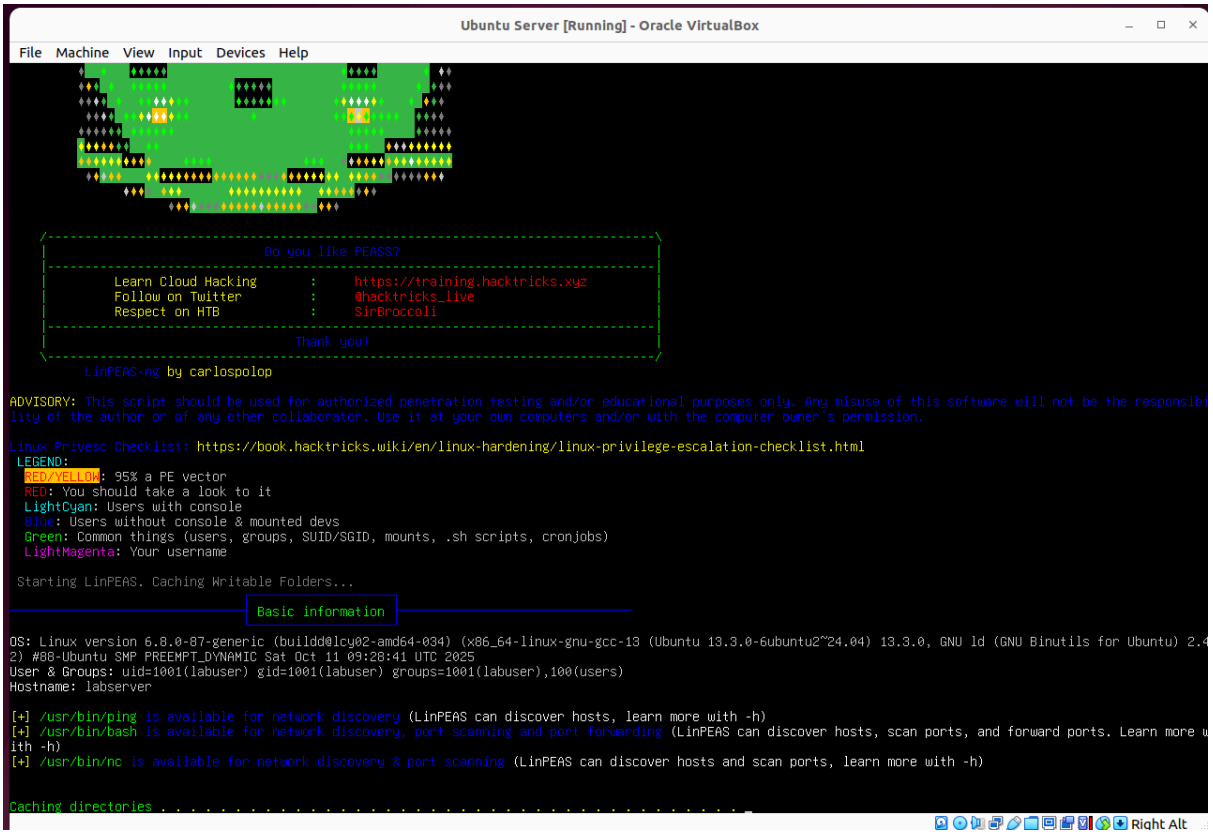


Cron tasks run automatically; if root runs a writable script, that becomes a privilege escalation vector. All cron entries are default and non-writable, so no cron-based escalation path exists.

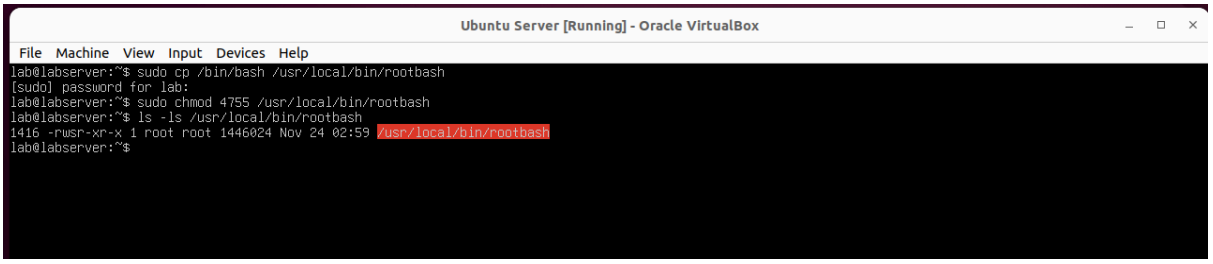
Next, I used linpeas which is a script which will search for all possible paths to escalate privileges on Linux hosts on Kali.
I downloaded linpeas on Kali, hosted with python3 -m http.server 8000, so ubuntu server could download linpeas.



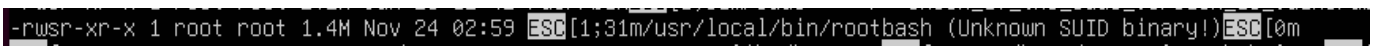
I ran ./linpeas.sh | tee linpeas.log



After completing manual enumeration and running linpeas on the Ubuntu server, there was nothing identified. It's because the machine is fully patched, and does not have common misconfigurations. To continue this lab, I decided to add a controlled vulnerability on the system. This allowed me to test the privesc process.



After I ran linpeas.sh, I got



This showed -rwsr-xr-x 1 root root 1.4M Nov 24 02:59 /usr/local/bin/rootbash (Unknown SUID binary!)

After adding the SUID misconfiguration on the Ubuntu server, the second linpeas scan correctly flagged `/usr/local/bin/rootbash` as an 'Unknown SUID binary.' Linpeas highlights this type of file because it is not part of the standard OS installation and it runs with elevated privileges due to the SUID bit. Any non-privileged user can execute the file and immediately obtain a root shell. This confirms that the SUID configuration represents an intentional privilege-escalation vulnerability.