MGS 650 Information Assurance
Lab 5: System Hardening
Submitted by: Akhilesh Anand Undralla
12/06/2021

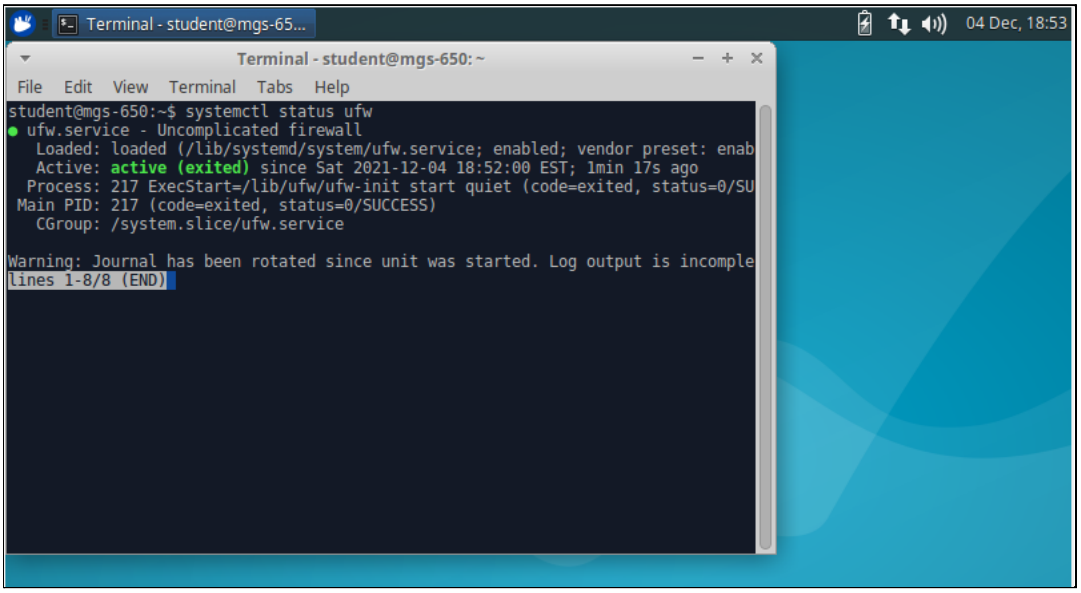


Figure 1.0: checking status of service using the command systemctl status ufw

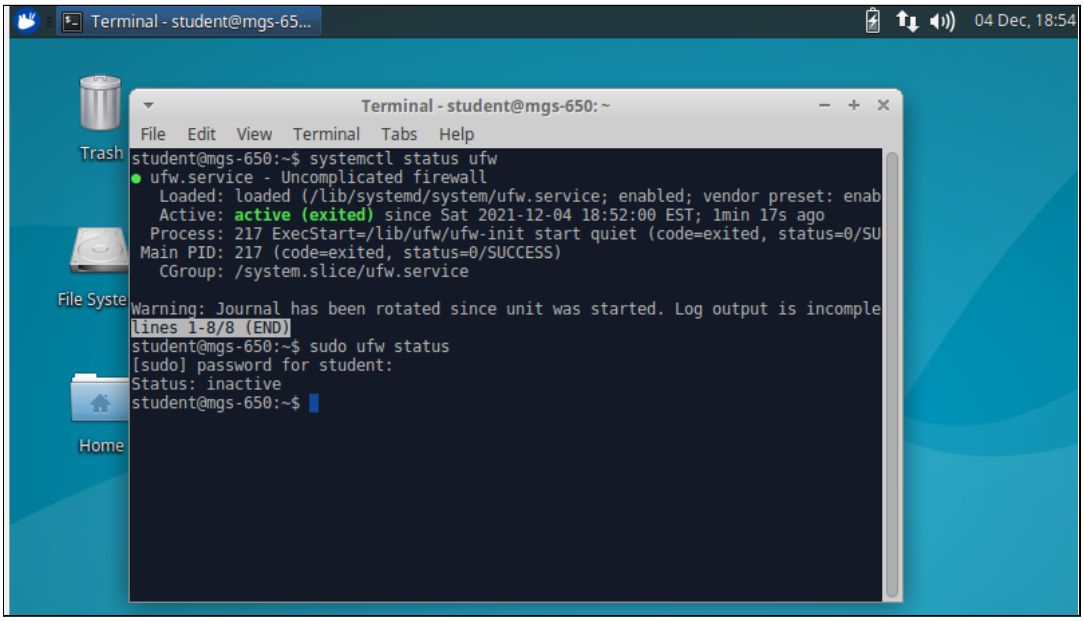


Figure 1.1: checking status of service and firewall using the command systemctl status ufw and sudo ufw status

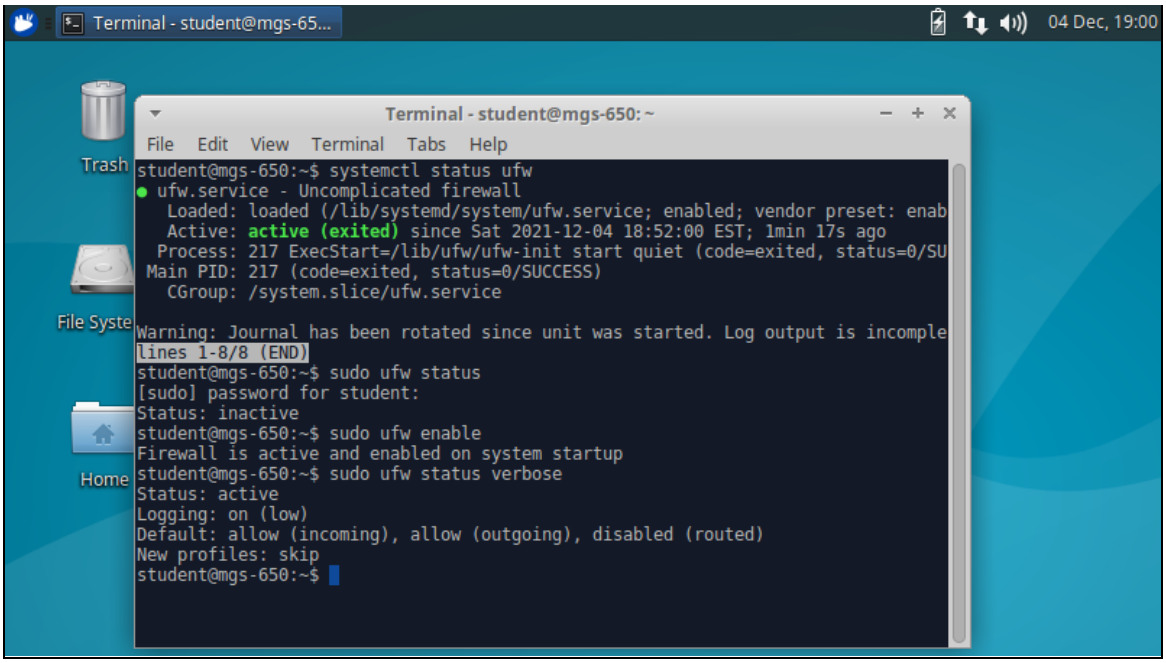


Figure 1.2: Enabling the firewall using sudo ufw enable and checking the status with the command sudo ufw status verbose

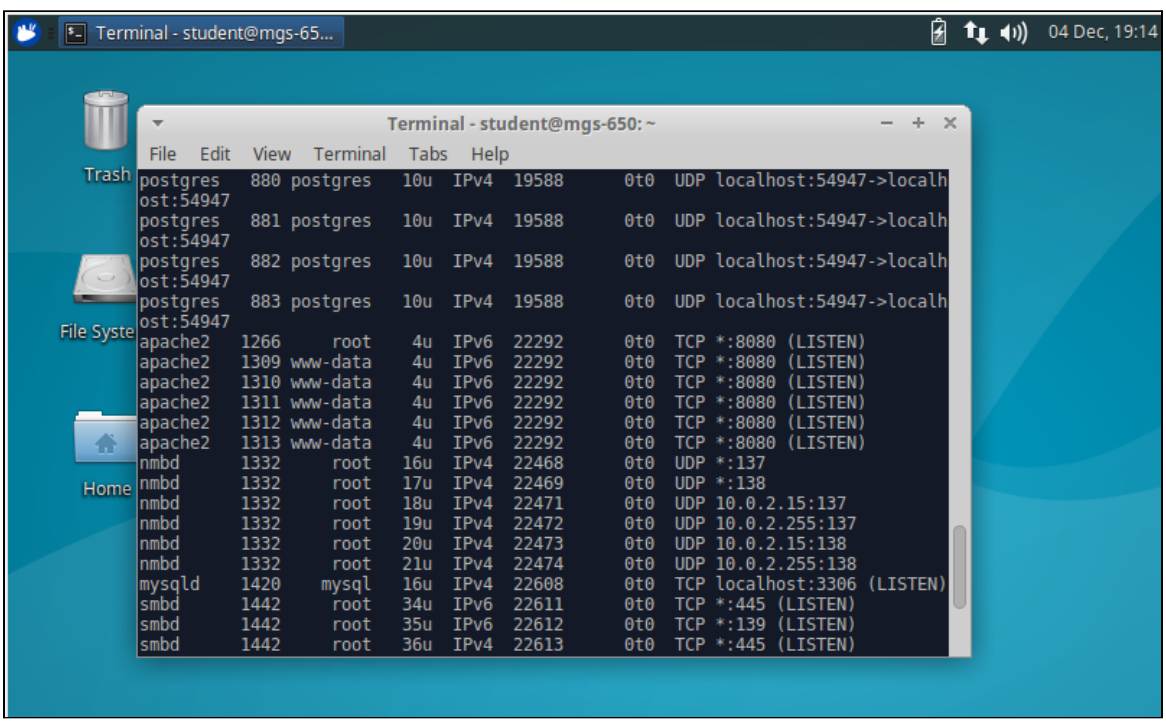


Figure 1.3: sudo lsof -i -P command shows currently-running programs that are using the network

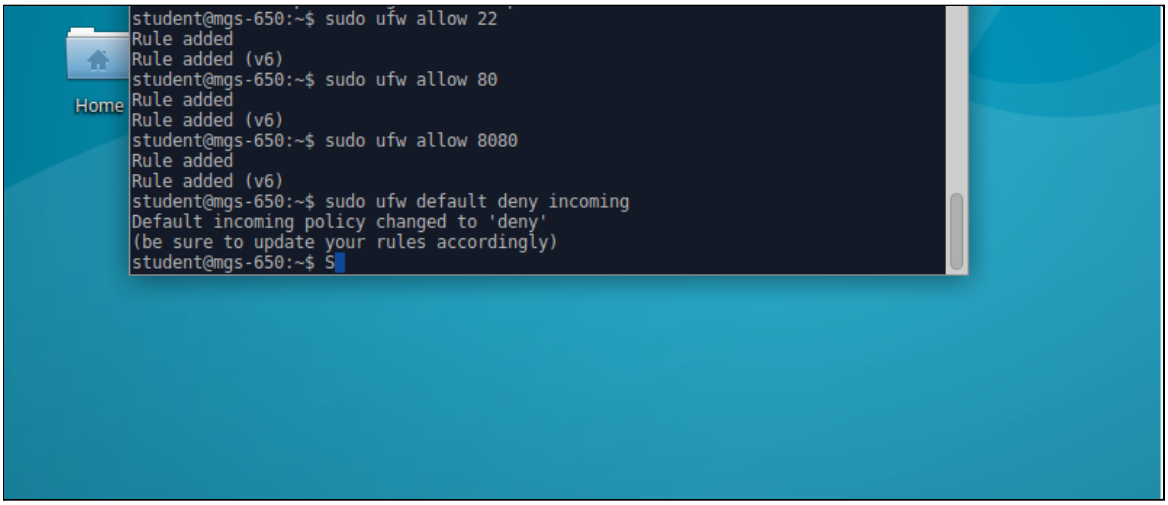


Figure 1.4: Firewall rule that allows inbound SSH traffic (22), web services(80 & 8080) with sudo ufw allow 22 and default policy to deny incoming traffic using sudo ufw default deny incoming

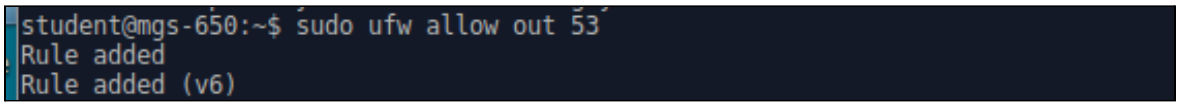


Figure 1.5: Allow the three ports 53 for outgoing communications with sudo ufw allow out 53

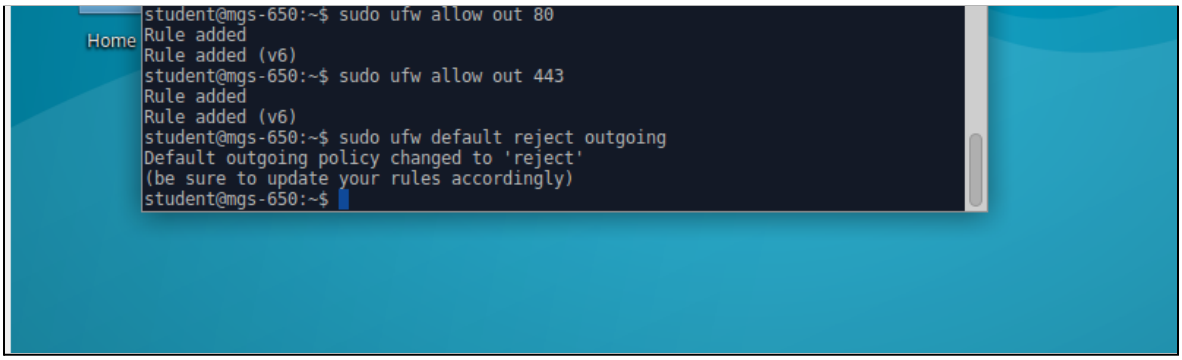


Figure 1.6: Allow the ports (80, 443) for outgoing communications and default policy to reject remaining with sudo ufw default reject outgoing

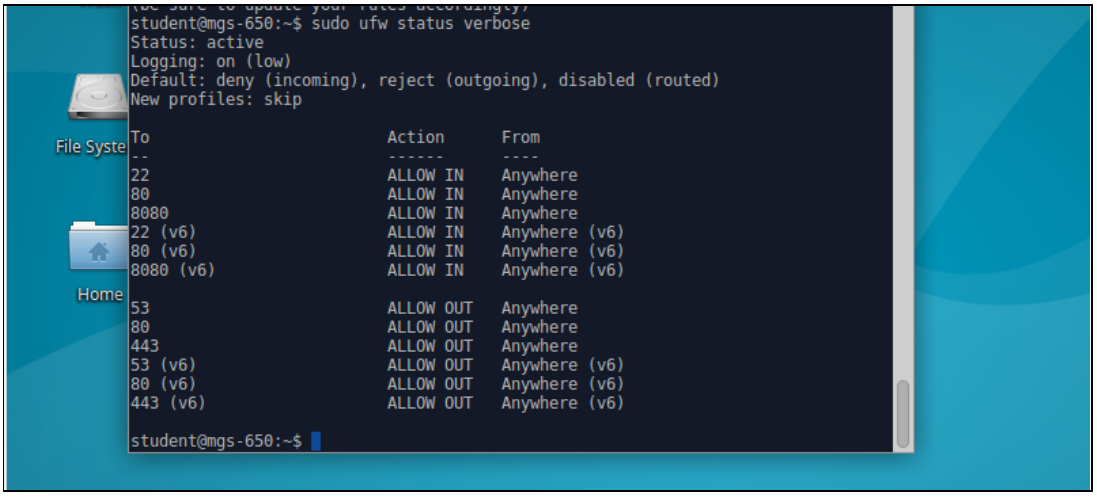


Figure 1.7: Current Firewall status after setting network rules (verbose)

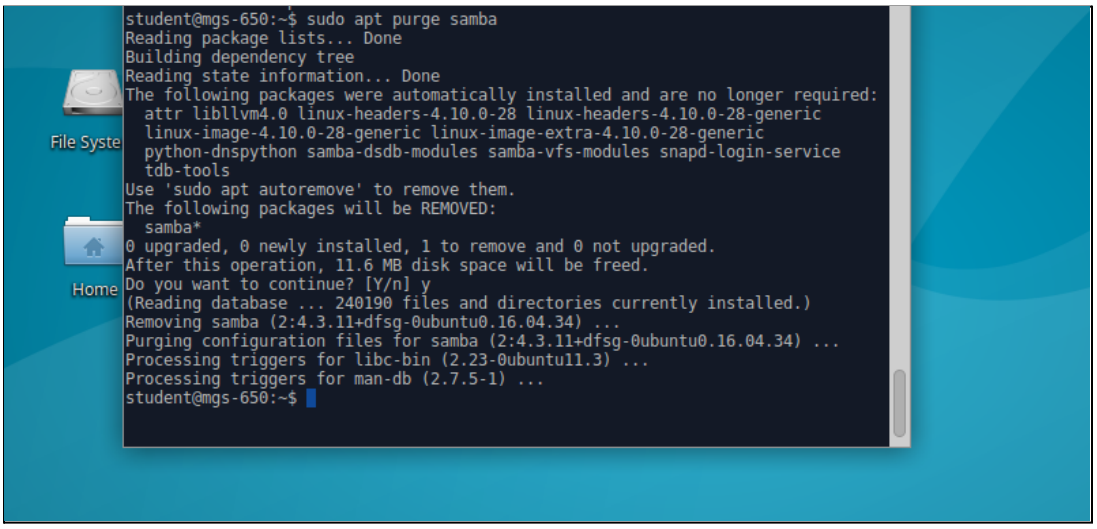


Figure 1.8: Disabling smbd as a running service using sudo apt purge samba

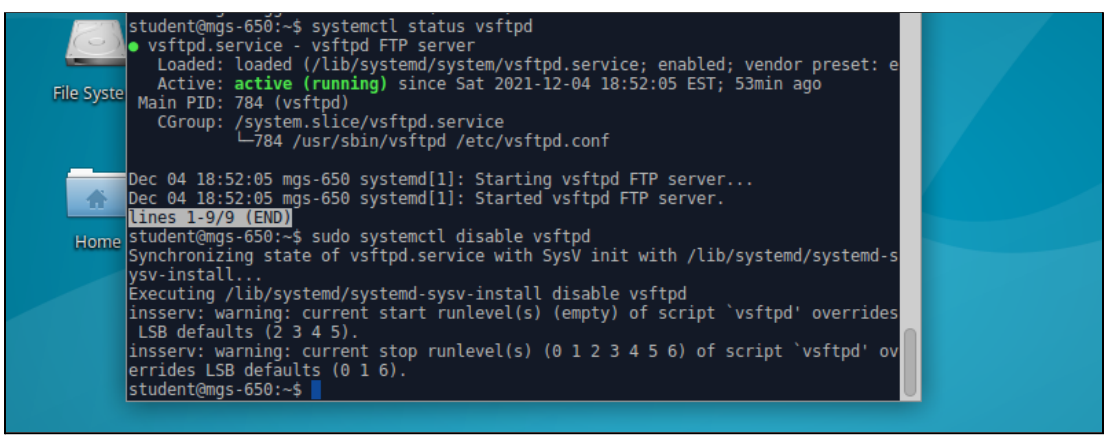
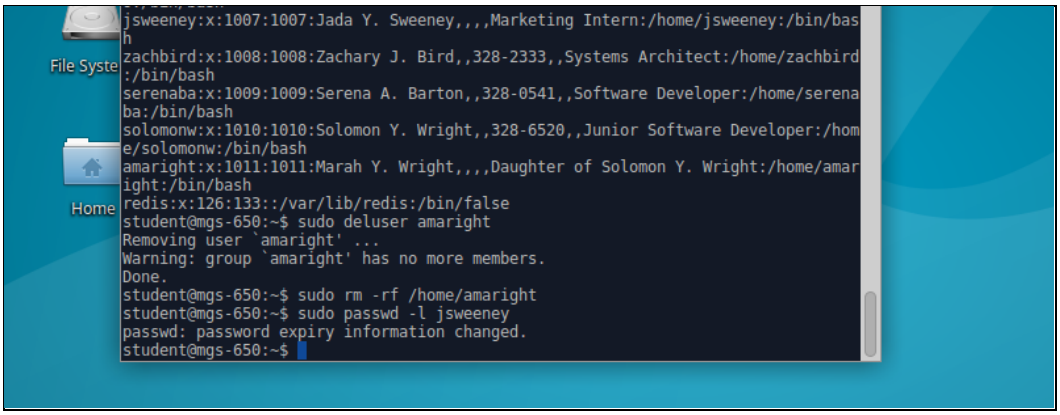
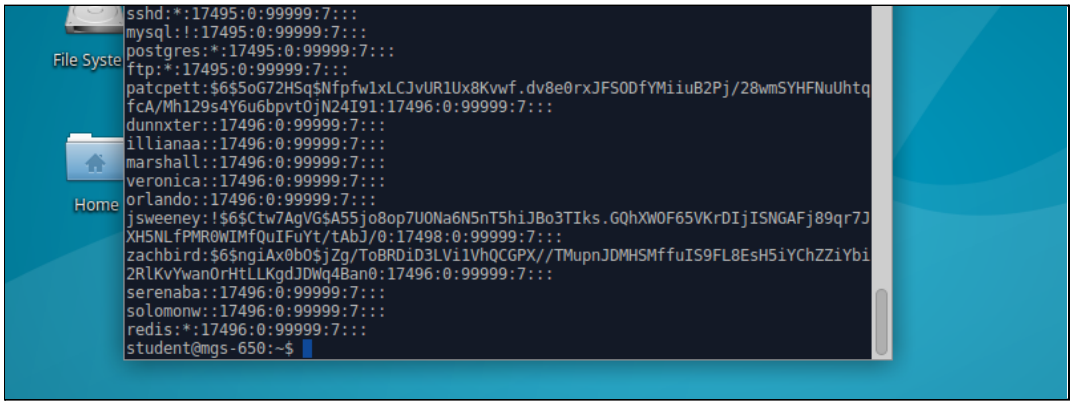


Figure 1.9: Disabling vsftpd as a running service using sudo apt purge vsftpd



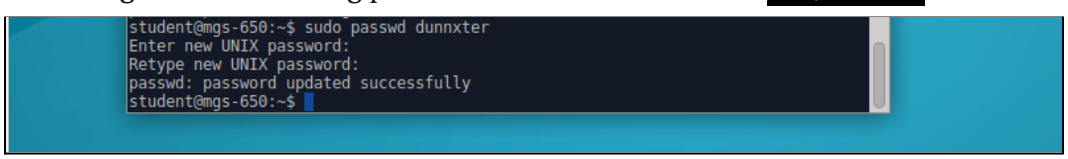
```
jsweeney:x:1007:1007:Jada Y. Sweeney,,,,Marketing Intern:/home/jsweeney:/bin/bas
h
zachbird:x:1008:1008:Zachary J. Bird,,328-2333,,Systems Architect:/home/zachbird
:/bin/bash
serenaba:x:1009:1009:Serena A. Barton,,328-0541,,Software Developer:/home/serena
ba:/bin/bash
solomonw:x:1010:1010:Solomon Y. Wright,,328-6520,,Junior Software Developer:/hom
e/solomonw:/bin/bash
amaright:x:1011:1011:Marah Y. Wright,,,,Daughter of Solomon Y. Wright:/home/amar
ight:/bin/bash
redis:x:126:133::/var/lib/redis:/bin/false
student@mgs-650:~$ sudo deluser amaright
Removing user `amaright' ...
Warning: group `amaright' has no more members.
Done.
student@mgs-650:~$ sudo rm -rf /home/amaright
student@mgs-650:~$ sudo passwd -l jsweeney
passwd: password expiry information changed.
student@mgs-650:~$
```

Figure 2.0: Deleting user account with sudo deluser amaright, and deleting home folder with sudo rm -rf /home/amaright



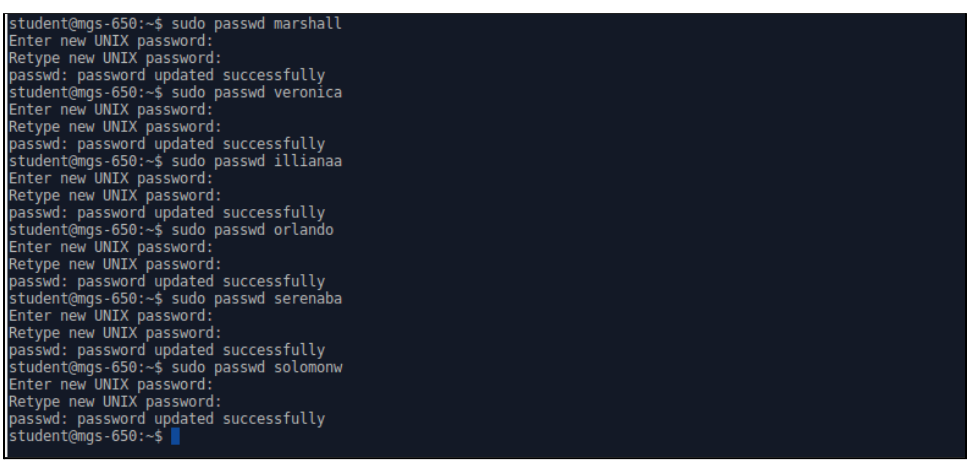
```
sshd:*:17495:0:99999:7:::
mysql:!:17495:0:99999:7:::
postgres:*:17495:0:99999:7:::
ftp:*:17495:0:99999:7:::
patcpett:$6$5oG72HSq$Nfpfw1xLCJvUR1Ux8Kvwf.dv8e0rxJFSODfYMiiuB2Pj/28wmSYHFNuUhtq
fcA/Mh129s4Y6u6bpvtOjN24I91:17496:0:99999:7:::
dunnxter::17496:0:99999:7:::
illianaa::17496:0:99999:7:::
marshall::17496:0:99999:7:::
veronica::17496:0:99999:7:::
orlando::17496:0:99999:7:::
jsweeney:!$6$Ctw7AgVG$A55jo8op7UONa6N5nT5hiJBo3TIks.GQhXWOF65VKrDIjISNGAFj89qr7J
XH5NLfPMR0WIMfQuIFuYt/tAbJ/0:17498:0:99999:7:::
zachbird:$6$ngiAx0bO$jZg/ToBRDiD3LVi1VhQCGPX//TMupnJDMHSMffuIS9FL8EsH5iYChZZiYbi
2RlKvYwanOrHtLLKgdJDWq4Ban0:17496:0:99999:7:::
serenaba::17496:0:99999:7:::
solomonw::17496:0:99999:7:::
redis:*:17496:0:99999:7:::
student@mgs-650:~$
```

Figure 2.1: Checking password hashes stored in the /etc/shadow file



```
student@mgs-650:~$ sudo passwd dunnxter
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
student@mgs-650:~$
```

Figure 2.2: Setting password for user who don't have a password using sudo passwd dunnxter



```
student@mgs-650:~$ sudo passwd marshall
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
student@mgs-650:~$ sudo passwd veronica
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
student@mgs-650:~$ sudo passwd illianaa
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
student@mgs-650:~$ sudo passwd orlando
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
student@mgs-650:~$ sudo passwd serenaba
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
student@mgs-650:~$ sudo passwd solomonw
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
student@mgs-650:~$
```

Figure 2.3: Setting password for users who don't have a password.

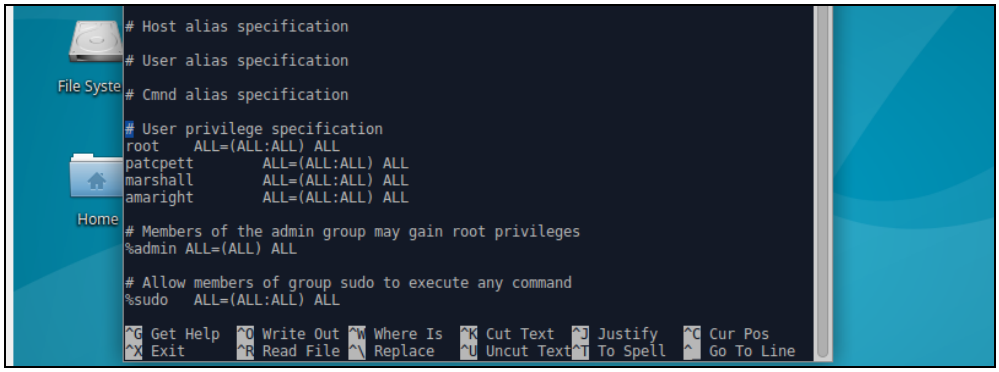


Figure 2.4: Checking user who can execute commands as root via sudo using sudo visudo

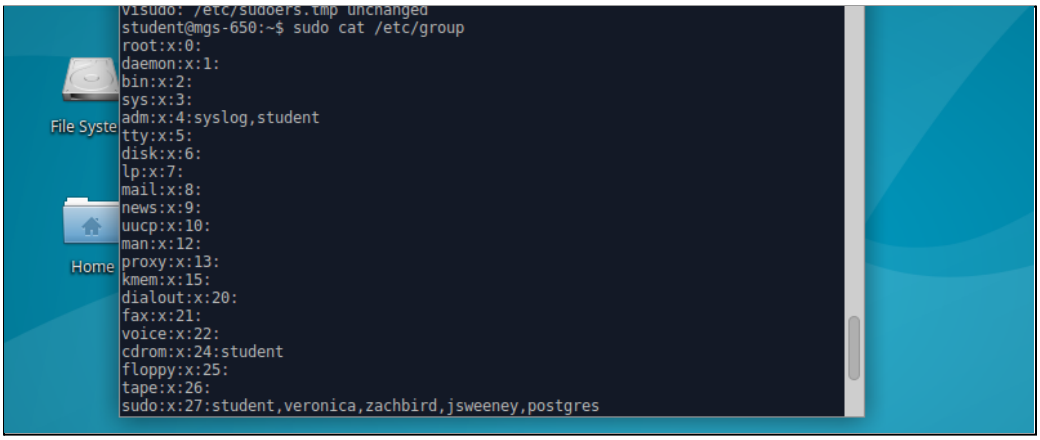


Figure 2.5: Checking users in each group using sudo cat /etc/group file

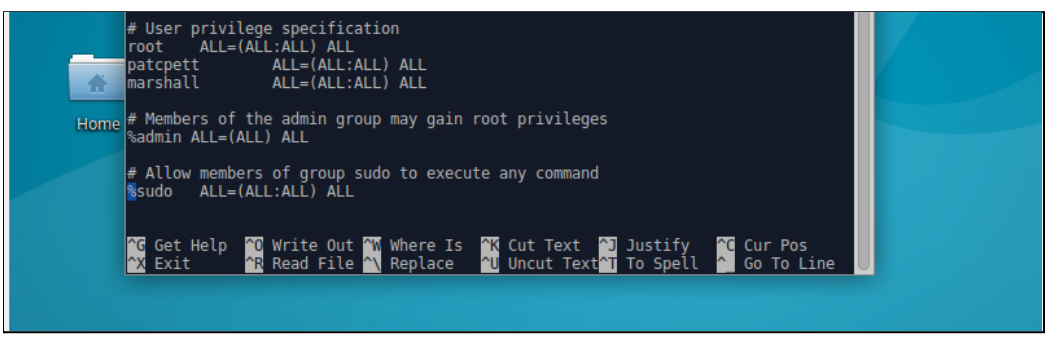


Figure 2.6: Checking sudo users details after removing amaright from sudoers file

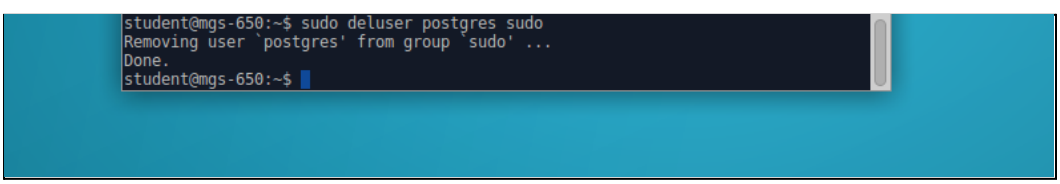


Figure 2.7: Removing postgres user from the sudo group with sudo deluser postgres sudo

Executive Summary

Initial steps taken before the security configuration assessment were to update all the software, configure firewall rules, remove unnecessary programs (samba, vsftpd), check user access and secure stored password hashes. This can partially and potentially can help towards system hardening by minimising attack vectors such as outdated packages with known security vulnerabilities, unnecessary network access [Figure 1.1 - 1.7], unnecessary running/installed programs [Figure 1.8 & 1.9], unrestricted user access [Figure 2.0, 2.2 - 2.7] and unencrypted data respectively [Figure 2.1].

The security configuration assessment using Center for Internet Security (CIS) Critical Security Controls (CSC) framework was performed to asses mgs650's internal network security posture. The CIS benchmark report generated has various configuration recommendations that applied for system hardening and auditing a Linux system. Overall completion of the assessment indicated recommendations for each control with scoring details for all the 6 controls defined for CIS Ubuntu Linux 16.04 LTS Benchmark v1.0.0. It is one of the CIS Controls that recommends secure configurations for hardware and software on mobile devices, laptops, workstations, and servers. The Center for Internet Security Configuration Assessment Tool (CIS-CAT) suggests security configuration benchmarks distributed by CIS as well as NIST under the Security Content Automation Protocol (SCAP) program, as an initiative to enable automation and standardization of technical security operations.

After the scan, six recommendation categories generated falls under the following configuration baselines:

- Initial Setup
- Services
- Network Configuration
- Logging and Auditing
- Access, Authentication and Authorization
- System Maintenance

For this scan, a Level 1 profile is chosen which is intended for servers and provides a practical and prudent way to secure a system within acceptable technology means and not too much performance impact.

Non-native file system types such as freevxfs, hfs, jffs32 FAT, hfsplus, squashfs, udf are supported under Linux's in-built functionality. Steps must be taken to ensure a check list of filesystems that are needed after considering the system environment since Internet access to cloud storage and standard network connectivity may use non-standard file system formats. Next, if a filesystem is not needed remove/disable support for unneeded file system types. This reduces the local attack surface of the system. An alerting system such as Advanced Intrusion Detection Environment (AIDE) must be set up to detect unauthorized changes to current file state/configuration files against a snapshot and prevent accidental or malicious misconfigurations. Steps must be taken to ensure permissions on bootloader config are configured preventing non-root users to read and write the boot parameters. This will reduce exploits security upon boot. Additional process hardening recommendation after initial setup of the system is to set up a hard limit on core dumps to prevent exposing confidential information from a core file.

Inorder to ensure legitimate incoming traffic from other machines, specific port communications i.e., port 22, port 80, port 8080 are allowed [Figure 1.4] and all the remaining ports are denied/blocked by changing the default policy to 'deny'. Similarly, outgoing traffic rules [Figure 1.6] are modified to allow port 43, port 50, port 443 that specifically allow port communications from our machine to other machines. However, this negates a report result 3.6.5 that was needed to ensure firewall rules exist for all open ports. Hence, this risk is accepted in this scenario.

Although there are password hashes starting "*", "!" which are invalid password hashes and a user cannot normally log in with those accounts [Figure 2.1] (here mysql, sshd, postgres), it is also recommended that the shell field in the password file be set to /sbin/nologin. This will also prevent the account from potentially being used to run any commands by locking the 'sync', 'shutdown', and 'halt' users. These users are traditionally shipped with a null password in unix and linux distributions which are used to properly shut down a system without having to provide the root password. This is useful in the case of a desktop workstation, but is detrimental in the case of a server system (also, this machine is assumed operating as a web server) and creates a risk of Denial of Service attack. So, these accounts must be either removed or locked like other default accounts to prevent use.