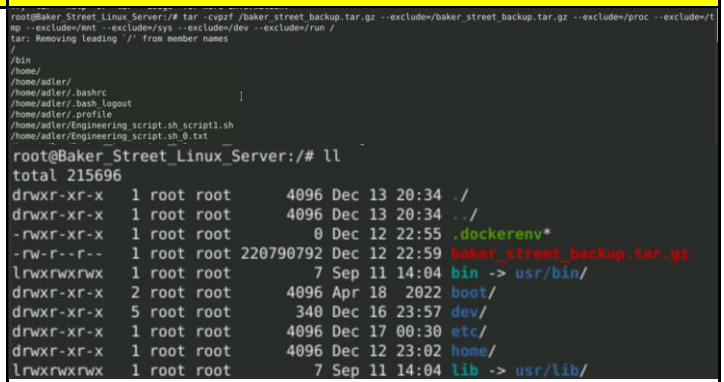
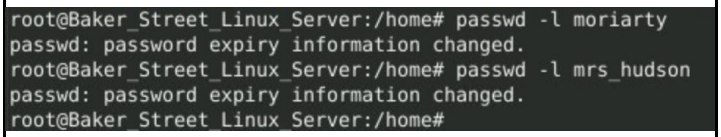
Cybersecurity

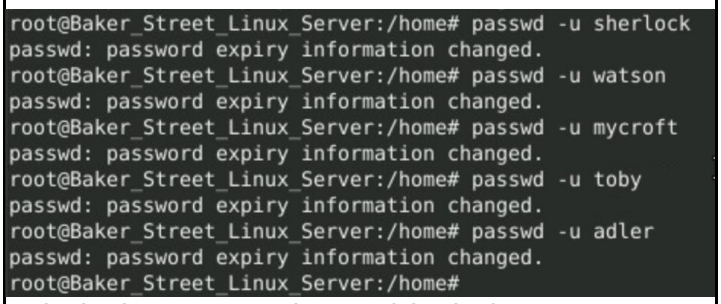
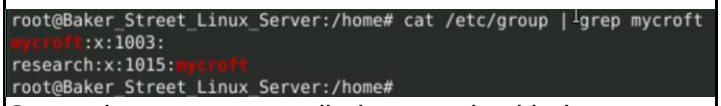
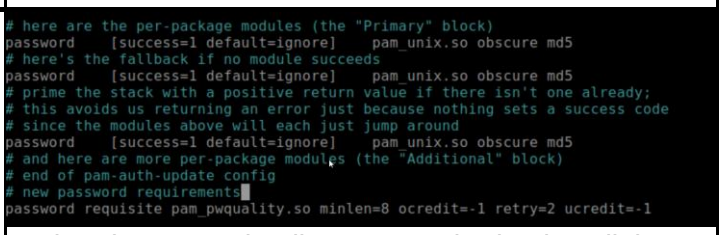
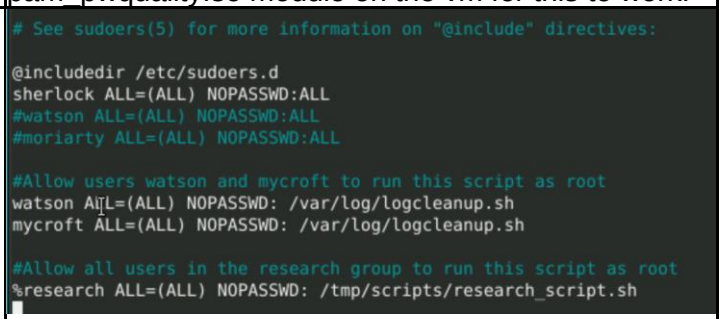
Project 1 Hardening Summary and Checklist

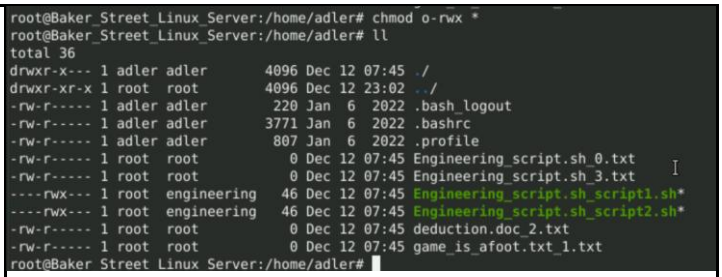
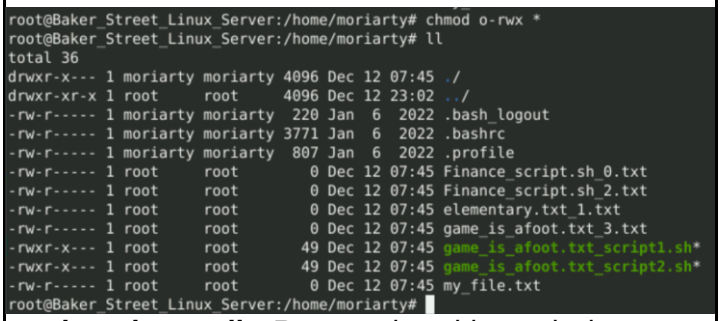
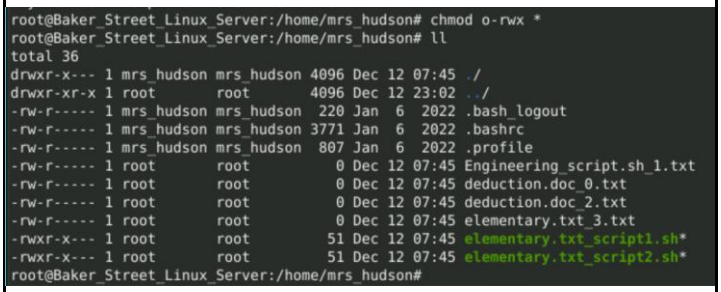
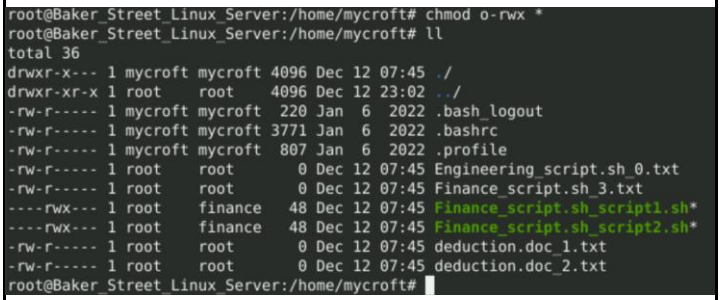
OS Information

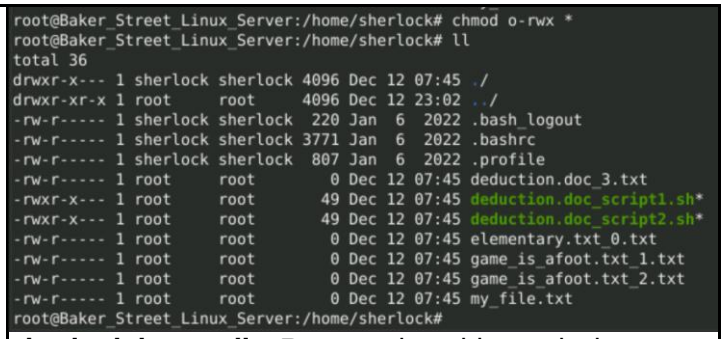
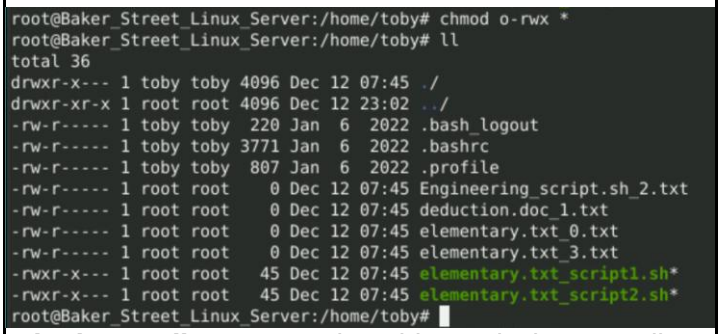
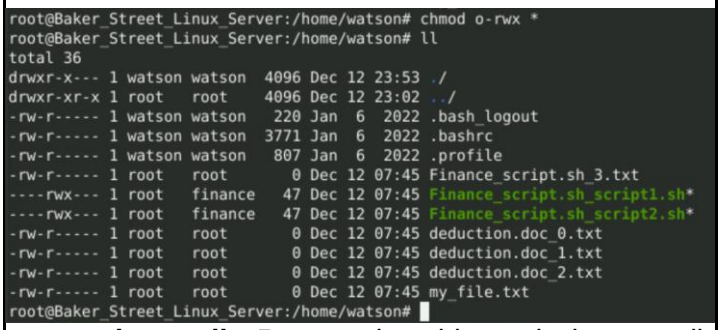
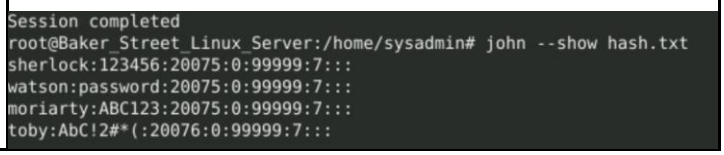
Customer	Baker Street Corporation
Hostname	Baker_Street_Linux_Server
OS Version	Ubuntu 24.04.5 LTS
Memory information	Total: 15Gi; used: 1.3Gi; free: 10Gi; shared: 184Mi; buff/cache: 3.2Gi; available: 13Gi
Uptime information	15:21:42 up 1:23, 0 users, load average: 0.27, 0.24, 0.25

Checklist

Completed	Activity	Script(s) used / Tasks completed / Screenshots
☑	OS backup	 Created OS backup “baker_street_backup.tar.gz”
☑	Auditing users and groups	 Deleted terminated employees (lestrade, irene, mary, gregson) with the command “userdel -r <username>”.  Locked accounts of employees on leave (moriarty,

		mrs_hudson).  Unlocked current employees (sherlock, watson, mycroft, toby, adler).  Created a new group called research with the command “groupadd research”. Moved user mycroft from the marketing group to the research group with the command “usermod -aG research mycroft”. Removed the marketing group with the command “groupdel marketing”.
☑	Updating and enforcing password policies	 Updated password policy. Note: I had to install the pam_pwquality.so module on the vm for this to work.
☑	Updating and enforcing sudo permissions	 Edited the sudoers file with the command “sudo visudo”. Ensured that sherlock was the only su user. Gave watson and mycroft sudo privileges to run /var/log/logcleanup.sh. Gave anyone in the research group sudo privileges to run /tmp/scripts/research_script.sh as root.

☑	Validating and updating permissions on files and directories	 **adler home dir:** Removed world permissions on all files. Set the two engineering scripts to be in the engineering group with the command “chown :engineering <filename>”. Changed the two engineering scripts to only allow rwx access to those in the engineering group using the command “chmod 070 <filename>”.  **moriarty home dir:** Removed world permissions on all files.  **mrs_hudson home dir:** Removed world permissions on all files.  **mycroft home dir:** Removed world permissions on all files. Set the two finance scripts to be in the finance group with the command “chown :finance <filename>”. Changed the two finance scripts to only allow rwx access to those in the finance group using the command “chmod 070 <filename>”.

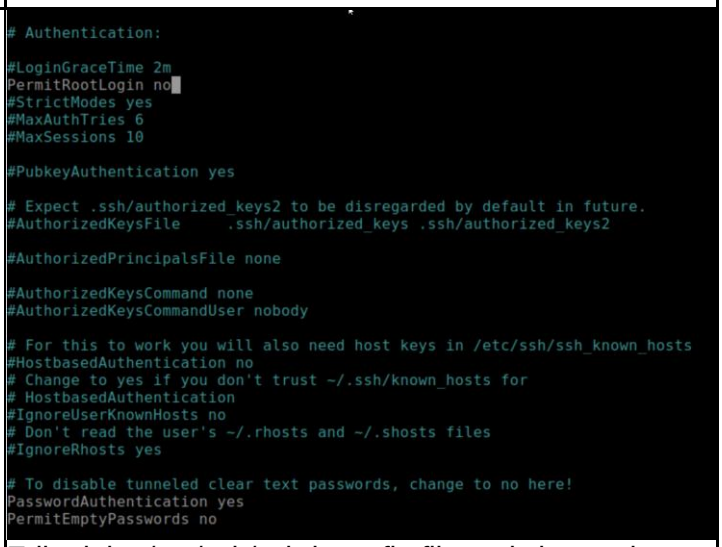
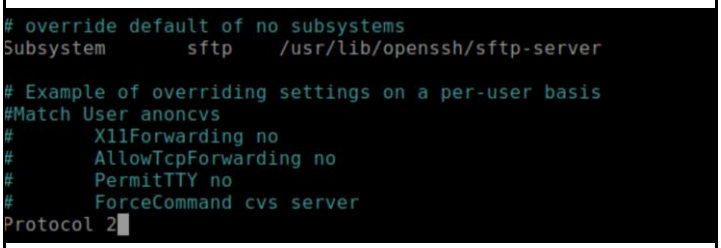
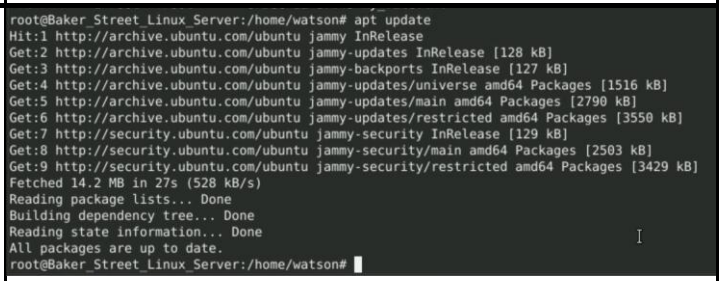
		 **sherlock home dir:** Removed world permissions on all files.  **toby home dir:** Removed world permissions on all files.  **watson home dir:** Removed world permissions on all files. Set the two finance scripts to be in the finance group with the command “chown :finance <filename>”. Changed the two finance scripts to only allow rwx access to those in the finance group using the command “chmod 070 <filename>”. Did not find any files in the home directory that had hidden passwords in them.
☑	Optional: Updating password hashing configuration	This optional task wasn’t included in the instructions or the activity files for this project, but I decided I would see if I could crack the passwords for the users on the system. I copied the users’ hashed passwords from the /etc/shadow file into a text file and then ran john using the wordlist “rockyou.txt”. john was able to crack 4 of the 7 passwords. Here are the results: 

```
root@Baker_Street_Linux_Server:/home/sherlock# chmod o-rwx *
root@Baker_Street_Linux_Server:/home/sherlock# ll
total 36
drwxr-x--- 1 sherlock sherlock 4096 Dec 12 07:45 ./
drwxr-xr-x 1 root     root     4096 Dec 12 23:02 ../
-rw-r----- 1 sherlock sherlock  220 Jan  6  2022 .bash_logout
-rw-r----- 1 sherlock sherlock 3771 Jan  6  2022 .bashrc
-rw-r----- 1 sherlock sherlock  807 Jan  6  2022 .profile
-rw-r----- 1 root     root        0 Dec 12 07:45 deduction.doc_3.txt
-rwxr-x--- 1 root     root       49 Dec 12 07:45 deduction.doc_script1.sh*
-rwxr-x--- 1 root     root       49 Dec 12 07:45 deduction.doc_script2.sh*
-rw-r----- 1 root     root        0 Dec 12 07:45 elementary.txt_0.txt
-rw-r----- 1 root     root        0 Dec 12 07:45 game_is_afoot.txt_1.txt
-rw-r----- 1 root     root        0 Dec 12 07:45 game_is_afoot.txt_2.txt
-rw-r----- 1 root     root        0 Dec 12 07:45 my_file.txt
root@Baker_Street_Linux_Server:/home/sherlock#
```

```
root@Baker_Street_Linux_Server:/home/toby# chmod o-rwx *
root@Baker_Street_Linux_Server:/home/toby# ll
total 36
drwxr-x--- 1 toby toby 4096 Dec 12 07:45 ./
drwxr-xr-x 1 root root 4096 Dec 12 23:02 ../
-rw-r----- 1 toby toby  220 Jan  6  2022 .bash_logout
-rw-r----- 1 toby toby 3771 Jan  6  2022 .bashrc
-rw-r----- 1 toby toby  807 Jan  6  2022 .profile
-rw-r----- 1 root root    0 Dec 12 07:45 Engineering_script.sh_2.txt
-rw-r----- 1 root root    0 Dec 12 07:45 deduction.doc_1.txt
-rw-r----- 1 root root    0 Dec 12 07:45 elementary.txt_0.txt
-rw-r----- 1 root root    0 Dec 12 07:45 elementary.txt_3.txt
-rwxr-x--- 1 root root   45 Dec 12 07:45 elementary.txt_script1.sh*
-rwxr-x--- 1 root root   45 Dec 12 07:45 elementary.txt_script2.sh*
root@Baker_Street_Linux_Server:/home/toby#
```

```
root@Baker_Street_Linux_Server:/home/watson# chmod o-rwx *
root@Baker_Street_Linux_Server:/home/watson# ll
total 36
drwxr-x--- 1 watson watson  4096 Dec 12 23:53 ./
drwxr-xr-x 1 root   root    4096 Dec 12 23:02 ../
-rw-r----- 1 watson watson   220 Jan  6  2022 .bash_logout
-rw-r----- 1 watson watson  3771 Jan  6  2022 .bashrc
-rw-r----- 1 watson watson   807 Jan  6  2022 .profile
-rw-r----- 1 root   root       0 Dec 12 07:45 Finance_script.sh_3.txt
----rwx--- 1 root   finance   47 Dec 12 07:45 Finance_script.sh_script1.sh*
----rwx--- 1 root   finance   47 Dec 12 07:45 Finance_script.sh_script2.sh*
-rw-r----- 1 root   root       0 Dec 12 07:45 deduction.doc_0.txt
-rw-r----- 1 root   root       0 Dec 12 07:45 deduction.doc_1.txt
-rw-r----- 1 root   root       0 Dec 12 07:45 deduction.doc_2.txt
-rw-r----- 1 root   root       0 Dec 12 07:45 my_file.txt
root@Baker_Street_Linux_Server:/home/watson#
```

```
Session completed
root@Baker_Street_Linux_Server:/home/sysadmin# john --show hash.txt
sherlock:123456:20075:0:99999:7:::
watson:password:20075:0:99999:7:::
moriarty:ABC123:20075:0:99999:7:::
toby:AbC!2#*(:20076:0:99999:7:::
```

		As you can see these passwords are weak, so it is a good thing that we strengthened the password policy on the server.
☑	Auditing and securing SSH	 Edited the /etc/ssh/sshd_config file and changed PermitRootLogin from yes to no and changed PerrmitEmptyPasswords from yes to no.  Removed all ports except Port 22 from sshd_config.  Changed from Protocol 1 to Protocol 2 in sshd_config. Restarted the ssh service with the command “service ssh restart”.
☑	Reviewing and updating system packages	 Ran apt update.

```
# Authentication:

#LoginGraceTime 2m
PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile     .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
PasswordAuthentication yes
PermitEmptyPasswords no
```

```
#Include /etc/ssh/sshd_config.d/*.conf

Port 22
#Port 2223
#Port 2224
#Port 2225
#Port 8967
```

```
# override default of no subsystems
Subsystem       sftp    /usr/lib/openssh/sftp-server

# Example of overriding settings on a per-user basis
#Match User anoncvs
#       X11Forwarding no
#       AllowTcpForwarding no
#       PermitTTY no
#       ForceCommand cvs server
Protocol 2
```

```
root@Baker_Street_Linux_Server:/home/watson# apt update
Hit:1 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:3 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1516 kB]
Get:5 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2790 kB]
Get:6 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [3550 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [2503 kB]
Get:9 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [3429 kB]
Fetched 14.2 MB in 27s (528 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
root@Baker_Street_Linux_Server:/home/watson#
```

```
root@Baker_Street_Linux_Server:/home/watson# apt upgrade -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@Baker_Street_Linux_Server:/home/watson#
```

Ran apt upgrade –y.

```
root@Baker_Street_Linux_Server:/home/sysadmin# ll
total 92
drwxr-x--- 1 sysadmin sysadmin  4096 Dec 13 22:16 ./
drwxr-xr-x 1 root     root      4096 Dec 12 23:02 ../
-rw-r----- 1 sysadmin sysadmin   220 Jan  6  2022 .bash_logout
-rw-r----- 1 sysadmin sysadmin  3771 Jan  6  2022 .bashrc
-rw-r----- 1 sysadmin sysadmin   807 Jan  6  2022 .profile
-rw-r----- 1 root     root      3062 Dec 13 21:32 Linux_Hardening1
-rw-r--r-- 1 root     root     28302 Dec 13 21:50 Linux_Hardening2
-rwxr----- 1 root     root      2623 Dec 13 22:13 hardening_script_1.sh*
-rwxr--r-- 1 root     root       991 Dec 13 22:16 hardening_script_2.sh*
-rw-r----- 1 root     root     22922 Dec 17 01:19 package_list.txt
root@Baker_Street_Linux_Server:/home/sysadmin# wc -l package_list.txt
322 package_list.txt
root@Baker_Street_Linux_Server:/home/sysadmin#
```

Created a file “package_list.txt” which contains a list of all installed packages. The command used was “apt list –installed > package_list.txt”. The command “wc –l package_list.txt” indicated that there are 322 packages installed.

```
root@Baker_Street_Linux_Server:/home/sysadmin# grep telnet package_list.txt
telnet/jammy,now 0.17-44build1 amd64 [installed]
root@Baker_Street_Linux_Server:/home/sysadmin# grep rsh package_list.txt
rsh-server/jammy,now 0.17-22 amd64 [installed]
root@Baker_Street_Linux_Server:/home/sysadmin#
```

Determined that telnet was on the installed packages list, but not rsh-client.

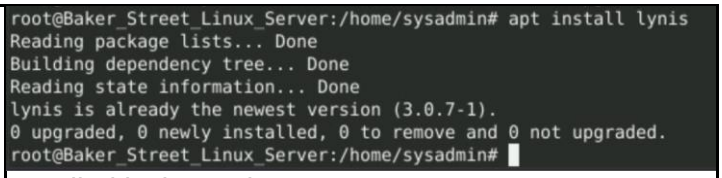
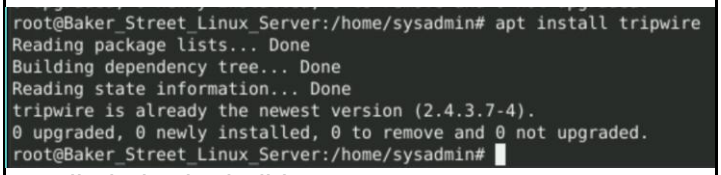
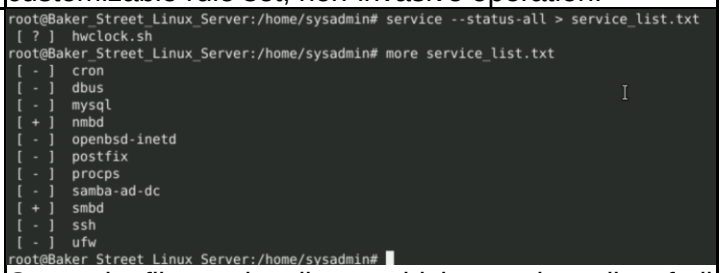
```
root@Baker_Street_Linux_Server:/home/sysadmin# apt-get remove telnet
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer requ
  attr dirmngr gnupg gnupg-l10n gnupg-utils gpg gpg-agent gpg-wks-client g
  libavahi-client3 libavahi-common-data libavahi-common3 libboost-iostream
  libcgi-pm-perl libclone-perl libcups2 libencode-locale-perl libevent-cor
  libfcgi0ldbl libgfapi0 libgfrpc0 libgfxdr0 libglusterfs0 libgpgme11 libh
  libhttp-date-perl libhttp-message-perl libibverbs1 libio-html-perl libja
  liblwp-mediatypes-perl libmecab2 libnl-3-200 libnl-route-3-200 libnpth0
  libtalloc2 libtdb1 libtevent0 libtimedate-perl liburi-perl liburing2 lib
  pinentry-curses python3-certifi python3-cffi-backend python3-chardet pyt
  python3-importlib-metadata python3-ldb python3-markdown python3-more-ite
  python3-requests-toolbelt python3-six python3-talloc python3-tdb python3
Use 'apt autoremove' to remove them.
The following packages will be REMOVED:
  telnet
0 upgraded, 0 newly installed, 1 to remove and 0 not upgraded.
After this operation, 158 kB disk space will be freed.
Do you want to continue? [Y/n] y
(Reading database ... 15996 files and directories currently installed.)
Removing telnet (0.17-44build1) ...
Processing triggers for menu (2.1.47ubuntu4) ...
```

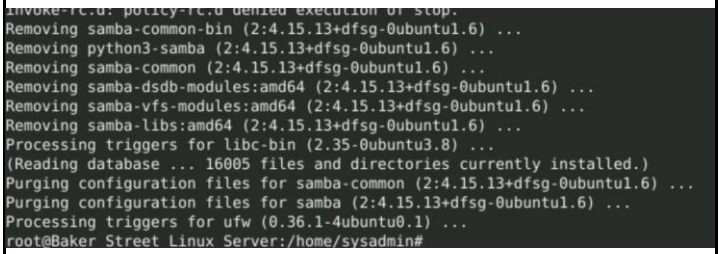
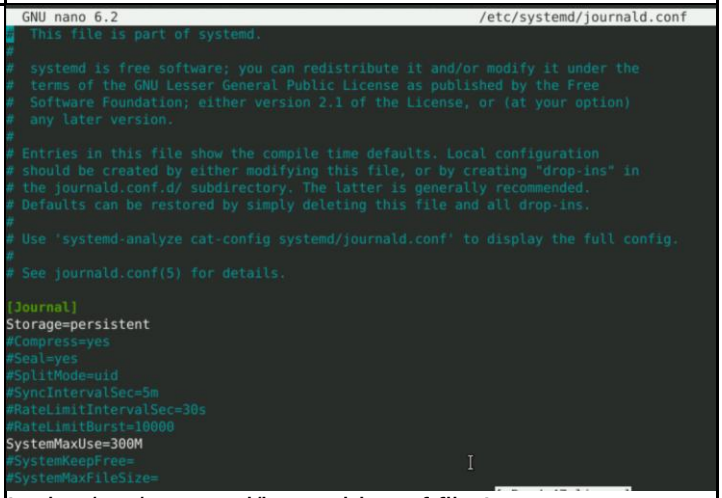
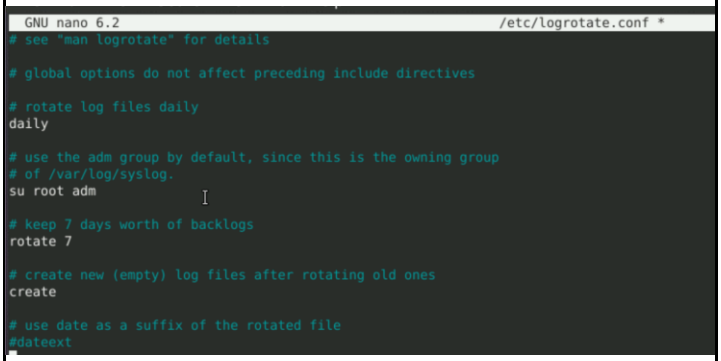
Removed telnet with the command “apt-get remove telnet”. Verified that telnet was removed.

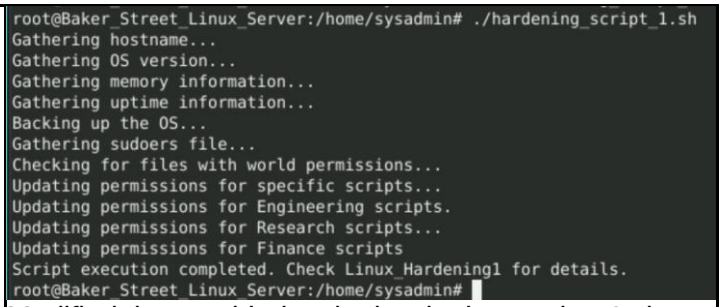
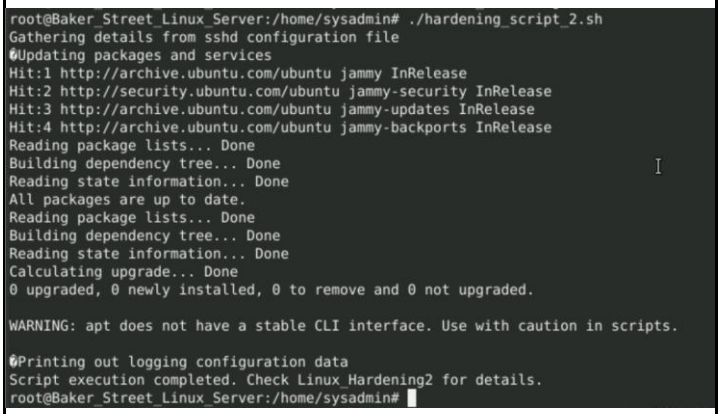
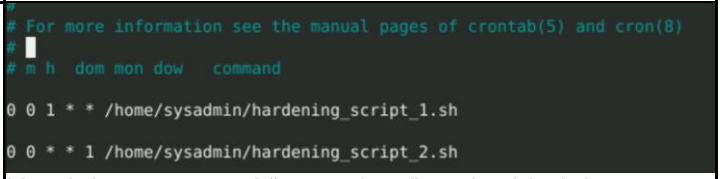
Removed unnecessary dependencies with the command “apt autoremove –y”.

```
root@Baker_Street_Linux_Server:/home/sysadmin# apt install ufw
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ufw is already the newest version (0.36.1-4ubuntu0.1).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@Baker_Street_Linux_Server:/home/sysadmin#
```

Installed ufw.

		 Installed lynis version 3.0.7-1.  Installed tripwire build 2.4.3.7-4. **Hardening features of ufw:** access control, rate limiting, application profiles, ipv6 support, logging, ip blocking, port blocking, service management. **Hardening features of lynis:** file and system integrity audits, user and group management, kernel and system hardening, filesystem security, networking and firewall audits, audit and logging, compliance checks, malware detection, authentication and authorization, backup and patch management. **Hardening features of tripwire:** file integrity monitoring, critical system file monitoring, audit logging and reporting, system integrity verification, alerting and notification, centralized monitoring, rootkit and malware detection, compliance monitoring, customizable rule set, non-invasive operation.
☑	Disabling unnecessary services	 Created a file service_list.txt which contains a list of all services (running and not running). Noticed that both mysql and samba-ad-dc were installed, but not running. The [-] indicates not running. Disabled the mysql service from starting on boot with the command “update-rc.d mysql disable”.  Removed mysql with the command “apt-get remove –purge mysql-server mysql-client mysql-common mysql-server-core-* mysql-client-core-*”. Removed the mysql data files with the command

		“rm –rf /etc/mysql /var/lib/mysql /var/log/mysql /var/log/mysql.*”. Disabled the samba service from starting on boot with the command “update-rc.d samba-ad-dc disable”.  Removed samba with the command “apt-get remove –purge samba*”.  Verified that both mysql and samba were no longer an available service.
☑	Enabling and configuring logging	 In the /etc/systemd/journald.conf file I set storage to persistent and systemmaxuse to 300M.  In the /etc/logrotate.conf file I changed log rotation from weekly to daily and the rotation schedule from every 4 weeks to every 7 days.

☑	Scripts created	 Modified the provided script hardening_script_1.sh, made it executable, and then tested it to make sure there were no errors. (Note: I commented out the tar command so I could show this abbreviated version of the output, but I did test the script with the tar command active). The script can be found here: https://drive.google.com/file/d/1JJ-vATO71N5A-NSYmrFPLxeJ_qD1vj71/view?usp=sharing The output report for script 1 can be found here: https://drive.google.com/file/d/11P16pM0AQ9ZeQ6eaXp43Sh6z1BoKX_OV/view?usp=sharing  Modified the provided script hardening_script_2.sh, made it executable, and then tested it to make sure there were no errors. The script can be found here: https://drive.google.com/file/d/18bFT7HrfQk8Ydd5AAFHawM8A04WBxf6x/view?usp=sharing The output report for script 2 can be found here: https://drive.google.com/file/d/18mDg47hinN6F3_XVbbsZsmjurDaybyND/view?usp=sharing
☑	Scripts scheduled with cron	 Used the command “crontab -e” and added these two lines to the file to automate the running of those scripts.

Summary Report
for
Baker Street Corporation
Linux System Hardening

This report summarizes the steps taken to confirm that a Linux server owned by Baker Street Corporation (BCS) was properly configured to protect them from security breaches. During this audit, several potential security issues were found. The discovered security issues are detailed below along with the steps taken to correct them.

After bringing the system online a backup of the operating system (OS) was created so that BSC would be able to restore their system in case of a crash or other harmful event. It was noted that several employees had been terminated, but still had accounts on the system. These accounts and any files associated with these employees were removed. It was also noted that two employees were on leave and did not need access to their accounts, so they were locked. All current employees' accounts were unlocked so that they would have access to their files. A new research group was created and user mycroft was added to this group. Mycroft had previously been a part of the marketing group which was disbanded. The old marketing group was removed from the system.

BCS's password policy was weak so it was updated to require a minimum of 8 characters, at least 1 uppercase character, at least 1 special character and a maximum of 2 retries before the user is locked out. The user sherlock was the only user given permission to have root privileges for the system. The users watson and mycroft were given root privileges to run a specific log cleanup script. Also, any user that is part of the research group was given root privileges to run a specific research script.

Because giving "world" permissions to user's files can be dangerous, world permissions were removed from all the files in each user's home directory. All "engineering" scripts located in the home directory were moved to the engineering group and read-write-execute permissions were given only to those in that group. All "finance" scripts located in the home directory were moved to the finance group and read-write-execute permissions were given only to those in that group. A thorough search was conducted of all the user files to see if there were any potential passwords stored in them. None were found.

Changes were made to the secure shell (ssh) service to prevent anyone from logging in as a root user or logging in with no password. All incoming ports for ssh were eliminated except port 22. Also, it was ensured that Protocol 2 was being used for the ssh service. Protocol 2 is the superior security protocol since it uses more secure cryptographic and encryption algorithms than Protocol 1. All packages installed on the system were updated and/or upgraded to ensure their latest versions were active. The telnet package was removed from the system because it is an old network protocol and has many known vulnerabilities including no encryption of data and susceptibility to common hacker attacks. Uncomplicated Firewall (ufw) was installed that provides an easy method to configure and manage the firewall by simplifying the process of defining rules and policies. Lynis was installed to allow for in-depth security scans to be done that provide a thorough assessment of system hardening and security measures. Also, tripwire was installed to detect and monitor unauthorized changes to files, directories, or system configurations in real-time. Tripwire is widely used by cybersecurity professionals for file integrity monitoring, intrusion detection, system hardening, and compliance checks.

A list of all installed services on the system was produced which showed that both MySQL and Samba were installed, but not actively running. MySQL is open-source software that is used to store and manage data. Its vulnerabilities include potentially having improperly configured file or directory permissions, which could allow unauthorized users to read or modify sensitive data. Also, SQL injection is a known server attack method where malicious SQL code could be written into a query that is executed by MySQL. For these reasons, MySQL was disabled and purged from the system. Samba

is also a free open-source tool that is used for file sharing and print services across different operating systems. Like MySQL Samba could allow improper file permissions on the server which could allow users to access or modify sensitive data. Denial-of-service (DoS) vulnerabilities have also affected Samba, allowing an attacker to overwhelm the Samba service by sending specially crafted requests or exploiting flaws in the way Samba processes requests. For these reasons, Samba was also disabled and purged from the system.

System wide logging services were set to be persistent so that log data is continuously saved to storage, ensuring that even if the system reboots or loses power, the log information remains accessible and is not lost. Logs were set to run daily and would maintain at least 7 days' worth of log records.

Finally, two scripts were written to provide administrators with documentation of the current state of the system at any given time and to periodically backup the system and monitor changes to important files, directories, packages and services. These scripts were automated to run on a weekly or monthly schedule.

End of report