

Project 1 Day 3 Scenario:

Today you will conclude your hardening tasks by scripting and scheduling the tasks you completed on Day 1 & 2.

Today's focus is on

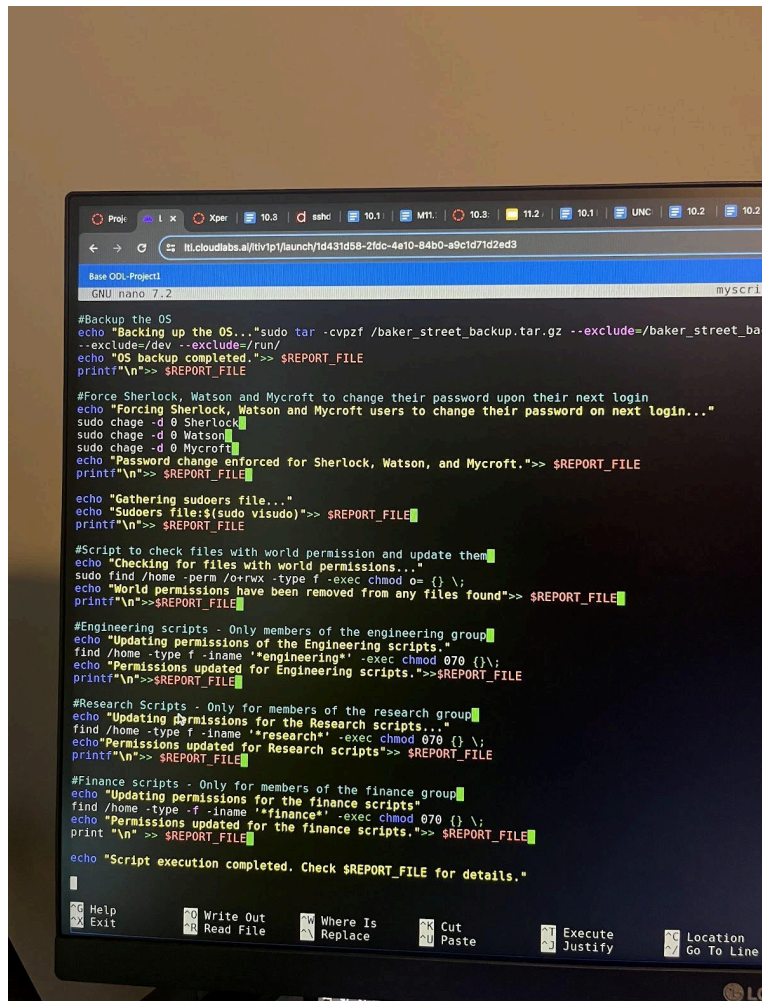
- (1) Scripting your hardening tasks
- (2) Scheduling your hardening scripts
- (3) Completing the summary report

Part 1: Scripting your hardening tasks

Over the last 2 class days you have done great work hardening and auditing BSC's environment. While you have protected BSC's linux server, you have done each part, one step at a time. In Part 1 of today's activities, you will build out 2 scripts to automate many of the tasks you completed over the last two days. The first script will be for the first day's tasks, and the second day will cover the second day's tasks. Additionally, you have been provided with script templates to guide you through the script development process. Once you have built out your scripts, test them out to confirm there are no errors.

Script 1:

- View and copy the following [script 1 template](#) , update the script everywhere you see placeholders in **red**. Be sure to remove all the placeholders!
 - Copy the script into a file called **hardening_script1.sh**
 - Update the permissions and run it to validate there are no errors.
1. Nano [myscript.sh](#) allows us to create a new script called [mscript.sh](#)
 2. cp [myscript.sh](#) `hardening_script1.sh` allows us to copy the script into the file `hardening_script1.sh`
 3. `chmod +x` allows us to execute this script so we can run it with no errors.



```
Base ODI-Project1
GNU nano 7.2 myscrip

#Backup the OS
echo "Backing up the OS..."sudo tar -cvpzf /baker_street_backup.tar.gz --exclude=/dev --exclude=/run/
--exclude=/dev --exclude=/run/
echo "OS backup completed.">> $REPORT_FILE
printf"\n">> $REPORT_FILE

#Force Sherlock, Watson and Mycroft to change their password upon their next login
echo "Forcing Sherlock, Watson and Mycroft users to change their password on next login..."
sudo chage -d 0 Sherlock
sudo chage -d 0 Watson
sudo chage -d 0 Mycroft
echo "Password change enforced for Sherlock, Watson, and Mycroft.">> $REPORT_FILE
printf"\n">> $REPORT_FILE

echo "Gathering sudoers file..."
echo "Sudoers file:(sudo visudo)">> $REPORT_FILE
printf"\n">> $REPORT_FILE

#Script to check files with world permission and update them
echo "Checking for files with world permissions..."
sudo find /home -perm /o+rwx -type f -exec chmod o={} \;
echo "World permissions have been removed from any files found">> $REPORT_FILE
printf"\n">> $REPORT_FILE

#Engineering scripts - Only members of the engineering group
echo "Updating permissions of the Engineering scripts."
find /home -type f -iname "engineering" -exec chmod 070 {} \;
echo "Permissions updated for Engineering scripts.">> $REPORT_FILE
printf"\n">> $REPORT_FILE

#Research Scripts - Only for members of the research group
echo "Updating permissions for the Research scripts..."
find /home -type f -iname "research" -exec chmod 070 {} \;
echo "Permissions updated for Research scripts.">> $REPORT_FILE
printf"\n">> $REPORT_FILE

#Finance scripts - Only for members of the finance group
echo "Updating permissions for the finance scripts"
find /home -type f -iname "finance*" -exec chmod 070 {} \;
echo "Permissions updated for the finance scripts.">> $REPORT_FILE
printf"\n">> $REPORT_FILE

echo "Script execution completed. Check $REPORT_FILE for details."
```

Script 2

- View and copy the following [script 2 template](#) , update the script everywhere you see placeholders in **red**. Be sure to remove all the placeholders!
 - Copy the script into a file called **hardening_script2.sh**
 - Update the permissions and run it to validate there are no errors.
1. Nano [myscript2.sh](#) allows us to create a script called [myscript2.sh](#)
 2. Cp [myscript2.sh](#) [hardening_script2.sh](#) allows us to copy this script into a file called [hardening_script2.sh](#)
 3. `chmod +x` allows us to execute this script so we can run it with no errors.

```
Base ODI-Project1
GNU nano 7.2 myscrip

#Backup the OS
echo "Backing up the OS..."sudo tar -cvpzf /baker_street_backup.tar.gz --exclude=/dev --exclude=/run/
--exclude=/dev --exclude=/run/
echo "OS backup completed.">> $REPORT_FILE
printf"\n">> $REPORT_FILE

#Force Sherlock, Watson and Mycroft to change their password upon their next login
echo "Forcing Sherlock, Watson and Mycroft users to change their password on next login..."
sudo chage -d 0 Sherlock
sudo chage -d 0 Watson
sudo chage -d 0 Mycroft
echo "Password change enforced for Sherlock, Watson, and Mycroft.">> $REPORT_FILE
printf"\n">> $REPORT_FILE

echo "Gathering sudoers file..."
echo "Sudoers file:(sudo visudo)">> $REPORT_FILE
printf"\n">> $REPORT_FILE

#Script to check files with world permission and update them
echo "Checking for files with world permissions..."
sudo find /home -perm /o+wx -type f -exec chmod o={} \;
echo "World permissions have been removed from any files found">> $REPORT_FILE
printf"\n">> $REPORT_FILE

#Engineering scripts - Only members of the engineering group
echo "Updating permissions of the Engineering scripts."
find /home -type f -iname "Engineering*" -exec chmod 070 {} \;
echo "Permissions updated for Engineering scripts.">> $REPORT_FILE
printf"\n">> $REPORT_FILE

#Research Scripts - Only for members of the research group
echo "Updating permissions for the Research scripts..."
find /home -type f -iname "Research*" -exec chmod 070 {} \;
echo "Permissions updated for Research scripts.">> $REPORT_FILE
printf"\n">> $REPORT_FILE

#Finance scripts - Only for members of the finance group
echo "Updating permissions for the finance scripts"
find /home -type f -iname "Finance*" -exec chmod 070 {} \;
echo "Permissions updated for the finance scripts.">> $REPORT_FILE
printf"\n">> $REPORT_FILE

echo "Script execution completed. Check $REPORT_FILE for details."
```

Part 2: Scheduling your hardening scripts

In Part 2 of today's activity, you will use cron to schedule the 2 scripts you just created.

Complete the following:

- Using cron, schedule script 1 to run Once a month on the first of the month
- Using cron, schedule script 2 to run Once a week every Monday

Be sure to note on your checklist what you have completed

1. `Crontab -e` allows us to edit the crontab file for the current user
2. `* * 1 1 * myscript.sh` allows us to run [myscript.sh](#) once a month on the first of the month.
3. `* * * * 1 myscript2.sh` allows you to run this script once a week every monday.

