Homework : Linux Week 1

Background

In the previous class activities, you acted as system administrator in order to troubleshoot a malfunctioning server.

The senior administrator was quite pleased with your work. Now, they would like you to prepare another server to replace this server. You are tasked with completing the following checklist in order to prepare a new server.

Set-up:

Please log into your local virtual machine. Use the following credentials:

- Username: `sysadmin`

- Password: `cybersecurity`

In order to get started with your tasks, you will need to open the `Terminal` within your Ubuntu VM. If you are unsure how to do it, within your Ubuntu VM, do the following:

- Open the Linux terminal by pressing `CTRL+ALT+T` for Windows users or `CTRL+OPTIONS+T` for Mac users.

- Alternatively, press the `Windows + A` key simultaneously (`Command + A` for Mac users), then type in "Terminal" in the search bar, and select the `Terminal` icon (not the `Xfce Terminal` icon)

Instructions:

The administrator has provided a checklist for you detailing all the steps that need to be done. As you solve each step, please fill out the [Submission File](SubmissionFile.md). This will be your homework deliverable.

Checklist:

For each of the following checklists involving system administration tasks, yoiu will need to run the correct command and confirm the results.

Ensure permissions on sensitive files:

Remember that the `/etc/` directory is where system configuration files exist. It would be a good starting point to navigate to this directory by running `cd /etc/`.

Inspect the file permissions of each of the below files. If they do not match the descriptions below, then please update the permissions.

- Permissions on `/etc/shadow` should allow only `root` read and write access.

- Permissions on `/etc/gshadow` should allow only `root` read and write access.

- Permissions on `/etc/group` should allow `root` read and `write` access, and allow everyone else `read` access only.

- Permissions on `/etc/passwd` should allow `root` read and `write` access, and allow everyone else `read` access only.

To get started, run the following command to view the file permissions:

- `ls -l <file1>`

If permissions need to be changed or modified, you'll need to use the `chmod` command.

Create user accounts:

The next section of the admin's checklist requires various users to be set up. These commands do not require you to be working from a specific directory.

- Add user accounts `sam`, `joe`, `amy`, `sara` and `admin`.

- In order for users to be added to the system, you need to run the command with `sudo`.

Next, we want to ensure that our user's passwords fit various requirements of length and complexity.

- Force users to create passwords with a minimum length of `16` characters that must incorporate all `4` types of character type classes (numbers, symbols, etc.).
 - **Note**: you will need to edit two settings for this.

Then, we want to enforce a password rotation policy. In old Linux terms, we might have called this process the **ch**ange **age** of for password expiration.

- Force passwords to expire, at a _maximum_, of every `90` days.

Lastly, we want to make sure that only the `admin` user has general `sudo` group access. This requires a command that will allow user modifications.

- Ensure that only the `admin` user has general sudo access.

Create user group:

Next up on our checklist, we want to execute the commands to fully set up a group on our system.

This requires a group to be made, users to be added to the group, a shared group folder to be created, the group folder owners to be set, and lastly, you'll need to set group ID sticky bits for these shared folders.

- Add the group, `engineers`, to the system.
- Add users `sam`, `joe`, `amy`, and `sara` to the managed group. This will be similar to how you added `admin` to the `sudo` group in the previous exercise.
- Create a shared folder for this group: `/home/engineers`.
- Change ownership on the new engineers' shared folder to the `engineers` group.
- Using `chmod`'s letter-based options, add the `SGID` bit _and_ the `sticky` bit to allow collaboration between engineers in this directory.

 Note: You will need to use `sudo` for all of these commands.

* **Note:** Please refer to the Day 3 Student Guide for a refresher on special bits.

Lynis auditing

Part of your administrator's checklist involves running an audit against the system in order to harden it. You'll be using the system and security auditing tool, `lynis`, in order to do so.

- Install the package, `lynis`, to your system if it is not already installed.
- Check the `lynis` documentation for instructions on how to run a system audit and then,
- Run a lynis system audit with `sudo`.
- Provide a report from `lynis` output on what more could be done to harden the system.

BONUS: chkrootkit scans

Lastly, despite claims from misled enthusiasts, Linux is _not_ immune to malware. You will need to install and run the application `chkrootkit`, to search for any potential rootkits installed on the system.

- Install the package, `chkrootkit`, to your system if it is not already installed.
- Check the `chkrootkit` documentation for instructions on how to run a scan to find system root kits.
- Run `chkrootkit` (with `sudo`) in expert mode to verify the system does not have a root kit installed.
- Provide a report from `chkrootkit` output on what more could be done to harden the system.

Vagrant Update

After you complete this homework, please make sure to pull the latest Vagrant virtual machine build. Please refer to the [Using Vagrant document](https://docs.google.com/document/d/1h-zcoKt6c4AnZBENHLPWsXJ1djs1c96-9zkf7Jsb_GQ/) on the commands you will need to run.

```
vagrant box update && vagrant destroy --force && vagrant up
```