Lab 3

Filesystems and Permissions

Outcomes

By the end of this lab, you will:

- Have a deeper knowledge of where to find different kinds of files on systems.
- Have a deeper understanding UNIX permissions including special permission bits.
- Be able to find files matching different characteristics and change permissions on those files.

Rubric (8 points total):

- Demonstrate your script changing filesystem permissions and finding files with different characteristics to the TA.
- Turn in this script to iLearn.
- Both the demonstration (checkoff) and the submission to iLearn are necessary in order to receive credit for the lab.

Procedure

Start up your virtual machine and use it to write a single script that has individual commands in it that perform the requested operations. The `find` command is the key to these commands. We suggest using the `find` man page and Google to find the usage and flags you need.

Portions of this lab will also require knowledge of how to view and modify file system permissions. You can use the man page for ls and chmod to get more information about these subjects as well as referencing the following slides from lecture.

[Access Control & Root]
[The Filesystem]

Script

The script should be written in a form where the TA can check off individual items. For example, if you are asked 'find all directories in /tmp/ that are subdirectories of /tmp, but no other files', and separately 'find all files in /var that are symbolic links' your script would look like this. The `read -p "Hit any key to continue”` is just a syntactic construct so your TA can easily parse through the output.

```
#!/bin/bash
```

```
echo 'finding directories in /tmp/ that are subdirectories of /tmp,
but no other files'
read -p "Hit any key to continue."
echo find /tmp -mindepth 1 -type d
read -p "Hit any key to continue."
echo 'find all files in /var/ that are symbolic links'
read -p "Hit any key to continue."
echo find /var -type s
```

Requirements

1. Find all files in /bin, /sbin, /usr/bin, and /usr/sbin that are setuid and owned by root. Why are these files potential security risks?
2. Find all files across the entire system that have setuid or setgid enabled (regardless of owner).
3. Find all files in /var that have changed in the last 20 minutes.
4. Find all files in /var that are regular files of zero length.
5. Find all files in /dev that are not regular files and also not directories. The same command should print a listing that includes permissions and modification times (at a minimum) for these files.
6. Find all directories in /home that are not owned by root. In the same command, change their permissions to ensure they have 711 (`-rwx--x--x`) permissions.
7. Find all regular files in /home that are not owned by root. In the same command, change their permissions to ensure they have 755 (`-rwxr-xr-x`) permissions.
8. Find all files (of all types) in /etc that have changed in the last 5 days.

Submission

In addition to demoing a script which accomplishes the above requirements to your TA you must also submit that script to iLearn in order to receive points for this lab. Because iLearn does not accept shell scripts, please rename the file with a .txt extension.