

COMPUTER SYSTEMS

UNIT7: OPERATING SYSTEM MANAGEMENT

Activity 1

**CFGS
DAM**

1. Display the contents of the **/etc/passwd** file, using **cat /etc/passwd**. Analyze the fields in each row.

Display the contents of the **/etc/group** file, using **cat /etc/group**. Analyze the fields in each row.

Analyze the contents of the **/etc/sudoers** file. Find out how can be edited its content. Is it necessary to edit it before a user can perform actions on behalf of other users?

2. Create 2 users with the **/etc/skel** configuration files. Deletes a user, keeping their files. Totally deletes another user. Check actions

Display the contents of the **/etc/skel** directory, using **ls -the /ect/skel**. Pay attention to each file.

3. Create a user with his/her own GID. Create multiple files with that user. Modify the user's GID. Assign the old files to the new GID.

4. Change a user's password using: **sudo passwd user**

5. Run **ls -l /sr/bin/passwd**. Note who the owner is and whether the **set-uid** is active in that file.

Run **ls -l /tmp**. Notice who owns it, the group, and whether the directory's sticky-bit is active.

6. Create a file named test.txt and apply the following octal permission masks on test.txt, checking its result: "rwsrwxrwt", "rwxrwxrw-" and "rw-r-srw-".

7. On the previous test.txt file, apply the following modifications on the permission mask checking its results.

Disable read permission for group and others

Enable execute permission for owner and disable set-gid

Grant read and write permissions to the user and group, overriding all other permissions. For the rest of the users, their permissions are maintained.

8. Get permissions on newly created files and directories by applying a 0022 permission mask.

9. Create a test_umask.txt file and a test_umask directory. Modify the permission mask to 0002. Create a new test_umask2.txt file and a new directory test_umask2 rechecking their permissions.

10. Create a TEST local variable with a value of "1". Verify that it is a local variable. Make it global and check it out. Delete it.

Create a TEST local variable with a value of "2". Verify that it is a local variable by launching the bash program (this program is the one of the command interpreter that, when launched from a shell, creates a child process for its execution). Return to the parent process of bash (exit) and makes the TEST variable global. Verify that it is a global variable by launching the bash program. Within bash, create the TEST_CHILD environment variable with a value of "3". Return to the father and find if it is recognized. Delete the TEST variable. Define a TEST environment variable with value "OK" permanently in the ~/.bashrc non-login shell configuration file and verify that we can view its contents from a login shell terminal.

COMPUTER SYSTEMS UNIT7: OPERATING SYSTEM MANAGEMENT Activity 1	CFGS DAM
---	---------------------

Search the Internet for information to edit PS1 and PS2 variables using escape sequences.

11. Create an alias named "li" that executes "ls -ltra". Check the creation of the alias and delete it.
12. Access the Windows "Computer Manager" and display all users (disabled or not) and groups of the system, carefully reading its description. Which users are disabled by default?
13. Practice with the different **ps** options studied here to collect different information from system processes. Analyze the information according to the header.

Study the real-time processes of the system. Filter real-time process information by users or by PID. Analyze the information according to the header in each case.
14. Using the **sleep** command (which blocks execution for a time expressed by default in seconds, such as: sleep 12), suspend the return of the shell prompt for 50 seconds. For that task or background and then to foreground, before it finishes.
15. Launch the **yes process > /dev/null** in the background with priority +10 with a common user. Try to increase the priority, is it possible? Performs the same procedure as superuser.
16. Access the Windows "Task Manager" and observe the running applications. Observe the resources consumed by all processes. What is the process that consumes the most memory? Open a word processor and finish running.
17. Launch several processes similar to those of the previous example, trying to send them different termination, pause and continuation signals, checking their status after each one.
18. Practice with all the performance and statistics tools specified in this section. If in doubt, read the help manuals.

Practice and analyze the "Performance", "Users", "Details" and "Services" tabs of the Microsoft Windows 10 "Task Manager".
19. Install, test and study at least one application corresponding to each section of this classification.