

CST8177 – Assignment 1

Managing File Access

Objectives

- To understand how to create a directory and file rights or permissions system that meets the requirements of all users who will be using the system;
- To plan for creating such a rights / permissions system;
- To test this rights / permission systems to ensure that it works as required;
- To explore what does work, what is possible, and what isn't possible for such a problem.
- Helpful commands: **passwd**, **useradd**, **grpadd**, **chgrp**, **groups**, **id**, **newgrp**, **chown**, **chmod**, **script**, and others

Submission

This project requires two submissions:

1. Gzipped-tar submission to your lab teacher by email. Name the file **A1-8177-<your-name>.tgz**.

Example: **A1-8177-Hawking.tgz** or

(for a team) **A1-8177-Hawking-Turing.tgz**.

You can combine **gzip** with **tar** by using the **tar** command **czf** to create the file, or separately create the tar file with **cf** and then use **gzip**. Be careful with file extensions.

Note: If the naming convention is not followed, this part of the submission will not be considered for marking.

The submission contains the following files from your Linux system:

- The excerpt of the **/etc/group** file after completing the assignment: the excerpt contains only the groups that were created for this assignment.
 - The excerpt of the **/etc/passwd** file after completing the assignment: the excerpt contains only the users that were created for this assignment.
 - One document with the output of both commands, clearly labelled:
 - **find /home/project -ls**
 - **tree -pug /home/project**
2. Paper-based submission in your lab teacher's physical dropbox (beside room T324). The submission must be stapled in the top left corner and may not be submitted in an envelope, folder, or equivalent.

The paper submission contains:

- Cover page listing the standard information (see Submission Standard on course web site);
- A brief summary of the requirements as you understand them;
- Some description about the solution you propose to use; some insight into your analysis and design of the solution.
- A printed copy of the emailed submission with the following files from your Linux system:
 - The excerpt of the **/etc/group** file after completing the assignment: the excerpt contains only the groups that were created for this assignment.
 - The excerpt of the **/etc/passwd** file after completing the assignment: the excerpt contains only the users that were created for this assignment.
 - One document with the output of both commands, clearly labelled:

```
find /home/project -ls
tree -pug /home/project
```
- One document with the following items, clearly labelled:
 - the table that lists all files and file attributes created for the setup
 - the completed Test Plan.
 - The script command (see **man script**) output of a complete and successful execution of the Test Plan.
 - The tables from the end of this document, completely filled out from your test results.
- Note: Plagiarism will not be tolerated. College plagiarism policy will be strictly enforced.
- You may work as pairs (teams of two). If a team is composed of students from different labs, you will have to decide how and when to work together.

Section A - Planning

Problem description

For this assignment, you will be the system administrator of a fictional company called AlgoTech, a medium-sized software development company. As such, you are in charge of the file server, which is implemented as a Fedora Linux server.

After their yearly review, the executive committee has proposed that some changes be made to allow for proper sharing and protection of information across the different departments of the company.

The company has identified three major user groups:

1. The Manager group (henceforth called **mgmt**): this group ensures that the timelines and budgetary constraints of the projects are observed.
2. The Project Leader group (henceforth called **pl**): this group oversees and administers projects.
3. The Development group (henceforth called **dev**): this group is responsible for the actual development (analysis, design & implementation) of the projects.

The executive committee has identified the following document sharing requirements. All files are:

- To be stored in the **/home/project** directory on the server
- Each group will have a series of subdirectories creating the document tree below the **/home/project** directory.
- There is a share directory for files shared by everyone (**/home/project/share**): every employee can create files and directories.
- mgmt group: Creates financial documents and reports of three categories: **approved**, **final**, and **secure**
 - Secure mgmt reports are modifiable by mgmt only
 - Final mgmt reports are modifiable by mgmt and readable by pl
 - Approved mgmt reports are modifiable by mgmt and readable by pl and dev
- dev group: Creates documents in three formats: **preliminary**, **revised** and **final**
 - Preliminary dev documents are modifiable by dev only
 - dev can modify revised dev documents which are readable by pl
 - Final dev documents are modifiable by dev and readable by mgmt and pl
- pl group: Does not create documents.

The executive committee, using the information above, has created a preliminary directory tree structure. All the directories listed are located in **/home/project: mgmt/secure, mgmt/final, mgmt/approved, dev/prelim, dev/rev, dev/final** and **share**.

Proposed solution

Plan the assignment of ownership (user, group) and permissions (for files and directories) to ensure that the directory tree structure, access rights, and access for each user/group is done to match the criteria listed above.

There may be more than one way to solve the problem.

NOTE: ACLs may NOT be used in this assignment.

You may use the table below, Table #1: Planning directory access, to determine the rights you will need to assign for each directory: place a check mark.

You may use the table below, Table #2: Planning file access, to determine the rights you will need to assign for each file: place a check mark.

Section B - Implementation and Testing

Implementation

- Create the users and groups and associate the right users to the groups.
 - Note: Keep the default login group that is created when using the **useradd** command and add additional groups as required by your solution..
 - When creating additional users and/or groups limit yourself to the minimum number of additional users and/or groups required. Keep both user and group names short but comprehensible.
- Do not overlook the permission flags **SUID**, **SGID**, and **Sticky**.
- Your solution may include extra "filter" directories and soft or hard links.
- Create two users per group for testing. Example: **mgmt1**, **mgmt2**
- Implement your proposed directory structure. Create non-empty text files in each of the directories.
- Set the ownership/permissions on all file objects as you have determined in the Planning section above.
- Example:

```
su - mgmt1
echo "management approved text"
    > /home/project/mgmt/approved/file.txt
exit
su - dev1
```

... and so on
- Set the ownership and permissions on all file objects as you have determined in the section above.
- Establish what **umask** setting each class of users requires at login time.

Testing

Test your setup. Log in as each user in turn and test the proper access rights to files that user needs access to, and nothing else, by accessing each directory and trying to read and write each file in each directory one at a time. To do this, fill out Sample Table #3: Test Plan for Files as suitably modified to handle all your users.

Develop the Test Plan you will follow, showing for each individual test at least:

- a test identifier
- the reason the for test
- actual commands to be used
- expected result of the commands

Use the **script** command (it's not a script, nor does it require you to write a script – it's just the name of a command and is short for "typescript", a now-obsolete word) to log all your final test activities for inclusion in your submissions. You may make light editing changes: for example, to clean up the prompts.

Note: All this testing effort may seem time consuming, but it is the only way to ensure that your solution is working properly. This is one of the places you would ordinarily use scripting, to develop a test script to avoid tedious and error-prone manual test entry.

You may create a test script if you wish but it's not a requirement.

Section C - Identifying problems and refining the solution

Identifying problems

- Potential problem #1: Directory management

Once you have determined proper file access, determine what capabilities users have in the various directories.

Can users who have read-only access to files in a given directory also create files in that same directory?

- Potential problem #2: File access

Does the owner of a file have to change ownership and/or permissions on every newly-created file?

Note: It is most effective to keep default permissions (based on **umask**) whenever possible: when a user creates a file the default file permissions (as set by the user's **umask**) should be sufficient. You can require a certain **umask** setting for each user of the system.

- Potential problem #3: File access

A user should only be able to delete their own files in the publicly shared folder **share**.

Refining the solution

- Adjust your solution by modifying the directory structure and/or ownership and/or permissions and/or anything else that helps in implementing a refined solution.

Below are samples for Tables 1, 2, and 3.

Not supplied, but required, is a similar testing table to summarize directory testing (what will you do to test reading, writing, and accessing a directory? Answer this and other questions in your paper submission).

Directories	Group mgmt			Group pl			Group dev
	R	W	X	R	W	X	R
mgmt/ secure							
mgmt/final							
mgmt/ approved							
dev/prelim							
dev/rev							
dev/final							
share							

Sample Table #1: Planning directory access suggestion

File	Group mgmt		Group pl		Group dev		others
	R	W	R	W	R	W	R
mgmt/secure/ text-file.ms							
mgmt/final/ text-file.mf							
mgmt/approved text-file.ma							
dev/prelim/ text-file.dp							
dev/rev/ text-file.dr							
dev/final text-file.df							
share/text-file							

Sample Table #2: Planning file access suggestion

Files	User mgmt2		User pl2		User dev2		Other users
	Write (echo)	Read (cat)	Write (echo)	Read (cat)	Write (echo)	Read (cat)	Write (echo)
mgmt/secure/ text-file.ms							
mgmt/final/ text-file.mf							
mgmt/approved text-file.ma							
dev/prelim/ text-file.dp							
dev/rev/ text-file.dr							
dev/final text-file.df							
share/text-file							

Sample Table #3: Testing suggestion for files