Mini-Project MA-ICR

A shared encrypted network file system

Titus Abele

MSE Computer Science
HES-SO Master
Lausanne, Switzerland
May 1, 2024

Contents

Listings

List of Figures

Introduction

Before jumping into the theory behind a shared encrypted network file system, it is important to lay a foundation about what a file system is, why it requires sharing capabilities and why security is paramount. A file system (FS) manages and provides access for resources. Generally, resources are composed of folders containing files or other folders. This way, an organizational hierarchy of resources can be established.

```
[drwxr-xr-x 4.0K]  .
  |--- [lrwxrwxrwx    7]  bin -> usr/bin
  |--- [drwxr-xr-x 4.0K]  boot
  |--- [drwxr-xr-x 3.5K]  dev
  |--- [drwxr-xr-x 4.0K]  etc
  |--- [drwxr-xr-x 4.0K]  home
  |--- [ rwxrwxrwx 2.0M]  init
  |--- [lrwxrwxrwx    7]  lib -> usr/lib
  |--- [lrwxrwxrwx    9]  lib32 -> usr/lib32
  |--- [lrwxrwxrwx    9]  lib64 -> usr/lib64
  |--- [lrwxrwxrwx   10]  libx32 -> usr/libx32
  |--- [drwx------  16K]  lost+found
  |--- [drwxr-xr-x 4.0K]  media
  |--- [drwxr-xr-x 4.0K]  mnt
  |--- [drwxr-xr-x 4.0K]  opt
  |--- [dr-xr-xr-x    0]  proc
  |--- [drwx------ 4.0K]  root
  |--- [drwxr-xr-x  680]  run
  |--- [lrwxrwxrwx    8]  sbin -> usr/sbin
  |--- [drwxr-xr-x 4.0K]  snap
  |--- [drwxr-xr-x 4.0K]  srv
  |--- [dr-xr-xr-x    0]  sys
  |--- [drwxrwxrwt  12K]  tmp
  |--- [drwxr-xr-x 4.0K]  usr
  |--- [drwxr-xr-x 4.0K]  var

23 directories, 1 file
```

Listing 1: Ubuntu root folder

Access should be handled so that some roles can access some resources, this is generally done by using dedicated roles such as administrators, owners, guests or even users. An example can be seen in Listing 1, it shows all folders and files contained within the root directory of a machine running Ubuntu, a popular Linux distribution. The root directory is the top-level directory in the file system's hierarchy. Left of the resource names (`boot`, `dev`, `etc`...) we can find their relative permissions. For instance, the `boot` directory is marked as `[drwxr-xr-x 4.0K] boot` which means:

- The `d` in first position indicates the nature of the resource, in this case a directory.
- The pattern `rwxr-xr-x` translates to the owner having read, write, and execute permissions, while the group and others have read and execute permissions only.

Each triplet (`rwx`) relates to a specific permission class. In a Unix-like file system such as the one depicted in Listing 1, these classes are defined in order as "Owner-Group-Others".

This means that for the `boot` directory:

- The Owner has read (`r`), write (`w`) and execute (`x`) permissions.
- The Group that is associated with the directory and all other users only have read and execute permissions.

This way any access to the directory can be easily limited and permissions can be revoked, approved and modified very easily.

```
seirios@T16:/$ touch foo.txt
touch: cannot touch 'foo.txt': Permission denied
```

Listing 2: Trying to create a file in the root folder without write permissions

1 Part b

etc