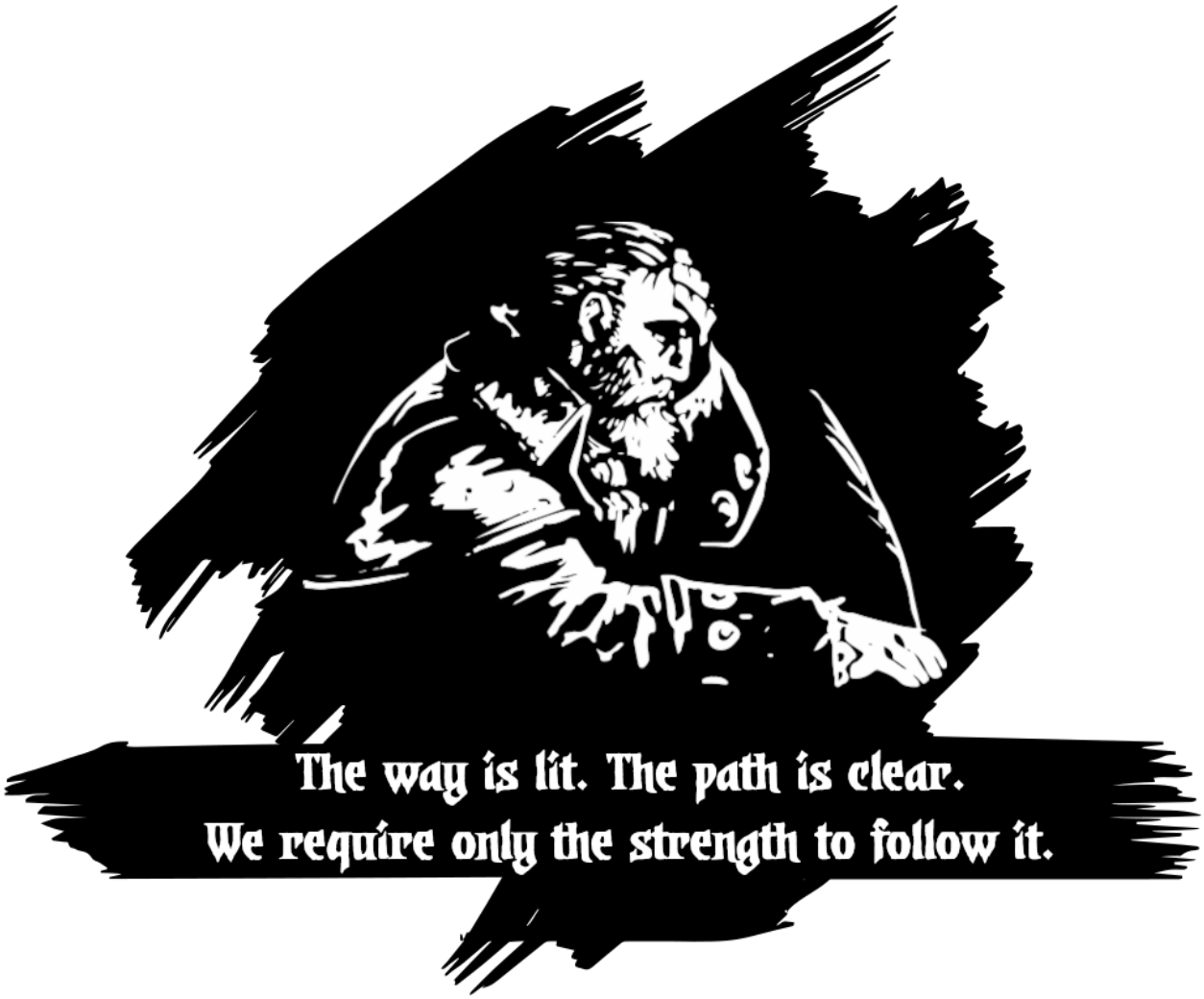
Exercises — The Right Tarball

version **#f6de30ca5ee18fb9a30e5fbfc9eb9a1de8e13354**



Assistants C/Unix 2023 <assistants@tickets.assistants.epita.fr>

Copyright

This document is for internal use at EPITA (website) only.

Contents

*https://intra.assistants.epita.fr

File Tree

```
right_tarball/
└── my_tarball.tar.gz  (to submit)
```

Authorized commands : You are only allowed to use the following commands

- builtins
- touch(1)
- mkdir(1)
- chmod(1)

Reminder : Grant execution permission to your scripts before pushing them

1 Goal

Generate a tarball with a specific architecture and permissions.

Start by creating a folder `my_tarball` at the root of your exercise directory.

2 File handling and permissions

This section must be done inside the `my_tarball` folder.

To check that you have understood notions about how to handle files, directories and their associated permissions correctly, you must:

- create an `AUTHORS` file and modify its permissions. An `AUTHORS` file contains any information about the author(s) of a project. It must have the `640` mode and must not be empty.
- create a new directory named `doc`. It must have the `700` mode.

Tips

A conventional `AUTHORS` file contains a line that can easily be parsed:

```
42sh$ cat -e AUTHORS
* prenom.nom$
```

Tips

If the octal permission system is still unclear to you, take time to read the chmod manual.

Once this is done, you must create a text file containing at least 5 lines, correctly wrapped at 80 columns - no line should exceed 80 characters. You can use the features of your text editor (Vim or Emacs) to handle the text formatting:

- this file must be named `rights`, located in the `doc` directory you just created, and have the `640` access rights.

If you lack imagination, you can write about the permission system and the different commands you can use to manipulate them.

Finally, we want you to write a script, this script should be placed as follow:

- you must create an `src` directory having the `755` mode.
- the script must be located in an `src` directory and named `authors.sh` with the `644` access rights - thus, not executable.

The script have to handles the *creation* and the *permission modification* of an `AUTHORS` file in the current working directory - the directory in which you are when running the script. The permission made by the script **must** be the same as the `AUTHORS` created before (so `640` access rights).

3 Assignment tarball

A tarball is often used for **backups** or **releases** purposes. In a release, you generally find information about the author(s) of the project, some documentation and source code. Let us make one.

To validate this exercise, you must compress your previous work in a tarball and submit it.

Your tarball must be called `my_tarball.tar.gz`. This file must be a tar archive compressed in the *gzip* format. Take time to read the manual of `tar` to find out how to create this archive.

The tarball must contain the directory `my_tarball` and all its content.

Be careful!

Do not include any trash or git related file.

Tips

You can check that your submission is correct by comparing it with this output:

```
42sh$ tar tf my_tarball.tar.gz
my_tarball/
my_tarball/AUTHORS
my_tarball/doc/
my_tarball/doc/rights
my_tarball/src/
my_tarball/src/authors.sh
```

The way is lit. The path is clear. We require only the strength to follow it.