SCALE FOR PROJECT FILESYSTEM (/PROJECTS/FILESYSTEM)

Introduction

To ensure this evaluation goes smoothly, please respect the following set of rules :

- Please remain courteous, polite, respectful and constructive at all times during this exchange. The trust bond between the school's communauty and yourself depends on it.

- Should you notice any malfunctions within the submitted project, make sure you take the time to discuss those with the student (or group of students) being graded.

- Keep in mind that some subjects can be interpreted differently. If you come accross a situation where the student you're grading has interpreted the subject differently than you, try and judge fairly whether their interpretation is acceptable or not, and grade them accordingly. Our peer-evaluation system can only work if you both take it seriously.

Guidelines

- You may only evaluate whatever is in the GiT submission directory of the student you are grading.

- Make sure to check wether the GiT submission directory belongs to the student (or group) you're grading, and that it's the right project.

- Make sure no mischievous aliases have been used to trick you into correcting something that is not actually in the official submitted directory.

- Any script created to make this evaluation session easier - whether it was produced by you or the student being graded - must be checked rigorously in order to avoid bad surprises.

- If the student who is grading this project hasn't done the project him/herself yet, he/she must read the whole topic before starting the evaluation session.

- Use the flags available to you on this scale in order to report a submission directory that is empty, non-functional, that contains a norm errors or a case of cheating, etc...
In this case, the evaluation session ends and the final grade is 0 (or -42, in case of cheating). However, unless the student has cheated, we advise you to go through the project together in order for the two (or more) of you to identify the problems that may have led for this project to fail, and avoid repeating those mistakes for future projects.

- Careful ! The code testing is done in the student hand made linux distribution.

Attachments

Subject (https://cdn.intra.42.fr/pdf/pdf/3985/lk_filesystem.en.pdf)

Main

Free points

Does the project contain a Makefile ?

Does the project compile ?

Yes No

Base

Insert the module into the kernel.

Then, do something like:

```
mount -o loop -t fortytwofs directory/ image
```

Is this command working ?

If it's needed, format the disk image with the student provided binary.
Create an image with something like:
```
dd if=/dev/zero of=image bs=2048
```
The size can be changed.

☑ Yes ☒ No

The real stuff

Enter the filesystem with cd and test the following:

- Directory creation (mkdir)
- File creation (touch)
- Ownership (chown)
- Rights (chmod)
- Links (ln -s)

If all of those things work, grade this question. Otherwise the scale stop here.

☑ Yes ☒ No

Code

Look at the student code.

Does the student use Superblocks and Inodes ?
Both Linux ones and customs ?

☑ Yes ☒ No

Bonus

Creative

Ask the student about his bonuses, and grade them at your discretion.

Rate it from 0 (failed) through 5 (excellent)

Ratings

Don't forget to check the flag corresponding to the defense

✔ Ok ★ Outstanding project

Empty work | Incomplete work | No author file | Invalid compilation

Norme | Cheat | Crash | Incomplete group | Forbidden function

Conclusion

Leave a comment on this evaluation

Preview!!!