Operating Systems
Jacobs University Bremen
Dr. Jürgen Schönwälder

Course: CO20-320202
Date: 2019-11-25
Due: 2019-12-02

OS 2019 Problem Sheet #6

Warning: Whenever you prefix a shell command with sudo, make sure you know what you are doing. And never work as root unless you know what you are doing.

Problem 6.1: *file systems* (1+1+1+1+1+1 = 6 points)

On Linux systems, you can create a file system in a regular file and then mount it into your file system tree:

```
dd of=vhd.ext3 bs=1k seek=4096 count=0
sudo mkfs -t ext3 vhd.ext3
sudo modprobe loop
mkdir ./mnt
sudo mount vhd.ext3 ./mnt
```

Your new file system will now appear under the `./mnt` directory.

a) The new file system is not empty. What is the purpose of the directory that is contained in the new file system?

b) Change the current working directory so that you are located in the new file system. Run the shell command `stat -f .` and explain the difference between free blocks and available blocks.

c) Change the current working directory such that it is outside the new file system. Delete the underlying file `vhd.ext3`. What happens to the mounted file system?

d) Change the current working directory such that you are located in the new file system and run `stat -f .`. Create a large file in the new file system using the following command:

```
sudo dd of=big.data bs=1k seek=4096 count=0
```

How large is the file that you have created? Run `stat -f .` again. How have the free block and free inode numbers changed? Explain what you observe.

e) Change the current working directory such that you are located in the new file system. Execute the following commands:

```
sudo chattr +i big.data
sudo rm big.data
```

Explain what you observe. Learn about tools that can display file attributes.

f) Change the current working directory such that you are located outside of the new file system. Install a statically linked version of `busybox` on your system (e.g., 'sudo apt install busybox-static' on a Debian of Ubuntu system). Now copy the busybox program into your new file system and the run a `chroot` command:

```
sudo mkdir -p mnt/bin
sudo cp /bin/busybox mnt/bin/busybox
sudo ln mnt/bin/busybox mnt/bin/sh
sudo chroot mnt /bin/sh
```

Explain what has happened when you executed the `chroot` command. Why was it important to copy a statically linked version of `busybox`?

Problem 6.2: *overlay file systems* (2+1+1 = 4 points)

Create an empty directory and change into it. Then execute the following shell commands:

```
mkdir lower
mkdir upper
mkdir work
mkdir over
sudo mount -t overlay overlay -olowerdir=lower,upperdir=upper,workdir=work over
```

a) Create a file `over/top`. Where is the file `over/top` actually stored? Create a file `lower/low`. What happens if you append data to `over/low`, i.e., where is the data actually stored? What happens if you unlink `over/low`? (How does the file system remember that `over/low` got unlinked?) Create a file `lower/lo`. What happens if you change the permissions of `over/lo`?

b) Describe at least two use cases for overlay file systems.

c) Does the overlay file system always copy data when metadata of a file in the lower layer is changed? Is it possible to stack multiple lower layers? Explain.