DAT151: Assignment 1 Report

Group: 2

Group Members: Soukup Jan, Fabienne Feilke

Date: February 10, 2026

Note: After monday's tutorial we shortened command line output texts so that the report isn't that long.

Part 1 - The Filesystem

Task 1: Filesystem basics

1. Describe some of the key differences between the Linux/UNIX filesystem and Windows. How are files and folders organised in Linux/UNIX?

Linux uses a single unified tree structure starting at the root (`/`), while Windows uses drive letters (C:, D:) for different partitions.

Linux filenames are case-sensitive (`File.txt` and `file.txt` are different), while Windows is generally case-insensitive.

Linux identifies file types by metadata/content rather than strictly relying on extensions like `.exe` or `.txt`.

2. What is a mount point?

A mount point is a directory in the existing filesystem hierarchy where a new filesystem (like a partition) is attached to make it accessible. It acts as an acces node, allowing the operating system to treat external storage as subdirectory

Command line instructions:

```
soukup@localhost:~$ sudo parted -s /dev/sda unit s print free
Model: ATA SAMSUNG MZ7PD256 (scsi)
Disk /dev/sda: 500118192s
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start       End         Size        File system  Name
Flags
    34s         2047s       2014s       Free Space
 1      2048s       1230847s    1228800s    fat32        EFI System
Partition  boot, esp
 2      1230848s    3327999s    2097152s    xfs
bls_boot
 3      3328000s    481042431s  477714432s
lvm
```

```
    481042432s  500118158s  19075727s   Free Space

soukup@localhost:~$ sudo parted -s /dev/sda mkpart
"dat151_assignment1_partition" xfs 481042432s 100%
soukup@localhost:~$ sudo mkfs -t xfs /dev/sda4
meta-data=/dev/sda4              isize=512    agcount=4, agsize=596096 blks
     =                       sectsz=512   attr=2, projid32bit=1
     =                       crc=1        finobt=1, sparse=1, rmapbt=1
     =                       reflink=1    bigtime=1 inobtcount=1 nrext64=1
     =                       exchange=0
data     =                       bsize=4096   blocks=2384384, imaxpct=25
     =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0, ftype=1, parent=0
log      =internal log           bsize=4096   blocks=16384, version=2
     =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
Discarding blocks...Done.

soukup@localhost:~$ sudo mkdir /opt/assignment_data
soukup@localhost:~$ sudo mount /dev/sda4 /opt/assignment_data
soukup@localhost:~$ df -h /opt/assignment_data
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda4       9.1G  210M  8.9G   3% /opt/assignment_data

soukup@localhost:~$ lsblk /dev/sda4
NAME MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda4   8:4    0  9.1G  0 part /opt/assignment_data
```

3. In most UNIX systems, there are seven types of files defined. What are these? Use the file command to display the information of a few files. In which of the seven categories do these files belong?

Seven types:

- Regular files (-) - Data files, executable programs, text files, or images
- Directories (d) - Files that act as containers for other files and folder structures
- Character devices (c) - Device files that handle data character-by-Character, such as terminals (/dev/tty) or printers.
- Block devices (b) -Device files that handle data in blocks, typical for storage devices like hard drives (/dev/sda).
- Local domain sockets(s) - Files used for inter-process communication, often over a network
- Named pipes (FIFO)(p) - Used for inter-process communication, allowing data to flow from one process to another
- Symbolic links (l) -Files that point to another file or directory (shortcuts)

Commands used:

```
ls -ld /etc/passwd /etc /dev/sda4 /dev/tty
```

Categorization:

- `/etc/passwd`: regular file (`-`).
- `/etc`: directory (`d`).
- `/dev/sda4`: block device (`b`).
- `/dev/tty`: character device (`c`).

4. Hard links vs. symbolic links

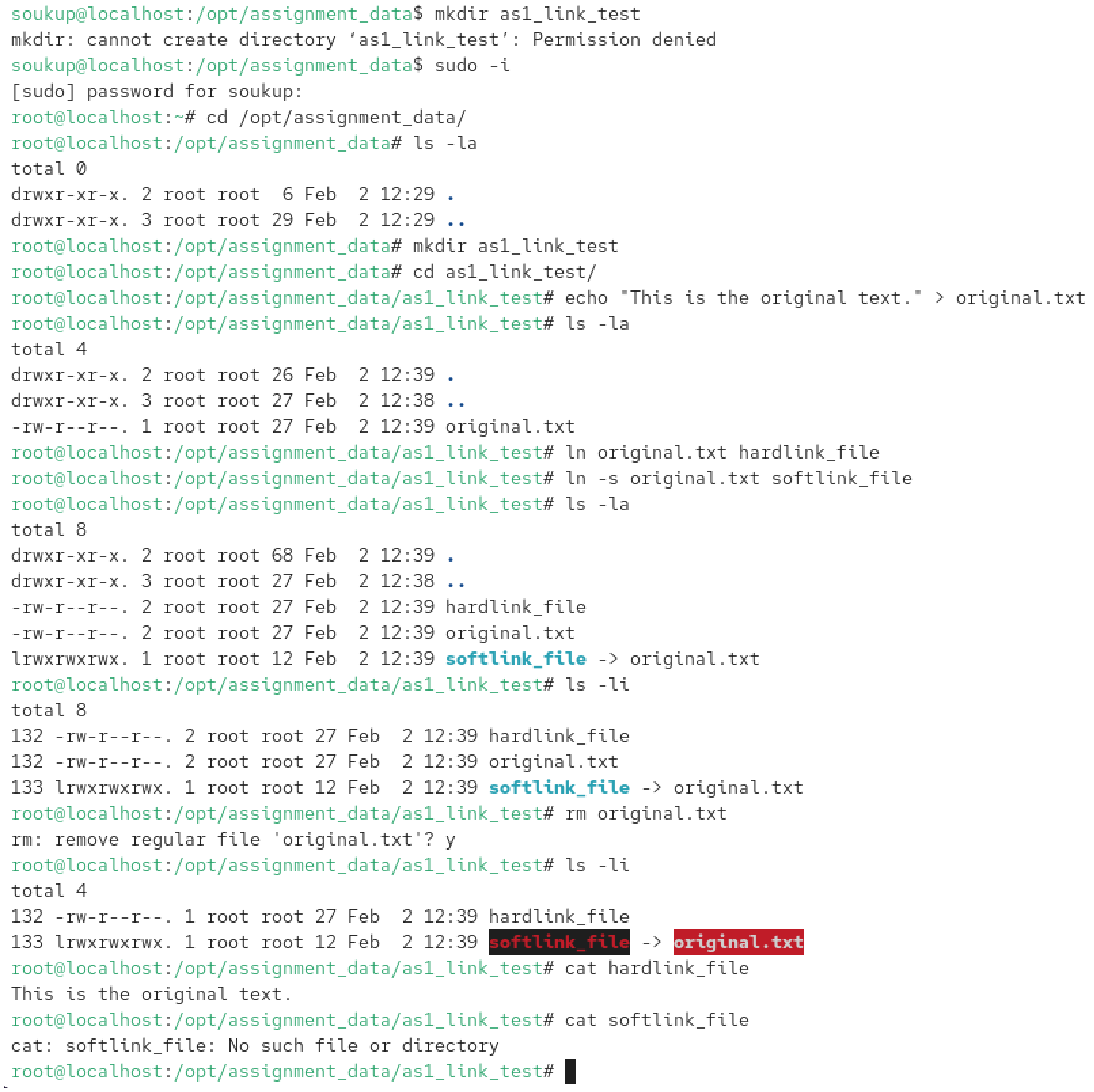
Commands used:

```
soukup@localhost:/opt/assignment_data$ sudo -i
root@localhost:/opt/assignment_data/as1_link_test# echo "This is the original text." > original.txt
root@localhost:/opt/assignment_data/as1_link_test# ls -la
total 4
drwxr-xr-x. 2 root root 26 Feb  2 12:39 .
drwxr-xr-x. 3 root root 27 Feb  2 12:38 ..
-rw-r--r--. 1 root root 27 Feb  2 12:39 original.txt
root@localhost:/opt/assignment_data/as1_link_test# ln original.txt hardlink_file
root@localhost:/opt/assignment_data/as1_link_test# ln -s original.txt softlink_file
root@localhost:/opt/assignment_data/as1_link_test# ls -li
total 8
132 -rw-r--r--. 2 root root 27 Feb  2 12:39 hardlink_file
132 -rw-r--r--. 2 root root 27 Feb  2 12:39 original.txt
133 lrwxrwxrwx. 1 root root 12 Feb  2 12:39 softlink_file -> original.txt
root@localhost:/opt/assignment_data/as1_link_test# rm original.txt
rm: remove regular file 'original.txt'? y
root@localhost:/opt/assignment_data/as1_link_test# ls -li
total 4
132 -rw-r--r--. 2 root root 27 Feb  2 12:39 hardlink_file
133 lrwxrwxrwx. 1 root root 12 Feb  2 12:39 softlink_file -> original.txt
root@localhost:/opt/assignment_data/as1_link_test# cat hardlink_file
This is the original text.
root@localhost:/opt/assignment_data/as1_link_test# cat softlink_file
cat: softlink_file: No such file or directory
```

Explanation of the code above

- Hard link still works because it references the same inode - data.
- Soft link is broken because it references a filename that has been deleted (does not exist).

Screenshot:



```
soukup@localhost:/opt/assignment_data$ mkdir as1_link_test
mkdir: cannot create directory ‘as1_link_test’: Permission denied
soukup@localhost:/opt/assignment_data$ sudo -i
[sudo] password for soukup:
root@localhost:~# cd /opt/assignment_data/
root@localhost:/opt/assignment_data# ls -la
total 0
drwxr-xr-x. 2 root root  6 Feb  2 12:29 .
drwxr-xr-x. 3 root root 29 Feb  2 12:29 ..
root@localhost:/opt/assignment_data# mkdir as1_link_test
root@localhost:/opt/assignment_data# cd as1_link_test/
root@localhost:/opt/assignment_data/as1_link_test# echo "This is the original text." > original.txt
root@localhost:/opt/assignment_data/as1_link_test# ls -la
total 4
drwxr-xr-x. 2 root root 26 Feb  2 12:39 .
drwxr-xr-x. 3 root root 27 Feb  2 12:38 ..
-rw-r--r--. 1 root root 27 Feb  2 12:39 original.txt
root@localhost:/opt/assignment_data/as1_link_test# ln original.txt hardlink_file
root@localhost:/opt/assignment_data/as1_link_test# ln -s original.txt softlink_file
root@localhost:/opt/assignment_data/as1_link_test# ls -la
total 8
drwxr-xr-x. 2 root root 68 Feb  2 12:39 .
drwxr-xr-x. 3 root root 27 Feb  2 12:38 ..
-rw-r--r--. 2 root root 27 Feb  2 12:39 hardlink_file
-rw-r--r--. 2 root root 27 Feb  2 12:39 original.txt
lrwxrwxrwx. 1 root root 12 Feb  2 12:39 softlink_file -> original.txt
root@localhost:/opt/assignment_data/as1_link_test# ls -li
total 8
132 -rw-r--r--. 2 root root 27 Feb  2 12:39 hardlink_file
132 -rw-r--r--. 2 root root 27 Feb  2 12:39 original.txt
133 lrwxrwxrwx. 1 root root 12 Feb  2 12:39 softlink_file -> original.txt
root@localhost:/opt/assignment_data/as1_link_test# rm original.txt
rm: remove regular file 'original.txt'? y
root@localhost:/opt/assignment_data/as1_link_test# ls -li
total 4
132 -rw-r--r--. 1 root root 27 Feb  2 12:39 hardlink_file
133 lrwxrwxrwx. 1 root root 12 Feb  2 12:39 softlink_file -> original.txt
root@localhost:/opt/assignment_data/as1_link_test# cat hardlink_file
This is the original text.
root@localhost:/opt/assignment_data/as1_link_test# cat softlink_file
cat: softlink_file: No such file or directory
root@localhost:/opt/assignment_data/as1_link_test#
```

Task 2: File attributes and permissions

1. In the traditional Linux/UNIX filesystem model, every file comes bundled with a set of 16 bits. What are these bits, and why are they needed? Explain.

The 16 bits include 9 permission bits (read/write/execute for user, group, others), 3 special mode bits (SUID, SGID, sticky bit), and 4 bits for file type. They define both access control and the file's type in the filesystem.

2. The nine permission bits are often represented using octal numbers. Explain what these are, and how the permissions bits 110 100 101 can be represented using octals. What permissions does a file with the octal value 745 have?

`110 100 101` in octal is **645**. Octal **745** corresponds to `rwxr--r-x` (user: rwx, group: r, others: rx).

3. Chmod commands

Commands used *(I created the folder using sudo, which is why i needed to use sudo in other commands as well)*:

```
soukup@localhost:/opt/assignment_data/file_permissions$ sudo touch
file_mnemonic.txt
soukup@localhost:/opt/assignment_data/file_permissions$ sudo chmod
u=rw,g=rx,o=rx file_mnemonic.txt
soukup@localhost:/opt/assignment_data/file_permissions$ ls -la
total 0
drwxr-xr-x. 2 root root 31 Feb  2 12:48 .
drwxr-xr-x. 3 root root 30 Feb  2 12:47 ..
-rw-r-xr-x. 1 root root  0 Feb  2 12:48 file_mnemonic.txt
soukup@localhost:/opt/assignment_data/file_permissions$ sudo touch
file_octal.txt
soukup@localhost:/opt/assignment_data/file_permissions$ sudo chmod 655
file_octal.txt
soukup@localhost:/opt/assignment_data/file_permissions$ ls -la
total 0
drwxr-xr-x. 2 root root 53 Feb  2 12:50 .
drwxr-xr-x. 3 root root 30 Feb  2 12:47 ..
-rw-r-xr-x. 1 root root  0 Feb  2 12:48 file_mnemonic.txt
-rw-r-xr-x. 1 root root  0 Feb  2 12:50 file_octal.txt
```

Why octal can be preferred: It's easy to understand, fast to write, more compact compared to other options.

Part 2 - Software installation and management

Task 3: Package management systems and lower level package management

1. What is a package management system, and why do we need one? What is the package format used by the package management system on your machine?:

A package management system that automates software installation, updates and removal of software. It also simplifies the distribution of software.

AlmaLinux uses the **.rpm** format.

2. RPM operations

For this task we forgot to capture a screenshot of the command line and save all of the output, however we both saw the behaviour of rpm installation and its problems.

Commands and output:

```
rpm -qa
libgcc-14.3.1-2.1.el10.alma.1.x86_64
fonts-filesystem-2.0.5-18.el10.noarch
...
```

```
emacs-29.4-12.el10.x86_64
mariadb-connector-c-config-3.4.4-1.el10.noarch

soukup@localhost:/opt/assignment_data/file_permissions$ rpm -qa | wc -l
1289
soukup@localhost:/opt/assignment_data/file_permissions$ rpm -q --requires bash
/usr/bin/sh
config(bash) = 5.2.26-6.el10
filesystem >= 3
libc.so.6()(64bit)
libc.so.6(GLIBC_2.11)(64bit)
libc.so.6(GLIBC_2.14)(64bit)
libc.so.6(GLIBC_2.15)(64bit)
libc.so.6(GLIBC_2.2.5)(64bit)
libc.so.6(GLIBC_2.25)(64bit)
libc.so.6(GLIBC_2.3)(64bit)
libc.so.6(GLIBC_2.3.4)(64bit)
libc.so.6(GLIBC_2.33)(64bit)
libc.so.6(GLIBC_2.34)(64bit)
libc.so.6(GLIBC_2.36)(64bit)
libc.so.6(GLIBC_2.38)(64bit)
libc.so.6(GLIBC_2.4)(64bit)
libc.so.6(GLIBC_2.8)(64bit)
libc.so.6(GLIBC_ABI_DT_RELR)(64bit)
libtinfo.so.6()(64bit)
rpmlib(BuiltinLuaScripts) <= 4.2.2-1
rpmlib(CompressedFileNames) <= 3.0.4-1
rpmlib(FileDigests) <= 4.6.0-1
rpmlib(PayloadFilesHavePrefix) <= 4.0-1
rpmlib(PayloadIsZstd) <= 5.4.18-1
rtld(GNU_HASH)
```

Result of installing with rpm: The install failed due to missing dependencies, as rpm does not resolve or download dependencies automatically - we have to install them ourselves.

Task 4: High-level package management

1. DNF system: DNF (Dandified YUM) locates software in repositories and automatically resolves dependencies.

2. What is a software repository? If you are using dnf as your high-level package management system, add the epel software repository to your system. If you are using something else try a different repository of choice.:

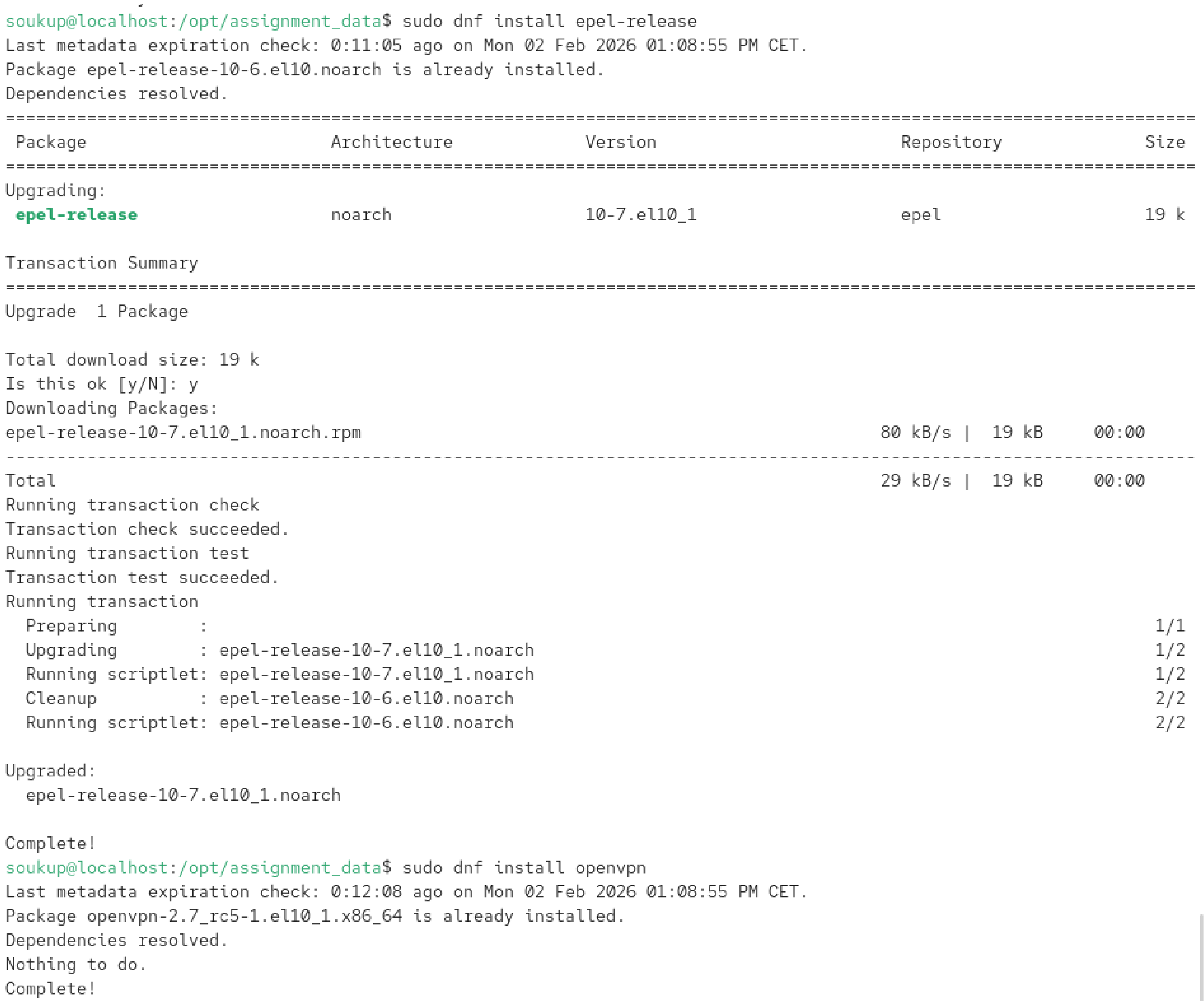
Software repository is a storage location for software packages (remote or local).

Command output:

```
soukup@localhost:/opt/assignment_data$ sudo dnf install epel-release
Last metadata expiration check: 0:11:05 ago on Mon 02 Feb 2026 01:08:55 PM
```

```
CET.
Package epel-release-10-6.el10.noarch is already installed.
Dependencies resolved.
============================================================================
==========================================
 Package                         Architecture             Version
Repository                Size
============================================================================
==========================================
Upgrading:
 epel-release                    noarch                   10-7.el10_1
epel                      19 k

Transaction Summary
============================================================================
==========================================
Upgrade  1 Package

Total download size: 19 k
Is this ok [y/N]: y
Downloading Packages:
epel-release-10-7.el10_1.noarch.rpm
80 kB/s |  19 kB     00:00
----------------------------------------------------------------------------
------------------------------------------
Total
29 kB/s |  19 kB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
    Preparing        :
1/1
    Upgrading        : epel-release-10-7.el10_1.noarch
1/2
    Running scriptlet: epel-release-10-7.el10_1.noarch
1/2
    Cleanup          : epel-release-10-6.el10.noarch
2/2
    Running scriptlet: epel-release-10-6.el10.noarch
2/2

Upgraded:
    epel-release-10-7.el10_1.noarch

Complete!
soukup@localhost:/opt/assignment_data$ sudo dnf install openvpn
Last metadata expiration check: 0:12:08 ago on Mon 02 Feb 2026 01:08:55 PM
CET.
Package openvpn-2.7_rc5-1.el10_1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
```

Screenshot:



```
soukup@localhost:/opt/assignment_data$ sudo dnf install epel-release
Last metadata expiration check: 0:11:05 ago on Mon 02 Feb 2026 01:08:55 PM CET.
Package epel-release-10-6.el10.noarch is already installed.
Dependencies resolved.
=====================================================================================================================
 Package                        Architecture             Version                       Repository               Size
=====================================================================================================================
Upgrading:
 epel-release                   noarch                   10-7.el10_1                   epel                     19 k

Transaction Summary
=====================================================================================================================
Upgrade  1 Package

Total download size: 19 k
Is this ok [y/N]: y
Downloading Packages:
epel-release-10-7.el10_1.noarch.rpm                                                      80 kB/s |  19 kB     00:00
---------------------------------------------------------------------------------------------------------------------
Total                                                                                    29 kB/s |  19 kB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                                                             1/1
  Upgrading        : epel-release-10-7.el10_1.noarch                                                             1/2
  Running scriptlet: epel-release-10-7.el10_1.noarch                                                             1/2
  Cleanup          : epel-release-10-6.el10.noarch                                                               2/2
  Running scriptlet: epel-release-10-6.el10.noarch                                                               2/2

Upgraded:
  epel-release-10-7.el10_1.noarch

Complete!
soukup@localhost:/opt/assignment_data$ sudo dnf install openvpn
Last metadata expiration check: 0:12:08 ago on Mon 02 Feb 2026 01:08:55 PM CET.
Package openvpn-2.7_rc5-1.el10_1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
```

3. Try using dnf to install the openvpn package. Does this differ in any way from task 3? If it installs, remove the package afterwards.:

Using the `dnf` to install the OpenVPN package also installs all of the required dependencies as well. This means we do not have to install them ourselves and the `dnf` package manager also pulls the correct versions of the identified dependencies.

We also tried the interactive OpenVPN installer that also automatically pulls all needed dependencies.

```
Welcome to this OpenVPN road warrior installer!
...
Finished!

The client configuration is available in:
/opt/assignment_data/jan_soukup.ovpn
New clients can be added by running this script again.

sudo dnf remove openvpn
```

4. Use dnf to update the packages on your system:

```
sudo dnf update
```

Part 3 - Systemd

Task 1: Systemd targets and units

1. a) What is the current target that ctrl-alt-del.target points to, and what are the consequences?

a) Current target: Points to `reboot.target`. This will restart the system on the button combination press.

Command line output:

```
soukup@localhost:/opt/assignment_data$ ls -l /usr/lib/systemd/system/ctrl-alt-del.target
lrwxrwxrwx. 1 root root 13 Aug 17 02:00 /usr/lib/systemd/system/ctrl-alt-del.target -> reboot.target
```

b) Modify the ctrl-alt-del.target to halt the computer when Control+Alt+Del is pressed.

b) Modification: We created a symlink to the `halt.target` instead, which will halt the tartget - shutdown the PC instead.

Command line output:

```
soukup@localhost:/opt/assignment_data$ sudo ln -sf /usr/lib/systemd/system/halt.target /usr/lib/systemd/system/ctrl-alt-del.target
soukup@localhost:/usr$ sudo systemctl daemon-reload
```

2. Set the systemd default.target to multi-user.target, then restart the computer.

Command line output:

Check current default

```
soukup@localhost:/usr$ systemctl get-default
graphical.target
```

Change to multi-user:

```
soukup@localhost:/usr$ sudo systemctl set-default multi-user.target
Removed '/etc/systemd/system/default.target'.
Created symlink '/etc/systemd/system/default.target' →
'/usr/lib/systemd/system/multi-user.target'.
```

After this step we restarted the PC, which then booted us in a terminal mode instead of GUI. After logging in with our credentials we then switched back to GUI with the following command:

```
sudo systemctl isolate graphical.target
```

3. Reboot the computer, and boot into the emergency.target by modifying the kernel line of the Grub stanza for you system. Mount the root file system (/) rw, and create a file in the /root directory. Then restart the computer.

First we restarted the PC. On the boot sequence, we pressed 'E' to get into edit mode of the boot. We added a `systemd.user=emergency.target` - later we found out only writing `emergency` also works (shortcut). This booted us to terminal emergency mode. We then remounted `/` as `rw` and created `/root/emergency_test.txt` - this was done because we didn't have write privileges, so we needed to remount first before we could create/edit files. Output:

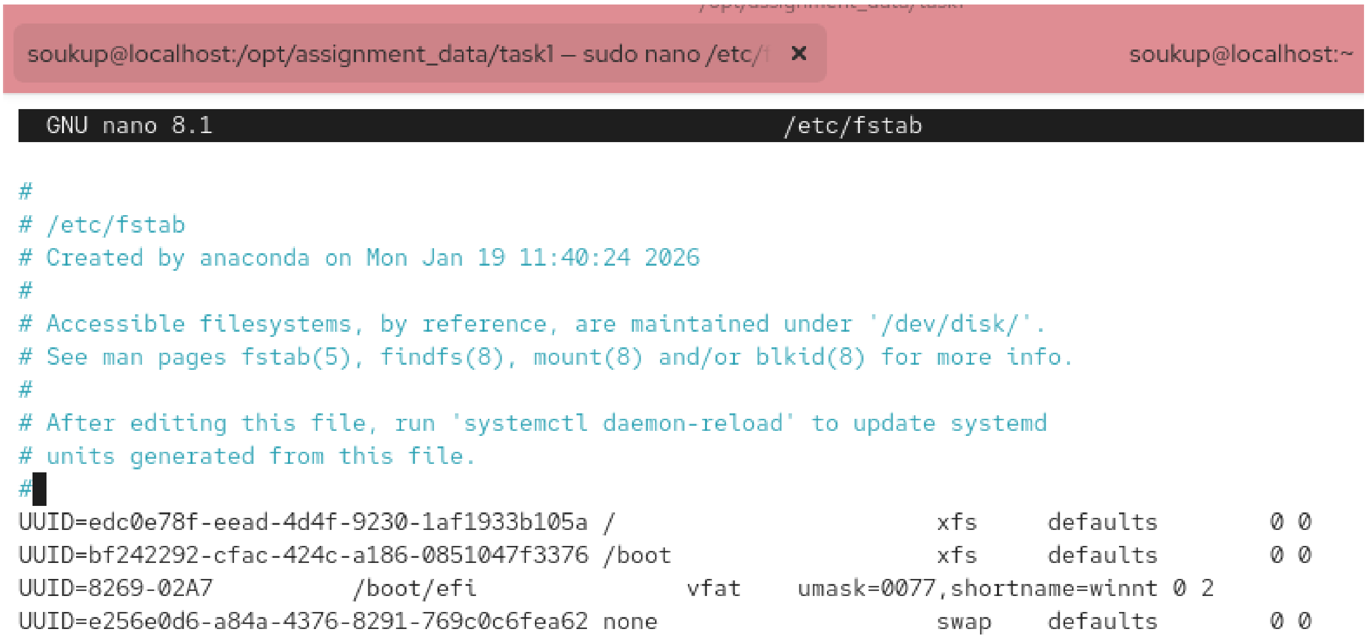
```
mount -o remount,rw /
touch /root/emergency_test.txt
```

The file `/root/emergency_test.txt` stayed at the file system even after rebooting.

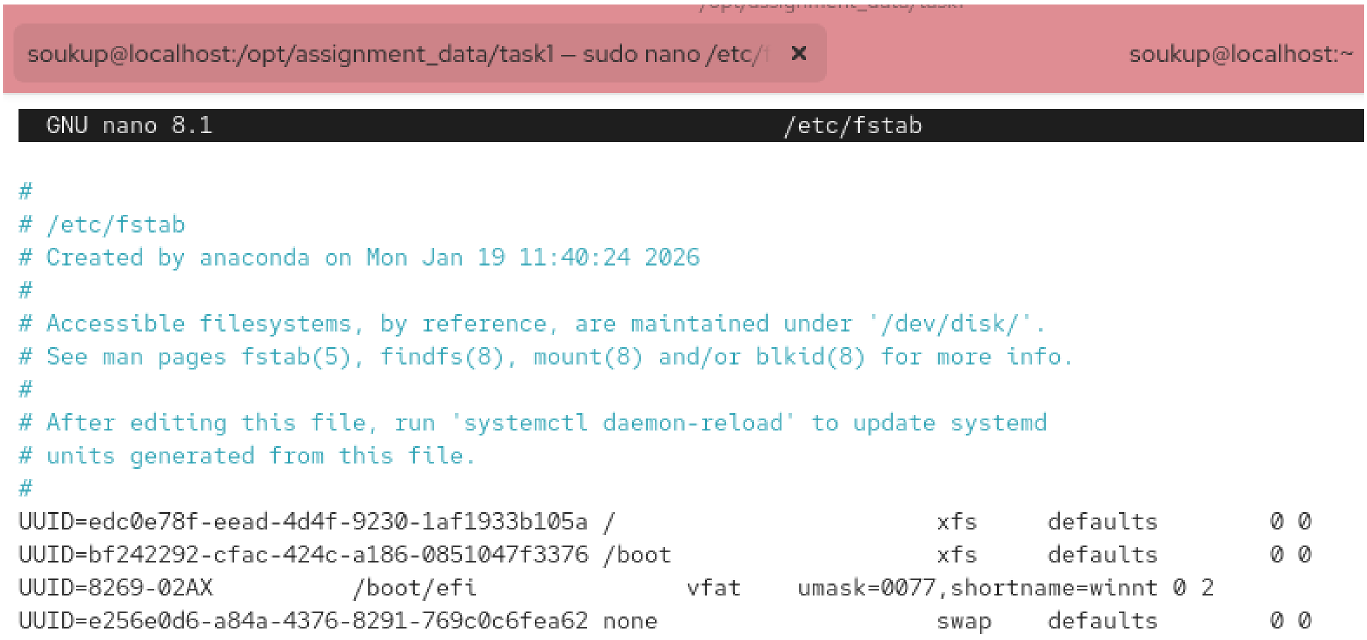
4. Only if the previous task was solved, modify the /boot stanza in the /etc/fstab file and use a wrong value.

We first made a copy of the `/etc/fstab` file. Then we changed one of the UUID letters to make it invalid, rebooted the PC which caused the boot to not finish. We again went to emergency mode, remounted and fixed the file (rewrote it with the copy of the original).

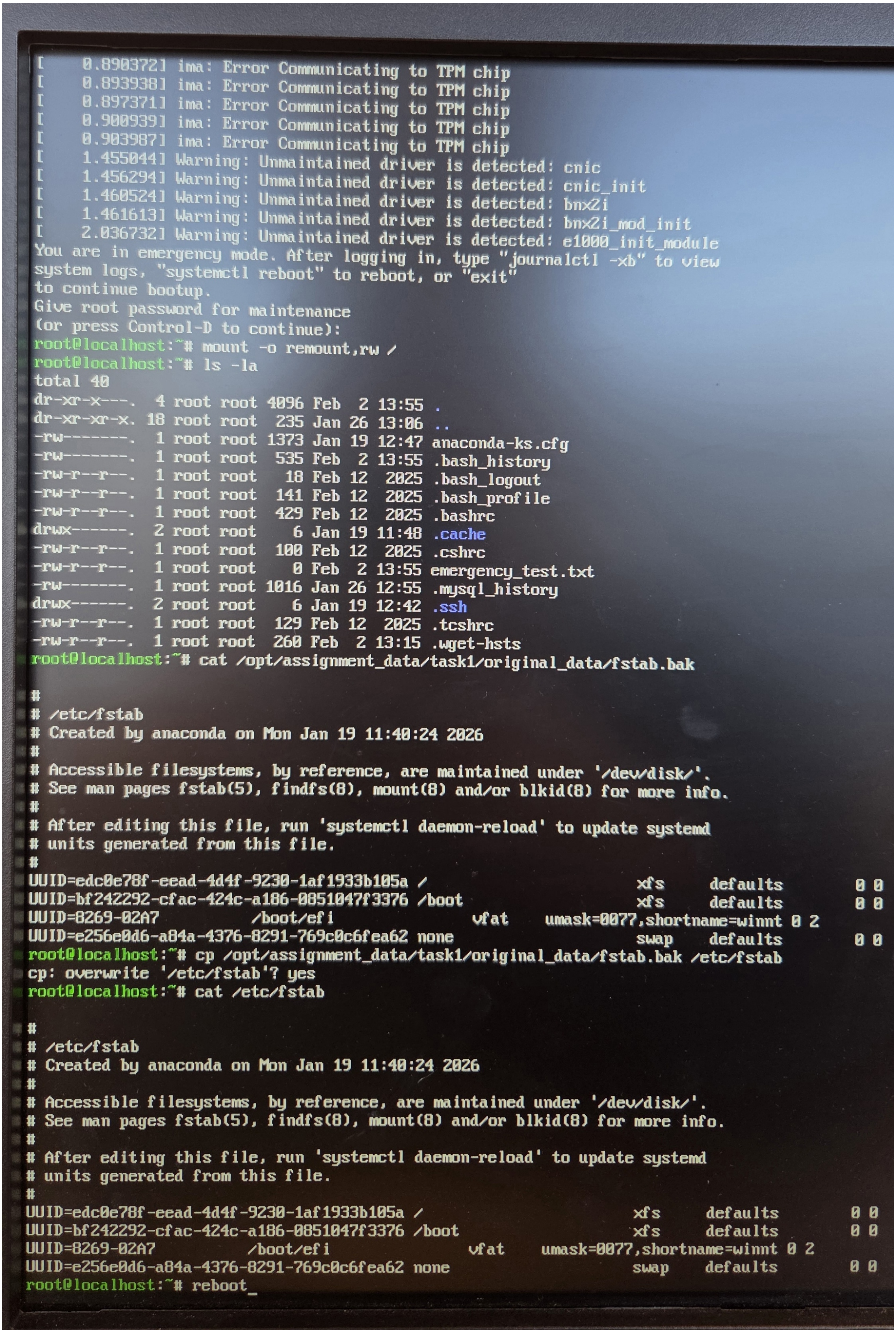
Screenshots:


soukup@localhost:/opt/assignment_data/task1 — sudo nano /etc/f ✕
soukup@localhost:~

```
  GNU nano 8.1                                           /etc/fstab

#
# /etc/fstab
# Created by anaconda on Mon Jan 19 11:40:24 2026
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=edc0e78f-eead-4d4f-9230-1af1933b105a /                       xfs     defaults        0 0
UUID=bf242292-cfac-424c-a186-0851047f3376 /boot                   xfs     defaults        0 0
UUID=8269-02A7          /boot/efi               vfat    umask=0077,shortname=winnt 0 2
UUID=e256e0d6-a84a-4376-8291-769c0c6fea62 none                    swap    defaults        0 0
```


soukup@localhost:/opt/assignment_data/task1 — sudo nano /etc/f ✕
soukup@localhost:~

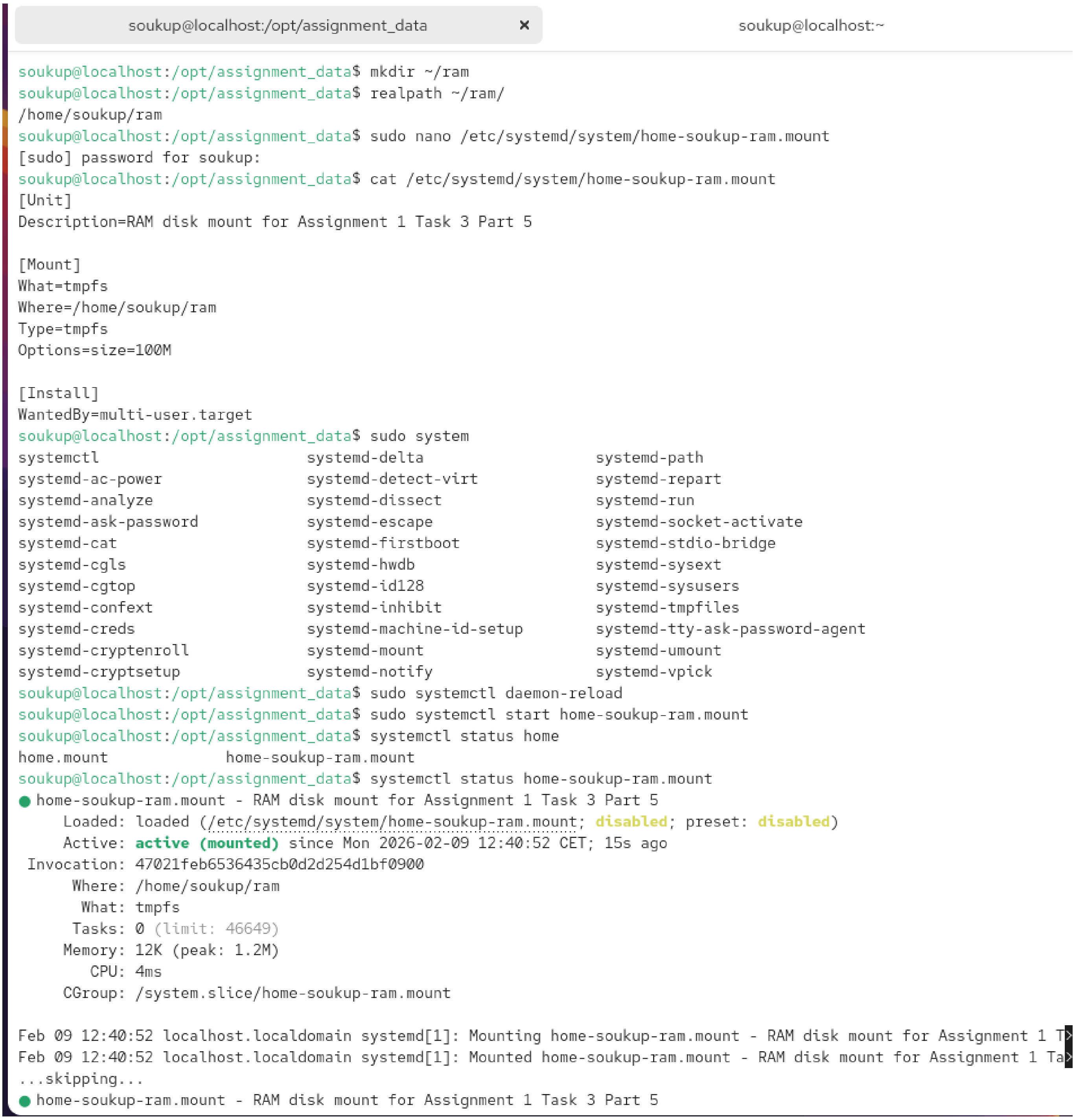
```
  GNU nano 8.1                                           /etc/fstab

#
# /etc/fstab
# Created by anaconda on Mon Jan 19 11:40:24 2026
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=edc0e78f-eead-4d4f-9230-1af1933b105a /                       xfs     defaults        0 0
UUID=bf242292-cfac-424c-a186-0851047f3376 /boot                   xfs     defaults        0 0
UUID=8269-02AX          /boot/efi               vfat    umask=0077,shortname=winnt 0 2
UUID=e256e0d6-a84a-4376-8291-769c0c6fea62 none                    swap    defaults        0 0
```



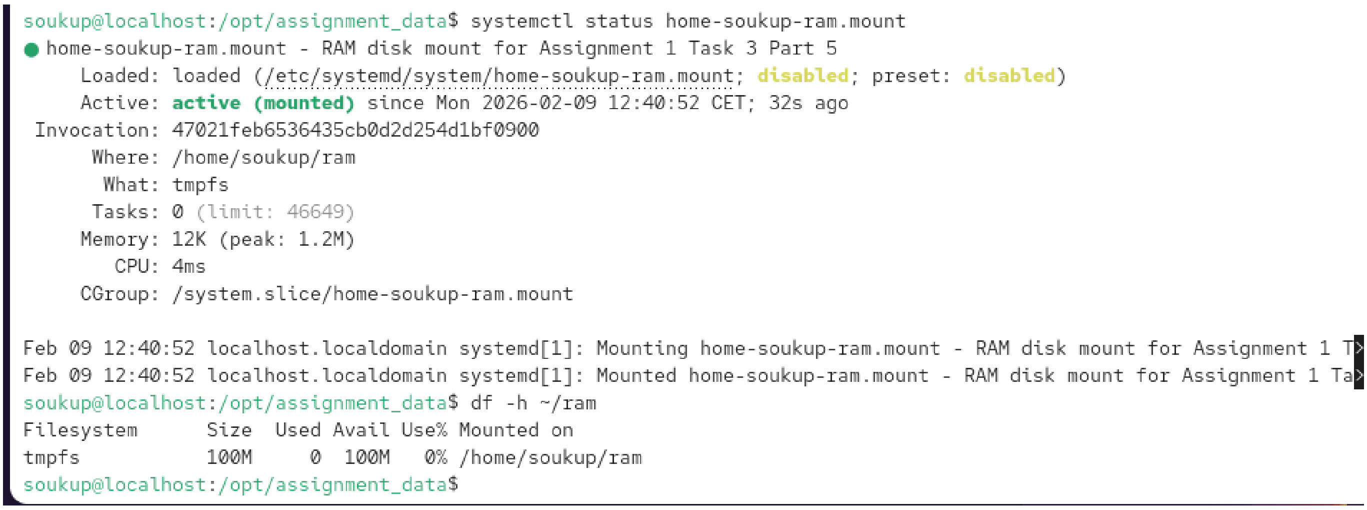
```
[    0.890372] ima: Error Communicating to TPM chip
[    0.893938] ima: Error Communicating to TPM chip
[    0.897371] ima: Error Communicating to TPM chip
[    0.900939] ima: Error Communicating to TPM chip
[    0.903987] ima: Error Communicating to TPM chip
[    1.455044] Warning: Unmaintained driver is detected: cnic
[    1.456294] Warning: Unmaintained driver is detected: cnic_init
[    1.460524] Warning: Unmaintained driver is detected: bnx2i
[    1.461613] Warning: Unmaintained driver is detected: bnx2i_mod_init
[    2.036732] Warning: Unmaintained driver is detected: e1000_init_module
You are in emergency mode. After logging in, type "journalctl -xb" to view
system logs, "systemctl reboot" to reboot, or "exit"
to continue bootup.
Give root password for maintenance
(or press Control-D to continue):
root@localhost:~# mount -o remount,rw /
root@localhost:~# ls -la
total 40
dr-xr-x---.  4 root root 4096 Feb  2 13:55 .
dr-xr-xr-x. 18 root root  235 Jan 26 13:06 ..
-rw-------.  1 root root 1373 Jan 19 12:47 anaconda-ks.cfg
-rw-------.  1 root root  535 Feb  2 13:55 .bash_history
-rw-r--r--.  1 root root   18 Feb 12  2025 .bash_logout
-rw-r--r--.  1 root root  141 Feb 12  2025 .bash_profile
-rw-r--r--.  1 root root  429 Feb 12  2025 .bashrc
drwx------.  2 root root    6 Jan 19 11:48 .cache
-rw-r--r--.  1 root root  100 Feb 12  2025 .cshrc
-rw-r--r--.  1 root root    0 Feb  2 13:55 emergency_test.txt
-rw-------.  1 root root 1016 Jan 26 12:55 .mysql_history
drwx------.  2 root root    6 Jan 19 12:42 .ssh
-rw-r--r--.  1 root root  129 Feb 12  2025 .tcshrc
-rw-r--r--.  1 root root  260 Feb  2 13:15 .wget-hsts
root@localhost:~# cat /opt/assignment_data/task1/original_data/fstab.bak

#
# /etc/fstab
# Created by anaconda on Mon Jan 19 11:40:24 2026
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=edc0e78f-eead-4d4f-9230-1af1933b105a /                       xfs     defaults        0 0
UUID=bf242292-cfac-424c-a186-0851047f3376 /boot                   xfs     defaults        0 0
UUID=8269-02A7          /boot/efi               vfat    umask=0077,shortname=winnt 0 2
UUID=e256e0d6-a84a-4376-8291-769c0c6fea62 none                    swap    defaults        0 0
root@localhost:~# cp /opt/assignment_data/task1/original_data/fstab.bak /etc/fstab
cp: overwrite '/etc/fstab'? yes
root@localhost:~# cat /etc/fstab

#
# /etc/fstab
# Created by anaconda on Mon Jan 19 11:40:24 2026
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=edc0e78f-eead-4d4f-9230-1af1933b105a /                       xfs     defaults        0 0
UUID=bf242292-cfac-424c-a186-0851047f3376 /boot                   xfs     defaults        0 0
UUID=8269-02A7          /boot/efi               vfat    umask=0077,shortname=winnt 0 2
UUID=e256e0d6-a84a-4376-8291-769c0c6fea62 none                    swap    defaults        0 0
root@localhost:~# reboot_
```

5. Create a directory ram below your home directory. Create a systemd mount unit that mounts a ramdisk on the ram directory.

We created a `/ram` directory, got the real path name to the directory, created a TOML configuration file to create a new mount. Then reloaded the `systemd` and started the mount with `systemctl start`. The process can be seen in the screenshots below:



```
soukup@localhost:/opt/assignment_data$ mkdir ~/ram
soukup@localhost:/opt/assignment_data$ realpath ~/ram/
/home/soukup/ram
soukup@localhost:/opt/assignment_data$ sudo nano /etc/systemd/system/home-soukup-ram.mount
[sudo] password for soukup:
soukup@localhost:/opt/assignment_data$ cat /etc/systemd/system/home-soukup-ram.mount
[Unit]
Description=RAM disk mount for Assignment 1 Task 3 Part 5

[Mount]
What=tmpfs
Where=/home/soukup/ram
Type=tmpfs
Options=size=100M

[Install]
WantedBy=multi-user.target
soukup@localhost:/opt/assignment_data$ sudo system
systemctl                       systemd-delta                   systemd-path
systemd-ac-power                systemd-detect-virt             systemd-repart
systemd-analyze                 systemd-dissect                 systemd-run
systemd-ask-password            systemd-escape                  systemd-socket-activate
systemd-cat                     systemd-firstboot               systemd-stdio-bridge
systemd-cgls                    systemd-hwdb                    systemd-sysext
systemd-cgtop                   systemd-id128                   systemd-sysusers
systemd-confext                 systemd-inhibit                 systemd-tmpfiles
systemd-creds                   systemd-machine-id-setup        systemd-tty-ask-password-agent
systemd-cryptenroll             systemd-mount                   systemd-umount
systemd-cryptsetup              systemd-notify                  systemd-vpick
soukup@localhost:/opt/assignment_data$ sudo systemctl daemon-reload
soukup@localhost:/opt/assignment_data$ sudo systemctl start home-soukup-ram.mount
soukup@localhost:/opt/assignment_data$ systemctl status home
home.mount             home-soukup-ram.mount
soukup@localhost:/opt/assignment_data$ systemctl status home-soukup-ram.mount
● home-soukup-ram.mount - RAM disk mount for Assignment 1 Task 3 Part 5
     Loaded: loaded (/etc/systemd/system/home-soukup-ram.mount; disabled; preset: disabled)
     Active: active (mounted) since Mon 2026-02-09 12:40:52 CET; 15s ago
 Invocation: 47021feb6536435cb0d2d254d1bf0900
      Where: /home/soukup/ram
       What: tmpfs
      Tasks: 0 (limit: 46649)
     Memory: 12K (peak: 1.2M)
        CPU: 4ms
     CGroup: /system.slice/home-soukup-ram.mount

Feb 09 12:40:52 localhost.localdomain systemd[1]: Mounting home-soukup-ram.mount - RAM disk mount for Assignment 1 T>
Feb 09 12:40:52 localhost.localdomain systemd[1]: Mounted home-soukup-ram.mount - RAM disk mount for Assignment 1 Ta>
...skipping...
● home-soukup-ram.mount - RAM disk mount for Assignment 1 Task 3 Part 5
```



```
soukup@localhost:/opt/assignment_data$ systemctl status home-soukup-ram.mount
● home-soukup-ram.mount - RAM disk mount for Assignment 1 Task 3 Part 5
     Loaded: loaded (/etc/systemd/system/home-soukup-ram.mount; disabled; preset: disabled)
     Active: active (mounted) since Mon 2026-02-09 12:40:52 CET; 32s ago
 Invocation: 47021feb6536435cb0d2d254d1bf0900
      Where: /home/soukup/ram
       What: tmpfs
      Tasks: 0 (limit: 46649)
     Memory: 12K (peak: 1.2M)
        CPU: 4ms
     CGroup: /system.slice/home-soukup-ram.mount

Feb 09 12:40:52 localhost.localdomain systemd[1]: Mounting home-soukup-ram.mount - RAM disk mount for Assignment 1 T>
Feb 09 12:40:52 localhost.localdomain systemd[1]: Mounted home-soukup-ram.mount - RAM disk mount for Assignment 1 Ta>
soukup@localhost:/opt/assignment_data$ df -h ~/ram
Filesystem      Size  Used Avail Use% Mounted on
tmpfs           100M     0  100M   0% /home/soukup/ram
soukup@localhost:/opt/assignment_data$
```