DV1466 UNIX and Linux
An Overview and Introduction.

Assignment 1310.
(1.5 ECTS, A-F)

Developed by Diego Navarro, Tek.Lic.

September 5th, 2024.

Deadline 1:	September 13th, 2024 18:00.
Deadline 2:	November 22nd, 2024 18:00.
Deadline 3:	January 10th, 2025 18:00.
Submission:	PDF file uploaded to Canvas.

---

1 Description

This assignment is based upon the theory and concepts presented during the lecture "*The Linux OS and its file hierarchy*". For this assignment, students are tasked with solving a set of problems, and documenting their respective solutions, based exclusively on information retrieved from academic sources.

The assignment expands over the aforementioned lecture by introducing advanced concepts of the GNU/Linux operative system. It also assesses the student capability to retrieve information from academic sources, to analyze it, and to propose and document solutions based on their theoretical findings.

2 Instructions

The following assignment presents a set of five problems that are related to advanced concepts of the GNU/Linux operative system. Students are tasked with proposing a solution to each of these problems, and presenting them in a written document.

The proposed solutions must be based on, and supported by, the information published in academic sources. This means that every claim, definition, or fact that is offered as an argument for the solution, must be backed up by an academic source. Therefore, students must perform an information retrieval activity to gather that information using, exclusively, the sources available through the BTH library databases. The O'Reilly and Springer databases are strongly recommended for the development of this assignment.

To verify the gathered information and the sources used to obtained it, students must also reference each source that was used in the development of each solution. The style chosen for referencing these sources is the ACM citation and reference formats. Students must provide detailed information about the references (e.g. chapter/subchapter number, page number, paragraph number, etc.), and they must include the source URL.

Lastly, students are free to address any number of problems, in accordance to their desired grade (see Section 3 for more details on the assignment assessment).

Example

The following is an example that illustrates the desired outcome for the problems presented in this assignment, in terms of presentation, argumentation, references, and length:

Problem

You have a system running the Windows operative system (OS). You want to prepare this system for dual-booting (i.e. running two different operative systems in the same machine), by installing GNU/Linux alongside Windows. For this, you have used three different hard drives. One drive stores the Windows OS files, another stores the Linux OS files, and the third one is used as a storage drive that may be accessed by both OS. When you run Windows, everything works correctly, and you can access the files in your storage drive. When you run GNU/Linux, you can see that the storage drive is being recognized by the OS, that it can be mounted correctly (when forced), and that you have full permission to access the drive and its content. Despite these, you cannot access any files in the storage drive. Why does this problem occur and how can it be addressed?.

Proposed Solution

This problem is most probably related to the file system used by the storage drive. Windows uses the NT File System (NTFS) as the default file system format for its hard drives [1]. Since Windows was already running in the machine, we can assume that the drives were formatted using NTFS. However, NTFS support in GNU/Linux is rather limited. The read-only support is considered to be reasonably reliable (reason why we may see the files in the storage drive), but read/write support is much less stable, and it could corrupt the NTFS partition, depending on the version of the Linux kernel used [2] (therefore limiting the access to files in the storage drive).

A potential solution for this is the **File System in User Space (FUSE)** feature, which is a user-space daemon that acts as an intermediary between the data source (i.e. the storage drive) and the Linux kernel [3]. Commonly, file systems are implemented within the kernel-space due to performance reasons [4]. However, since FUSE is implemented in user-space, we could load (or develop) an NTFS-like file system outside the kernel, to safely access the files within the storage drive, and bypassing the compatibility limitation between the Linux kernel and NTFS.

References

[1] Mike Halsey and Andrew Bettany. 2015. An introduction to the Windows File Systems (sec. 1, par. 5) in *Windows File System Troubleshooting (1st. ed.)*. Apress, U.S.A. Retrieved September 5th, 2024, from `https://learning.oreilly.com/library/view/windows-file-system/9781484210161/A334509_1_En_1_Chapter.html`.

[2] Roderick W. Smith. 2005. File and Filesystem Compatibility (sec.1, par.4) in *Linux in a Windows World (1st. ed.)*. O'Reilly Media, Inc. Revised September 5th, 2024, from `https://learning.oreilly.com/library/view/linux-in-a/0596007582/apbs04.html`.

[3] Brian Ward. 2021. Network File Transfer and Sharing (Sec. 12.7, p.2) in *How Linux Works (3rd. ed.)*. No Starch Press, U.S.A. Retrieved September 5th, 2024, from `https://learning.oreilly.com/library/view/how-linux-works/9781098128913/c12.xhtml#h1-500402c12-0007`.

[4] Michael Hausenblas. 2022. File Systems (sec. 1, par. 7) in *Learning Modern Linux (1st. ed.)*. O'Reilly Media, Inc. Revised September 5th, 2024, from `https://learning.oreilly.com/library/view/learning-modern-linux/9781098108939/ch05.html#fs-basics`.

3 Submission and Assessment

The development of this assignment is individual and, despite the desired grade, every submission must:

- Present a single written document in .pdf format.
- Include the student name and intended grade in the written document.
- Present the proposed solutions to the addressed problems, and their respective argumentation, in the written document.
- Clearly link a solution to the respective problem it addresses, in the written document.
- Provide all the necessary references to support the proposed solutions, in the written document. Also, references must be complete (i.e. include source details and URL).
- Use only those sources provided through the BTH Library databases.

Each problem that is solved correctly awards students one point. The final grade for this assignment is determined by the number of points achieved by the student, based on the grading criteria shown in Table 1.

Table 1: Grading criteria for Assignment 1310.

Points	**Grade**
1	E
2	D
3	C
4	B
5	A

If a solution is incorrect or incomplete (i.e. it lacks argumentation or there are missing details from references), no point will be awarded for it. A submission will be graded Fx only when just one of the provided answers is correct, but it lacks proper argumentation or referencing. Also, in accordance with BTH regulations, submissions graded with Fx can only be complemented up to E grade.

Submissions will only be accepted through Canvas and within deadline limits. Lastly, plagiarism is prohibited and any found case of plagiarism will have disciplinary actions taken against the students involved.

Assignment 1310.

Problem 1 (1 Point).

You own a GNU/Linux server that runs a user authentication system. The authentication system verifies user information and grants access to an external e-mail service. Your user authentication system uses single-factor authentication (i.e. username + password combination). The company providing the e-mail service wants you to update the authentication system to offer multifactor authentication (i.e. fingerprint, QR codes, facial recognition, SMS tokens, etc.). However, you do not have the resources (neither money nor infrastructure) to develop, maintain, and interconnect multiple custom authentication systems independently. How may the GNU/Linux OS running in your server help you address this issue?

Problem 2 (1 Point).

You are the system administrator of a computing server, running GNU/Linux, in a web development company. The company is currently executing three different projects in parallel. Each project has a dedicated team working on its development, and all of them require access to processing power and memory allocation from the computing server, simultaneously. There has been complaints that one team is starving the others by monopolizing the resources from the computing server. How can you monitor the computing requirements of each team and configure GNU/Linux to ensure that every team has fair access to the resources they need?

Problem 3 (1 Point).

A video game company (definitively not CD Project Red) is about to release a major update for one of its video games, that was released 9 years ago (definitively not The Witcher III Next-Gen Update). The update improves upon some of its core systems (i.e. rendering, physics, I/O, etc.), and it is planned to be offered for free to all customers that already own the game. The company uses GNU/Linux for the development of the game, the update, and its services. Before its release, the company would like to test if the minimum hardware requirements listed for the original version of the game may be capable of running appropriately the game update, since the updated core systems are more computationally demanding. However, all testing computers available in the game company are significantly superior when compared with the minimum hardware requirements listed for the original version. How can the company

tests if the game update runs appropriately with the aforementioned hardware requirements without having to purchase new hardware or physically modifying their existing computers?

Problem 4 (1 Point).

A bank has recently established a new banking system that is composed of three elements: a web server that hosts the banking interface, an application server that computes the transaction logic, and a database server that hosts customer information. All these servers run GNU/Linux. After initial tests, the bank's IT security department has identified that, when initializing a transaction in the application server, the system grants employees access to all customer information and transaction records in the database server. This is considered a security risk since employees can, accidentally or intentionally, modify customer records or facilitate data tampering. After seeing this, the bank wants to update the banking system to restrict employees' access to the information stored in the database server, allowing them to browse only the information that relates with their respective position and intended transaction. However, the bank doesn't have the resources to make any changes in the infrastructure of the new banking system, and they cannot modify the existing databases. How can the IT department address this issue using GNU/Linux?

Problem 5 (1 Point).

A company uses a server to monitor and manage the network traffic generated by its employees. This server runs GNU/Linux. The company has discovered that some services employees use while working (such as streaming music or videos, or reviewing social media) can sometimes saturate the company's bandwidth, negatively affecting other services that are considered priorities (such as video-conferencing or transferring large files). The company doesn't have the resources to improve their bandwidth or network speed. Also, the company doesn't want to ban the access to services employees commonly use. How can this issue be addressed from the server perspective?