OBL4-OS

April 5, 2019

This is a mandatory assignment. Use resources from the course to answer the following questions. **Take care to follow the numbering structure of the assignment in your submission**. Some questions may require a little bit of web searching. Some questions require you to have access to a Linux machine, for example running natively or virtually on your own PC, or by connecting to `gremlin.stud.iie.ntnu.no` over SSH (Secure Shell). Working in groups is **permitted**, but submissions must be **individual**.

1 File systems

1. Name two factors that are important in the design of a file system.
2. Name some examples of file metadata.

2 Files and directories

1. Consider a Fast File System (FFS) like Linux's ext4.
   (a) Explain the difference between a hard link and a soft link in this file system. What is the length of the content of a soft link file?
   (b) What is the minimum number of references (hard links) for any given folder?
   (c) Consider a folder `/tmp/myfolder` containing 5 subfolders. How many references (hard links) does it have? Try it yourself on a Linux system and include the output. Use `ls -ld /tmp/myfolder` to view the reference count (hint, it's the second column in the output).
   (d) Explain how spatial locality is acheived in a FFS.
2. NTFS - Flexible tree with extents
   (a) Explain the differences and use of *resident* versus *non-resident* attributes in NTFS.
   (b) Discuss the benefits of NTFS-style extents in relation to blocks used by FAT or FFS.
3. Explain how copy-on-write (COW) helps guard against data corruption.

3 Security

1. Authentication
   (a) Why is it important to hash passwords with a unique salt, even if the salt can be publicly known?
   (b) Explain how a user can use a program to update the password database, while at the same time does not have read or write permissions to the password database file itself. What are the caveats of this?
2. Software vulnerabilities

(a) Describe the problem with the well-known `gets()` library call. Name another library call that is safe to use that accomplishes the same thing.

(b) Explain why a microkernel is statistically more secure than a monolithic kernel.