Lab 1: Introduction to Xubuntu

50.020 Security

Hand-out: January 28
Hand-in: February 4, 11am

1 Objectives

- To create a new user account
- To practice the usage of Ubuntu shell commands
- To understand and modify a simple Python script
- If you took 50.012 Networks, you only need to look at 6.1 onwards

Note: Don't be discouraged by the length of this handout. It is intended to give you a brief summary of important command line tools in Linux. Read through the command descriptions, and try out the different commands to familiarize yourself. Section 7.1 contains the main practical part of this exercise.

2 First time login

After you boot from USB, you will be presented with a login with name `student`. Use the following password to login: `password`.

3 Create new user account in Xubuntu

Use the `start menu/settings/users and accounts` to create a new user account with Administrative rights. The initial user account is common to all and, therefore, has the same common password. You are strongly encouraged to create a new user account with your own password. Log out, and back in as your new user.

4 How to open shell in a terminal

In Linux, the command line (or terminal) is usually used for many tasks. A large number of useful commands exist, that enable the user to quickly perform simple and complex tasks.

We will start by opening a terminal (similar to `cmd` in Windows). For this class, we will use `xfce4-terminal`. You can open it by pressing Alt + F2 and typing `xfce4-terminal`. Or you can use the keyboard shortcut by pressing

- Ctrl + Alt + T

Opening the terminal will automatically start a *shell*, and interactive environment. By default, we use `bash`, which supports a command history (up/down) keys, and auto-completion of commands and many options via tab key.

5 Common commands

5.1 Find the usage of commands

If you don't know how to use the shell commands, you can use command `man` (which stands for MANual)to show the usage of them.

```
$ man <command name>
```

If you want to quite while reading the manual of the command, press q

5.2 Change the directory

The `cd` command allows you to change your current directory to any accessible directory on the system.

```
$ cd <directory>
```

Using the command `cd` without applying any parameter will bring you back to your home directory.

5.3 Listing the contents of a directory

The ls command can be used to view the contents of the current directory.

```
$ ls [option] <filename>
```

To know more about the field `[option]`, use the command `man ls` . It is frequently used with the options `-al` to list hidden files as well (their name start with .), and provide more information. Type `cd /`, `ls` and `ls -al` in the command line, what is the result?

5.4 locate, less

Command `locate` can be used to search files in Xubuntu. For example, if you want to find all text files, you can use:

```
$ locate *.txt
```

Command `less` can be used to display the content of file. Example:

```
$ less myfile.txt
```

Another way to display file is to use command `cat`, example:

```
$ cat myfile.txt
```

5.5 grep

Command `grep` can be used to search for strings or patterns in text files. For example, if you want to find all occurences of "foo" in file A.txt

```
$ grep foo A.txt
```

Command `grep` can be also be used to filter output of other commands using *piping* symbol "|". The following will locate all python scripts on the current machine, but display only the ones in the home directory. Depending on the content of your home directory, the output might be empty.

```
$ locate *.py | grep home
```

5.6 mkdir

`mkdir` can be used to create directories. To create `dir1` in the current location, type:

```
 $ mkdir dir1
```

6 File management Ubuntu commands

6.1 File access control in Linux

Linux uses per-file access control. There are three different types of access to a file: reading (r), writing (w), and execution (x). These three types of access can be set for the file owner, a group of users, and everyone else. As a result, the access rights for each of those three classes of users can be described in a short string, e.g "r-x" for read and execution rights. The rights of all three types of users together yield a longer string, e.g. "rw-r–r–" for a file which can be read and written by the owner, but only read by everyone else.

Try using `ls -al` and `cd` to look around on your system, and see who has what kind of access control to which file. The content of `/etc/` could be especially interesting.

6.2 Copying files and directories

To copy files, you can use the `mv` or `cp` commands. The command line

```
$ mv file1 file2
```

allows one to move `file1` to `file2`. After the move, `file1` will no longer exist. Command `cp` on the other hand copies one file to another.

```
$ cp file1 file2
```

Note that if `file2` does not exist, it will be created; however if it does exist, it will be overwritten. There is NO undo command in Linux. For copying directories, you can use the `cp` and `mv` commands just like you use them with files. If you want to use `cp` copy directory, you need to use `-r`

```
$ cp -r dir1 dir2
```

6.3 Deleting files and directories

The `rm` command is used for removing files. To remove a file:

```
$ rm file1
```

There are two commands you can use for removing directories. If the directory is empty, you can use `rmdir`:

```
$ rmdir dir1
```

Or you can use `rm` with the `-r` switch to recursively delete.

```
$ rm -r dir1
```

6.4 chmod

We introduced Linux file access control in Section 6.1. Now we discuss how to change to rights to files. Please note that you have to be either root (or use sudo, see below) or the owner of a file to change the file's rights.

`chmod` is an abbreviation of change mode, and can be used to change the file permissions.

```
chmod <people><+/-><permissions>
```

Example: `chmod o-w file1` (deny others from editing the file)
Example: `chmod u+rwx file1` (give the owner full control)
Exmaple: `chmod +rwx file1` (give everyone full control)
Example: `chmod +x file1` (allow anyone to execute the file. Allows direct execution of python scripts if a *shebang* is present [#!/usr/bin/env python])

6.5 Sudo and root

Your normal user does not have administrator rights (e.g. to install software system wide). In Linux, the administrator account is called "root" or "superuser". Your system is set up to allow you to perform commands as root by using `sudo` in front of the command. Example: `less /etc/shadow` as normal user fails :

```
$ less /etc/shadow
/etc/shadow: Permission denied
```

The reason for this is your missing read rights on the file:

```
$ ls -al /etc/shadow
-rw-r----- 1 root shadow 1166 Aug 11 13:22 /etc/shadow
```

But you can see the content using `sudo`:

```
$ sudo less /etc/shadow
```

You will be asked to enter a password, enter the password you chose earlier.

7 Python warmup and Shift Cipher

7.1 Part II: SwapEvenCase

We provide the `ex1.py` script skeleton on eDimension You are required to modify this code to convert every even position letters in `sherlock.txt` to upper case characters. Store these changes to a new output file. After modifying this code, you can run in terminal and check the content of output file. Command to run:

```
$ ./ex1.py -i inputfile -o outputfile
```

7.2 Part II: Shift Cipher

- Based on ex1.py, write a Python script that can be run from the command prompt or shell as the following:

  ```
  $ ./shiftcipher.py -i [input filename] -o [output filename] -k [key]
  -m [mode]
  ```

- The key must be between 0 and 255, otherwise an error message is produced. The key specifies how many characters the original letter should be shifted. For encryption, the key is added to the original letter. For decryption, the key is subtracted from the original letter. For example:

  ```
  'A'=65, k=11
  E_k('A')=65+11=76='L'
  D_k('L')=65
  ```

- The mode must either be 'd' (Decryption), 'e' (Encryption), or their capital letter, otherwise an error message is produced.
- Use the text file given 'sherlock.txt' and encrypt it with some key, and then decrypt it. Check whether the text remain the same after the process of encryption and decryption.
- Consider the full range of 256 ASCII values in your encryption and decryption implementation.