Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences

Linux overview

Foundation Course @H-BRS, March 2024
By: Vedika Chauhan(MAS)

What is Linux?

Linux as a kernel

People often say that Linux is an operating system; it’s actually a kernel.

When we talk about Linux as an operating system, we’re typically referring to one of its distributions (for example Ubuntu).

Linux file system

- Linux organizes files in a hierarchical tree, where relationships are thought of in terms of parents and children.
- Directories can contain other directories as well as regular files, which are the "leaves" of the tree
- In Linux, everything is a file, including directories!

```
$ tree
```

Relative path vs absolute path

- Absolute path: complete details needed to locate a file or directory
  - Example : /home/user/Documents/path/to/some/file
- Relative path: points to a specific location in relation to the current location
  - Example: ./path/to/some/file (assuming you’re located in /home/user/Documents)

Some good to know symbols when referring to locations in a filesystem tree include:

1. “~”: user’s home directory
2. “..” directory immediately above the current one
3. “.” current directory

Traversing file system

For traversing a file system, here are some of the commands you should be familiar with:

- `ls` lists directory contents.
- `cd` stands for "change directory."
- pwd (print working directory) outputs the absolute path of the current working directory.

Text editors

The two popular text editors are nano and vi (or vim). There's others too, like Emacs… In principle, all text editors work the same, it's a matter of preference which one you'd use.

```
$ nano [filename]
$ vi [filename]
```

Let's try to edit file!

Viewing files

cat

- `cat` simply prints the content of the file to standard display i.e. your screen. It cannot be simpler than this, can it?
- `cat` becomes powerful when combined with options, for that we can use our best friend `-h`.

What is “-h”? Whenever you’re stuck, most CLI commands come with “help” on how to use them. For that, append “-h” to a command name. For example:

```
$ cat -h
```

head

head command is another way of viewing text file but with a slight difference: it displays the first 10 lines of a text file by default.

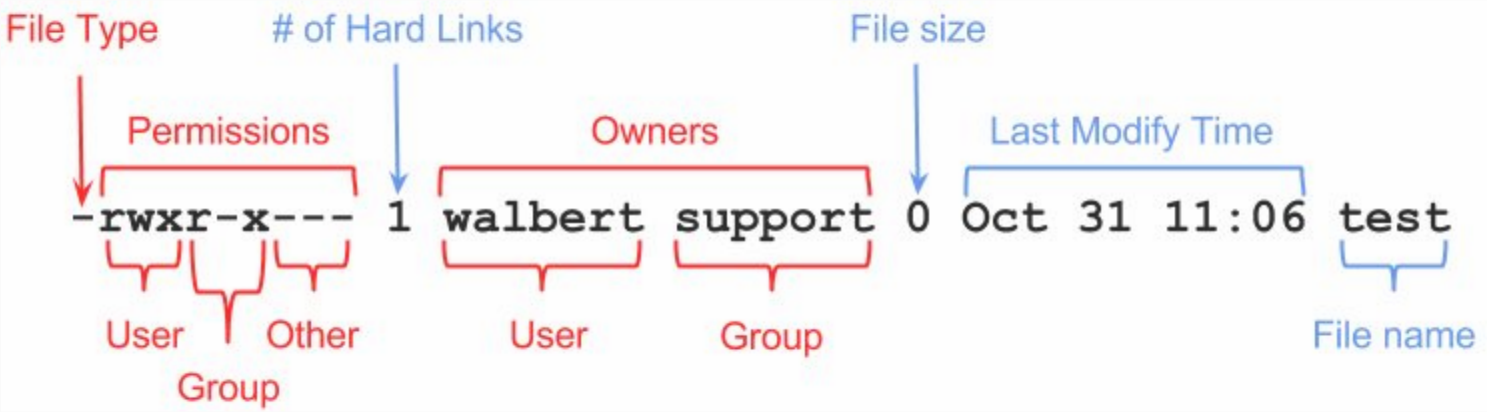
tail

`tail` is opposite of head i.e it will display last 10 lines of a text file by default.

File management

File permissions

In Linux, access to data objects like files and directories, is handled via permissions. Typical permission settings for a file/directory can for example look like this:



```
-rwxr-x--- 1 walbert support 0 Oct 31 11:06 test
```

File permissions

In simple term chown is used to change the ownership of a file while chmod is used for changing the file mode bits.

- chown defines who owns the file
- chmod defines who can do what with the file

Their usage is pretty similar:

```
$ chown [user] [file]
$ chmod [mode] [file]
```

Administrative access

Administrative access is needed to execute tasks which require elevated privileges. Those tasks usually involve system-wide changes that can affect other users or the stability and security of the system.

There are two ways administrative (or privileged, or root) access can be given:

1. sudo: grants administrative access to this particular command
2. su: switches to a root user account, from where every issued command has elevated privileges

CRUD operations

CRUD operations are operations to copy (`cp`), move/rename (`mv`), and delete (`rm`) files.

- “cp” command is used to copy files and directories
- “mv” command is used to move or rename files and directories
- “rm” command is used to remove files and directories

General structure of commands:

```
$ cp file1 file2
$ mv file1 file2
$ rm -r dir1
```

Filtering input

grep can be used to search for specific patterns within files.

- Open `welcome.txt`
- Type either $ grep "string" [filename] or $ [filename] grep "string"

The grep command can be used together with pipes for getting distinct output. For example:

```
$ dpkg -L | grep [package-name]
$ ifconfig | grep -B 4 eth
```

I/O redirection and pipes

Input and output in the Linux environment is distributed across three streams. These streams are:

- standard input (stdin)
- standard output (stdout)
- standard error (stderr)

The streams are also numbered:

- stdin (0)
- stdout (1)
- stderr (2)

I/O redirection and pipes cont.

- Commands with a single bracket overwrite the destination’s existing contents.
- Commands with a double bracket append to the destination’s existing contents

```
$ echo “Write to a new file” > data.txt
$ echo “Append contents to an existing file” >> data.txt
```

Bonus topics

Package management

What is package management? Package management refers to a set of tools and processes for managing software packages (and their dependencies), including their installation, upgrade, configuration, and removal.

Distributions like Ubuntu (and other Debian-based ones) use APT.

- APT is a package management system
- It uses dpkg under-the-hood

```
$ sudo apt install nginx
$ sudo apt purge apache2
$ apt list --installed
```

Processes

- One can display a running process, change their priorities level, kill them, etc.
- `ps` lists running processes
- `top` does the same, but it displays resource usage in real-time

General structure :

```
$ ps
$ top
```

Managing processes

Killing a process means stopping it:

```
$ kill [pid]
```

Or you can be nice about it…

```
$ nice -n [value] [process name]
```

Networking

Here are some commands useful for troubleshooting and managing network connections:

```
$ ping [ip-address]
$ ipconfig
$ curl https://api.ipify.org
$ wget [location]
```

Any questions?