Working with Linux File System

Diptesh Kanojia
IITB-Monash Research Academy

e-Yantra Team
ERTS Lab, IIT Bombay

IIT Bombay
March 29, 2021

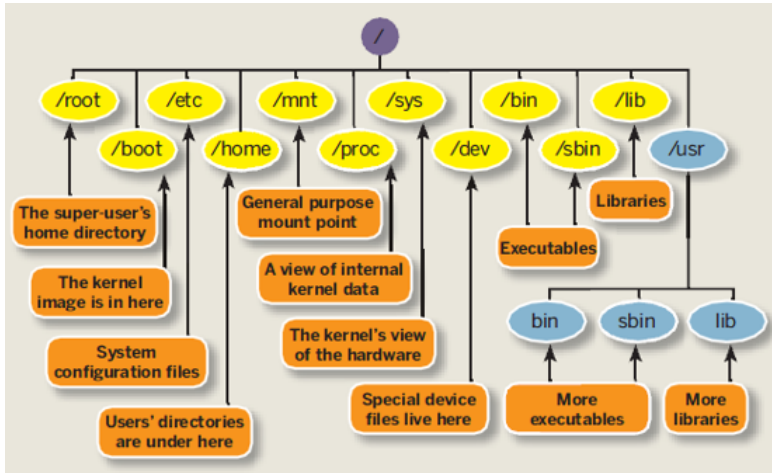# Agenda for Discussion

Hierarchy of Linux File System

Figure 1: Hierarchy of Linux File System

What can we do with the file system ?

- How to move around ?

What can we do with the file system ?

- How to move around ?
- How to see which files and directories we have ?

What can we do with the file system ?

- How to move around ?
- How to see which files and directories we have ?
- How to create new files and directories ?

What can we do with the file system ?

- How to move around ?
- How to see which files and directories we have ?
- How to create new files and directories ?
- How to create, copy, and delete files and directories ?

What can we do with the file system ?

- How to move around ?
- How to see which files and directories we have ?
- How to create new files and directories ?
- How to create, copy, and delete files and directories ?
- How to edit files ?

Copy files

Let's say you have downloaded `code.py`, inside the directory `~/Downloads`. You have a new directory called `project` inside `~/Desktop/`, so the path to the project directory would be `~/Desktop/project`.

How to copy `code.py` to your project folder ?

```
cp ~/Downloads/code.py ~/Desktop/project/
```

Use `ls` command to see what's inside `~/Desktop/project/`

```
code.py
```

Copy directories

Let's say you have a downloaded a folder `data`, inside the directory `~/Downloads`, which contains the files for processing. And you want to move it to the directory called `project` inside `~/Desktop/`. The path to the project directory would be `~/Desktop/project`.

How to copy the folder`data` to your project folder ?

```
cp -r ~/Downloads/data/ ~/Desktop/project/
```

Use `ls` command to see what's inside `~/Desktop/project/`

```
code.py data
```

Deleting file

Let's say after you have copied `code.py`, which was inside the directory `~/Downloads`, to the directory called `project` inside `~/Desktop/`. The path to the project directory would be `~/Desktop/project`.

How to delete `code.py` inside the folder `~/Downloads/` ?

```
rm ~/Downloads/code.py
```

Use `ls` command to whether its deleted or not from `~/Downloads/`

```
You should not see code.py
```

Deleting folder

Let's say after you have copied the folder `data`, which was inside the directory `~/Downloads`, to the directory called `project` inside `~/Desktop/`. The path to the project directory would be `~/Desktop/project`.

How to delete the folder `data` inside the folder `~/Downloads/` ?

```
rm -r ~/Downloads/code
```

Use `ls` command to whether its deleted or not from `~/Downloads/`

```
You should not see the folder data
```

Moving file

Copying and deleting is a lot of work, to minimise work you can directly move the downloaded files and folders using the command `mv`.
Let's say you want to move the file `code.py`, which was inside the directory `~/Downloads`, to the directory called `project` inside `~/Desktop/`. The path to the project directory would be `~/Desktop/project`.

How to move the file `code.py` to your project folder ?

```
mv ~/Downloads/code.py ~/Desktop/project
```

Use `ls` command to whether its moved or not ?

```
code.py
```

Moving folder

Let's say you want to move the folder data, which was inside the
directory ~/Downloads, to the directory called project inside
~/Desktop/. The path to the project directory would be
~/Desktop/project.

How to move the folder data to your project folder ?

```
mv ~/Downloads/data ~/Desktop/project
```

Use ls command to whether its moved or not ?

```
code.py data
```

Nano, Vi/VIM or Emacs

Several choices which are available:

Nano, Vi/VIM or Emacs

Several choices which are available:

- Nano: Basic terminal based editor, for which control commands are diplayed at the bottom of the screen.

Nano, Vi/VIM or Emacs

Several choices which are available:

- Nano: Basic terminal based editor, for which control commands are displayed at the bottom of the screen.
- Vi / VIM: Standard UNIX editor, which can be extended to act like an IDE.

Nano, Vi/VIM or Emacs

Several choices which are available:

- Nano: Basic terminal based editor, for which control commands are displayed at the bottom of the screen.
- Vi / VIM: Standard UNIX editor, which can be extended to act like an IDE.
- Emacs: Another standard editor with longest history of development. It is also an extensible editor which can act like an IDE, mail client, calendar planning, and also to play Tetris.

References

- Working with files and directories
- LearnShell.org
- TLDR pages for Shell commands
- e-Yantra Homepage

Thank You!

Author: Diptesh Kanojia

Contributor: Prashant Sharma

Post your queries at: resources@e-yantra.org