System Administration

DAY 1 CONTENT

- Linux History
- Free/Open source Software and Licenses
- What is Linux?
- Distros
- Why Linux?
- Installation
- Basic Commands
- File & Directory Manipulation
- File globing

LINUX HISTORY

- Unix first version created in Bell Labs in 1969
  - Multics(Assembly) – UNICS (B) – UNIX (C)

- Bell Labs licensed their OS with NDA “Non-Distribution Agreement” to sell their OS
- Unix flavors
  - IBM → AIX
  - Sun → Solaris
  - Apple → MAC

- Linus Torvalds in 1991
  - Created Linux kernel based on Unix OS
  - Relicensed the project under the *GNU General Public License*, declaring it to be open-source software.

FOSS

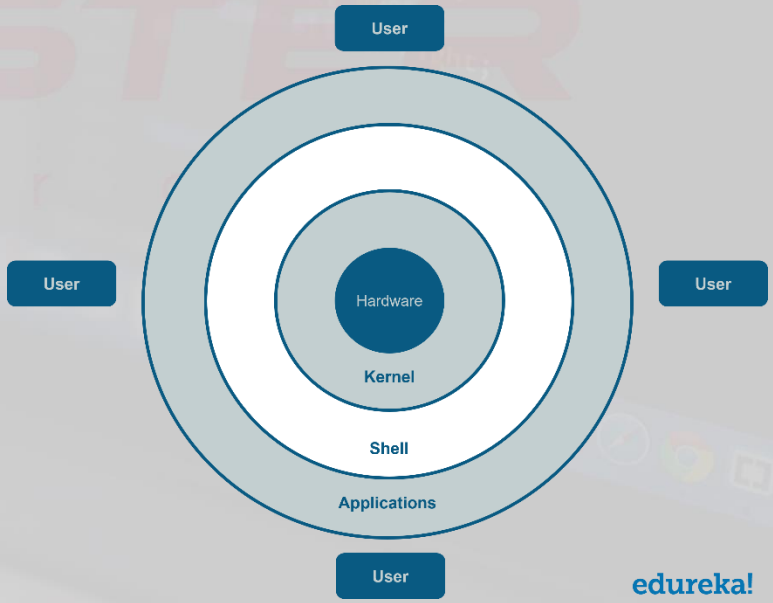
Free **Open-Source** Software license

An open-source license is a type of licenses for computer software and other products that allows the source code, blueprint or design to be used, modified and/or shared under defined terms and conditions.

- Free/Open Source Software (FOSS) provides many freedoms, including the ability to:
  - View the source code used to compile programs
  - Make modifications
  - Distribute these modifications.

What is Linux?

- **LINUX** is an open-source and community-developed operating system for computers, servers, mainframes, mobile devices, and embedded devices. It has support on almost every major computer platform
- The kernel is a program at the heart of the Linux operating system that takes care of fundamental stuff, like letting hardware communicate with software.
  - Storage mgmt.
  - Memory mgmt.
  - CPU mgmt.
  - Jobs scheduling



Distributions

Why Linux?

- Free
- Open Source
- Secure
- Stability and Performance

Installation

Linux Components

- ***Kernel***
  - *core of the operating system.*
  - *Contains components like device drivers.*
  - *It loads into RAM when the machine boots and stays resident in RAM until the machine powers off.*
- ***Shell***
  - *Provides an interface by which the user can communicate with the kernel.*
  - *"**bash**" is the most commonly used shell on Linux.*
  - *The shell parses commands entered by the user and translates them into logical segments to*
- ***Terminal***
  - *Gives the shell a place to accept typed commands and to display*
  - *be executed by the kernel or other utilities.*

Linux Commands

- Commands have the following syntax

Command [options] [arguments]

- *Options* are used to modify command behavior and most probably are having (–) before them
- *Arguments* are file names or other info needed by the command
- Each item is separated by a space
- Separate commands with semicolon (;)

Examples

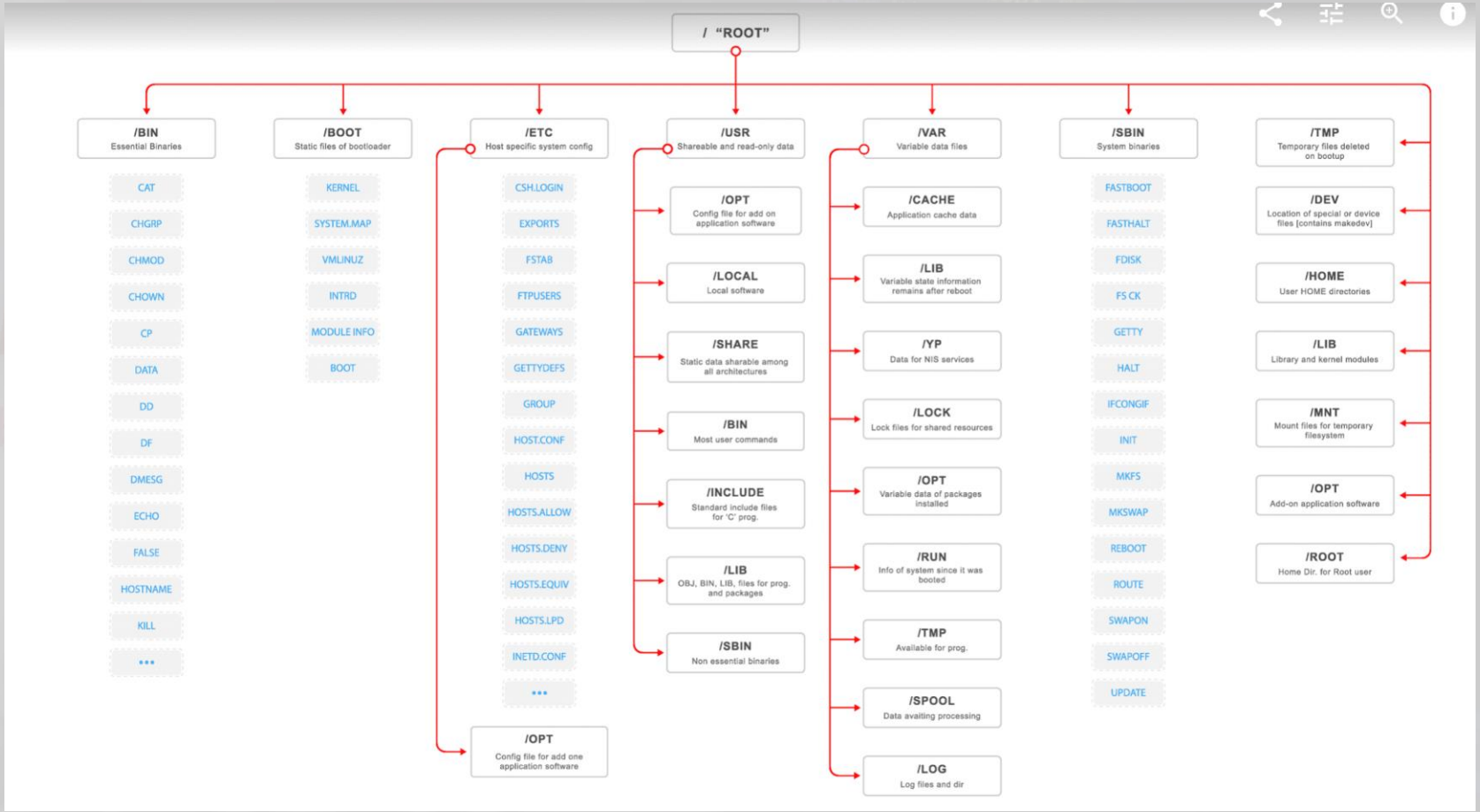
```
cal
cal 5
cal 5 2020

whoami
cal;whoami
```

Directories tree

Absolute pathname
/home/nagger/Downloads

Relative pathname
./Downloads



Basic Commands

To Displays the current working directory of the terminal.

```
pwd
```

To move from directory to directory on the system

```
cd
```

```
cd /home/nagger/Downloads
cd ~
cd ..
```

To list directory content

```
ls command
```

Getting help

- *man command*
- *whatis command*

  Shows the commands one line description
- command **--help** Option
  - Another way to get help about a command.
  - help is built in the command itself (if supported).

Viewing File Content

```
cat fname
```

```
more fname
```

- Spacebar: moves forward on screen
- b: move back one screen
- q: quit and return to the shell prompt

```
head -n fname
```

```
tail -n fname
```

File Naming

- File names may be up to 255 characters.
- There are no extensions in Linux
- Avoid special characters as >< ? * # '
- File names are case sensitive
- Everything in linux is a file
- Directory is a special type of file (you can’t have a directory and file with the same name in the same path)

FILE & DIR. MANIPULATION

To create files

```
touch file(s)_name
```

To create directories

```
mkdir [-p] dir(s)_name
```

To remove files

```
rm[-i] file(s)_name
```

To remove directories

```
rmdir dir(s)_name
rm [-r] dir(s)_name
```

FILE & DIR. MANIPULATION

To Copy Files and Directories

```
cp [options] source(s) target
```

- -i Prevents you from accidentally overwriting existing files or directories
- -r Copy a directory including the contents of all subdirectories

Moving and Renaming Files and Directories

```
mv [options] source(s) target
```

- -i Prevents you from accidentally overwriting existing files or directories

File Globing

Asterisk(*): represents **0 or more character**, except leading (.)

Question mark(?) character represents **any single character** except the leading (.)

Square bracket([]): represent a **range of characters** for a single character position.

Questions?!

Let’s get connected!

/mohammedalnaggar

0020 1010-8595-07

Thank YOU!

boost your career