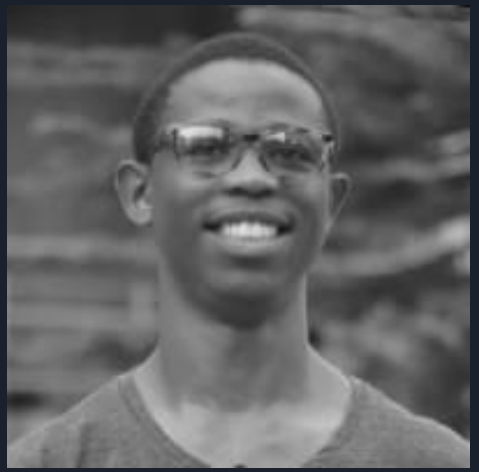
Day 3: Command-line Basics (part 2)

Nemuel Wainaina

27th March 2024

> whoami



Nemuel Wainaina

- Software Engineer
- Security Researcher
- Technical Instructor (1.7k+ students)

Github : https://github.com/nemzyxt

LinkedIn : https://linkedin.com/in/nemuel-wainaina

Twitter : https://twitter.com/nemuelwainaina

Medium : https://medium.com/@nemzyxt

> more commands ...

Managing users:

- **whoami :** display the current username
- **who :** display ***information*** about users who are currently logged in
- **users :** display a ***list*** of currently logged on users
- **adduser, useradd :** create a new user
- **passwd :** change user password
- **sudo <commands> :** execute the command(s) with root privileges

> more commands ...

Software/package management:

- **apt-get / yum / dnf :** install, update, or remove software packages
- **apt-get install :** install a software package
- **apt-get update :** update the local package index
- **apt-get upgrade :** upgrade installed packages to their latest versions
- **apt-get dist-upgrade :** upgrade installed packages (intelligently)
- **apt-get remove :** remove the package but retain config files
- **apt-get purge :** remove both the package and config files
- **apt-cache search :** search for package name in online package database
- **apt-get do-release-upgrade :** upgrade to newer distro release

> more commands ...

Networking operations:

- **ping :** check network connectivity
- **ifconfig :** display and configure network interfaces
- **netstat :** display network connections, routing tables and interface statistics
- **ssh :** securely connect to remote machines
- **scp :** securely copy files between machines
- **wget :** download files from the internet

> managing processes

Managing processes:

- **ps :** display information about running processes
- **top :** display dynamic real-time information about running processes
- **kill :** send a signal to a process, default being the TERM (terminate) signal
- **killall :** kill processes by name
- **pgrep :** look up processes by name and other attributes
- **pkill :** send signal to processes based on certain criteria

> managing permissions

- Permissions dictate who can read, write or execute files & directories on the system
- They are divided into 3 sets: **User** (U), **Group** (G), **Others** (O)

- There are 3 primary types of permissions:
  - **Read (r) :** allows the user to view the contents of the file or directory
  - **Write (w) :** allows the user to modify the contents of the file or directory
  - **Execute (x) :** allows the user to execute the file if it is a program or script

- The underlying system only understands 1s and 0s language …
  - r w x => 1 1 1 => 4 2 1 Therefore: 4 + 2 + 1 = 7 (all permissions granted)

> more on permissions

Examples :

- rw- : 4 + 2 = 6
- r-x : 4 + 1 = 5
- **---** : 0 + 0 + 0 = 0

To change file permissions:

- **chmod :** change file or directory permissions

> a note on bash scripting

- **Bash (Bourne Again Shell) :** a command-line interpreter (shell) for Linux-based systems. It's the default shell even on MacOS!

- **Bash scripting :** involves writing scripts that contain a series of commands to be executed by the Bash shell. These scripts can be used to automate repetitive tasks, perform system administration tasks, etc.

- To create one, create a new file, with a **'.sh'** extension. Add some commands to it. Give it **execute (x)** permissions. Run it with:
  - ***./scriptname.sh***

Q&A