**Born2beroot**

**- Set up sudo and users/groups**

Sudo allows the user to run a program as another user, most often the root user.

\*su commando:

The su (short for substitute or switch user) utility allows you to run commands with another user’s privileges, by default the root user.

\* exit

Go back to user console with normal privileges.

\* apt update & apt upgrade

apt is a command-line utility for installing, updating, removing, and otherwise managing deb packages.

\* sudo apt update

This will pull the latest changes from the APT repositories.

\* sudo apt upgrade

Upgrade the installed packages to their latest versions run.

\* apt install sudo

Install sudo if you don't want to keep switching to the root user., cause it's not automatically installed with Debian. (Lsblk to show the block devices just installed. Block devices are nonvolatile mass storage devices whose information can be accessed in any order. Hard disks, floppy disks, and CD-ROMs are examples of block devices).

\* sudo usermod -aG sudo "username"

This command adds the user to the sudo group, providing them with sudo privileges.

\* usermod: a command in Linux that is used to change the properties of a user.

\* -a: Add the the group to the list of groups that the user is part of.

\* -G: Groups and tells usermod that the following argument is a group.

\* getent group sudo

Checks the user of a particular group.

\* sudo groupadd "name"

Create a new user group.

\* sudo useradd -g "group name" "username".

Add a new user into the group

\* sudo usermod -aG "group name" "existing\_user"

Add an existing user to a group.

\* sudo apt install vim

Install vim if nano isn't sufficient.sudo

\* sudo useradd "username"

Creates a new user.

\* Use the -m (--create-home) option to create the user home d directory as /home/username

**Set up SSH server**

SSH is a protocol for securely exchanging data between two computers over an untrusted network. SSH protects the privacy and integrity of the transferred identities, data, and files. The SSH server listens for incoming connection requests (usually on TCP port 22(that we will change to 4242) on the host system) and responds to them.

The main difference between ssh and sshd: sshd is a server (like a web server serving https) and SSH is a client (think of a web browser). The client/user authenticates itself against the server using the users credentials.

\* sudo apt install openssh-server

Installs the server.

\* sudo systemctl status ssh

Verifies the server and its activity.

\* systemctl: Command to manage services and control when they start.

\* ip a

Command to retrieve your IP address.

\* cd /etc/ssh, sudo vim sshd\_config. Change port 22 to port 4242. (Don't forget tor remove the hashtag)adds port 4242

**Uncomplicated firewall (UFW)**

A firewall is network security system that monitors a controls incoming and outgoing network traffic based on predetermined security rules. A firewall typically establishes a barrier between a trusted and an untrusted network like the internet. Ufw is used through the command line.

\* sudo apt install ufw

installs uncomplicated firewall

\* sudo ufw enable

activate the firewall

\* sudo ufw status

check if it's active

To change the port in the os:

Go to VirtualBox -> Choose the VM -> Select "Settings" -> Choose "Network" -> "Adapter 1" -> "Advance" -> "Port Forwarding". Set host and guest to 4242 and use "sudo reboot" in the terminal to reboot.

\* sudo ufw allow 4242(/tcp)

allows the port on the firewall

Sudo ufw status numbered

To gain remote access to the SSH server run either of the following two commands.

- (sudo) ssh -p 4242 username@localhost

- (sudo) ssh -p 4242 username@127.0.0.1

**Setting up a password policy**

The password and authentication-related configuration files are stored in /etc/pam.d/ directory in Debian- based systems. And the password policies are defined in /etc/pam.d/common-password file.

\* sudo apt install libpam-pwquality

Installs password quality checking library.

\* sudo vim /etc//login.defs

change the password expiration values in login.defs.

\* go to /etc/pam.d and run "sudo vim common-password"

\* look for password requisite pam\_pwquality.so retry=3 by using "/" in vim

\* set at least one upper-case letter in the pw add ucredit=-1

\* set at least one lower-case letter in the pw add lcredit=-1

\* set at least one digit in the pw add dcredit=-1

\* set the minimum length in the pw add minlen=10

\* set at max consecutive identical chars in the pw add maxrepeat=3

\* check if the password contains the username in some form add usercheck=0

\* set a minimum number of chars that must be different from the old pw add difok=7

\* make the root password comply to this policy add enforce\_for\_root

- Configure Sudo group

\* sudo visudo

\* you have to run this command from /et/sudoers.d

The visudo command edits the sudoers file, which is used by the sudo command, to change what users and groups are allowed to run sudo, run visudo.

In the defaults sections add the following.

\* Defaults requiretty

- enables TTY

Linux operating system represents everything in a file system, the hardware devices that we attach are also represented as a file. The terminal is also represented as a file. There a command exists called tty which displays information related to terminal. The tty command of terminal basically prints the file name of the terminal connected to standard input. tty is short for teletype, but popularly known as a terminal it allows you to interact with the system by passing on the data (you input) to the system, and displaying the output produced by the system.

\* Defaults badpass\_message="error message"

When an incorrect password is given.sudos

\* Defaults logfile="/var/log/sudo/sudo.log"

Creating the log file that archives both inputs and outputs actions using sudo.

Linux systems typically save their log files under the /var/log directory. This works fine, but check if the application saves under a specific directory under /var/log. If it does, great. If not, you may want to create a dedicated directory for the application under /var/ log. Why? Because other applications save their log files under /var/log as well, and if your application saves more than one log file‚Äîp erhaps once every day or after each service restart‚Äîit may be difficult to crawl through a large directory to find the file you want.

\* Defaults passwd\_tries=3

set the maximum password tries to 3.

\* Defaults log\_input, log\_output (no specifiers necessary because you are in the sudoers file)

- Construct a script

\*The #! shebang is used to tell the kernel which interpreter should be used to run the commands present in the file

\* sudo vim /usr/local/bin/monitoring.sh

\*creates a shell script.

A Bourne shell script is a text file intended to be executed as a set of commands for the Bourne shell, which is a shell (command line interpreter) for Unix-style operating systems. While a number of other shells exist for such operating systems, the Bourne shell (and descendants such as Bash) has been the normal default since Version 7 Unix in 1977, so it is what most computer people think of when discussing "Unix commands".

\*wall

a command utility to display a message on all terminals of logged in users.

\* uname

A command-line utility that prints basic information about the operating system name and system hardware.

\* -a prints all information.

\* grep

grep, which means global regular expression print, is a small family of commands that search input files

for a search string and print any lines that match it

\* architecture

\*uname -a

\* Physical processor

grep "physical id" /proc/cpuinfo | sort | uniq | wc -l

\* wc

wc is a utility to count words, lines, or bytes of the provided input, either a list of files or standard input.

\* -l

Suppress normal output; instead print the name of each input file from which output would

normally have been printed. The scanning will stop on the first match.

\* sort

The sort command prints the output of a file in given order. This command processes on your data (the content of the file or output of any command) and reorders it in the specified way, which helps us to read the data efficiently.

\* uniq

the tool that helps to detect the adjacent duplicate lines and also deletes the duplicate lines. uniq filters out the adjacent matching lines from the input file(that is required as an argument) and writes the filtered data to the output file.

\*Virtual processor

grep "^processor" /proc/cpuinfo | wc -l

\* The current available RAM on your server and its utilization rate as a percentage.

\* df shows disk space

\* -m = shows it in MB

\* -Bg = shows it in GB

\* ^ = beginning of the line.

\* $ = end of the line.

\* -v = to deselect the line.

\* awk = allows the user to use variables, numeric functions, string functions, and logical operators

\* Free server space and usage

\* df -Bm | grep '^/dev/' | grep -v '/boot$' | awk '{fdisk += $3} {tdisk += $2} END {printf("%.2f"), fdisk/tdisk\*100}' \*$ refers to the columns

\*Free -m

Shows the current available ram in MN

\* Cpu usage

\*top -bn1 | grep '^%Cpu' | awk '{printf("%.1f%%"), $2}'

\* top

top command is used to show the Linux processes. It provides a dynamic real-time view of the running system. Usually, this command shows the summary information of the system and the list of processes or threads which are currently managed by the Linux Kernel.

\* -n = exit top command after specific repetitions.

\* -b = send output from top to file or other programs.

\* Last reboot

\* who -b | awk '{print $3 " " $4}' (only gives us the third and fourth column)

\* who = who are currently logged in.

\* -b = show last time of system reboot

\* Check if LVM is active

\* $(lsblk | grep 'lvm' | if ($1) echo "yes" else echo "no";exit;')

\* lsblk = show partitions

\* check the active connections

\* $(ss -t | grep ESTAB | wc -l)

\* ss -t = investigate and display sockets

\* ESTAB = established socket

\* the number of users using the server

\* $(who | cut -d " " -f 1 | sort -u | wc -l)

\* cut -d = delimiter with " "

\* -f = only field limited by -d

\* sort -u to only output the first one

\* The IPv4 address of your server and its MAC (Media Access Control) address

\* ip link show | grep ether | awk '{print $2}'

\* ip link show = display network device.

\* Number of sudo commands

\* journalctl \_COMM=sudo | grep COMMAND | wc -l

\* journalctl\_COMM To look for log messages from a specific application. modifier.

\* Make the shell appear on on boot and every 10 minutes

Cron allows Linux and Unix users to run commands or scripts at a given date and time. You can schedule scripts to be executed periodically.

\* sudo crontab -e

Creates and/or makes your own cronjob

syntax = 1 2 3 4 5 /root/backup.sh

\* 1 = minutes (0-59)

\* 2 = hours (0-23)

\* 3 = days (0-31)

\* 4 = month (0-12)

\* 5 = days (of the week) (0-7)

\* sudo crontab -e

\*/10 \* \* \* \* /usr/local/bin/monitoring.sh

@reboot sleep 12; sh /usr/local/bin/monitoring.sh

- Get Digital signature

The main difference is that sshd is a server (like a web server serving https) and SSH is a client (think of a web browser). The client/user authenticates itself against the server using the users credentials.