BORN2BEROOT

Table of Contents

1. Description
2. Installation
3. Commands

Description

This project aims to introduce you to the wonderful world of virtualization.

Skills
Rigor Network & system administration

Installation

This is Debian 11, codenamed bullseye, netinst, for 64-bit PC (amd64) Download.

Commands

See the partition:

```
# lsblk
```

Be root:

```
# su
```

User add high level:

```
# adduser <user_name>
```

User add low level:

```
# useradd <user_name>
```

User delete:

```
# deluser <username>
```

See Groups:

```
# getent group
```

```
# getent group <group_name>
```

Group create:

```
# groupadd <groupname>
```

Group delete:

```
# groupdel <groupname>
```

See user groups:

```
# groups <username>
```

Adding users to groups:

```
# usermod -aG <group_name> <user_name>
```

Removing users from groups:

```
# gpasswd --delete <user_name> <group_name>
```

Install SSH:

```
# apt install openssh-server
```

SSH initialization:

```
# systemctl start ssh
# systemctl enable ssh
```

SSH query:

```
# systemctl status ssh
```

#Port 22 -> Port 4242 #PermitRootLogin prohibit-password -> PermitRootLogin no

```
# nano /etc/ssh/sshd_config
```

SSH service restart:

```
# service sshd restart
```

Virtual Machine restart:

```
# reboot
```

Connect from physical machine:

```
# ssh your_42user_name@localhost -p 4242
```

Install UFW:

```
# apt install UFW
```

Deny all incoming requests.:

```
# ufw default deny incoming
```

Accept outgoing requests.:

```
# ufw default allow outgoing
```

Enable UFW:

```
# ufw enable
```

Check UFW:

```
# ufw status
```

```
# ufw status numbered (sequential rules)
```

Allow port 4242:

```
# ufw allow 4242
```

Deny port 4242:

```
# ufw deny 4242
```

See all system information:

```
# uname -a
```

Change the Hostname:

```
# hostnamectl set-hostname <new-name>
# vim /etc/hosts
$ 127.0.1.1 <new-name>
```

Delete the allowed rule:

```
# ufw delete allow 4242
```

Delete disallowed rule:

```
# ufw delete deny 4242
```

Delete first Rule:

```
# ufw delete 1
```

Install Sudo:

```
# apt install sudo
```

Adding users to Sudo:

```
# usermod -aG sudo <user_name>
```

Let's open configuration file:

```
# -sudo visudo -f /etc/sudoers.d
```

```
Defaults     passwd_tries=3 Defaults     badpass_message Defaults
requiretty Defaults     logfile="/var/log/sudo/sudo.log" Defaults
log_input, log_output      Defaults     iolog_dir="/var/log/sudo/"
Defaults     secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap
```

Let's edit password policies:

```
# nano /etc/login.defs
```

```
PASS_MAX_DAYS   30 PASS_MIN_DAYS    2 PASS_WARN_AGE   7
```

See password policies:

```
# chage -l <username>
# chage -l root
```

Install password policies:

```
# apt install libpam-pwquality
```

Add password policies:

```
# nano /etc/security/pwquality.conf
```

```
difok = 7  minlen = 10  dcredit= -1  ucredit= -1  enforce_for_root
enforcing= 1 maxrepeat= 3 usercheck = 1 dictcheck = 1
```

Changing users password:

```
# passwd root
# passwd <user_name>
```

Getting system information every 10 minutes:

```
# crontab -u root -e
*/10 * * * * bash /your/monitoring.sh_path
```

Open the Monitoring.sh:

```
# vim /usr/local/sbin/monitoring.sh
```

Edit file "Monitoring.sh":

```
ARCH=$(uname -a)
PCPU=$(cat /proc/cpuinfo | grep cpu\ cores | uniq | wc -l)
VCPU=$(cat /proc/cpuinfo | grep processor | wc -l)
CPUUSG=$(top -b -n1 | grep "Cpu(s)" | awk '{print($4)"%"}')
LASTBOOT=$(who -b | awk '{print $3,$4}')
lvmrtn=$(lsblk | grep "lvm" | wc -l)
LVMGET=$(if [ $lvmrtn -eq 0 ]; then echo no; else echo yes; fi)
TCPCNT=$(netstat | grep ESTABLISHED | wc -l)
TCPEST=$(netstat | grep ESTABLISHED | awk '{print($6)}')
USRCNT=$(who | wc -l)
NTWRKMAC=$(ip link show | grep link/ether | awk '{print($2)}')
NTWRKIP=$(hostname -I)
SUDOCNT=$(journalctl _COMM=sudo | grep COMMAND | wc -l)
MEMUSG=$(free -m | grep Mem: | awk '{print($3)}')
MEMUSGTTL=$(free -m | grep Mem: | awk '{print($2)}')
MEMUSGPRCNT=$(free -m | grep Mem: | awk '{printf("%.2f"),($3/$2*100)}')
DSKUSGMB=$(df -BM --total | grep total | awk '{print($3)}' | tr -d M)
DSKUSGGB=$(df -BG --total | grep total | awk '{print($3)}' | tr -d G)
DSKUSGPRCNT=$(df --total | grep root | awk '{print($5)}')
```

```
wall "
        #Architecture: $ARCH
        #CPU physical : $PCPU
        #vCPU : $VCPU
        #Memory Usage: $MEMUSG/$MEMUSGTTL"MB" ($MEMUSGPRCNT"%")
        #Disk Usage: $DSKUSGMB/$DSKUSGGB"Gb" ($DSKUSGPRCNT)
        #CPU load: $CPUUSG
        #Last boot: $LASTBOOT
        #LVM use: $LVMGET
        #Connexions TCP : $TCPCNT $TCPEST
        #User log: $USRCNT
        #Network: "IP" $NTWRKIP ($NTWRKMAC)
        #Sudo : $SUDOCNT cmd
```

See cron status:

```
# sudo systemctl status cron
```

Stop the cron:

```
# sudo systemctl stop cron
```

Disable the cron:

```
# sudo systemctl disable cron
```

Get disk signature:

```
# shasum <your_virtual_machine>.vdi
```