

UNIVERSIDAD DE LAS FUERZAS ARMADAS-ESPE SEDE SANTO DOMINGO

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN - DCCO-SS

CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN

PERIODO : Mayo 2020 – Septiembre 2020

ASIGNATURA : Sistemas Operativos

TEMA : Componentes y conceptos de un Sistema Operativo

NOMBRES : Jonathan Landa

NIVEL-PARALELO : Tercer “B”

DOCENTE : Germán Rodríguez

FECHA DE ENTREGA : 24/05/2020

SANTO DOMINGO - ECUADOR

2020

1. Introducción

El sistema operativo es el software que coordina y dirige todos los servicios y aplicaciones que utiliza el usuario, por eso es el más importante y fundamental en una computadora. Se trata de programas que permiten y regulan los aspectos más básicos del sistema. Los sistemas operativos más utilizados son Windows, Linux, OS/2 y DOS

2. Sistemas de Objetivos

2.1. Objetivo General: Instalar un Sistema Operativo en nuestro ordenador a través de una maquina virtual.

2.2. Objetivos Específicos:

2.2.1. Instalar un sistema operativo basado en Linux

2.2.2. Analizar todos los componentes del sistema operativo instalado

2.2.3. Comprobar la fundamentación teórica recibida en clases sobre este sistema operativo

3. Desarrollo

3.1

- **Virtual Box**

VirtualBox es un software de virtualización para arquitecturas x86/x64, creado originalmente por la empresa alemana innotek GmbH. Actualmente es desarrollado por Oracle Corporation. Con esta aplicación es posible instalar sistemas operativos adicionales, conocidos como «Sistemas Invitados», dentro de otro sistema operativo «Anfitrión», cada uno con sus recursos virtuales.

Entre los sistemas operativos que soportada el modo anfitrión se encuentran GNU/Linux, Mac OS X, OS/2 Warp , Microsoft Windows, y Solaris/OpenSolaris, dentro de estos es

posible virtualizar los sistemas operativos FreeBSD, GNU/Linux, OpenBSD, OS/2 Warp, Windows, Solaris, MS-DOS, entre muchos otros.

- Instalación de la Máquina Virtual (“Virtual Box”)
- Descargamos e instalamos nuestra máquina virtual (“Virtual Box”) desde su página oficial <https://www.virtualbox.org/wiki/Downloads>



- Una vez descargado la máquina virtual procedemos a su instalación.

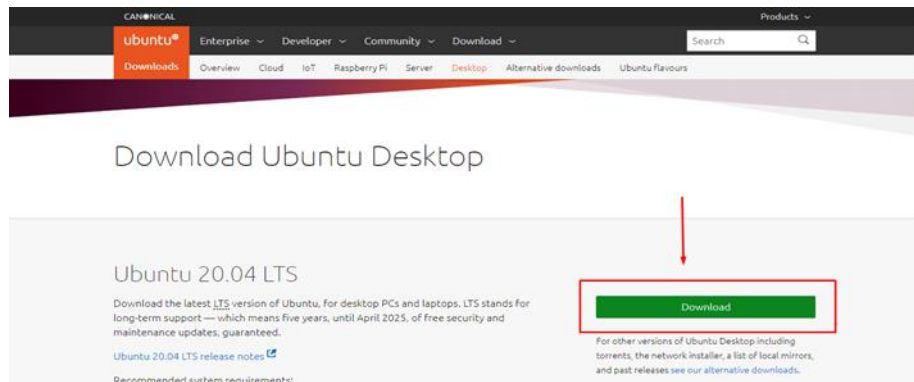


- **UBUNTU**

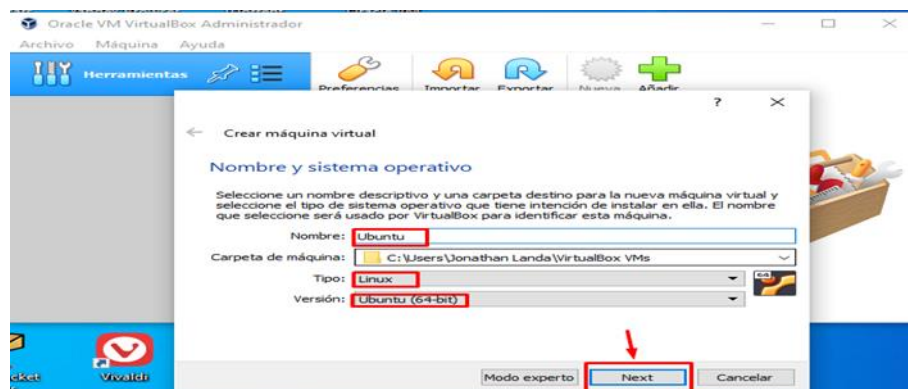
Ubuntu es un sistema operativo de código abierto para computadores. Es una distribución de Linux basada en la arquitectura de Debian. Actualmente corre en computadores de escritorio y servidores, en arquitecturas Intel, AMD y ARM.

- Procedemos a descargar nuestro S.O desde su página oficial (“Ubuntu”)

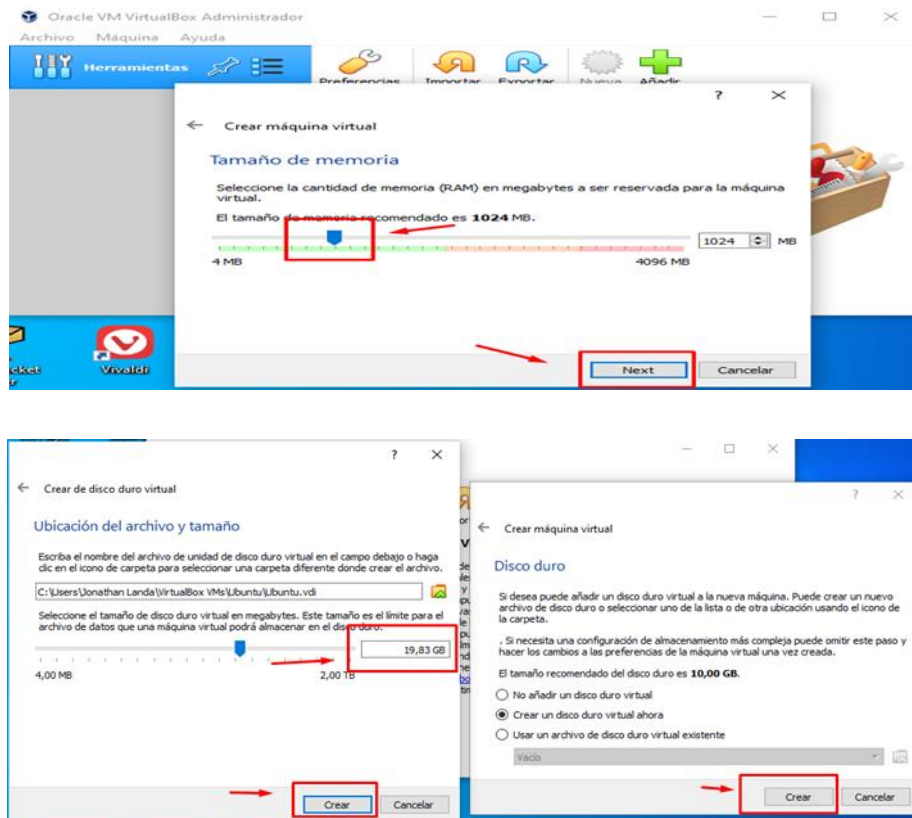
<https://ubuntu.com/download/desktop>



- Una vez descargado nuestro S.O procedemos a instalar en nuestra maquina virtual.
- Seleccionamos la pestaña nueva “Nueva” para crear nuestra máquina virtual el cual instalaremos nuestra S.O “Ubuntu”



- Seleccionamos la cantidad de memoria Ram queremos que se consuma en nuestro ordenador, también fijar la ubicación y tamaño del archivo.



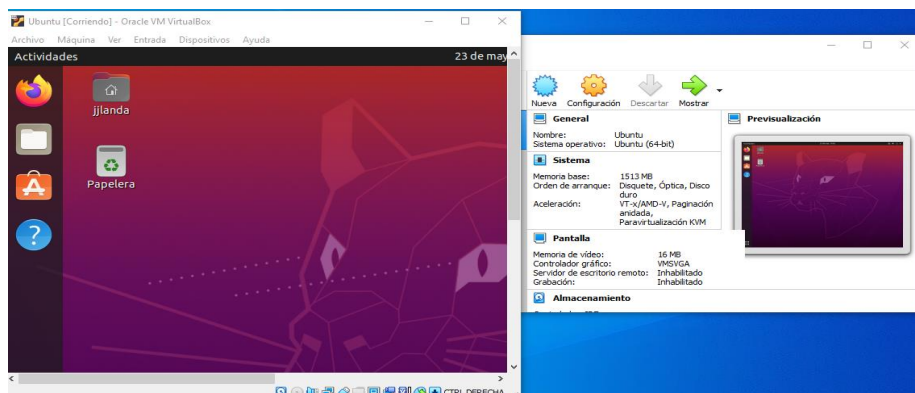
- Una vez creada procedemos a iniciar y seleccionamos cargar la imagen ISO en la carpeta donde la guardamos.



- Seleccionamos el idioma de nuestra preferencia y procedemos a instalar.



- Terminada la instalación del SO en su máquina virtual procedemos a reiniciar su Virtual Box y ya tenemos instalado Ubuntu en nuestro ordenador.

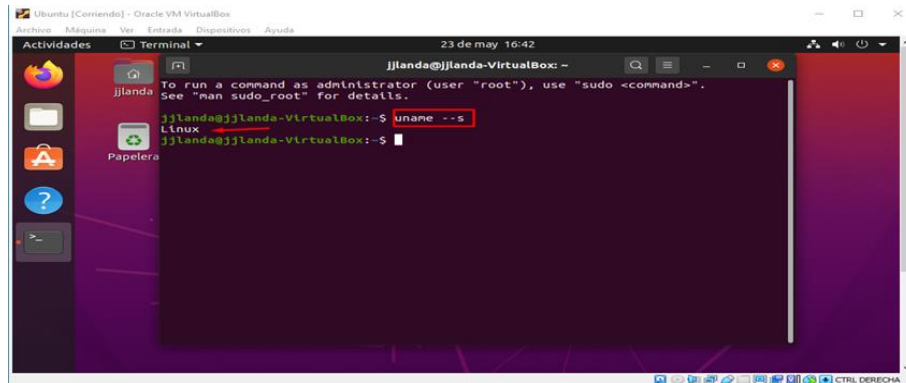


Análisis de los componentes y recursos del SO.

1. Dar clic en Applications, Systems Tools, Terminal
2. Se abrirá un terminal para ingresar los siguientes comandos:

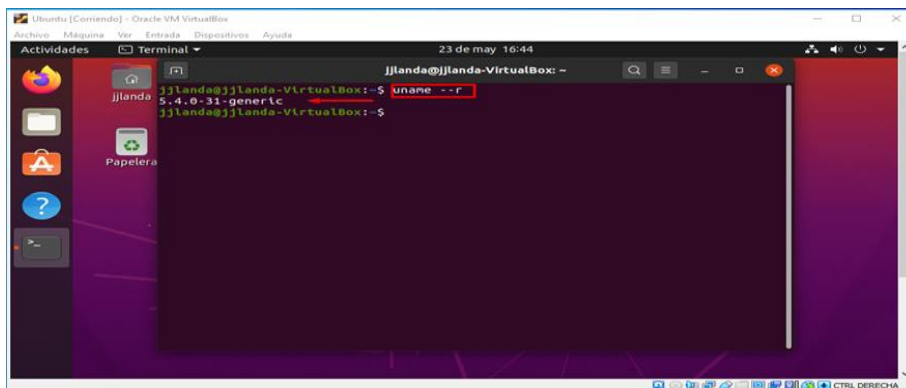
3. Imprimir el kernel (núcleo del SO) instalado, ingrese el siguiente comando (capture una pantalla del resultado):

\$uname -s



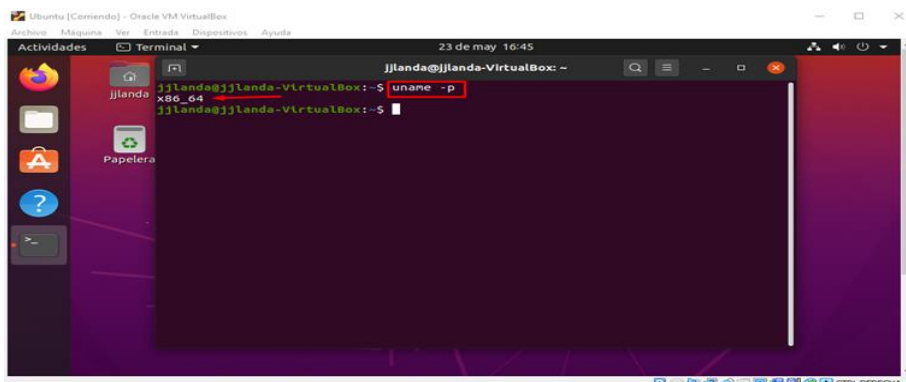
4. Imprimir el reléase del kernel del SO instalado (capture una pantalla del resultado):

\$uname -r



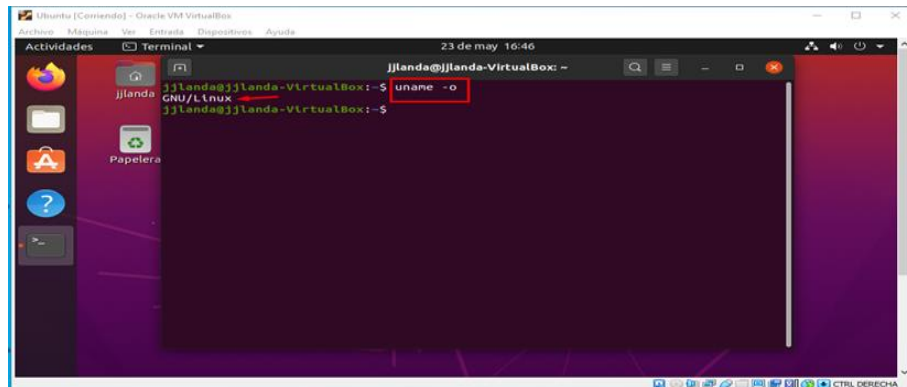
5. Imprimir el tipo del procesador (capture una pantalla del resultado):

\$uname -p



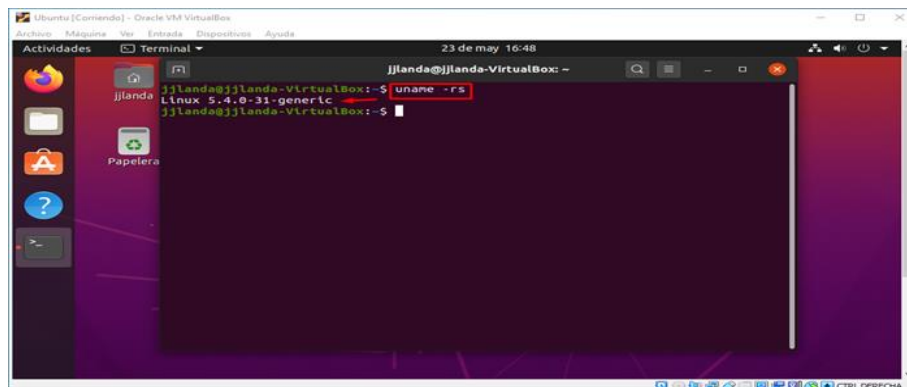
6. Imprimir el nombre del SO utilizado (capture una pantalla del resultado):

`$uname -o`



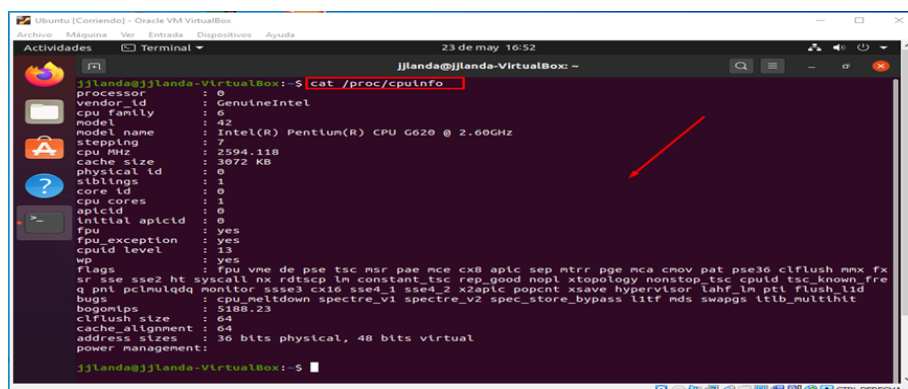
7. Para determinar la versión del kernel del SO instalado (capture una pantalla del resultado):

`$uname -rs`



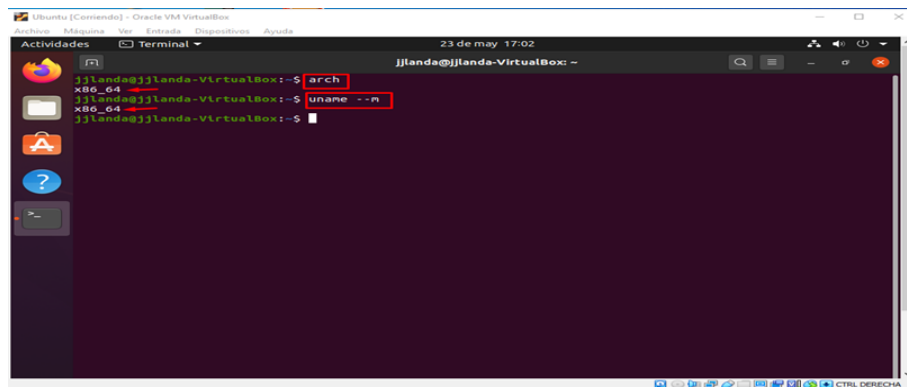
8. Mostrar información del CPU, (capture una pantalla del resultado):

`$cat /proc/cpuinfo`



9. Mostrar la arquitectura de la máquina, (capture una pantalla del resultado):

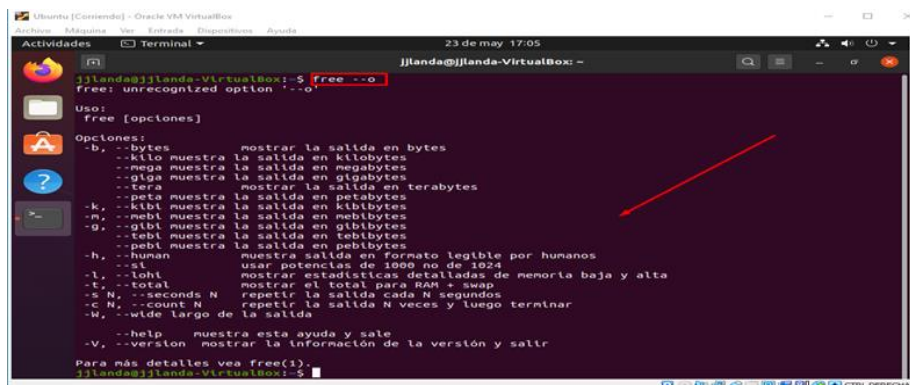
\$arch



```
jjlanda@jjlanda-VirtualBox:~$ arch
x86_64
```

10. Mostrar el total de memoria RAM y la partición SWAP, (capture una pantalla del resultado):

\$free -o -m



```
jjlanda@jjlanda-VirtualBox:~$ free -o -m
free: unrecognized option '-o'
Usa:
free [opciones]

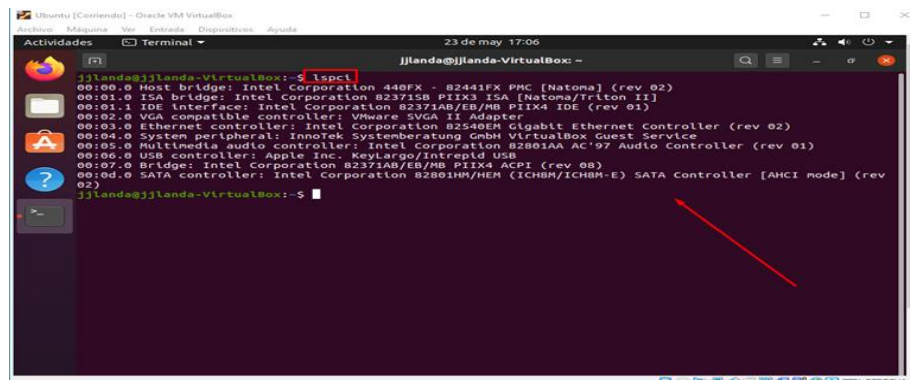
Opciones:
-b, --bytes          mostrar la salida en bytes
-k, --kilo           mostrar la salida en kilobytes
-m, --mega           mostrar la salida en megabytes
-g, --giga           mostrar la salida en gigabytes
-t, --tera           mostrar la salida en terabytes
-p, --peti           mostrar la salida en petabytes
-k, --kibi           mostrar la salida en kibibytes
-m, --mebi           mostrar la salida en mebibytes
-g, --gibi           mostrar la salida en gibibytes
-t, --tebi           mostrar la salida en tebibytes
-p, --pebi           mostrar la salida en pebibytes
-h, --human          muestra salida en formato legible por humanos
-l, --lsl            usar potencias de 1000 no de 1024
-i, --lohi           mostrar estadísticas detalladas de memoria baja y alta
-t, --total          mostrar el total para RAM + swap
-s N, --seconds N   repetir la salida cada N segundos
-c N, --count N      repetir la salida N veces y luego terminar
-W, --wide           ancho largo de la salida

--help muestra esta ayuda y sale
-V, --version muestra la información de la versión y salir

Para más detalles vea free(1)
jjlanda@jjlanda-VirtualBox:~$
```

11. Listar los dispositivos PCI / PCIe, (capture una pantalla del resultado):

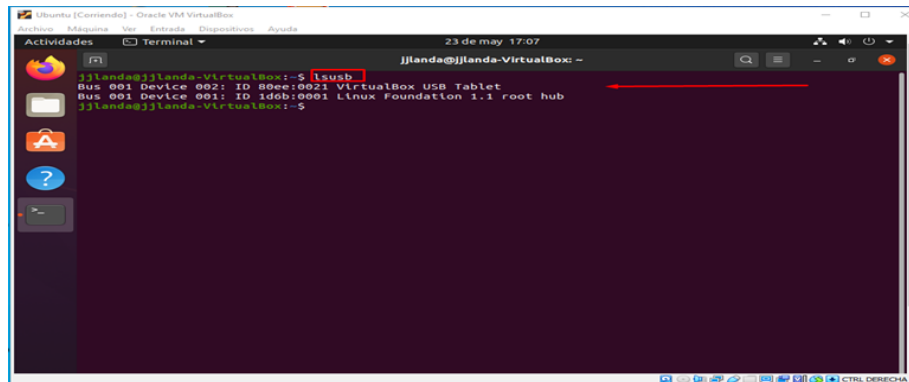
\$lspci



```
jjlanda@jjlanda-VirtualBox:~$ lspci
00:00.0 Host bridge: Intel Corporation 440FX - 82441FX PMC [Natoma] (rev 02)
00:01.0 ISA bridge: Intel Corporation 82371SB PIIX3 ISA [Natoma/Triton II]
00:01.1 IDE interface: Intel Corporation 82371AB/EB/MB PIIX4 IDE (rev 01)
00:02.0 VGA compatible controller: VMware SVGA II Adapter
00:03.0 Ethernet controller: Intel Corporation 82540EM Gigabit Ethernet Controller (rev 02)
00:04.0 System peripheral: Intel Corporation 82540EM Gigabit Ethernet Controller (rev 02)
00:05.0 Multimedia audio controller: Intel Corporation 82801AA AC'97 Audio Controller (rev 01)
00:06.0 USB controller: Apple Inc. KeyLargo/Intrepid USB
00:07.0 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 08)
00:0d.0 SATA controller: Intel Corporation 82801HM/HEM (ICH8M/ICH8M-E) SATA Controller [AHCI mode] (rev 02)
jjlanda@jjlanda-VirtualBox:~$
```

12. Listar los dispositivos USB, (capture una pantalla del resultado):

\$lsusb



```
jjlanda@jjlanda-VirtualBox:~$ lsusb
Bus 001 Device 002: ID 80ee:0021 VirtualBox USB Tablet
Bus 001 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
jjlanda@jjlanda-VirtualBox:~$
```

13. Instalar los paquetes para ejecutar el siguiente comando: lshw (listar hardware)

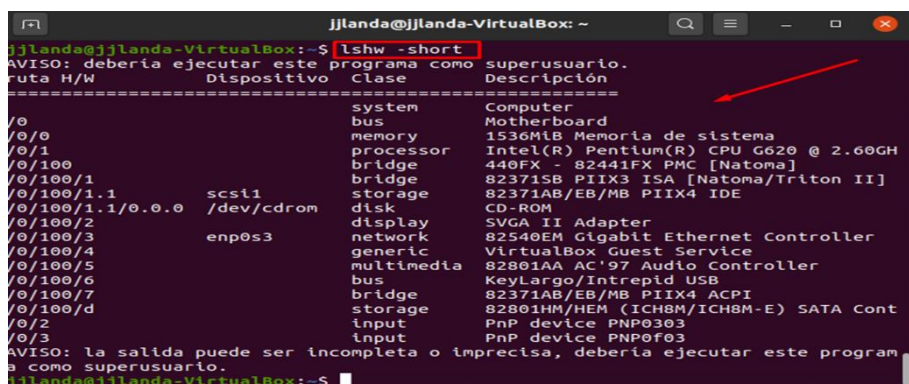
13.1 Primero acceder como usuario root (superusuario) con el comando \$sudo

13.2 Ingresar el password que se configuró inicialmente (laboratorio)

13.3 Ingresar el siguiente comando: #yum install lshw

13.4 Una vez instalado el comando ejecutar el siguiente comando, (capture una pantalla del resultado):

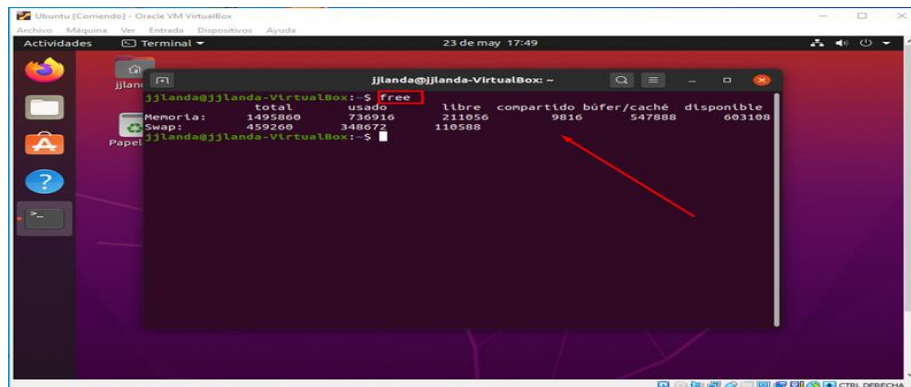
#lshw -short



```
jjlanda@jjlanda-VirtualBox:~$ lshw -short
Aviso: debería ejecutar este programa como superusuario.
Descripción
=====
/0 system Computer
/0/0 bus Motherboard
/0/1 memory 1536MiB Memoria de sistema
/0/100 processor Intel(R) Pentium(R) CPU G620 @ 2.60GH
/0/100/1 bridge 440FX - 82441FX PMC [Natoma]
/0/100/1.1 storage 82371SB PIIX3 ISA [Natoma/Triton II]
/0/100/1.1/0.0.0 /dev/cdrom disk 82371AB/EB/MB PIIX4 IDE
/0/100/2 display SVGA II Adapter
/0/100/3 network 82540EM Gigabit Ethernet Controller
/0/100/4 generic VirtualBox Guest Service
/0/100/5 multimedia 82801AA AC'97 Audio Controller
/0/100/6 bus KeyLargo/Intrepid USB
/0/100/7 bridge 82371AB/EB/MB PIIX4 ACPI
/0/100/d storage 82801HM/HEM (ICH8M/ICH8M-E) SATA Cont
/0/2 input PnP device PNP0303
/0/3 input PnP device PNP0F03
Aviso: la salida puede ser incompleta o imprecisa, debería ejecutar este program
a como superusuario.
jjlanda@jjlanda-VirtualBox:~$
```

14. Comprobar la memoria RAM libre, usada y total del sistema, (capture una pantalla del resultado):

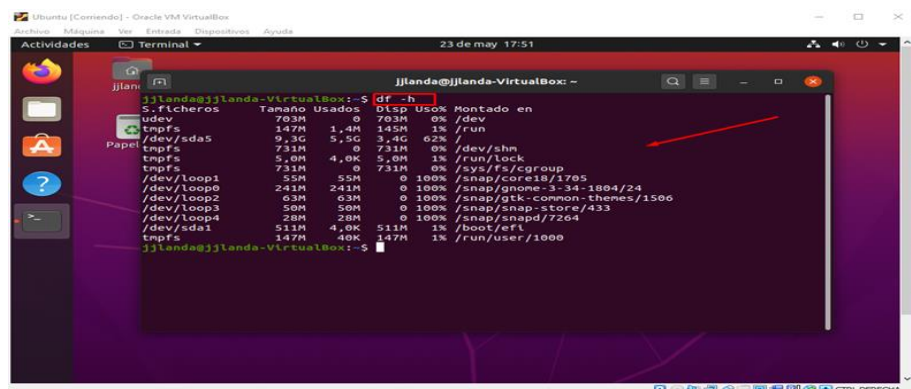
#free



```
jjlanda@jjlanda-VirtualBox:~$ free
              total        usado        libre   compartido búfer/caché disponible
Memoria:    1495860      736916      211056
Swap:       459260      348072      110588
```

15. Conocer el espacio usado y disponible en las particiones:

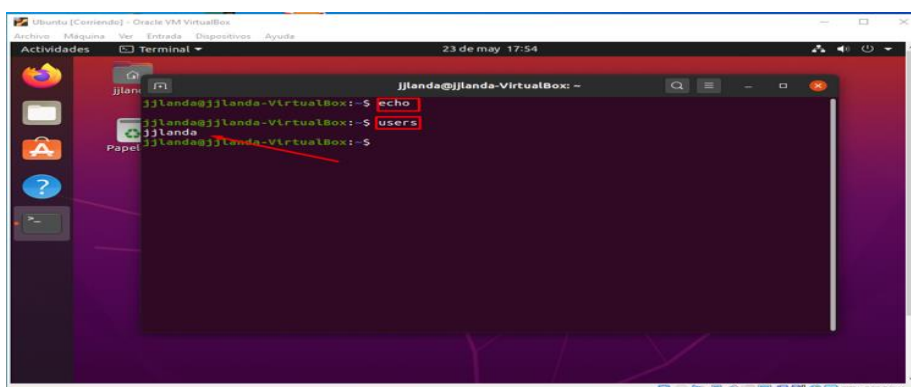
#df -h



```
jjlanda@jjlanda-VirtualBox:~$ df -h
Filesystem      Tamaño Usados  Dispo  Uso% Montado en
udev            783M    0      783M   0% /dev
tmpfs           147M    0      147M   0% /run
/dev/sda5       9.3G    5.5G    3.4G  62% /
tmpfs           731M    0      731M   0% /dev/shm
tmpfs           5.0M    4.0K    5.0M   1% /run/lock
tmpfs           731M    0      731M   0% /sys/fs/cgroup
/dev/loop0      241M    0      241M   0% /snap/core15/1705
/dev/loop2      63M    0      63M   0% /snap/gtk-common-themes/1506
/dev/loop3      50M    0      50M   0% /snap/snap-store/433
/dev/loop4      28M    0      28M   0% /snap/snapd/7264
/dev/sda1       511M    4.0K    511M   1% /boot/efi
tmpfs           147M    0      147M   0% /run/user/1000
```

16. Visualizar el nombre del usuario actual, (capture una pantalla del resultado):

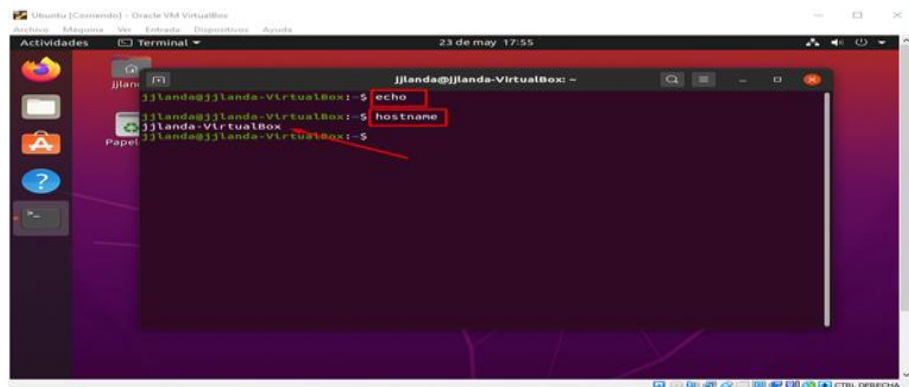
\$echo \$USER



```
jjlanda@jjlanda-VirtualBox:~$ echo $USER
jjlanda
```

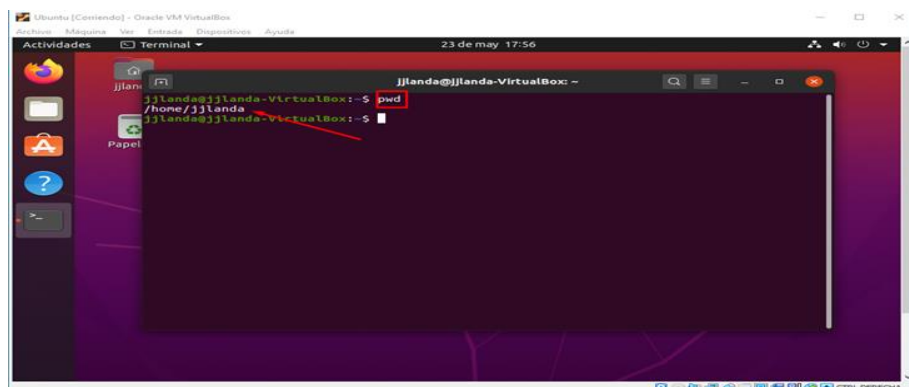
17. Visualizar el nombre del equipo, (capture una pantalla del resultado):

`$echo $HOSTNAME`



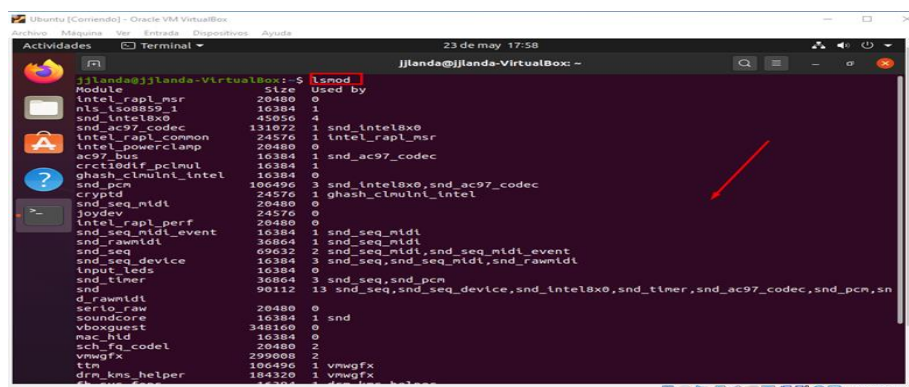
18. Visualizar el nombre del directorio actual, (capture una pantalla del resultado):

`$pwd`



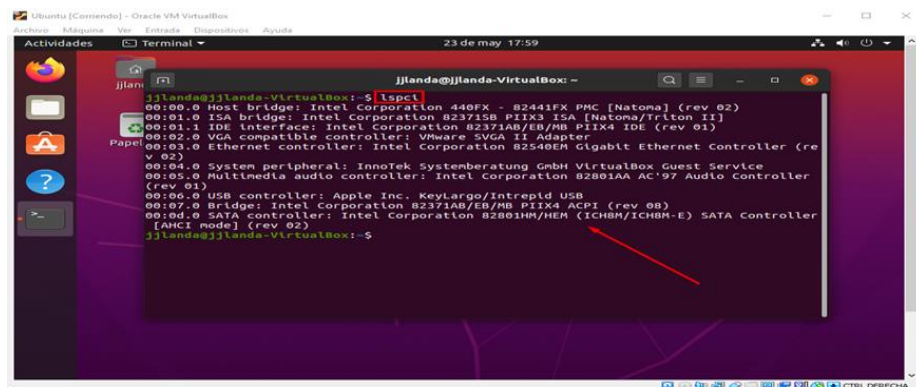
19. Listar todos los módulos que cargó el sistema, (capture una pantalla del resultado):

`#lsmod | less`



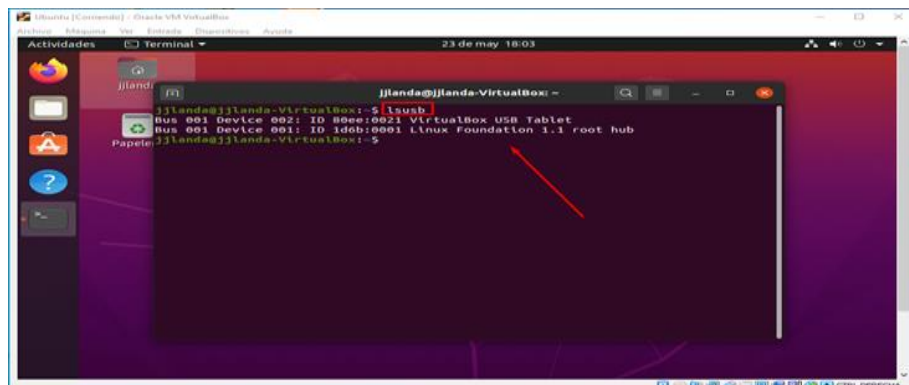
20. Listar los dispositivos de redes alámbricas PCI, (capture una pantalla del resultado):

`$lspci | grep -i ethernet`



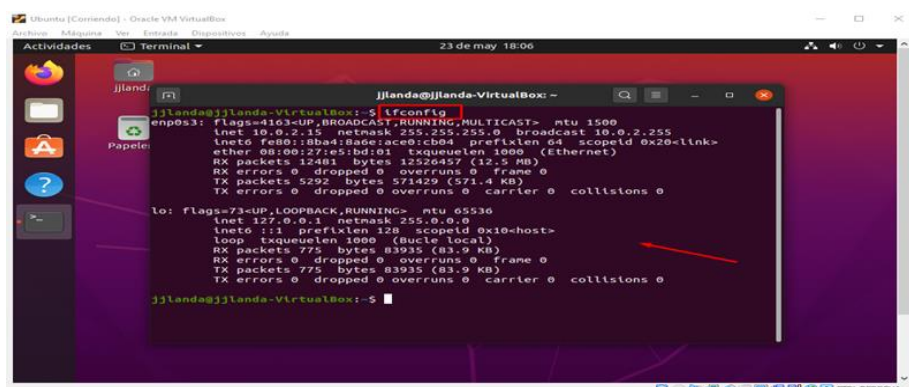
22. Listar los dispositivos de red USB, (capture una pantalla del resultado):

`$lsusb | grep -i ethernet ; lsusb | grep -i network`



23. Mostrar la configuración ip de los dispositivos de red

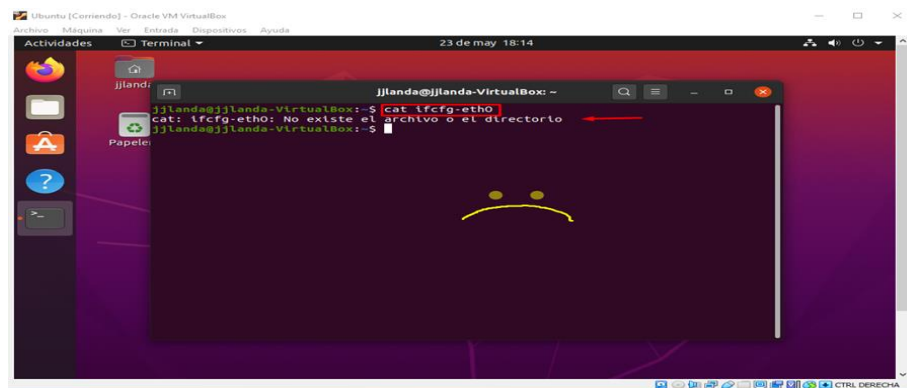
`$ifconfig`



24. En Centos, visualizar el archivo de configuración de un dispositivo de red (eth0)

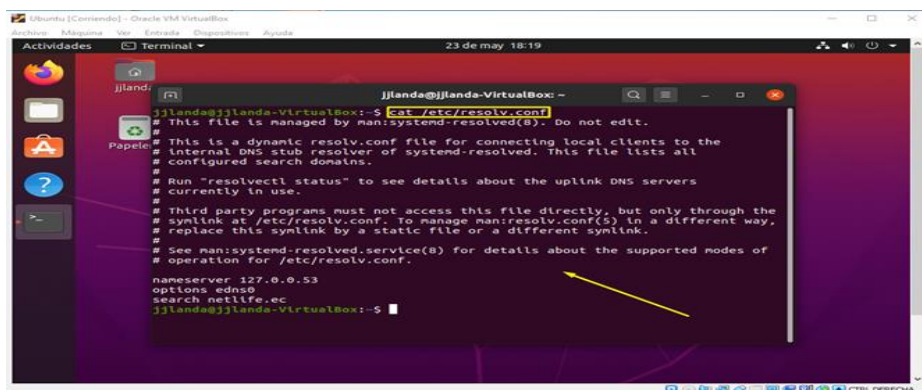
`$cd /etc/sysconfig/network-scripts`

`$cat ifcfg-eth0`



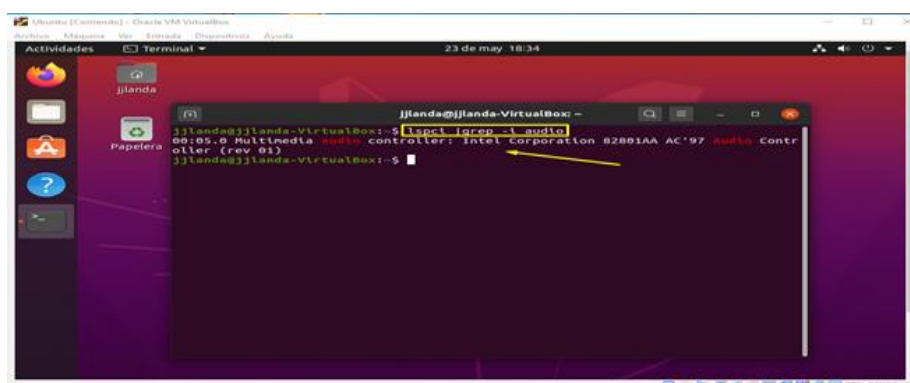
25. Visualizar los servidores configurados como DNS

`$cat /etc/resolv.conf`



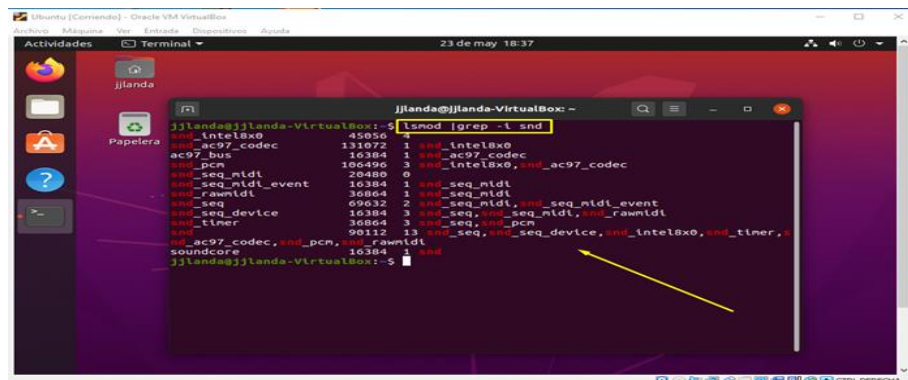
26. Listar hardware de audio

`$ lspci | grep -i audio`



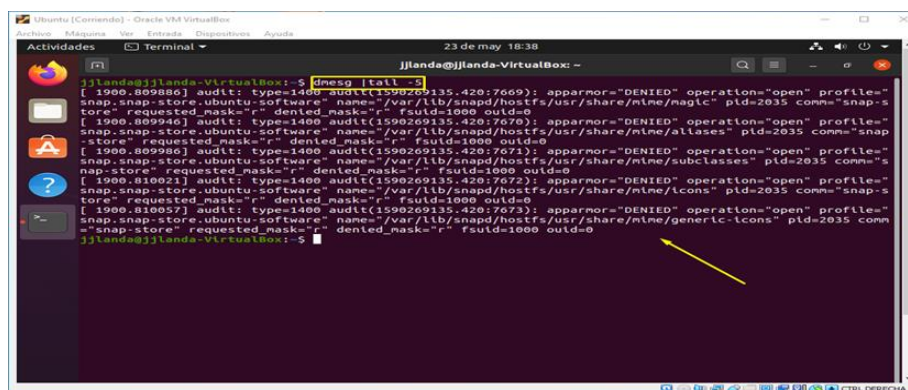
27. Módulos que cargó el sistema para ser usados por los dispositivos de sonido

\$ **lsmod | grep -i snd**



28. Visualizar las 5 últimas líneas del log del kernel.

\$ **dmesg | tail -5**



EJERCICIOS EN CASA

1. De todos los comandos ejecutados, distribúyalos de acuerdo a la siguiente tabla, añadir funciones si hace falta:

Comando(s)	Función (muestra información)
uname -p	Información del procesador
free -o -m	Memoria RAM y particiones SWAP
uname -rs	Kernel
Lspci grep -i network	Redes
Lspci grep -i audio	HW de audio
Dmesg tail -5	Logs

4. Conclusiones

- Instalar un Sistema Operativo con una máquina virtual en un ordenador de pocos recursos lleva su tiempo ya que este suele sufrir fallos.
- Se pudo concluir que la ejecución de los comandos básicos en la terminal de nuestro SO se realizó con éxito con las indicaciones dictadas por el docente.

5. Recomendaciones

- Las clases virtuales tienen su complejidad, pero el método que está llevando para su materia está siendo llevado correctamente.

6. Bibliografía/ Referencias

- ElTecnólogoEM, P. (2019, February 9). Que es y para que sirve VirtualBox. Retrieved from <https://eliezermolina.net/que-es-y-para-que-sirve-virtualbox/>
- Cuesta, G. (2019, October 8). ☒ ¿Qué es Ubuntu? " Linux en español. Retrieved from <https://www.xn--linuxenespaol-skb.com/distribuciones/ubuntu/>

7. Anexos

