Desarrollo de actividad sobre Linux, equipo 6

Descripción del SO

Linux. Linux es un sistema operativo de software libre (no es propiedad de ninguna persona o empresa), por ende no es necesario comprar una licencia para instalarlo y utilizarlo en un equipo informático.

● ¿Es open source o con licencia ?

Linux es un sistema operativo open source libre que se lanzó bajo la Licencia Pública General (GPL) de GNU y se convirtió en el proyecto de software open source más importante del mundo.

● ¿Cuales son los recursos de Hard que tiene la MV? Si no se pueden ver, buscar requisitos mínimos en internet

instalar sus SO. Lo mínimo que requiere el software libre es: Procesador 386X en adelante. Mínimo 4MB de RAM, aunque es recomendable disponer de 8 Mb. Controlador de Disco duro con 20 Mb de espacio libre, aunque es recomendable 100 Mb.

● ¿Cómo se accede a la ventana de comandos?

Ventana de comando Ctrl + Alt + t al mismo tiempo

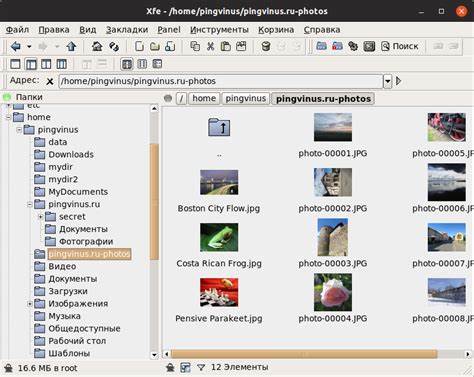
● ¿Puedo instalar aplicaciones?¿Por que?

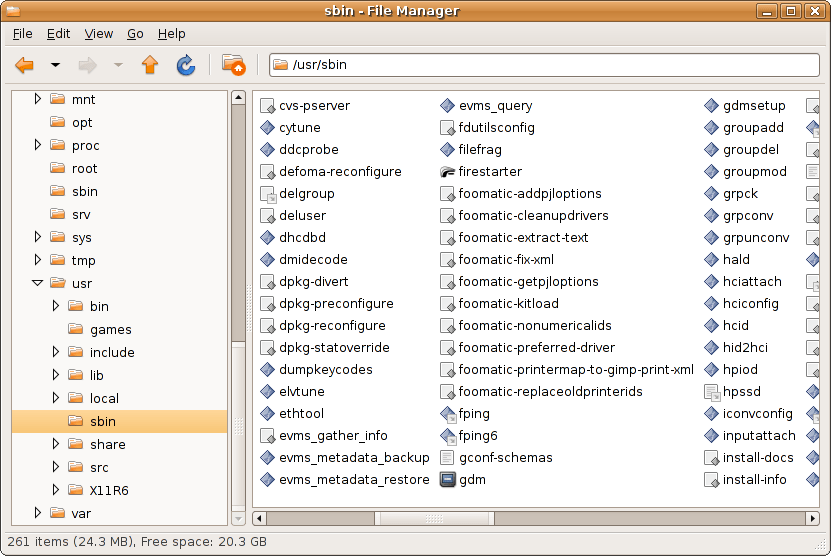
Si, porque me lo permite y son compatibles. Existen una serie de herramientas de interfaz gráfica o GUI que varían dependiendo de las diferentes distribuciones de Linux. Por ejemplo: 1.**Elementary OS:Appstore** 2.**Ubuntu Linux:Ubuntu** Software, entre otros

● ¿Hay juegos instalados?

Sí, dependiendo de la compatibilidad

● Capturar una imagen del file explorer (ejemplo





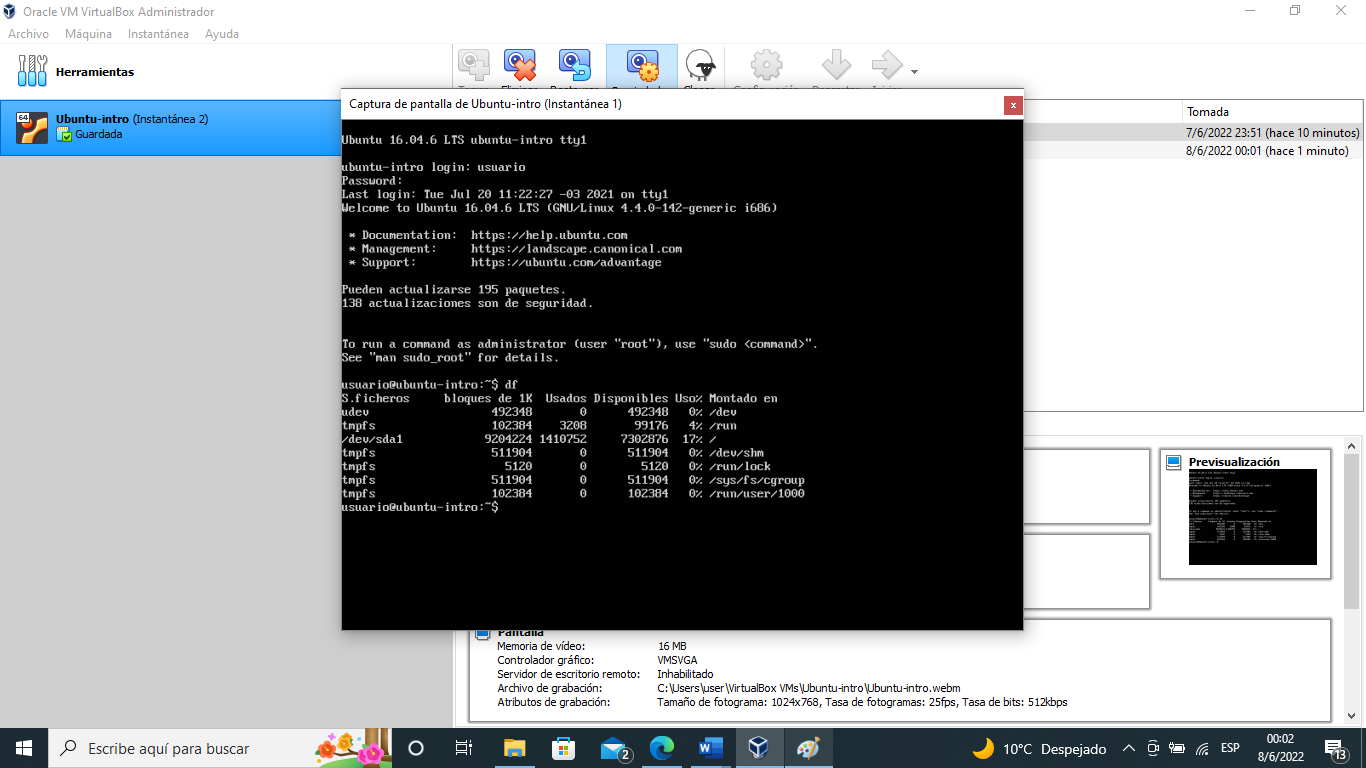
Ejercitación

En las mesas de trabajo debemos resolver los siguientes puntos con nuestra máquina virtual:

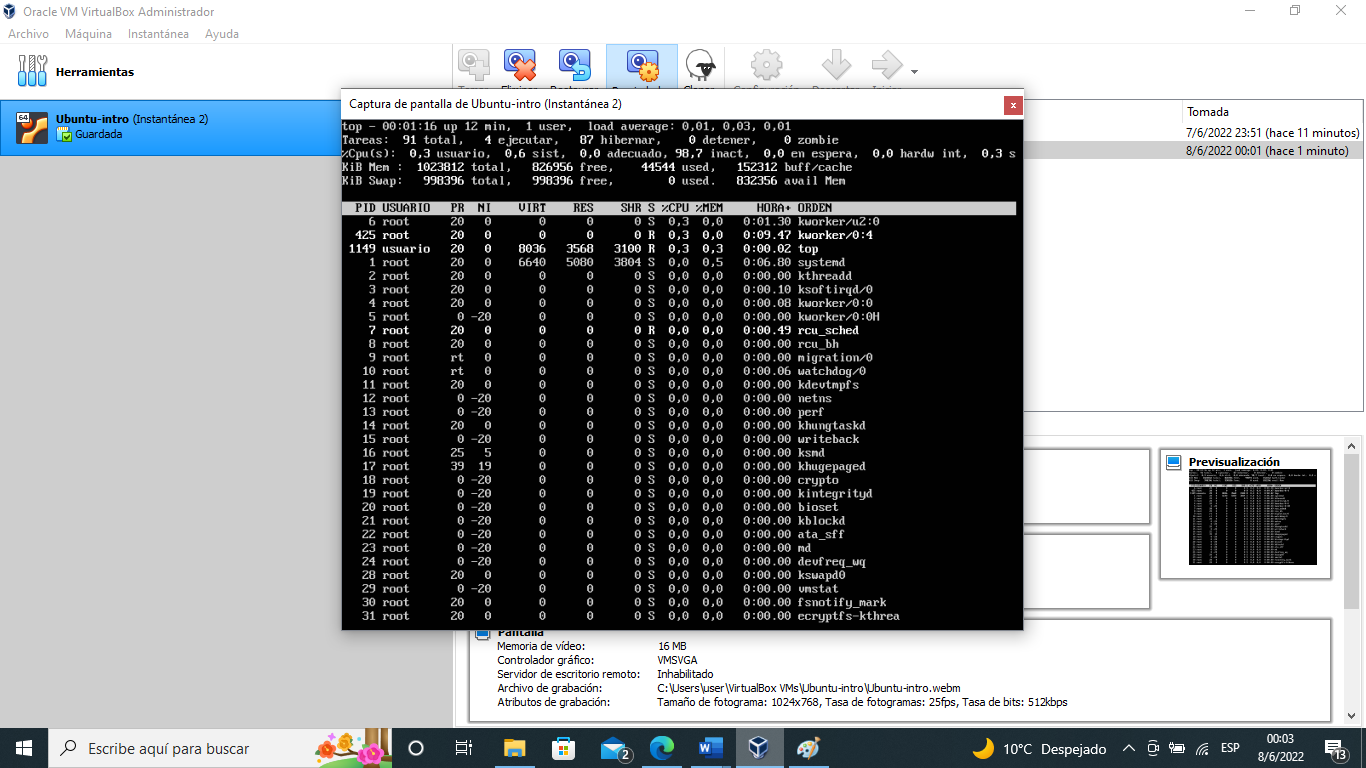
● Crear un archivo en Google Documents o Word en la computadora fuera de la máquina virtual.

Dentro de la máquina virtual:

● Escribir en la terminal el comando df, tomar print de pantalla.



● Escribir en la terminal el comando top, tomar print de pantalla.



● Apagar la máquina virtual con el comando poweroff.

● En base a los print de y comandos. .¿Cuáles son las funciones de estos comandos usados?

df:se utiliza para**mostrar la cantidad de espacio de disco que está libre en los sistemas de archivos**

**top:** Veremos una interfaz de trabajo en modo texto que se va a ir refrescando cada 3 segundos. Esta interfaz nos muestra un resumen del estado de nuestro sistema y la lista de procesos que estan funcionando.

**1. Tiempo de funcionamiento y media de carga del sistema**

En la primera línea nos muestra:

* Hora de hoy.
* El tiempo que ha estado el sistema encendido.
* Número de individuos haciendo uso (root).
* Además de mostrar el promedio de carga media en minutos.

**2. Tareas**

Tasks: 145 total, 1 running, 142 sleeping, 1 stopped, 1 zombie

La segunda línea nos muestra el total de tareas y procesos, los cuales tienen la posibilidad de estar en diferentes estados. Yo lo tengo en inglés pero perfectamente tu lo podrías tener en español:

* **Running**: Son los procesos que están funcionando en estos instantes o que se encuentran preparados para ejecutarse.
* **Sleeping**: Son los procesos dormidos aguardando que ocurra algo para ejecutarse.
* **Stopped**: Son los procesos los cuales su ejecución ha sido detenida.
* **Zombie**: Cuándo el proceso no esta en funcionamiento. Estos procesos se quedan en este estado cuando el desarrollo que los comenzó muere (padre).

**3. Estados de la CPU**

%Cpu(s): 0.1 us, 0.2 sy, 0.0 ni, 99.2 id, 0.0 wa, 0.0 hi, 0.2 si, 0.3 st

* **us**: tiempo de CPU de usuario.
* **sy**: tiempo de  de CPU del kernel.
* **id**: tiempo de CPU en procesos inactivos.
* **wa**: tiempo de espera para los procesos de la CP.
* **hi**: interrupciones de hardware.
* **si**: tiempo de CPU empleado en interrupciones a nivel de software.

**4. Memoria física**

KiB Mem : 8008868 total, 5880240 free, 1097232 used, 1031396 buff/cache

* Memoria total.
* Memoria usada.
* Memoria libre.
* Memoria usada por buffer.

**5. Memoria virtual**

KiB Mem : 8008868 total, 5865364 free, 1112092 used, 1031412 buff/cache

* Memoria total.
* Memoria utilizada.
* Memoria libre.
* Memoria en archivos temporales.

**6. Columnas**

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND

9 root 20 0 0 0 0 S 0.3 0.0 54:09.42 rcu\_sched

3749 root 20 0 162252 6760 5092 S 0.3 0.1 0:00.34 sshd

3794 root 20 0 113180 1576 1320 S 0.3 0.0 0:00.19 bash

4118 root 20 0 62816 16808 2736 S 0.3 0.2 3:10.58 tailwatchd

4248 root 20 0 160100 2228 1556 R 0.3 0.0 0:00.02 top

14217 root 20 0 587276 37856 29112 S 0.3 0.5 0:52.72 php-fpm

18905 mysql 20 0 1638776 179132 9116 S 0.3 2.2 50:59.11 mysqld

1 root 20 0 191144 4108 2604 S 0.0 0.1 5:44.43 systemd

2 root 20 0 0 0 0 S 0.0 0.0 0:00.23 kthreadd

3 root 20 0 0 0 0 S 0.0 0.0 0:18.12 ksoftirqd/0

5 root 0 -20 0 0 0 S 0.0 0.0 0:00.00 kworker/0:0H

7 root rt 0 0 0 0 S 0.0 0.0 0:04.05 migration/0

8 root 20 0 0 0 0 S 0.0 0.0 0:00.00 rcu\_bh

10 root 0 -20 0 0 0 S 0.0 0.0 0:00.00 lru-add-drain

11 root rt 0 0 0 0 S 0.0 0.0 0:01.77 watchdog/0

12 root rt 0 0 0 0 S 0.0 0.0 0:01.45 watchdog/1

13 root rt 0 0 0 0 S 0.0 0.0 0:03.78 migration/1

14 root 20 0 0 0 0 S 0.0 0.0 0:20.37 ksoftirqd/1

16 root 0 -20 0 0 0 S 0.0 0.0 0:00.00 kworker/1:0H

17 root rt 0 0 0 0 S 0.0 0.0 0:01.95 watchdog/2

18 root rt 0 0 0 0 S 0.0 0.0 0:02.82 migration/2

19 root 20 0 0 0 0 S 0.0 0.0 0:14.93 ksoftirqd/2

21 root 0 -20 0 0 0 S 0.0 0.0 0:00.00 kworker/2:0H

22 root rt 0 0 0 0 S 0.0 0.0 0:01.42 watchdog/3

23 root rt 0 0 0 0 S 0.0 0.0 0:06.48 migration/3

* **PID**: es el identificador de desarrollo. Cada desarrollo tiene un identificador exclusivo.
* **USER (USUARIO)**: usuario dueño del desarrollo.
* **PR**: prioridad del desarrollo. Si pone RT es que se está corriendo en el mismo instante.
* **NI**: asigna la prioridad. Si tiene un valor bajo (hasta -20) significa que tiene más prioridad que otro con valor prominente (hasta 19).
* **VIRT**: proporción de memoria virtual usada por el desarrollo.
* **RES**: proporción de memoria RAM física que usa el desarrollo.
* **SHR**: memoria compartida.
* **S (ESTADO)**: estado del desarrollo.
* **%CPU**: porcentaje de CPU central de procesamiento usado desde la más reciente renovación.
* **%MEM**: porcentaje de memoria física usada por el desarrollo desde la más reciente renovación.
* **TIME+ (HORA+)**: tiempo **total**de CPU de procesamiento que ha utilizado el desarrollo desde su inicio.
* **COMMAND**: comando usado para comenzar el desarrollo.