

Parcial 1 Sistemas Operativos

Brayan Stiven Tigreros Gonzalez

Primera Parte

Describe los estados de los procesos.

R//

Los estados de los procesos son:

- **En ejecución:** Proceso que se encuentra en ejecución usando la CPU
- **En espera:** Proceso que está esperando un evento
- **Detenido:** Proceso que fue pausado, ejemplo: Ctrl + Z
- **Zombi:** Proceso que ya terminó su ejecución, pero cuya entrada en la tabla de procesos aún no ha sido eliminada porque el proceso padre no ha leído su estado de salida
- **Muerto:** Proceso que ha sido terminado y ha sido eliminado del todo, no suele ser visible para los usuarios

Con sus propias palabras explique que es Kernel Monolítico y de varios ejemplos del mismo.

R//

Un kernel monolítico es uno de los tipos de núcleo del sistema operativo, el cual se encarga de controlar y administrar los servicios y peticiones de recursos y hardware con respecto a uno o varios procesos pero con la diferencia de que todo el sistema operativo está estructurado en un solo bloque, lo que lo hace mas rapido en ejecución por la comunicación directa entre sus componentes pero lo hace más sensible a fallos que puedan afectar todo el sistema

Segunda Parte

Para un sistema informático la denominación UEFI a que se refiere(conceptúe con sus propias palabra, no chat gpt)

R//

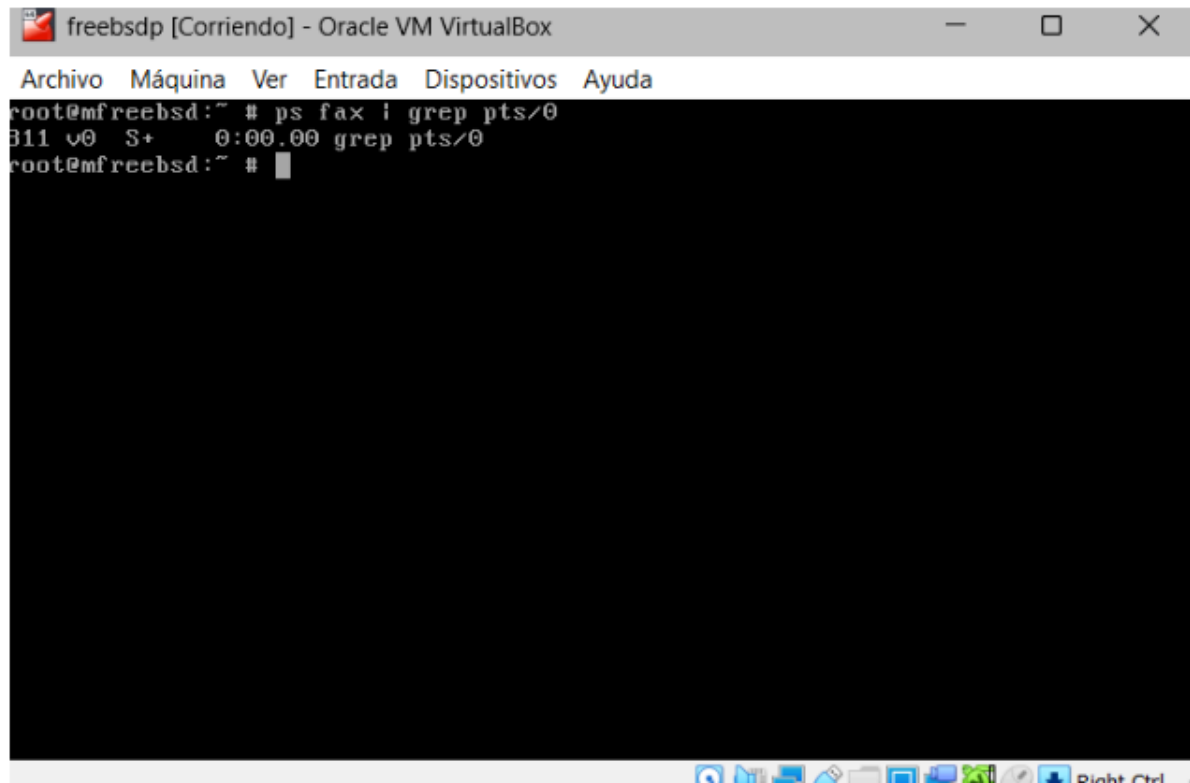
UEFI es el programa básico que inicia el computador, es como el motor de arranque para la PC, es lo que asegura que el hardware esté bien, configura el equipo como discos duros, memoria y tarjetas de video, y luego entrega el sistema operativo para comenzar.

UEFI es básicamente el enlace inteligente entre el hardware de la computadora y el sistema operativo, asegurándose de que todo salga bien y sin problemas.

Interprete el resultado de ps fax | grep pts/0 y muestre en con imagen.

R//

ps fax | grep pts/0

A screenshot of a terminal window titled 'freebsd [Corriendo] - Oracle VM VirtualBox'. The terminal shows the command 'ps fax | grep pts/0' being executed. The output is '311 v0 S+ 0:00.00 grep pts/0'. The terminal has a menu bar with 'Archivo', 'Máquina', 'Ver', 'Entrada', 'Dispositivos', and 'Ayuda'. The prompt is 'root@freebsd:~ #'.

```
freebsd [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
root@freebsd:~ # ps fax | grep pts/0
311 v0 S+ 0:00.00 grep pts/0
root@freebsd:~ #
```

311	Es el número de PID
v0	Es la terminal asociada al proceso
S+	Es el estado del proceso y + indica que esta de primero en la terminal
0:00:00	Tiempo que la CPU consume el proceso
grep pts/0	Nombre del proceso

Tercera Parte

Establezca la diferencia entre Proceso, Servicio, Daemon.

R//

Proceso:

Es cualquier programa en ejecución. Por ejemplo, si abres el navegador, se crea un proceso en memoria para que funcione.

Servicio:

Es un proceso especial que normalmente corre en segundo plano y brinda una función al sistema o a los usuarios (ejemplo: el servicio de red o de impresión).

Daemon:

Es un tipo de servicio en Linux/Unix que se ejecuta en segundo plano sin necesidad de interacción directa del usuario. Suelen iniciarse al arrancar el sistema

Si pudiéramos dar un resumen de todo esto sería que, proceso es lo general, servicio es un proceso que atiende funciones específicas y Daemon es un servicio en segundo plano característico de Linux

Establezca la Diferencia entre Script y Programa

R//

Script:

Es un archivo de texto con una serie de instrucciones escritas en un lenguaje interpretado (como Python). Se ejecuta línea por línea con un intérprete.

Programa:

Es un conjunto de instrucciones escritas en un lenguaje compilado o interpretado, que usualmente requiere ser compilado antes de ejecutarse y puede ser más complejo que un script.

Establezca la Diferencia entre JOBS y Proceso

R//

Proceso:

Es la ejecución de un programa en el sistema, con su propio espacio de memoria, PID (Process ID) y recursos asignados.

Jobs:

Es una tarea que el usuario lanza desde la terminal, y puede estar en **primer plano** o en **segundo plano** (background). Un job es una manera de manejar procesos dentro de la shell.

Cuales son los tipos de daemons en linux

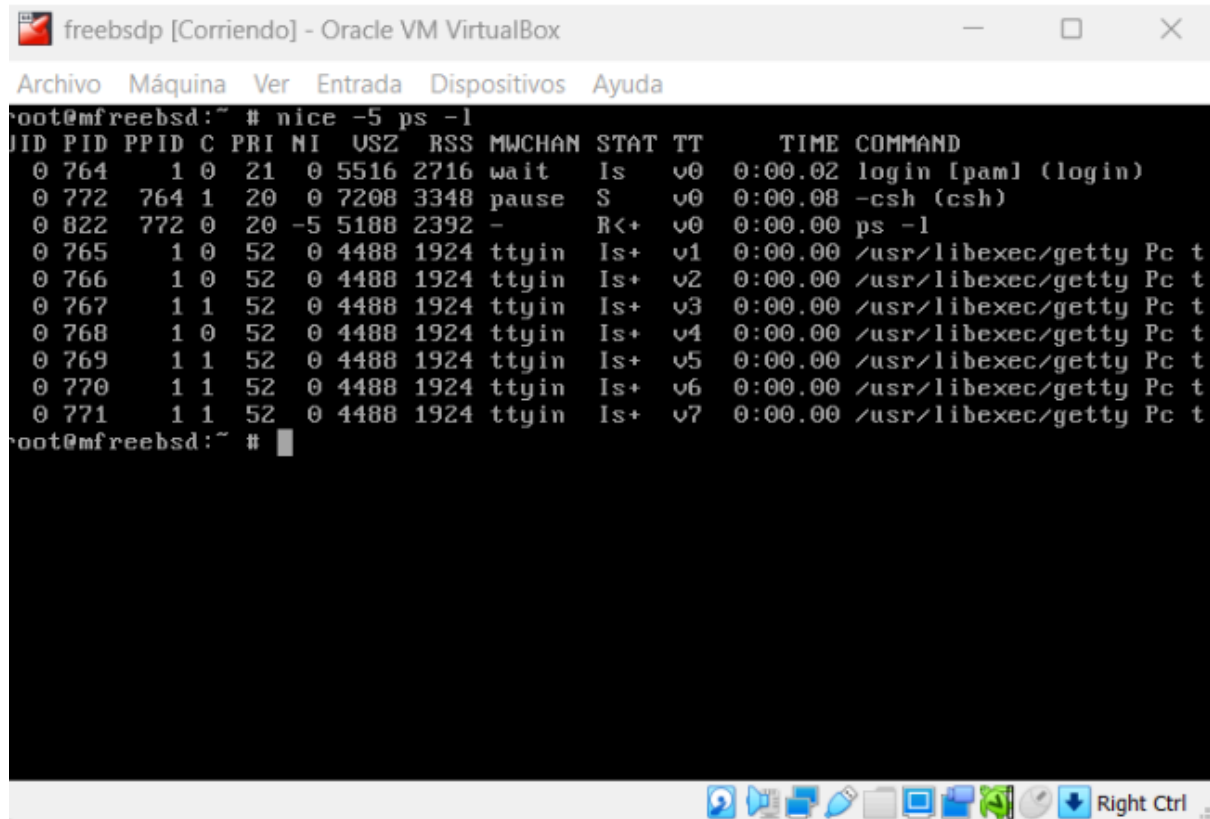
R//

- Daemons del sistema
- Daemons de red
- Daemons de impresión
- Daemons de registro
- Daemons de cron y tareas programadas

Evidencie un cambio de prioridades con el comando nice(adjunte imagen)

R//

nice -5 ps -l



```
freebsd [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
root@freebsd:~ # nice -5 ps -l
UID PID PPID C PRI NI   USZ  RSS  MWCHAN  STAT  TT      TIME  COMMAND
0 764  1 0 21 0 5516 2716 wait  Is   v0  0:00.02 login [pam] (login)
0 772 764 1 20 0 7208 3348 pause  S    v0  0:00.08 -csh (csh)
0 822 772 0 20 -5 5188 2392 -      R<+  v0  0:00.00 ps -l
0 765  1 0 52 0 4488 1924 ttyin  Is+   v1  0:00.00 /usr/libexec/getty Pc t
0 766  1 0 52 0 4488 1924 ttyin  Is+   v2  0:00.00 /usr/libexec/getty Pc t
0 767  1 1 52 0 4488 1924 ttyin  Is+   v3  0:00.00 /usr/libexec/getty Pc t
0 768  1 0 52 0 4488 1924 ttyin  Is+   v4  0:00.00 /usr/libexec/getty Pc t
0 769  1 1 52 0 4488 1924 ttyin  Is+   v5  0:00.00 /usr/libexec/getty Pc t
0 770  1 1 52 0 4488 1924 ttyin  Is+   v6  0:00.00 /usr/libexec/getty Pc t
0 771  1 1 52 0 4488 1924 ttyin  Is+   v7  0:00.00 /usr/libexec/getty Pc t
root@freebsd:~ #
```

Ejecuta el comando ps -l con un valor de nice = -5, lo cual le da más prioridad al proceso.

El valor nice va de -20 (mayor prioridad) a 19 (menor prioridad).

Esto permite que un proceso tenga preferencia en el uso de CPU.

Cuarta Parte

Se implementaron unos servidores los cuales están montados sobre un sistema operativo Solaris. Pero se han estado presentado problemas de desempeño, usted debe analizar dichos problemas aplicando una serie de conocimientos de su clase de sistemas operativos.

1.1. **ifconfig**

Muestra la configuración de las interfaces de red, incluyendo la dirección IP y el estado de la conexión.

1.2. **uptime**

Indica el tiempo que lleva encendido el sistema, el número de usuarios conectados y la carga promedio.

1.3. **format**

Presenta cómo está particionado el disco, mostrando tamaños y tipos de

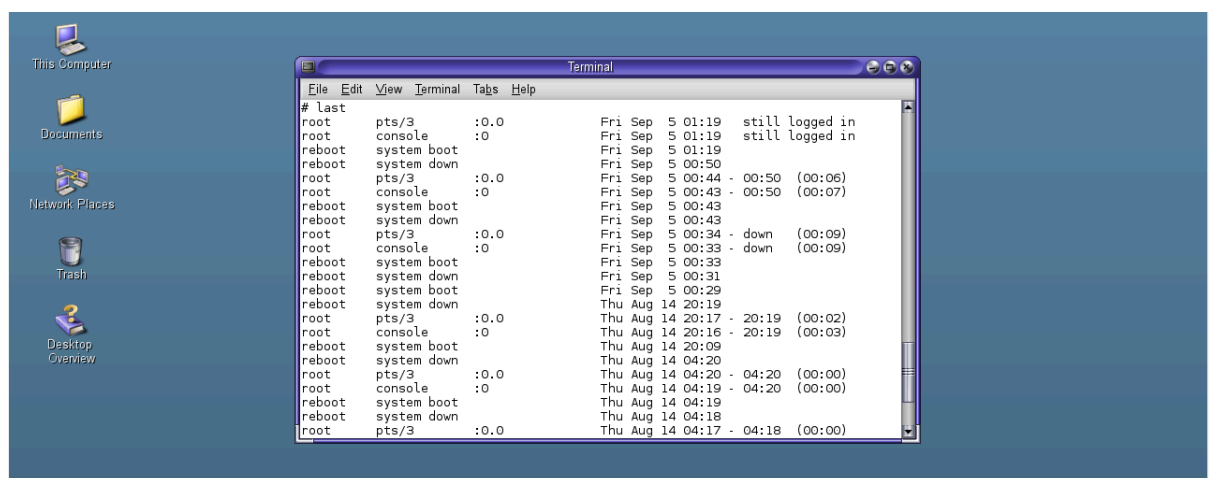
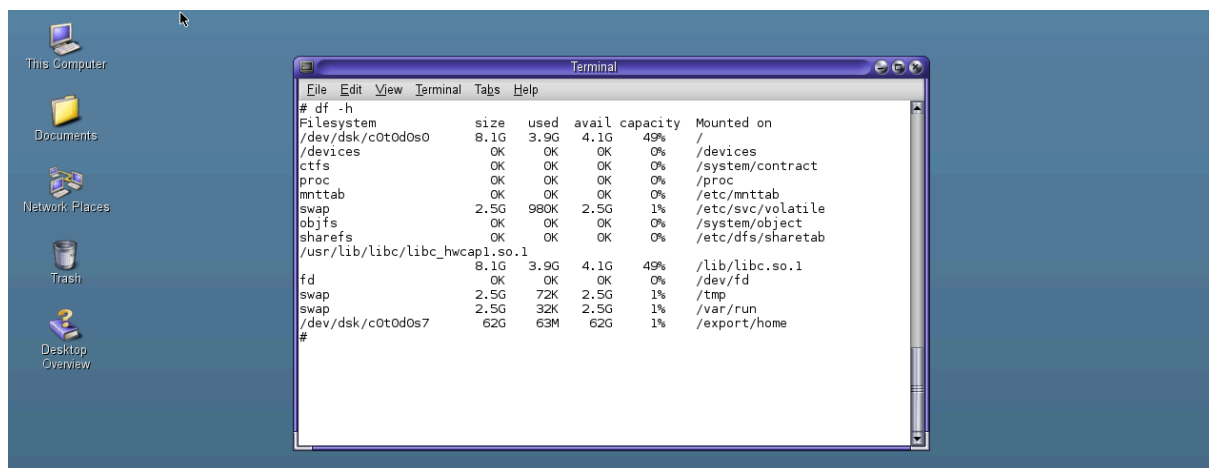
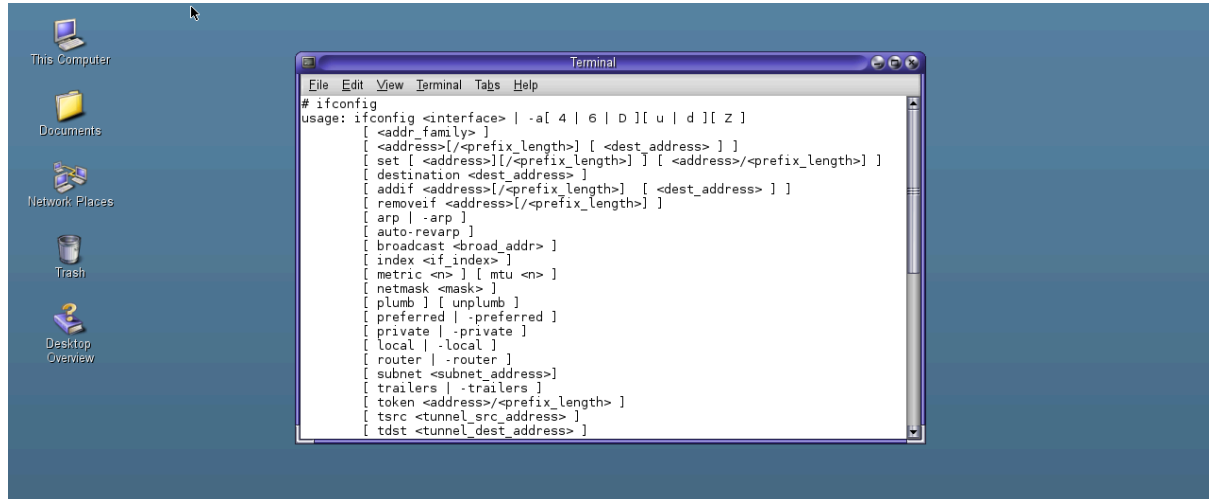
particiones.

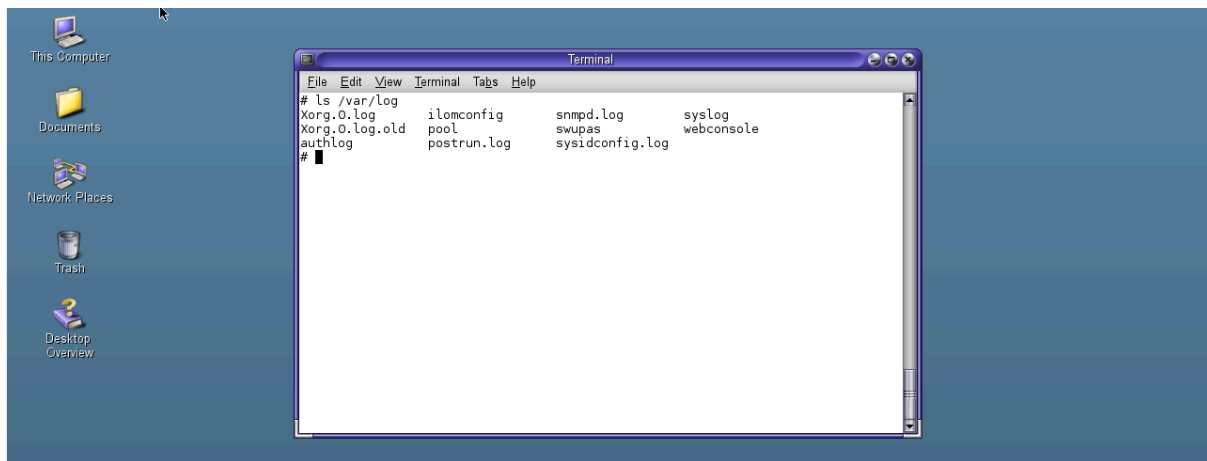
- 1.4. **df -h**
Informa el uso del disco duro y el espacio disponible en cada partición en formato legible (GB/MB).
- 1.5. **ps -ef**
Lista todos los procesos en ejecución con detalles de usuario, PID, consumo de CPU y memoria.
- 1.6. **who**
Muestra los usuarios que están conectados actualmente al sistema.
- 1.7. **w**
Detalla los usuarios conectados y qué procesos están ejecutando.
- 1.8. **netstat -an -f inet**
Lista las conexiones activas y los puertos abiertos en IPv4.
- 1.9. **netstat -an**
Presenta todas las conexiones de red y puertos en escucha (listening).
- 1.10. **last**
Muestra el historial de accesos de los usuarios al sistema, incluyendo conexiones y cierres de sesión.
- 1.11. **vmstat -P**
Genera estadísticas de memoria, procesos y CPU, y las guarda en un archivo de log.
- 1.12. **ls /var/log**
Lista los archivos de registro del sistema que permiten monitorear la actividad y errores.
- 1.13. **wc -l /var/log/webconsole**
Cuenta cuántas líneas tiene el archivo de log seleccionado (en este caso, webconsole).
- 1.14. **swap -l**
Indica el estado de la memoria de intercambio (swap), mostrando cuánto se usa y cuánto está libre.

1.15. man ps

Abre el manual del comando ps para consultar sus diferentes opciones y ejemplos de uso.

Algunos ejemplos de los comandos en solaris





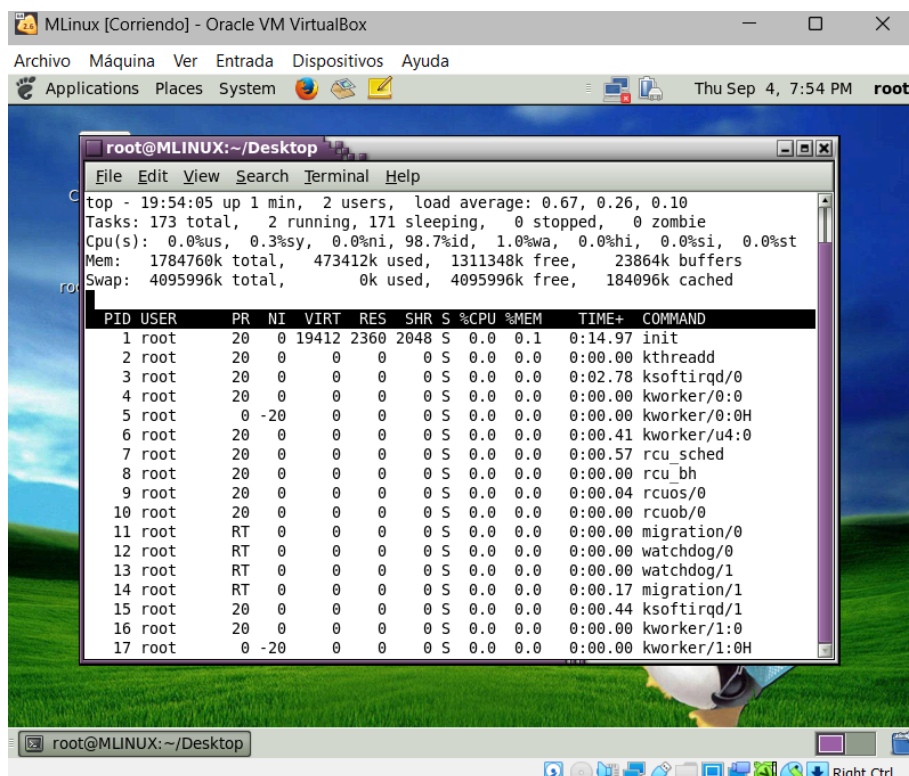
Fastboot

Reinicia el sistema inmediatamente, sin ejecutar la verificación de discos (fsck). Se utiliza cuando se necesita un reinicio rápido, pero no es recomendable en servidores en producción porque puede omitir la revisión de errores en los discos.

Init 5

Apaga el sistema de forma segura e inmediata es el equivalente a **shutdown -p now** en **freebsd**

En su máquina Oracle Linux y lance el comando **top** y describa cada uno de los ÍTEMS que aparecen en cada fila hasta la columna 6(OJO) revisen bien filas y columnas. Muestre la evidencia desde su máquina BSD(imagen)



1. Process ID (PID)

1.1. Es el número único que el sistema asigna a cada proceso en ejecución.

2. USER

2.1. Usuario propietario del proceso.

3. PR (Priority – Prioridad)

3.1. Prioridad del proceso según el planificador de Linux.

3.2. Un número más bajo significa mayor prioridad.

4. NI (Nice value – Valor de cortesía)

4.1. Es un ajuste que permite al usuario modificar la prioridad del proceso.

4.2. Rango: -20 (alta prioridad) hasta +19 (baja prioridad).

5. VIRT (Virtual memory – Memoria virtual)

5.1. Cantidad total de memoria virtual usada por el proceso.

5.2. Incluye memoria usada + memoria mapeada a disco + librerías compartidas.

6. RES (Resident memory – Memoria residente)

6.1. Es la parte de la memoria física (RAM) que el proceso tiene ocupada en este momento.

Evidencien los resultados de los siguientes comandos:

ps auxf

Muestra todos los procesos del sistema con detalle estilo BSD (aux) y además en formato de árbol jerárquico (f) para ver relaciones padre-hijo


```
root@MLINUX:~/Desktop
File Edit View Search Terminal Help

[root@MLINUX Desktop]# ps auxf
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         2  0.0  0.0      0     0 ?        S    19:52   0:00 [kthreadd]
root         3  0.4  0.0      0     0 ?        S    19:52   0:02 \_ [ksoftirqd]
root         5  0.0  0.0      0     0 ?        S<   19:52   0:00 \_ [kworker/0]
root         6  0.0  0.0      0     0 ?        S    19:52   0:00 \_ [kworker/u]
root         7  0.0  0.0      0     0 ?        S    19:52   0:00 \_ [rcu_sched]
root         8  0.0  0.0      0     0 ?        S    19:52   0:00 \_ [rcu_bh]
root         9  0.0  0.0      0     0 ?        S    19:52   0:00 \_ [rcuos/0]
root        10  0.0  0.0      0     0 ?        S    19:52   0:00 \_ [rcuob/0]
root        11  0.0  0.0      0     0 ?        S    19:52   0:00 \_ [migration]
root        12  0.0  0.0      0     0 ?        S    19:52   0:00 \_ [watchdog/]
root        13  0.0  0.0      0     0 ?        S    19:52   0:00 \_ [watchdog/]
root        14  0.0  0.0      0     0 ?        S    19:52   0:00 \_ [migration]
root        15  0.0  0.0      0     0 ?        S    19:52   0:00 \_ [ksoftirqd]
root        17  0.0  0.0      0     0 ?        S<   19:52   0:00 \_ [kworker/1]
root        18  0.0  0.0      0     0 ?        S    19:52   0:00 \_ [rcuos/1]
root        19  0.0  0.0      0     0 ?        S    19:52   0:00 \_ [rcuob/1]
root        20  2.0  0.0      0     0 ?        S    19:52   0:14 \_ [kworker/0]
root        21  0.0  0.0      0     0 ?        S<   19:52   0:00 \_ [khelper]
root        22  0.0  0.0      0     0 ?        S    19:52   0:00 \_ [kdevtmpfs]
root        23  0.0  0.0      0     0 ?        S<   19:52   0:00 \_ [netns]
root        24  0.0  0.0      0     0 ?        S<   19:52   0:00 \_ [perf]
```

ps -l

Muestra los procesos en formato larga lista, con campos adicionales como F (flags), S (estado), PRI (prioridad), NI (nice), ADDR, etc

```
[root@MLINUX Desktop]# ps -l
F S   UID    PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
4 S   0    2767  2765  0  80   0 - 27104 do_wai pts/0    00:00:00 bash
0 R   0    2835  2767  0  80   0 - 27041 -      pts/0    00:00:00 ps
[root@MLINUX Desktop]#
```

ps j

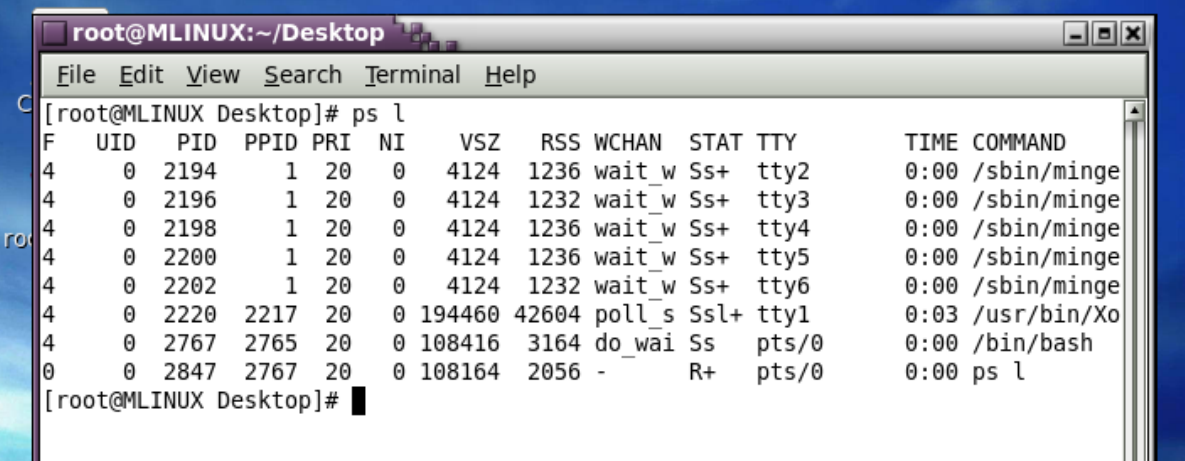
Muestra procesos en formato job control, incluyendo columnas como PPID (proceso padre), PGID (grupo de procesos), SID (session ID) y estado.

```
root@MLINUX:~/Desktop
File Edit View Search Terminal Help

[root@MLINUX Desktop]# ps j
PPID  PID  PGID  SID  TTY          TPGID STAT   UID   TIME COMMAND
    1  2194  2194  2194 tty2          2194 Ss+    0     0:00 /sbin/mingetty /dev/tty
    1  2196  2196  2196 tty3          2196 Ss+    0     0:00 /sbin/mingetty /dev/tty
    1  2198  2198  2198 tty4          2198 Ss+    0     0:00 /sbin/mingetty /dev/tty
    1  2200  2200  2200 tty5          2200 Ss+    0     0:00 /sbin/mingetty /dev/tty
    1  2202  2202  2202 tty6          2202 Ss+    0     0:00 /sbin/mingetty /dev/tty
  2217 2220  2220  2220 tty1          2220 Ssl+   0     0:03 /usr/bin/Xorg :0 -br -v
  2765 2767  2767  2767 pts/0         2838 Ss     0     0:00 /bin/bash
  2767 2838  2838  2767 pts/0         2838 R+     0     0:00 ps j
[root@MLINUX Desktop]#
```

ps l

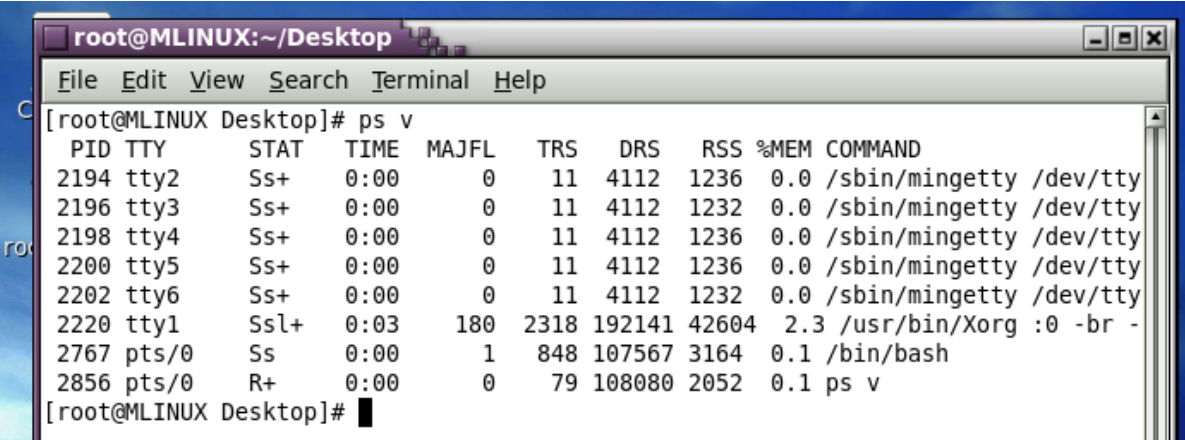
Similar a ps -l pero estilo BSD, también muestra información extendida de los procesos.



```
root@MLINUX:~/Desktop
File Edit View Search Terminal Help
[root@MLINUX Desktop]# ps l
 F  UID    PID  PPID  PRI  NI     VSZ   RSS  WCHAN  STAT TTY          TIME COMMAND
 4    0   2194     1   20    0    4124   1236 wait_w  Ss+  tty2        0:00 /sbin/minge
 4    0   2196     1   20    0    4124   1232 wait_w  Ss+  tty3        0:00 /sbin/minge
 4    0   2198     1   20    0    4124   1236 wait_w  Ss+  tty4        0:00 /sbin/minge
 4    0   2200     1   20    0    4124   1236 wait_w  Ss+  tty5        0:00 /sbin/minge
 4    0   2202     1   20    0    4124   1232 wait_w  Ss+  tty6        0:00 /sbin/minge
 4    0   2220   2217   20    0 194460 42604 poll_s  Ssl+ tty1        0:03 /usr/bin/Xo
 4    0   2767   2765   20    0 108416   3164 do_wai  Ss    pts/0      0:00 /bin/bash
 0    0   2847   2767   20    0 108164   2056 -       R+    pts/0      0:00 ps l
[root@MLINUX Desktop]#
```

ps v

Formato orientado a memoria, incluye campos como SIZE, RSS, SHARE, STAT, etc. Sirve para analizar el uso de memoria de procesos.



```
root@MLINUX:~/Desktop
File Edit View Search Terminal Help
[root@MLINUX Desktop]# ps v
  PID TTY          STAT TIME  MAJFL  TRS   DRS   RSS %MEM COMMAND
 2194 tty2        Ss+   0:00      0    11  4112  1236  0.0 /sbin/mingetty /dev/tty
 2196 tty3        Ss+   0:00      0    11  4112  1232  0.0 /sbin/mingetty /dev/tty
 2198 tty4        Ss+   0:00      0    11  4112  1236  0.0 /sbin/mingetty /dev/tty
 2200 tty5        Ss+   0:00      0    11  4112  1236  0.0 /sbin/mingetty /dev/tty
 2202 tty6        Ss+   0:00      0    11  4112  1232  0.0 /sbin/mingetty /dev/tty
 2220 tty1        Ssl+  0:03   180 2318 192141 42604  2.3 /usr/bin/Xorg :0 -br -
 2767 pts/0        Ss    0:00      1    848 107567 3164  0.1 /bin/bash
 2856 pts/0        R+    0:00      0     79 108080 2052  0.1 ps v
[root@MLINUX Desktop]#
```

ps u

Muestra procesos en formato de usuario, con el nombre del usuario, %CPU, %MEM, tiempo, comando, etc. Es más legible para administradores.

```
[root@MLINUX Desktop]# ps u
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	2194	0.0	0.0	4124	1236	tty2	Ss+	19:53	0:00	/sbin/mingetty
root	2196	0.0	0.0	4124	1232	tty3	Ss+	19:53	0:00	/sbin/mingetty
root	2198	0.0	0.0	4124	1236	tty4	Ss+	19:53	0:00	/sbin/mingetty
root	2200	0.0	0.0	4124	1236	tty5	Ss+	19:53	0:00	/sbin/mingetty
root	2202	0.0	0.0	4124	1232	tty6	Ss+	19:53	0:00	/sbin/mingetty
root	2220	0.3	2.3	194460	42604	tty1	Ssl+	19:53	0:03	/usr/bin/Xorg :
root	2767	0.0	0.1	108416	3164	pts/0	Ss	19:53	0:00	/bin/bash
root	2857	0.0	0.1	110264	2236	pts/0	R+	20:10	0:00	ps u

```
[root@MLINUX Desktop]#
```

ps -e

Lista todos los procesos en el sistema (equivalente a ps -A).

```
root@MLINUX:~/Desktop
```

File Edit View Search Terminal Help

```
[root@MLINUX Desktop]# ps -e
```

PID	TTY	TIME	CMD
1	?	00:00:00	init
2	?	00:00:00	kthreadd
3	?	00:00:02	ksoftirqd/0
5	?	00:00:00	kworker/0:0H
6	?	00:00:00	kworker/u4:0
7	?	00:00:00	rcu_sched
8	?	00:00:00	rcu_bh
9	?	00:00:00	rcuos/0
10	?	00:00:00	rcuob/0
11	?	00:00:00	migration/0
12	?	00:00:00	watchdog/0
13	?	00:00:00	watchdog/1
14	?	00:00:00	migration/1
15	?	00:00:00	ksoftirqd/1
17	?	00:00:00	kworker/1:0H
18	?	00:00:00	rcuos/1
19	?	00:00:00	rcuob/1
20	?	00:00:14	kworker/0:1
21	?	00:00:00	khelper
22	?	00:00:00	kdevtmpfs
23	?	00:00:00	netns

ps -C system

Muestra procesos cuyo nombre del comando sea exactamente system.

```
[root@MLINUX Desktop]# ps -C system
```

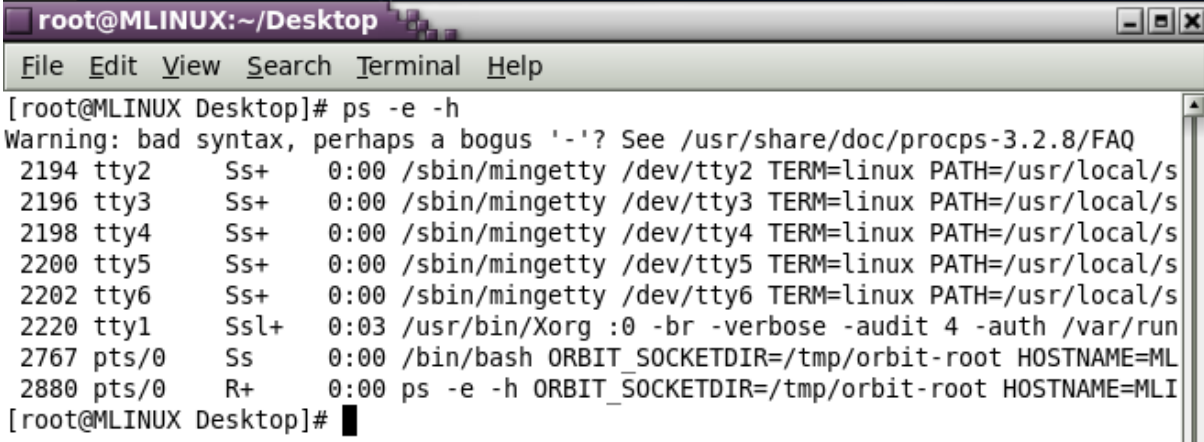
PID	TTY	TIME	CMD
-----	-----	------	-----

```
[root@MLINUX Desktop]#
```

ps -e -h

Lista todos los procesos (-e) pero sin encabezados (-h).

Útil si quieres procesar la salida con otro comando (ej: grep, awk).



```
root@MLINUX:~/Desktop
File Edit View Search Terminal Help
[root@MLINUX Desktop]# ps -e -h
Warning: bad syntax, perhaps a bogus '-'? See /usr/share/doc/procps-3.2.8/FAQ
2194 tty2      Ss+    0:00 /sbin/mingetty /dev/tty2 TERM=linux PATH=/usr/local/s
2196 tty3      Ss+    0:00 /sbin/mingetty /dev/tty3 TERM=linux PATH=/usr/local/s
2198 tty4      Ss+    0:00 /sbin/mingetty /dev/tty4 TERM=linux PATH=/usr/local/s
2200 tty5      Ss+    0:00 /sbin/mingetty /dev/tty5 TERM=linux PATH=/usr/local/s
2202 tty6      Ss+    0:00 /sbin/mingetty /dev/tty6 TERM=linux PATH=/usr/local/s
2220 tty1      Ssl+   0:03 /usr/bin/Xorg :0 -br -verbose -audit 4 -auth /var/run
2767 pts/0     Ss      0:00 /bin/bash ORBIT_SOCKETDIR=/tmp/orbit-root HOSTNAME=ML
2880 pts/0     R+      0:00 ps -e -h ORBIT_SOCKETDIR=/tmp/orbit-root HOSTNAME=MLI
[root@MLINUX Desktop]#
```