

CONSIDERACIONES IMPORTANTES

TPs S.O

COMANDOS TP 1

★ `open(AT_FDCWD, ".", O_RDONLY|O_NONBLOCK|O_CLOEXEC|O_DIRECTORY)=3`

Es una llamada al sistema `open()` que intenta abrir el directorio actual "." (indicado con `AT_FDCWD`), con las opciones:

- `O_RDONLY`: abre el archivo en modo solo lectura.
- `O_NONBLOCK`: intenta abrir el archivo de manera no bloqueante (aunque no siempre es relevante para directorios).
- `O_CLOEXEC`: asegura que el descriptor de archivo se cierre automáticamente si se ejecuta un nuevo programa.
- `O_DIRECTORY`: especifica que se espera abrir un directorio y no un archivo regular.

✓ `=3`: el retorno de la llamada es 3, lo que significa que el descriptor de archivo 3 se ha abierto correctamente.

★ `getdents64(3, 0x58134bca7ab0 /* 24 entries */, 32768)=768`: Sirve para leer entradas en el directorio.

- 3: Descriptor de archivo del directorio, previamente abierto con `open()`.
- `0x58134bca7ab0`: Dirección en memoria del búfer donde `getdents64` almacena las entradas.
- `/* 24 entries */`: 24 entradas de directorio leídas.
- 32768: Tamaño del búfer (en bytes).
- `=768`: 768 bytes fueron leídos y almacenados en el búfer.

★ `write()`: Para escribir la salida a la terminal.

★ `statx(AT_FDCWD, ".", AT_STATX_SYNC_AS_STAT|AT_SYMLINK_NOFOLLOW|AT_NO_AUTOMOUNT, STATX_MODE|STATX_NLINK|STATX_UID|STATX_GID|STATX_MTIME|STATX_SIZE, {stx_mask=STATX_BASIC_STATS|STATX_MNT_ID, stx_attributes=0, stx_mode=S_IFDIR|0755, stx_size=4096, ...})=0`

✓ La función `statx` obtiene información detallada sobre un archivo o directorio en Linux.

✓ `AT_FDCWD`: Indica que el directorio de referencia es el actual.

- `(".", "/*`: Nombre del archivo o directorio sobre el que se consulta (en este caso, el directorio padre ..).
- `AT_STATX_SYNC_AS_STAT|AT_SYMLINK_NOFOLLOW|AT_NO_AUTOMOUNT`: Opciones que controlan el comportamiento de `statx`, como no seguir enlaces simbólicos (`SYMLINK_NOFOLLOW`) y no montar automáticamente (`NO_AUTOMOUNT`).
- `STATX_MODE|STATX_NLINK|STATX_UID|STATX_GID|STATX_MTIME|STATX_SIZE`: Campos específicos a recuperar, como permisos (`MODE`), número de enlaces (`NLINK`), ID de usuario (`UID`), ID de grupo (`GID`), marca de tiempo de última modificación (`MTIME`) y tamaño (`SIZE`).
- `{...}`: Estructura donde se almacenan los datos obtenidos. Algunos valores dentro:
 - `stx_mask=STATX_BASIC_STATS|STATX_MNT_ID`: Máscara de campos llenados.
 - `stx_mode=S_IFDIR|0755`: Tipo de archivo y permisos (directorio con permisos 0755).
 - `stx_size=4096`: Tamaño en bytes (en este caso, 4096).

✓ `=0`: Retorno que indica éxito en la operación.

★ `read()`: para leer contenido de archivos.

★ `close()`: para cerrar archivos.

COMANDOS TP 2

★ `ps`: Muestra información sobre los procesos en ejecución en el sistema. Es útil para ver qué procesos están activos y obtener detalles básicos como:

- `PID`: Identificador único del proceso.
- `TTY`: Terminal asociado al proceso (si aplica).
- `TIME`: Tiempo total de CPU consumido por el proceso.
- `CMD`: Comando que inició el proceso.

★ `ps -e`: Muestra todos los procesos en el sistema.

★ `ps -ef`: Muestra todos los procesos con información detallada, incluyendo el usuario, el PID, el PPID, el tiempo de inicio y el comando completo.

✓ Sus salidas son:

- `UID`: Usuario que ejecuta el proceso.
- `PID`: Identificador único del proceso.
- `PPID`: ID del proceso padre.
- `C`: Porcentaje de uso de CPU.
- `STIME`: Hora de inicio del proceso.
- `TTY`: Terminal asociado (si existe).
- `TIME`: Tiempo total de CPU utilizado.

```
facuu6@ubuntu6:~$ ps
  PID TTY          TIME CMD
 2680 pts/0    00:00:00 bash
 3011 pts/0    00:00:00 ps
```

```
facuu6@ubuntu6:~$ ps -e
  PID TTY          TIME CMD
    1 ?           00:00:02 systemd
    2 ?           00:00:00 kthreadd
    3 ?           00:00:00 pool_workqueue_release
    4 ?           00:00:00 kworker/R-rcu_g
    5 ?           00:00:00 kworker/R-rcu_p
    6 ?           00:00:00 kworker/R-slub_
    7 ?           00:00:00 kworker/R-netns
   10 ?           00:00:00 kworker/0:0H-kblockd
   12 ?           00:00:00 kworker/R-mm_pe
   13 ?           00:00:00 rcu_tasks_kthread
   14 ?           00:00:00 rcu_tasks_rude_kthread
   15 ?           00:00:00 rcu_tasks_trace_kthread
   16 ?           00:00:00 ksoftirqd/0
   17 ?           00:00:00 rcu_preempt
   18 ?           00:00:00 migration/0
   19 ?           00:00:00 idle_inject/0
   20 ?           00:00:00 cpuhp/0
   21 ?           00:00:00 kdevtmpfs
```

```
facuu6@ubuntu6:~$ ps -ef
UID          PID    PPID  C   STIME TTY          TIME CMD
root         1      0  0  00:42 ?        00:00:02 /sbin/init
root         2      0  0  00:42 ?        00:00:00 [kthreadd]
root         3      2  0  00:42 ?        00:00:00 [pool_workqueue_release]
root         4      2  0  00:42 ?        00:00:00 [kworker/R-rcu_g]
root         5      2  0  00:42 ?        00:00:00 [kworker/R-rcu_p]
root         6      2  0  00:42 ?        00:00:00 [kworker/R-slub_]
root         7      2  0  00:42 ?        00:00:00 [kworker/R-netns]
root        10      2  0  00:42 ?        00:00:00 [kworker/0:0H-kblockd]
root        12      2  0  00:42 ?        00:00:00 [kworker/R-mm_pe]
root        13      2  0  00:42 ?        00:00:00 [rcu_tasks_kthread]
```

- **PID:** Identificador único del proceso.
- **PPID:** ID del proceso padre.
- **C:** Porcentaje de uso de CPU.
- **STIME:** Hora de inicio del proceso.
- **TTY:** Terminal asociado (si existe).
- **TIME:** Tiempo total de CPU utilizado.
- **CMD:** Comando completo usado para iniciar el proceso.

```
root      5      2  0 08:42 ?        00:00:00 [kworker/R-rcu_p]
root      6      2  0 08:42 ?        00:00:00 [kworker/R-slub_]
root      7      2  0 08:42 ?        00:00:00 [kworker/R-netsn]
root     10      2  0 08:42 ?        00:00:00 [kworker/0:0H-kblockd]
root     12      2  0 08:42 ?        00:00:00 [kworker/R-mm_pe]
root     13      2  0 08:42 ?        00:00:00 [rcu_tasks_kthread]
root     14      2  0 08:42 ?        00:00:00 [rcu_tasks_rude_kthread]
root     15      2  0 08:42 ?        00:00:00 [rcu_tasks_trace_kthread]
```

- ★ **ps tree:** Muestra el proceso padre (systemd), y todos sus procesos hijos

```
facuu6@ubuntu6:~$ pstree
systemd───ModemManager───3*[{ModemManager}]
          └─NetworkManager───3*[{NetworkManager}]
             └─accounts-daemon───3*[{accounts-daemon}]
                └─agetty
                   └─avahi-daemon───avahi-daemon
                      └─colord───3*[{colord}]
                         └─cron
                            └─cups-browsed───3*[{cups-browsed}]
                               └─cupsd───dbus
                                  └─dbus-daemon
                                     └─fwupd───5*[{fwupd}]
                                        └─gnome-remote-de───3*[{gnome-remote-de}]
```

- ★ **ps -p<PID> -f:** Funciona como el **ps**, pero en este caso muestra información de un proceso específico (lo especificamos con el <PID>)

```
facuu6@ubuntu6:~$ ps -p2680 -f
UID          PID    PPID  C STIME TTY          TIME CMD
facuu6       2680    2670  0 08:52 pts/0    00:00:00 bash
```

- ★ **top:** Muestra una vista en tiempo real de los procesos activos en el sistema, actualizándose constantemente para reflejar el uso actual de recursos.

- **PID:** Identificador del proceso.
- **USER:** Usuario que ejecuta el proceso.
- **PR:** Prioridad del proceso.
- **NI:** Valor de "nice", que ajusta la prioridad.
- **VIRT:** Memoria virtual total usada por el proceso.
- **RES:** Memoria física (RAM) usada por el proceso.
- **SHR:** Memoria compartida entre procesos.
- **S:** Estado del proceso (S = Dormido, R = En ejecución, Z = Zombi, T = Detenido).
- **%CPU:** Porcentaje de CPU que usa el proceso.
- **%MEM:** Porcentaje de memoria que consume.
- **TIME+:** Tiempo total de CPU usado.
- **COMMAND:** Comando o nombre del proceso.

- ✓ **top** es útil para monitorear el uso de CPU y memoria de cada proceso, así como el rendimiento general del sistema.

```
top - 09:48:23 up 1:06, 1 user, load average: 0.00, 0.00, 0.00
Tasks: 178 total, 2 running, 176 sleeping, 0 stopped, 0 zombie
%Cpu(s): 1,3 us, 1,0 sy, 0,0 ni, 97,7 id, 0,0 wa, 0,0 hi, 0,0 si, 0,0 st
MiB Mem : 1968,2 total, 252,0 free, 913,9 used, 973,0 buff/cache
MiB Swap: 2048,0 total, 2048,0 free, 0,0 used. 1054,3 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM    TIME+  COMMAND
 1248 root        20   0   341556   95908   60244 S   1,0   4,8    0:05.64 Xorg
 1711 facuu6     20   0 3390208 328792 139444 S   1,0  16,3    0:13.49 gnome-s+
 2670 facuu6  20   0   764872   54132   43244 S   0,3   2,7    0:01.63 gnome-t+
 3199 facuu6  20   0   11900    5888    3712 R   0,3   0,3    0:00.02 top
    1 root        20   0   23444    14528   9408 S   0,0   0,7    0:02.14 systemd
    2 root        20   0         0         0      0 S   0,0   0,0    0:00.00 kthreadd
    3 root        20   0         0         0      0 S   0,0   0,0    0:00.00 pool_wo+
    4 root         0 -20         0         0      0 I   0,0   0,0    0:00.00 kworker+
    5 root         0 -20         0         0      0 I   0,0   0,0    0:00.00 kworker+
    6 root         0 -20         0         0      0 I   0,0   0,0    0:00.00 kworker+
    7 root         0 -20         0         0      0 I   0,0   0,0    0:00.00 kworker+
   10 root         0 -20         0         0      0 I   0,0   0,0    0:00.00 kworker+
   12 root         0 -20         0         0      0 I   0,0   0,0    0:00.00 kworker+
   13 root        20   0         0         0      0 I   0,0   0,0    0:00.00 rcu_tas+
   14 root        20   0         0         0      0 I   0,0   0,0    0:00.00 rcu_tas+
   15 root        20   0         0         0      0 I   0,0   0,0    0:00.00 rcu_tas+
   16 root        20   0         0         0      0 S   0,0   0,0    0:00.34 ksofttir
```

- ★ **htop:** Monitorea el sistema en tiempo real.

- **CPU:** Muestra uso de la CPU.
 - Verde: Procesos de usuario.
 - Rojo: Procesos del sistema.
 - Azul: Baja prioridad.
 - Púrpura: Espera de I/O.
- **MEM:** Muestra uso de la RAM.
 - Verde: Aplicaciones.
 - Azul: Caché de disco.
 - Amarillo: Buffers.
- **SWP:** Muestra uso de Swap.
 - Azul: Swap usada.
 - Rojo: Swap en uso reciente.

Ventajas de htop sobre top:

- ✓ Interfaz visual e intuitiva.
- ✓ Navegación y control con el teclado (para filtrar, ordenar, o matar procesos).
- ✓ Colorido y más detalles para el uso de CPU, RAM y Swap.
- ✓ Vista en árbol de procesos, que facilita ver relaciones entre ellos.

```
CPU[|||||] 11.8% Tasks: 106, 318 thr, 73 kthr; 1 running
Mem[|||||] 776M/1.92G Load average: 0.00 0.00 0.00
Swp[|||||] 0K/2.00G Uptime: 01:26:48

Main I/O
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM    TIME+  Command
 1711 facuu6     20   0 3311M   321M   136M S   6.6  16.3    0:15.62 /usr/bin/gnome-shell
 3313 facuu6     20   0   8668   4864   3584 R   2.0   0.2    0:00.61 htop
 1248 root        20   0   338M  99.1M  60656 S   1.3   5.0    0:06.40 /usr/lib/xorg/Xorg -core :0
 1269 root        20   0   338M  99.1M  60656 S   0.7   5.0    0:00.72 /usr/lib/xorg/Xorg -core :0
    1 root        20   0 23444 14528  9408 S   0.0   0.7    0:02.16 /sbin/init
  315 root        20   0 67240 21996 20460 S   0.0   1.1    0:37.95 /usr/lib/systemd/systemd-journal
  356 root        RT   0 282M  27392  8704 S   0.0   1.4    0:00.10 /sbin/multipathd -d -s
  369 root        RT   0 282M  27392  8704 S   0.0   1.4    0:00.00 /sbin/multipathd -d -s
  373 root        RT   0 30596  8828  4988 S   0.0   0.4    0:00.28 /usr/lib/systemd/systemd-udevd
  374 root        RT   0 282M  27392  8704 S   0.0   1.4    0:00.00 /sbin/multipathd -d -s
  375 root        RT   0 282M  27392  8704 S   0.0   1.4    0:00.00 /sbin/multipathd -d -s
  376 root        RT   0 282M  27392  8704 S   0.0   1.4    0:00.00 /sbin/multipathd -d -s
  377 root        RT   0 282M  27392  8704 S   0.0   1.4    0:00.27 /sbin/multipathd -d -s
  378 root        RT   0 282M  27392  8704 S   0.0   1.4    0:00.00 /sbin/multipathd -d -s
  744 systemd-ne  20   0 18992  9216  8064 S   0.0   0.5    0:00.04 /usr/lib/systemd/systemd-networkd
  775 systemd-oo  20   0 17556  7680  6784 S   0.0   0.4    0:01.33 /usr/lib/systemd/systemd-oomd

F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8Nice + F9Kill F10Quit
```


★ **strace:** muestra las llamadas al sistema de un proceso, incluyendo:

- E/S de archivos: read, write.
- Memoria: mmap, malloc.
- Procesos: fork, execve.
- Red: connect, send.

✓ Es útil para **depuración** y entender cómo el proceso interactúa con el sistema.

```
CPU[ ] 4.0% Tasks: 106, 318 thr, 71 kthr; 1 running
Mem[ ] 785M/1.92G Load average: 0.21 0.12 0.04
Swp[ ] 0K/2.00G Uptime: 01:34:41

Main I/O
PID USER IO DISK R/W DISK READ DISK WRITE SWP% IOD% Command
1527 facuu6 B1 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/pipewire -c fil
1530 facuu6 B1 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/wireplumber
1531 facuu6 B1 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/pipewire-pulse
1532 facuu6 B4 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/gnome-keyring-d
1536 facuu6 B4 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/gnome-keyring-d
1537 facuu6 B4 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/gnome-keyring-d
1538 facuu6 B4 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/gnome-keyring-d
1539 facuu6 B4 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/gnome-keyring-d
1540 facuu6 B4 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/dbus-daemon --s
1541 facuu6 B4 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/pipewire
1542 facuu6 B4 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/wireplumber
1543 facuu6 B4 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/wireplumber
1544 facuu6 B4 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/wireplumber
1545 facuu6 B4 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/pipewire-pulse
1546 facuu6 B4 0.00 B/s 0.00 B/s 0.00 B/s N/A N/A /usr/bin/pipewire -c fil

F1:help F2:Setup F3:Search F4:Filter F5:Tree F6:SortBy F7:Nice F8:Nice + F9:Kill F10:Quit
```

★ **ls /proc/<PID>:** Lista los archivos y subdirectorios en el directorio /proc/<PID> de un proceso específico, mostrando los elementos que contienen información sobre el proceso.

- Archivos como status, cmdline, stat: información básica del proceso (estado, comando, estadísticas).
- Subdirectorios como fd/: descriptores de archivos abiertos por el proceso.

```
facuu6@ubuntu6:~$ ls /proc/3436/
arch_status environ maps patch_state statm
attr exe mem personality status
autogroup fd mountinfo projid_map syscall
auxv fdinfo mounts root task
cgroup gid_map mountstats sched timens_offsets
clear_refs io net schedstat timers
cmdline ksm_merging_pages ns sessionid timerslack_ns
comm ksm_stat numa_maps setgroups uid_map
coredump_filter latency oom_adj smaps wchan
cpu_resctrl_groups limits oom_score smaps_rollup
cpuset loginuid oom_score_adj stack
cwd map_files pagemap stat

Every 1.0s: ps -e ubuntu6: Thu Nov 7 10:39:09 2024

PID TTY TIME CMD
1 ? 00:00:02 systemd
2 ? 00:00:00 kthreadd
3 ? 00:00:00 pool_workqueue_release
4 ? 00:00:00 kworker/R-rcu_g
5 ? 00:00:00 kworker/R-rcu_p
6 ? 00:00:00 kworker/R-slab_
7 ? 00:00:00 kworker/R-netns
10 ? 00:00:00 kworker/0:0H-kblockd
12 ? 00:00:00 kworker/R-mm_pe
13 ? 00:00:00 rcu_tasks_kthread
14 ? 00:00:00 rcu_tasks_rude_kthread
15 ? 00:00:00 rcu_tasks_trace_kthread
```

★ **pidstat -p \$(pgrep -u username firefox) 1:** Muestra el uso de CPU y otros recursos para un proceso específico, ejecutado por el usuario.

- UID: Identificador de usuario del proceso.
- PID: Identificador del proceso (Process ID).
- %usr: Porcentaje de tiempo de CPU utilizado en el espacio de usuario (programas de usuario).
- %system: Porcentaje de tiempo de CPU utilizado en el espacio del núcleo (operaciones del sistema).
- %guest: Porcentaje de CPU en modo de "invitado" (en sistemas virtualizados).
- %wait: Porcentaje de tiempo que el proceso espera por I/O.
- %CPU: Porcentaje total de uso de CPU.
- CPU: Núcleo de la CPU donde se ejecuta el proceso.
- Command: Nombre del comando/proceso.

```
facuu6@ubuntu6:~$ pidstat -p $(pgrep facuu6 firefox) 1
pgrep: only one pattern can be provided
Try 'pgrep --help' for more information.
Linux 6.8.0-47-generic (ubuntu6) 07/11/24 _x86_64_ (1 CPU)

10:44:47 UID PID %usr %system %guest %wait %CPU CPU Command
10:44:47 0 1 0.01 0.02 0.00 0.04 0.03 0 systemd
```

★ **tasklist (Windows):** Muestra todos los procesos activos, incluyendo su nombre, PID (ID del proceso), sesión y uso de memoria. Permite filtros y formatos específicos para gestionar procesos fácilmente.

```
C:\Users\atn>tasklist

Nombre de imagen PID Nombre de sesión Núm. de ses Uso de memor
-----
System Idle Process 0 Services 0 8 KB
System 4 Services 0 144 KB
Registry 124 Services 0 56.340 KB
csrss.exe 456 Services 0 1.200 KB
csrss.exe 728 Services 0 6.436 KB
csrss.exe 524 Services 0 7.840 KB
```

★ **Get-Process (Windows):** Muestra información detallada sobre los procesos en ejecución en el sistema.

- Handles: Número de manejadores abiertos por el proceso.
- NPM(K): Memoria no paginada usada, en KB.
- PM(K): Memoria paginada usada, en KB.
- WS(K): Conjunto de trabajo, o memoria en uso activo, en KB.
- VM(M): Memoria virtual asignada al proceso, en MB.
- CPU(s): Tiempo total de CPU utilizado por el proceso, en segundos.
- Id: ID del proceso (PID).
- ProcessName: Nombre del proceso.

```
PS C:\Users\atn> Get-Process

Handles NPM(K) PM(K) WS(K) CPU(s) Id SI ProcessName
-----
214 13 2404 11444 0.17 10552 2 AdobeIPCBroker
213 13 9056 16308 23.83 2364 0 audiodg
193 12 11348 30524 0.13 1244 2 blitr
320 18 58112 36168 4.11 8804 2 blitr
281 17 31892 45764 3.53 9600 2 blitr
940 38 86736 58216 166.13 11580 2 blitr
443 25 90124 43672 2.48 13364 2 blitr
268 40 62172 47584 0.20 824 2 Canva
166 11 14872 28788 0.02 10492 2 Canva
50 4 616 3728 0.00 12656 2 COXProcess
303 17 13806 35073 0.26 656 2 chrome
```

★ **Stop-Process -Id <PID>:** Comando de PowerShell para matar un proceso por su ID.

★ **kill <PID>:** Mata el proceso especificando su ID.