



Lab 20.2: Examinar la actividad de System V IPC

System V IPC es un método bastante antiguo de Comunicación Entre Procesos (**IPC**) que se remonta a los primeros días de **UNIX**. Este involucra tres mecanismos:

1. Segmentos de memoria compartida
2. Semáforos
3. Colas de mensajes

Algunos programas modernos tienen a usar los métodos de **POSIX IPC** para esos tres mecanismos, pero todavía se puede encontrar un montón de aplicaciones **System V IPC**.

Ejecute el siguiente comando para obtener un resumen general de la actividad **System V IPC** en su sistema:

```
$ ipcs
```

```
----- Message Queues -----
key          msqid      owner      perms      used-bytes  messages

----- Shared Memory Segments -----
key          shmid      owner      perms      bytes       nattch     status
0x01114703  0          root       600        1000        6
0x00000000  98305     coop       600        4194304    2          dest
0x00000000  196610    coop       600        4194304    2          dest
0x00000000  23068675  coop       700        1138176    2          dest
0x00000000  23101444  coop       600        393216     2          dest
0x00000000  23134213  coop       600        524288     2          dest
0x00000000  24051718  coop       600        393216     2          dest
0x00000000  23756807  coop       600        524288     2          dest
0x00000000  24018952  coop       600        67108864   2          dest
0x00000000  23363593  coop       700        95408      2          dest
0x00000000  1441811   coop       600        2097152    2          dest

----- Semaphore Arrays -----
key          semid      owner      perms      nsems
0x00000000  98304     apache     600        1
0x00000000  131073    apache     600        1
0x00000000  163842    apache     600        1
0x00000000  196611    apache     600        1
0x00000000  229380    apache     600        1
```

Tenga en cuenta que casi todos los segmentos de memoria compartida actualmente en ejecución tienen 0 en el campo key (también conocido como **IPC_PRIVATE**), lo cual significa que solo se comparten entre procesos en una relación padre/hijo. Además, todos con la excepción de uno, están marcados para destrucción, ya que no tienen procesos asociados.

Es posible obtener más información acerca de los procesos que han creado los segmentos y que se han asociado a ellos con:

```
$ ipcs -p
```

```

----- Message Queues PIDs -----
msqid      owner      lspid      lrpid

----- Shared Memory Creator/Last-op PIDs -----
shmid      owner      cpid       lpid
0           root       1023       1023
98305      coop       2265       18780
196610     coop       2138       18775
23068675   coop       989        1663
23101444   coop       989        1663
23134213   coop       989        1663
24051718   coop       20573      1663
23756807   coop       10735      1663
24018952   coop       17875      1663
23363593   coop       989        1663
1441811    coop       2048       20573

```

Por lo tanto, haciendo:

```
$ ps aux |grep -e 20573 -e 2048
```

```

coop      2048  5.3  3.7 1922996 305660 ?      Rl   Oct27  77:07 /usr/bin/gnome-shell
coop      20573 1.9  1.7 807944 141688 ?      Sl   09:56   0:01 /usr/lib64/thunderbird/thunderbird
coop      20710 0.0  0.0 112652  2312 pts/0  S+   09:57   0:00 grep --color=auto -e 20573 -e 2048

```

vemos que **thunderbird** está usando un segmento de memoria compartida creado por **gnome-shell**.

Realice estos pasos en su sistema e identifique los diversos recursos que están siendo usados y por quién. ¿Hay alguna **fuga potencial** en el sistema (recursos compartidos que no están en uso por ningún proceso)? Por ejemplo:

```
$ ipcs
```

```

....
----- Shared Memory Segments -----
key          shmid      owner      perms      bytes      nattch     status
....
0x00000000  622601    coop       600        2097152    2          dest
0x0000001a  13303818  coop       666        8196       0
....

```

muestra un segmento de memoria compartida que no tiene procesos asociados y que no está marcado para ser destruido. Por lo anterior, si el proceso continúa así para siempre y no se asocia a ningún proceso, podría significar una fuga de memoria.