

Using GNU Screen to Manage Persistent Terminal Sessions

MIGUEL CÓRDOBA ARANDA

1º ASIR

IMPLANTACIÓN DE SISTEMAS OPERATIVOS

GNU SCREEN

Herramienta que nos permite ejecutar múltiples consolas que pueden ser minimizadas y re-adjuntadas a voluntad, compartidas, guardadas, divididas, etcétera... . Sobre todo se usa en servidores cuando se dejan tareas programadas.

ENTENCIENDO SCREEN

La “**sesión**” → Es un área de trabajo, que se crea cuando ejecutas el comando *screen*.

Una **ventana** → Shell abierta dentro de una sesión de screen, una sesión de screen puede contener múltiples ventanas.

Una **región** → Es una división, realizada a una ventana, que dentro contiene otra ventana, uno puede dividir una sesión de screen en 2 o más partes, horizontal o verticalmente y en cada **split** (porción de una región) desplegar una ventana.

INSTALAR SCREEN

En algunas distribuciones *screen* ya viene por defecto instalado, en aquellas donde no lo estuviera, con ejecutar → **#apt-get install screen** ó **#yum install screen**

TRABAJANDO CON SESIONES SCREEN

Crear una sesión de screen con nombre:

#screen -S [nombre] → Entrarás en una sesión de *screen* con el nombre indicado (nota: si se tienen muchísimas sesiones de screen activas, es muy útil identificarlas por nombre).

```
debianiso@debian:~$ screen -S Administrador
[detached from 2909.Administrador]
debianiso@debian:~$ screen -l
[detached from 2921.pts-0.debian]
debianiso@debian:~$ screen -ls
There are screens on:
    2921.pts-0.debian      (27/12/18 12:30:41)    (Detached)
    2909.Administrador   (27/12/18 12:29:51)    (Detached)
2 Sockets in /run/screen/S-debianiso.
debianiso@debian:~$
```

OPERACIONES CON SCREEN

Todas las operaciones en SCREEN se activan con la secuencia **CTRL+a** seguida de alguna letra y/o combinación de letras.

Ejemplo:

CTRL+a K → Matará un programa dentro de una *screen*.

Screen también admite “comandos”, los comandos se escriben luego de escribir “:” (dos puntos), ejemplo:

CTRL+a “:multiuser off” → Detiene el multi-usuario de una sesión y **on** lo vuelve a activar.

CTRL+a SHIFT+H → permite guardar el historial de comandos realizados por una ventana screen.

DETTACHING, ATTACHING Y RE-ATTACHING

Una de las cosas más útiles de SCREEN es la posibilidad de ejecutar un comando muy largo y dejarlo ejecutando dentro de una consola y luego minimizar la misma y re-conectarse luego.

CTRL+a d → Nos permitirá minimizar una sesión, no aparecerá DETACH, para volver a la sesión introducimos **#screen -r [nombre_sesión]**.

```
debianiso@debian:~$ screen -S Administrador
[detached from 2909.Administrador]
debianiso@debian:~$ screen -l
[detached from 2921.pts-0.debian]
debianiso@debian:~$ screen -ls
There are screens on:
  2921.pts-0.debian      (27/12/18 12:30:41)   (Detached)
  2909.Administrador    (27/12/18 12:29:51)   (Detached)
2 Sockets in /run/screen/S-debianiso.
debianiso@debian:~$
```

De existir más de una consola “minimizada”, veremos un listado como este:

CTRL+a y luego “c” → Esto habilitará otra ventana.

Num	Name	Flags
0	bash	\$
1	bash	\$

Para cambiar entre ventanas:

CTRL+a del 0 al 9 → cambiar a la ventana según su ordinal (posición).

CTRL+a “n” → “next window”, próxima ventana.

CTRL+a ” → Una lista de todas las ventanas creadas.

Esta lista se puede “navegar” (flechas arriba y abajo) y ENTER para seleccionar una ventana.

PARA DARLE NOMBRE A UNA “VENTANA”

CTRL+a Mayúsculas+a → Aparece un mensaje debajo solicitando el nombre de la screen.

Borren la palabra “bash” y pongan el nombre que deseen.

El nombre correspondiente de la ventana (muy útil para saber de qué se trata cada ventana).

Num	Name	Flags
0	Ventana0	\$
1	Ventana1	\$

CRTL+a SHIFT-s: permite dividir una screen en dos regiones.

```
debianiso@debian:~$ 
0 Ventana0
--
```

Luego, podemos cambiar el foco a la nueva región con:

CTRL+a y TAB → moverá el foco a la región siguiente

Al estar en esta región podemos realizar las siguientes operaciones:

CTRL+a SHIFT-C → Crear una nueva ventana en la región

CTRL+a n → Nos mueve a la ventana actual (la que está arriba), podemos igual cambiar a las diferentes ventanas ya existentes.

Para cerrar una región, simplemente se ejecuta:


CTRL+a SHIFT-x : cierra la región actual

CTRL+a SHIFT+q : al contrario, cierra todas las regiones, salvo la actual.

PANTALLA MULTIUSUARIO

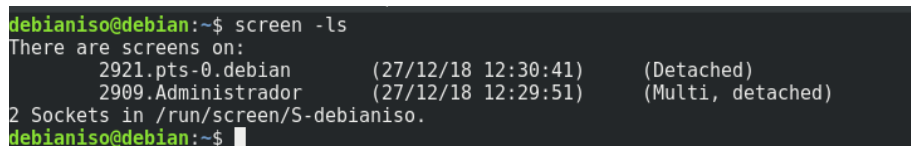
Una de las ventajas de usar screens, es que una consola pueda ser operada por varios usuarios a la vez, se utiliza normalmente.

CTRL+a y escribir “:multiuser on”



```
debianiso@debian:~$  
  
Multiuser mode enabled
```

En este momento, la sesión puede ser accedida por múltiples personas, pero primero, debemos autorizarlos.



```
debianiso@debian:~$ screen -ls  
There are screens on:  
  2921.pts-0.debian      (27/12/18 12:30:41)    (Detached)  
  2909.Administrador    (27/12/18 12:29:51)    (Multi, detached)  
2 Sockets in /run/screen/S-debianiso.  
debianiso@debian:~$
```

Para conectarse a una consola multiusuario → **#screen -x [nombre de la screen]** ó **#screen -r [PID de la sesión]**

Para autorizar usuarios → **CTRL+a “:acladd [lista de usuarios]”**. La lista de usuarios puede ser uno o varios nombres de usuario, separados por comas → **CTRL+a “:acladd miguel,pepe”**.

El usuario que creó la sesión de screen, más los usuarios “miguel” y “pepe” tienen derecho a acceder a la sesión screen.

Permisos en la consola screen multi-usuario → **w:escritura, r:lectura y x:ejecución** con los valores “+” (**agregar**) y “-” (**quitar**), luego de los permisos viene el número de la ventana donde recibirá los permisos (**el símbolo “#” indica “todas las ventanas”**).

Ejemplo → **CTRL+a :aclchg soporte -w “#”**

Para ver los usuarios conectados a una consola screen → **CTRL+a SHIFT-“+” (CTRL+a “*)**

Mensaje a todas las ventanas activas → **CTRL+a “:wall [mensaje]”**

CONCLUSIONES

SCREEN nos permite trabajar de manera cooperativa y colaborativa con consolas activas en servidores. Algo tan sencillo como SSH y una consola SCREEN compartida, nos permite operar con confianza contra nuestros servidores Linux.