Creación de Scripts.

Por: John A. Pérez B. ~ 20186748

Este tutorial es parte del siguiente video:

https://youtu.be/eAHCPRv-CCc

Los Shell Scripts

Para entender el funcionamiento de los Shell Scripts, es necesario que entendamos la forma en la que funciona la Shell. Esta es un interprete de comando, esto significa que al ingresar un comando esta buscara en la ruta establecida el programa relacionado al comando que ingresamos. Para saber donde nuestro intérprete busca los comandos colocamos el comando **$PATH**. En la siguiente imagen veremos una demostracion.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

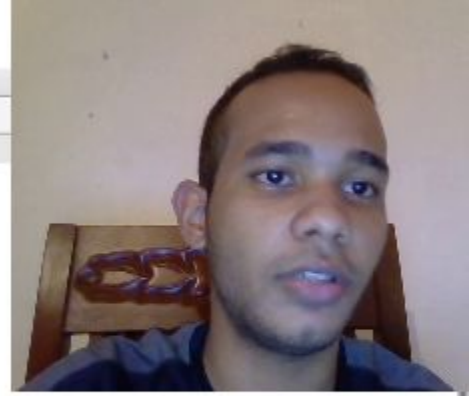
File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~

File Edit View Search Terminal Help

```
[root@localhost ~]# $PATH
bash: /usr/local/bin:/usr/local/sbin:/usr/bin:/usr/sbin:/bin:/sbin:/root/bin: No such file or directory
[root@localhost ~]# echo $PATH
/usr/local/bin:/usr/local/sbin:/usr/bin:/usr/sbin:/bin:/sbin:/root/bin
[root@localhost ~]#
```



root@localhost:~

1 / 4

Right Ctrl

Al crear un script, es necesario incluir la extensión **.sh,** de tal manera identificamos el archivo como un ShellScript, y el sistema lo identifica como tal al momento de trabajar con este.

Todos los Shell scripts poseen algo en comun, al principio se incluye la direccion de un interprete, luego de **#!.** Esto le indica a la Shell que interprete utilizara para dicho script. Este se coloca de la siguiente manera.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: sc.sh

```
#!/bin/bash
```

[Read 1 line]

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos

^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

root@localhost:~/Desktop/scrt

1 / 4

Right Ctrl

De igual forma en las que la Bash interpreta los comandos y los relaciona con un programa. Al colocar un script si contamos con los permisos necesarios, esta comenzara a interpretar dicha secuencia de comandos. Por lo que para ejecutarlo es necesario contar con dichos permisos. Estos se asignan con el comando **chmod.** Una vez contemos con los permisos el archivo cambia de color en la terminal

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

```
[root@localhost scrt]# nano sc.sh
[root@localhost scrt]# ls
sc.sh
[root@localhost scrt]# chmod 700 sc.sh
[root@localhost scrt]# ls
sc.sh
[root@localhost scrt]#
```

root@localhost:~/Desktop/scrt

1 / 4

Right Ctrl

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

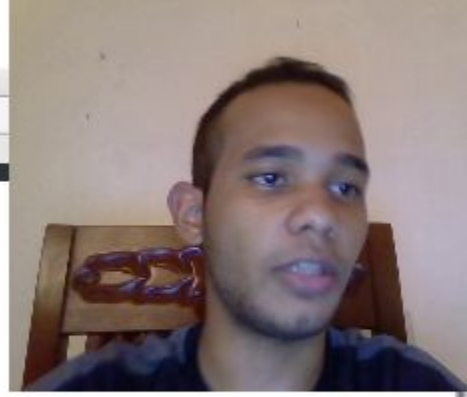
Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: sc.sh

```
#!/bin/bash

tst='Hola'

echo $tst
echo $tst
```



Estos también admiten elementos comunes en los lenguajes de programación como variables, comparadores, condicionales y ciclos.

^Y Prev Page ^K Cut Text ^C Cur Pos
^V Next Page ^U UnCut Text ^T To Spell

1 / 4

Right Ctrl

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

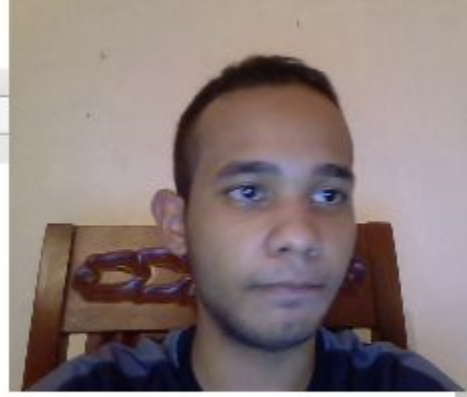
File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

```
[root@localhost scrt]# nano sc.sh
[root@localhost scrt]# ls
sc.sh
[root@localhost scrt]# ./sc.sh
Hola
Hola
[root@localhost scrt]#
```



Para ejecutarlos en la terminal simplemente colocamos el directorio en que se encuentran y su nombre.

Right Ctrl

Script 1: Calculadora

Paso 1

Generamos el archivo con el que vamos a trabajar con extensión **sh** utilizando el comando **touch** y una vez abrimos el archivo, colocamos los parámetros que vimos en el primer ejemplo.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

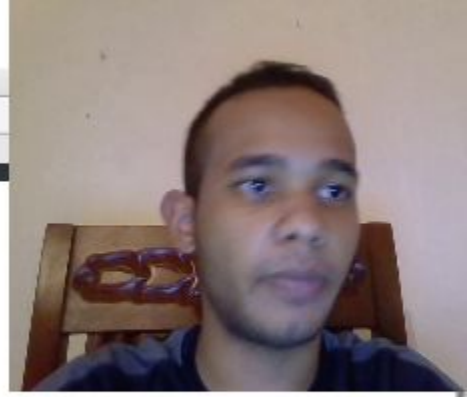
File Edit View Search Terminal Help

GNU nano 2.3.1 File: calc.sh

```
#!/bin/bash
```

[Read 0 lines]

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell



root@localhost:~/Desktop/scrt

1 / 4

Right Ctrl

Paso 2

Diseñamos la interfaz de nuestro programa. esta le dirá al usuario que puede hacer. para esto utilizamos el comando **echo** para imprimir estos datos en la Shell.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: calc.sh

```
#!/bin/bash

echo '*==============================*'
echo '| calculadora (elija una opcion)|'
echo '|------------------------------|'
echo '| [1]        suma              |'
echo '|------------------------------|'
echo '| [2]        resta             |'
echo '|------------------------------|'
echo '| [3]     multiplicacion       |'
echo '|------------------------------|'
echo '| [4]       Division           |'
echo '*==============================*'
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

root@localhost:~/Desktop/scrt

1 / 4

Right Ctrl

Paso 3

Aplicamos los permisos, y probamos nuestra interfaz. Luego le agregaremos la parte funcional.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

```
[root@localhost scrt]# ls
calc.sh  sc.sh
[root@localhost scrt]# nano calc.sh
[root@localhost scrt]# ls
calc.sh  sc.sh
[root@localhost scrt]# chmod 777 calc.sh
[root@localhost scrt]# ./calc.sh
*==============================*
| calculadora (elija una opcion)|
|------------------------------|
| [1]        suma              |
|------------------------------|
| [2]        resta             |
|------------------------------|
| [3]     multiplicacion       |
|------------------------------|
| [4]       Division           |
*==============================*
[root@localhost scrt]#
```

root@localhost:~/Desktop/scrt

1 / 4

Right Ctrl

Paso 4

Para que el usuario pueda elegir una opción es necesario colocar una entrada de texto. esto lo hacemos con la palabra reservada **read**, y luego colocamos el nombre de la variable que almacenará la entrada.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: calc.sh

```
#!/bin/bash

echo '*================================*'
echo '| calculadora (elija una opcion)|'
echo '|--------------------------------|'
echo '| [1]        suma                |'
echo '|--------------------------------|'
echo '| [2]        resta               |'
echo '|--------------------------------|'
echo '| [3]     multiplicacion         |'
echo '|--------------------------------|'
echo '| [4]       Division             |'
echo '*================================*'
read opc
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

root@localhost:~/Desktop/scrt

1 / 4

Right Ctrl

Paso 5

Para agregar la parte funciona de esta calculadora utilizaremos un switch, este funciona como comparador, donde se compara cada una de las opciones que agreguemos con el parámetro que le indiquemos al principio, para iniciarlo usamos **case** y como break utilizamos ;;. en este caso crearemos otro read con dos variables, e iniciaremos con la operación de suma. Utilizaremos el formato **[opcion #]) echo $(($n1 [operación] $n2));;.** Donde el símbolo de dólar indica que estas variables debe de ser tratadas de forma especial y no como texto plano.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: calc.sh

```
#!/bin/bash

echo '*================================*'
echo '| calculadora (elija una opcion)|'
echo '|--------------------------------|'
echo '| [1]        suma                |'
echo '|--------------------------------|'
echo '| [2]        resta               |'
echo '|--------------------------------|'
echo '| [3]     multiplicacion         |'
echo '|--------------------------------|'
echo '| [4]       Division             |'
echo '*================================*'
read opc
echo 'Coloque dos numeros'
read n1 n2
case $opc in
1) echo $(($n1 + $n2))
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

root@localhost:~/Desktop/scrt

1 / 4

Right Ctrl

Paso 6

Para agregar la demás funciones utilizamos el mismo formato **[opcion #]) echo $(($n1 [operación] $n2));;.** Y completamos hasta la división. para cerrar el switch utilizamos su nombre de forma inversa, es decir **esac.**

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: calc.sh

```
#!/bin/bash

echo '*================================*'
echo '| calculadora (elija una opcion)|'
echo '|--------------------------------|'
echo '| [1]        suma                |'
echo '|--------------------------------|'
echo '| [2]        resta               |'
echo '|--------------------------------|'
echo '| [3]     multiplicacion         |'
echo '|--------------------------------|'
echo '| [4]       Division             |'
echo '*================================*'

read opc

echo 'Coloque dos numeros'

read n1 n2

echo 'El resultado es:'

case $opc in
1) echo $(($n1 + $n2));;
2) echo $(($n1 - $n2));;
3) echo $(($n1*$n2));;
4) echo $(($n1/$n2));;
esac
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

root@localhost:~/Desktop/scrt

1 / 4

Right Ctrl

Paso 7

Probamos que el programa funcione correctamente.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

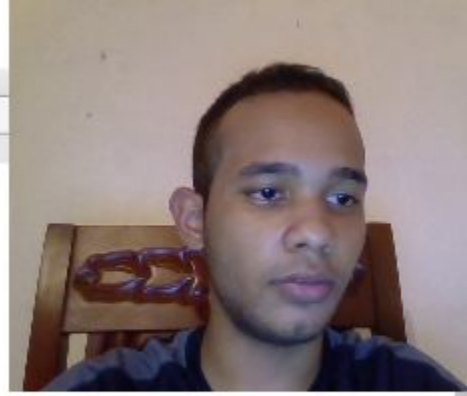
File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

```
[root@localhost scrt]# ./calc.sh
*==============================*
| calculadora (elija una opcion)|
|------------------------------|
| [1]        suma              |
|------------------------------|
| [2]        resta             |
|------------------------------|
| [3]     multiplicacion       |
|------------------------------|
| [4]       Division           |
*==============================*
1
Coloque dos numeros
5 5
El resultado es:
10
[root@localhost scrt]# ./calc.sh
*==============================*
| calculadora (elija una opcion)|
|------------------------------|
| [1]        suma              |
|------------------------------|
| [2]        resta             |
|------------------------------|
| [3]     multiplicacion       |
|------------------------------|
| [4]       Division           |
*==============================*
2
Coloque dos numeros
10 5
El resultado es:
5
[root@localhost scrt]#
```



root@localhost:~/Desktop/scrt

1 / 4

Right Ctrl

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

```
| [2]        resta              |
|-------------------------------|
| [3]     multiplicacion        |
|-------------------------------|
| [4]       Division            |
*===============================*
2
Coloque dos numeros
10 5
El resultado es:
5
[root@localhost scrt]# ./calc.sh
*===============================*
| calculadora (elija una opcion)|
|-------------------------------|
| [1]        suma               |
|-------------------------------|
| [2]        resta              |
|-------------------------------|
| [3]     multiplicacion        |
|-------------------------------|
| [4]       Division            |
*===============================*
3
Coloque dos numeros
2 5
El resultado es:
10
[root@localhost scrt]# ./calc.sh
*===============================*
| calculadora (elija una opcion)|
|-------------------------------|
| [1]        suma               |
|-------------------------------|
| [2]        resta              |
|-------------------------------|
| [3]     multiplicacion        |
|-------------------------------|
| [4]       Division            |
*===============================*
4
Coloque dos numeros
10 2
El resultado es:
5
[root@localhost scrt]#
```

root@localhost:~/Desktop/scrt

1 / 4

Right Ctrl

Script 2: Creador de usuarios

Paso 1

Generamos el archivo con el que vamos a trabajar con extensión **sh** utilizando el comando **touch** y una vez abrimos el archivo, colocamos los parámetros que vimos en el primer ejemplo.
E imprimimos un mensaje en la terminal para almacenar el nombre del usuario, y lo almacenamos en una variable.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: usr.sh

```
#!/bin/bash

echo 'Coloque el nombre del usuario'
read usr
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

root@localhost:~/Desktop/scrt

1 / 4

Right Ctrl

Paso 2

Utilizamos la variable almacenada como parametro del comando **adduser** al que tambien le agragaremos un comando anidado con **&&** para que se solicite la contraseña de dicho usario.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: usr.sh

```
#!/bin/bash

echo 'Coloque el nombre del usuario'
read usr

adduser $usr && passwd
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

root@localhost:~/Desktop/scrt

1 / 4

Right Ctrl

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: usr.sh

```
#!/bin/bash

echo 'Coloque el nombre del usuario'
read usr

adduser $usr && passwd $usr && echo 'Usuario creado'|| echo 'No fue posible crear el usuario'
```

Paso 3

Ahora agregamos un manejador de errores, este nos notificará si el comando se ejecutó o no. Para esto utilizamos la doble conjunción && y la || en caso de que surja un error.

^Y Prev Page ^K Cut Text ^C Cur Pos
^V Next Page ^U UnCut Text ^T To Spell

1 / 4

Right Ctrl

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

```
[root@localhost scrt]# ls
calc.sh  sc.sh  usr.sh
[root@localhost scrt]# ./usr.sh
Coloque el nombre del usuario
nuevo-centos-usr
Changing password for user nuevo-centos-usr.
New password:
BAD PASSWORD: The password fails the dictionary check - it is too simplistic/systematic
Retype new password:
passwd: all authentication tokens updated successfully.
Usuario creado
[root@localhost scrt]#
```

Paso 4

Agregamos los permisos a nuestro script y lo probamos.

Right Ctrl

Script 3: Creador de grupos

Paso 1

Generamos el archivo con el que vamos a trabajar con extensión **sh** utilizando el comando **touch** y una vez abrimos el archivo, colocamos los parámetros que vimos en el primer ejemplo. E imprimimos un mensaje con la solicitud del nombre y el ID del nuevo grupo, y los almacenamos en sus respectivas variables.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

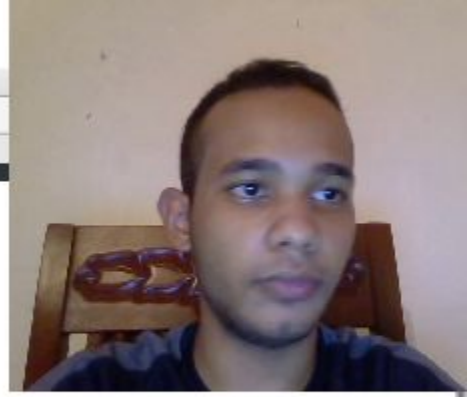
root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: groups.sh

```
#!/bin/bash

echo 'Ingrese el nombre y el ID del nuevo grupo'
read nm id
```

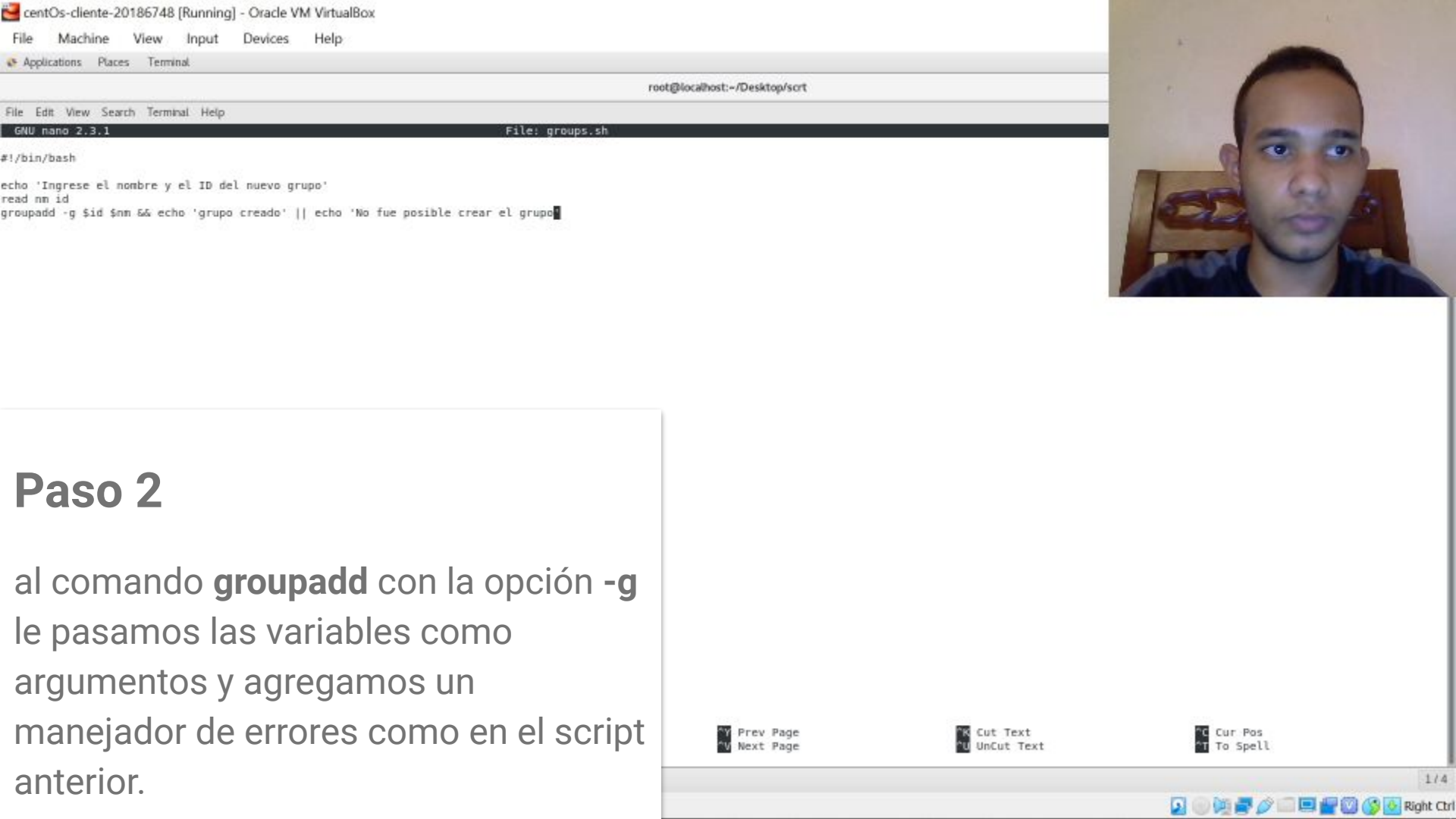
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell



root@localhost:~/Desktop/scrt

1 / 4

Right Ctrl



Paso 2

al comando **groupadd** con la opción **-g** le pasamos las variables como argumentos y agregamos un manejador de errores como en el script anterior.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

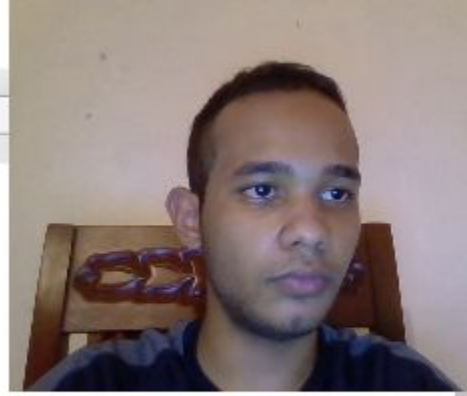
File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

```
[root@localhost scrt]# nano groups.sh
[root@localhost scrt]# ls
calc.sh  groups.sh  sc.sh  usr.sh
[root@localhost scrt]# chmod 777 groups.sh
[root@localhost scrt]# ./groups.sh
Ingrese el nombre y el ID del nuevo grupo
grupo-centos-tst 43224
grupo creado
[root@localhost scrt]#
```



Paso 3

Aplicamos los permisos y probamos

Right Ctrl

Script 4:
Generador de Backups

Paso 1

Generamos el archivo con el que vamos a trabajar con extensión **sh** utilizando el comando **touch** y una vez abrimos el archivo, colocamos los parámetros que vimos en el primer ejemplo. En el mismo colocamos dos variables, y le insertamos los datos de la fuente y el destino del backup con el nombre finalizado en **.tar.gz**. Ademas de una variable extra que tendrá como valor el comando **date +%b-%d-%y**, de tal manera que el nombre de nuestro backup posea la fecha actual

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: backup.sh

```
#!/bin/bash

dt=`date +%b-%b-%y`
src=~/Desktop/scrt
dest=~/Desktop/share/backup-$dt.tar.gz
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

root@localhost:~/Desktop/scrt

1 / 4

Right Ctrl

Paso 2

Utilizamos el comando **tar** con las opciones -**zcpf** las cuales nos permitiran comprimir archivos y directorios, creando un nuevo archivo con dicha informacion. Luego colocamos el destino y la fuente como argumentos.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: backup.sh

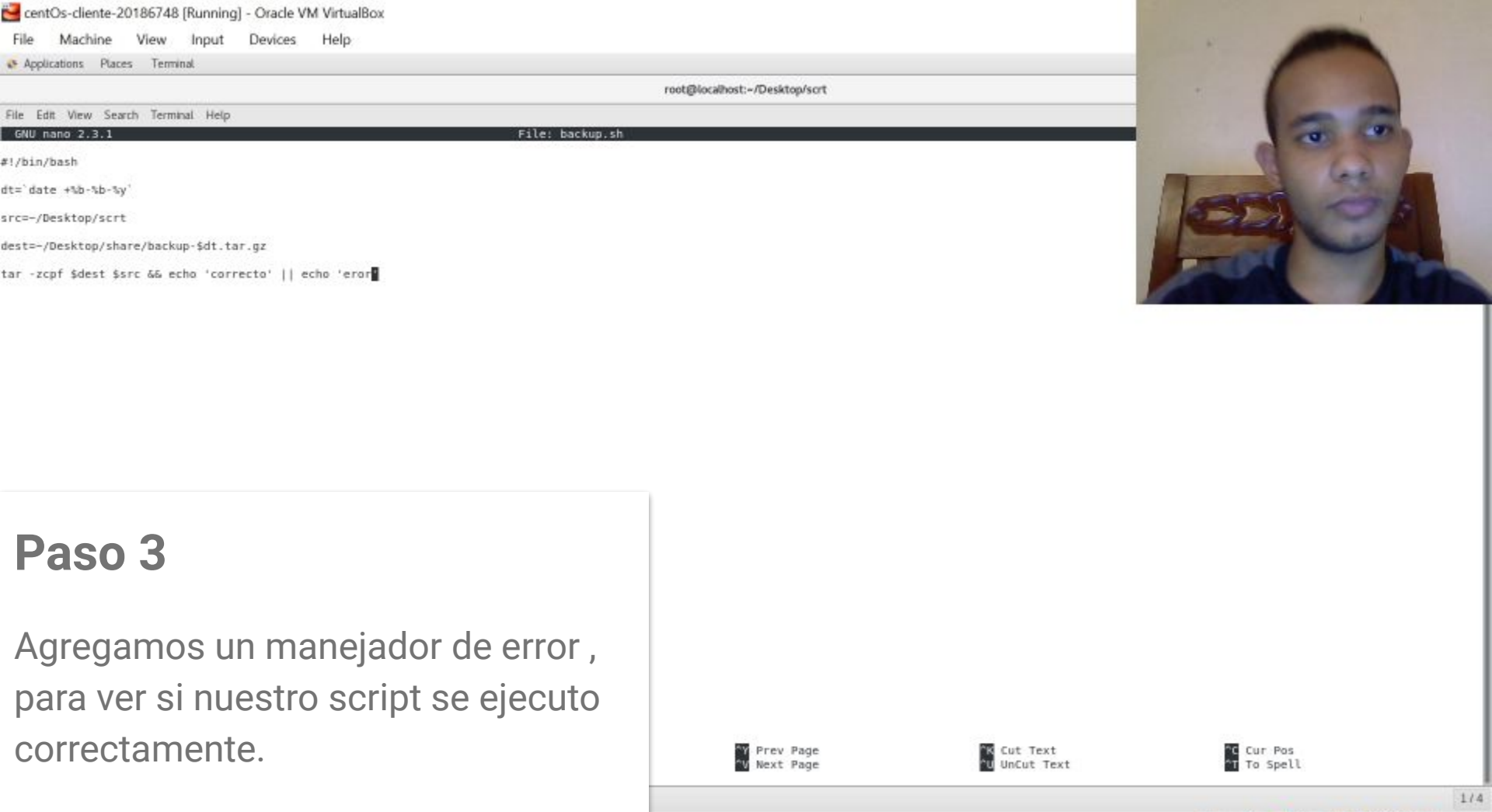
```
#!/bin/bash

dt=`date +%b-%b-%y`
src=~/Desktop/scrt
dest=~/Desktop/share/backup-$dt.tar.gz
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

root@localhost:~/Desktop/scrt

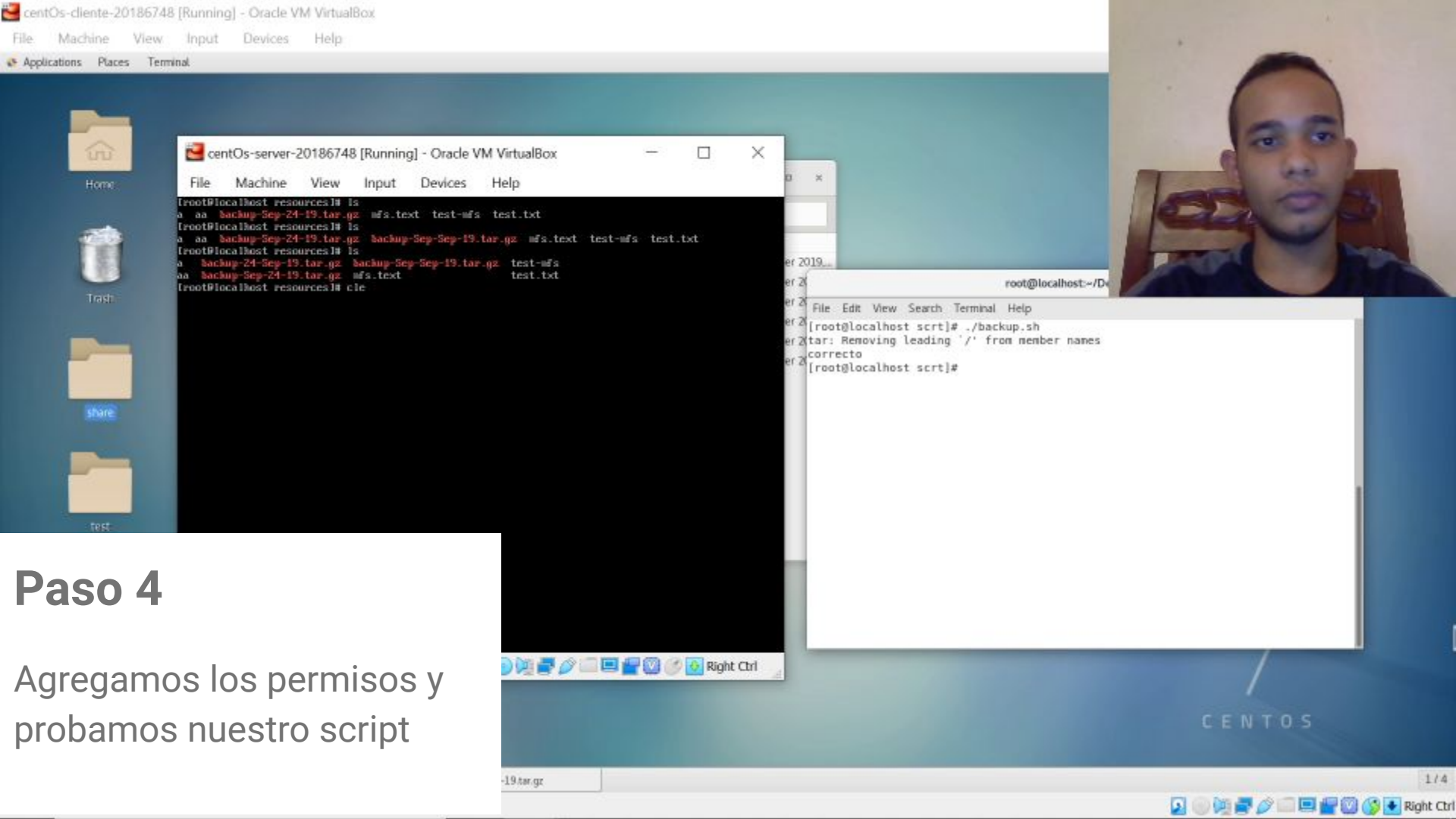
1 / 4

Right Ctrl



Paso 3

Agregamos un manejador de error , para ver si nuestro script se ejecuto correctamente.



Paso 4

Agregamos los permisos y probamos nuestro script

Script 5:
Transferencia de datos

Paso 1

Generamos el archivo con el que vamos a trabajar con extensión **sh** utilizando el comando **touch** y una vez abrimos el archivo, colocamos los parámetros que vimos en el primer ejemplo. E imprimimos un mensaje que le solicite al usuario colocar la ruta del archivo, la cual almacenaremos en una variable.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: transfer.sh

```
#!/bin/bash

echo 'coloque la ruta del archivo a transferir'
read rt
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

root@localhost:~/Desktop/scrt [share]

1 / 4

Right Ctrl

Paso 2

Con el comando **mv**, transferimos el archivo desde donde se encuentre hasta nuestro directorio compartido en la red NFS, para que este pueda ser accedido por nuestro otro computador. Y al igual que en los demás scripts agregamos un manejador de errores.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: transfer.sh

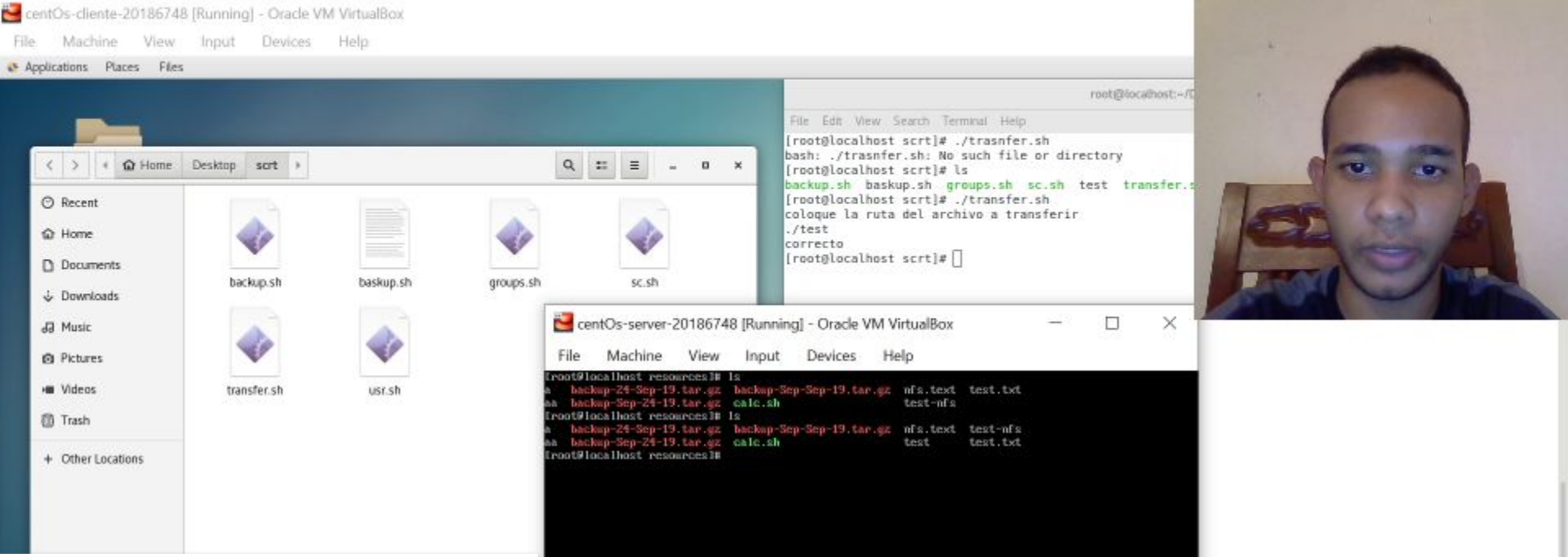
```
#!/bin/bash

echo 'coloque la ruta del archivo a transferir'
read rt

mv $rt ~/Desktop/share && echo 'correcto' || echo 'ha ocurrido un error'
```

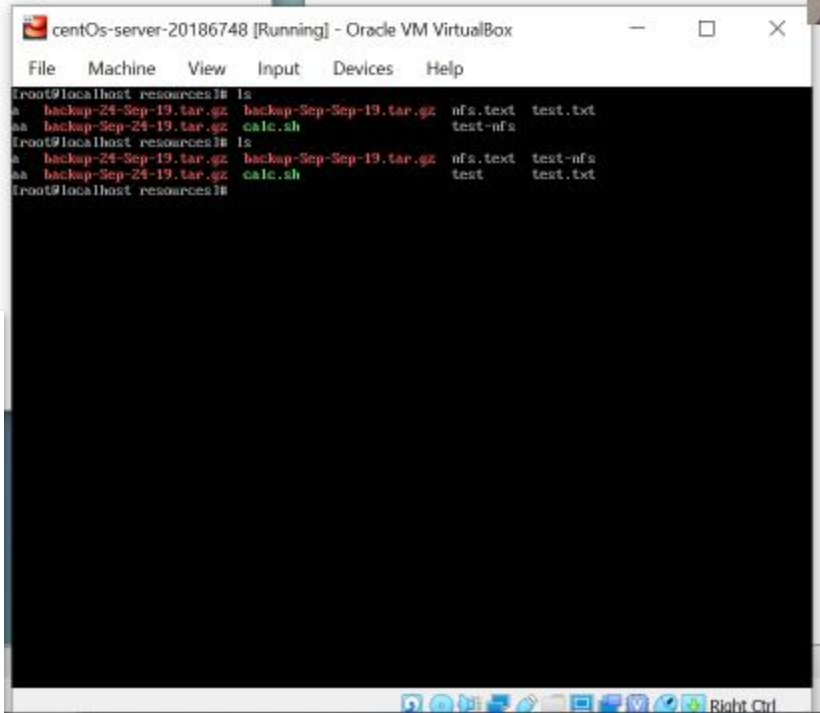
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

root@localhost:~/Desktop/scrt [share]

1 / 4

Right Ctrl





Paso 3

Aplicamos los permisos y probamos

Script 6: Manejador de servicios(iniciar, detener y reiniciar)

Paso 1

Generamos el archivo con el que vamos a trabajar con extensión **sh** utilizando el comando **touch** y una vez abrimos el archivo, colocamos los parámetros que vimos en el primer ejemplo. Luego generamos la interfaz, donde le solicitamos al usuario el nombre del servicio y la opcion a ejecutar

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: service.sh

```
#!/bin/bash
echo 'Nombre del servicio'
read servicio
echo 'opciones'
echo '[1] iniciar'
echo '[2] detener'
echo '[3] reiniciar'
read opc
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

root@localhost:~/Desktop/scrt [share] scrt 1 / 4

Right Ctrl

Paso 2

Nuevamente con un switch definimos el comando que se va a ejecutar en cada uno de los caso, sea el de iniciar, detener o reiniciar el servicio.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

GNU nano 2.3.1 File: service.sh

```
#!/bin/bash
echo 'Nombre del servicio'
read servicio
echo 'opciones'
echo '[1] iniciar'
echo '[2] detener'
echo '[3] reiniciar'
read opc

case $opc in
1) systemctl start $servicio.service;;
2) systemctl stop $servicio.service;;
3) systemcrl reload $servicio.service;;
esac
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

root@localhost:~/Desktop/scrt [share] scrt 1 / 4

Right Ctrl

Paso 3

Procedemos a probar, nuestro script, y luego utilizamos el comando **journalctl -u** y el nombre del archivo, para comprobar si el estado del servicio fue modificado por el script.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

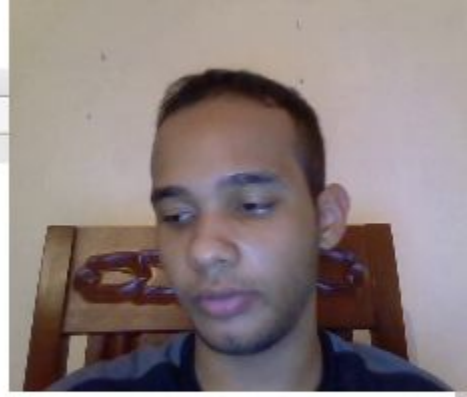
File Machine View Input Devices Help

Applications Places Terminal

root@localhost:~/Desktop/scrt

File Edit View Search Terminal Help

```
Nombre del servicio
nfs-server
opciones
[1] iniciar
[2] detener
[3] reiniciar
3
Job for nfs-server.service invalid.
[root@localhost scrt]# ./service.sh
Nombre del servicio
nfs-server
opciones
[1] iniciar
[2] detener
[3] reiniciar
1
[root@localhost scrt]# ./service.sh
Nombre del servicio
nfs-server
opciones
[1] iniciar
[2] detener
[3] reiniciar
3
[root@localhost scrt]# ./service.sh
Nombre del servicio
nfs-server
opciones
[1] iniciar
[2] detener
[3] reiniciar
2
[root@localhost scrt]# journalctl -u nfs-server.service
-- Logs begin at Tue 2019-09-24 17:23:59 AST, end at Tue 2019-09-24 19:01:54 AST. --
Sep 24 19:00:07 localhost.localdomain systemd[1]: Starting NFS server and services...
Sep 24 19:00:07 localhost.localdomain systemd[1]: Started NFS server and services.
Sep 24 19:00:37 localhost.localdomain systemd[1]: Stopping NFS server and services...
Sep 24 19:00:37 localhost.localdomain systemd[1]: Stopped NFS server and services.
Sep 24 19:01:18 localhost.localdomain systemd[1]: Unit nfs-server.service cannot be reloaded because it is inactive.
Sep 24 19:01:29 localhost.localdomain systemd[1]: Starting NFS server and services...
Sep 24 19:01:29 localhost.localdomain systemd[1]: Started NFS server and services.
Sep 24 19:01:40 localhost.localdomain systemd[1]: Reloading NFS server and services.
Sep 24 19:01:40 localhost.localdomain systemd[1]: Reloaded NFS server and services.
Sep 24 19:01:54 localhost.localdomain systemd[1]: Stopping NFS server and services...
Sep 24 19:01:54 localhost.localdomain systemd[1]: Stopped NFS server and services.
[root@localhost scrt]#
```



root@localhost:~/Desktop/scrt [share] scrt 1 / 4

Right Ctrl

Automatización de scripts con cron

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

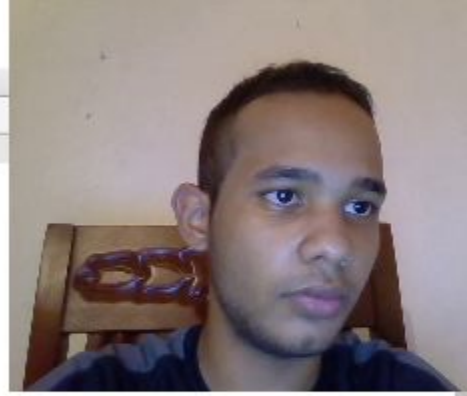
File Machine View Input Devices Help

Applications Places Terminal

root@localhost:/etc

File Edit View Search Terminal Help

```
[root@localhost etc]# ls | grep cron
anacrontab
cron.d
cron.daily
cron.deny
cron.hourly
cron.monthly
crontab
cron.weekly
[root@localhost etc]#
```



Paso 1

Utilizamos un editor para abrir el archivo crontab que se encuentra en el directorio /etc

Right Ctrl

Paso 2

Al iniciar crontab vemos un tipo de template que nos muestra como programas las tareas, estas nos permite la programación semanal, mensual, anual, o personalizadas a través de parámetros establecidos. En este caso para demostrar su funcionamiento y las diferente formas de programar las tareas, ejecutaremos el backup.sh cada minuto simplemente colocando * en todas las opciones de tiempo, le colocamos root como usuario, y asignamos la dirección de nuestro script. De esta manera lo programamos cada minuto.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Terminal

root@localhost:/etc

File Edit View Search Terminal Help

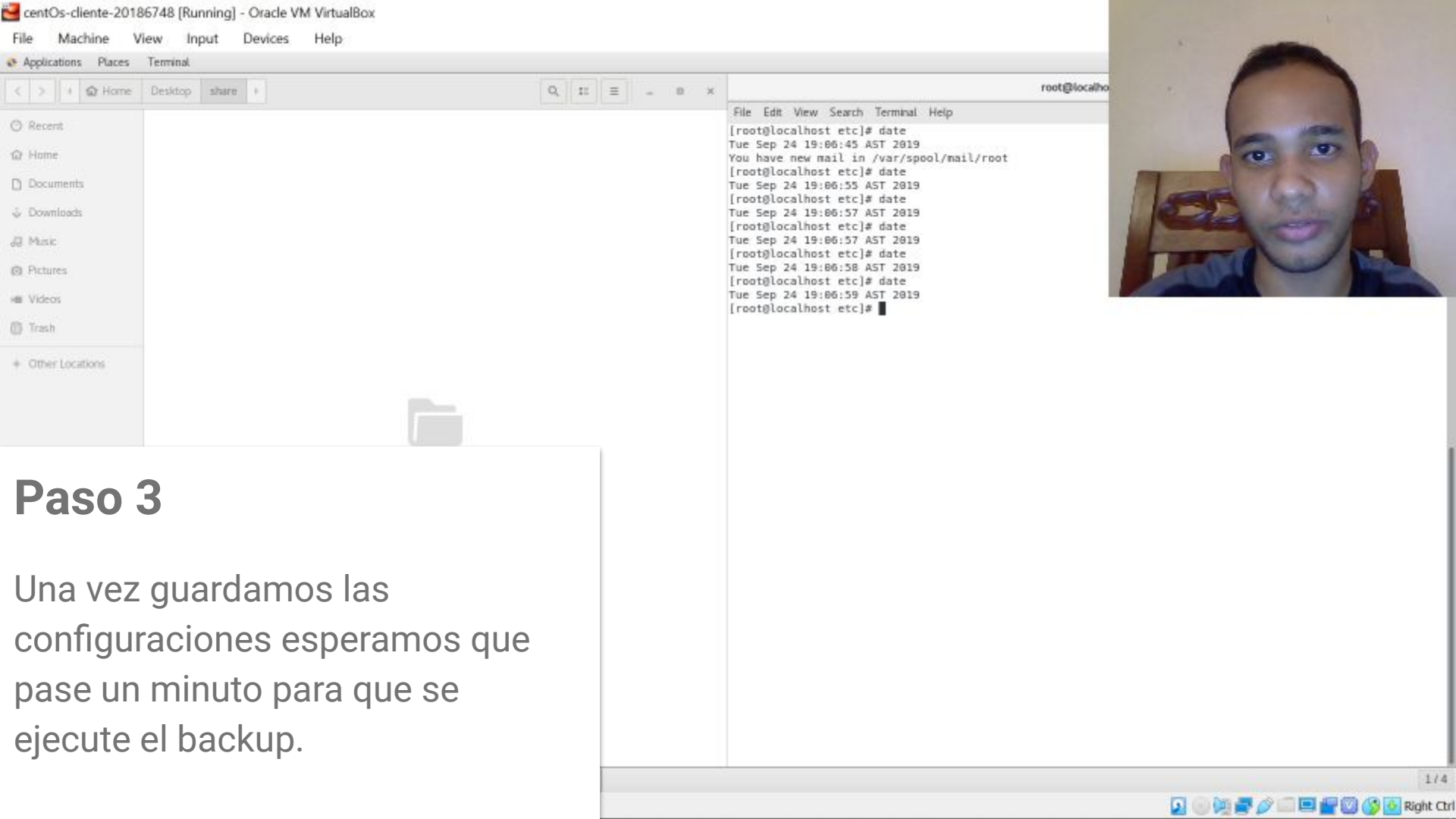
GNU nano 2.3.1 File: crontab

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root

# For details see man 4 crontabs

# Example of job definition:
# .---------------- minute (0 - 59)
# |  .------------- hour (0 - 23)
# |  |  .---------- day of month (1 - 31)
# |  |  |  .------- month (1 - 12) OR jan,feb,mar,apr ...
# |  |  |  |  .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# |  |  |  |  |
# *  *  *  *  * user-name  command to be executed
  *  *  *  *  * root       ~/Desktop/scrt/backup.sh
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

root@localhost:/etc [share] scrt 1 / 4

Right Ctrl



Paso 3

Una vez guardamos las configuraciones esperamos que pase un minuto para que se ejecute el backup.

centOs-cliente-20186748 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications Places Files

Home Desktop share

Recent
Home
Documents
Downloads
Music
Pictures
Videos
Trash
Other Locations

backup-24-Sep-19.tar.gz

root@localhost

File Edit View Search Terminal Help

```
[root@localhost etc]# date
Tue Sep 24 19:06:45 AST 2019
You have new mail in /var/spool/mail/root
[root@localhost etc]# date
Tue Sep 24 19:06:55 AST 2019
[root@localhost etc]# date
Tue Sep 24 19:06:57 AST 2019
[root@localhost etc]# date
Tue Sep 24 19:06:57 AST 2019
[root@localhost etc]# date
Tue Sep 24 19:06:58 AST 2019
[root@localhost etc]# date
Tue Sep 24 19:06:59 AST 2019
[root@localhost etc]# date
Tue Sep 24 19:07:00 AST 2019
[root@localhost etc]#
```

z" selected (773 bytes)

1 / 4

Right Ctrl

Y como vemos ,este se ha ejecutado exactamente después de un minuto. Esto significa que los parámetros establecidos fueron correctos