



Instituto Politécnico Nacional Escuela Superior de Cómputo

Desarrollo de Sistemas Distribuidos
Actividad: Instalación de NFS en la nube
Curso impartido por el profesor: Pineda Guerrero Carlos
Grupo: 4CV1
21/01
Alumno: Adrian González Pardo



Ultima fecha modificado: 16 de diciembre de 2020

1. Desarrollo

Para esta práctica si bien no necesitamos de ningún lenguaje para programar estas tareas, nos apoyamos de un script en bash para la realización de la instalación configuración y muestra de datos solicitados en la práctica

2. Códigos y scripts para funcionamiento

2.1. Script ejemplo clase:

```
1 #!/usr/bin/env bash
2
3 # @author Adrian Gonzalez Pardo
4 if [ $# -ne 3 ];then
5     echo "Usage script $0 <IP-Server> <IP-Client1> <IP-Client2>"
6     exit 1
7 fi
8 server=$1
9 client1=$2
10 client2=$3
11 user="adrian"
12 password="adrianPardo_99"
13
14 echo "Instalacion y configuracion remota de software y servidor"
15 sshpass -p ${password} ssh ${user}@${server} "sudo apt update && sudo apt
    install nfs-kernel-server -y && sudo mkdir /var/nfs/prueba -p && sudo ls -l
    /var/nfs && sudo chown nobody:nogroup /var/nfs/prueba && sudo chmod 777 /
    var/nfs/prueba && sudo ls -l /var/nfs && sudo echo -e '/var/nfs/prueba ${
    client1}(rw,sync,no_subtree_check)\n/var/nfs/prueba ${client2}(rw,sync,
    no_subtree_check)' | sudo tee -a /etc/exports && sudo exportfs -ra && sudo
    exportfs && sudo systemctl restart nfs-kernel-server && touch /var/nfs/
    prueba/texto.txt"
16 echo "Instalacion de clientes con IP ${client1} y ${client2}"
17 sshpass -p ${password} ssh ${user}@${client1} "sudo apt update && sudo apt
    install nfs-common -y && sudo mkdir -p /nfs/prueba && sudo mount -v -t nfs $
    {server}:/var/nfs/prueba /nfs/prueba && echo -e "\t\tTexto del cliente 1 con
    IP: ${client1}\n\t\tEsta es una prueba cliente 1\" > /nfs/prueba/texto.txt"
18 var=$(sshpass -p ${password} ssh ${user}@${server} "cat /var/nfs/prueba/texto.
    txt")
19 sshpass -p ${password} ssh ${user}@${client2} "sudo apt update && sudo apt
    install nfs-common -y && sudo mkdir -p /nfs/prueba && sudo mount -v -t nfs $
    {server}:/var/nfs/prueba /nfs/prueba && echo -e "\t\tTexto del cliente 2 con
    IP: ${client2}\n\t\tEsta es una prueba cliente 2\" >> /nfs/prueba/texto.txt"
20 var2=$(sshpass -p ${password} ssh ${user}@${server} "cat /var/nfs/prueba/texto.
    txt")
21 echo -e "Texto obtenido del cliente 1 en archivo del server nfs:\n${var}"
22 echo -e "Texto obtenido del cliente 1 y cliente dos en archivo server nfs:\n${
    var2}"
```

2.2. Script parte 1:

```
1 #!/usr/bin/env
2
3 # @author Adrian Gonzalez Pardo
4 if [ $# -ne 3 ];then
5     echo "Usage script $0 <IP-Server> <IP-Client1> <IP-Client2>"
6     exit 1
7 fi
8 LB='\033[1;34m'
9 N='\033[0m'
10 server=$1
11 client1=$2
12 client2=$3
13 echo "Conexion por primera vez a las vm"
```

```

14 user="adrian"
15 password="adrianPardo_99"
16 directorio_servidor="/var/nfs/servidor"
17 directorio_cliente="/nfs/cliente"
18 echo "Instalacion y configuracion remota de software y servidor"
19 sshpass -p ${password} ssh ${user}@${server} "sudo apt update && sudo apt
    install nfs-kernel-server -y && sudo mkdir ${directorio_servidor} -p && sudo
    ls -l /var/nfs && sudo chown nobody:nogroup ${directorio_servidor} && sudo
    chmod 777 ${directorio_servidor} && sudo ls -l /var/nfs && sudo echo -e '${
    directorio_servidor} ${client1}(rw, sync, no_subtree_check)\n${
    directorio_servidor} ${client2}(rw, sync, no_subtree_check)' | sudo tee -a /
    etc/exports && sudo exportfs -ra && sudo exportfs && sudo systemctl restart
    nfs-kernel-server && sudo systemctl enable nfs-kernel-server"
20 echo -e "Instalacion de clientes con IP ${LB}${client1}${N} y ${LB}${client2}${N}
    )"
21 sshpass -p ${password} ssh ${user}@${client1} "sudo apt update && sudo apt
    install nfs-common -y && sudo mkdir -p ${directorio_cliente} && sudo mount -
    v -t nfs ${server}:${directorio_servidor} ${directorio_cliente}"
22 sshpass -p ${password} ssh ${user}@${client2} "sudo apt update && sudo apt
    install nfs-common -y && sudo mkdir -p ${directorio_cliente} && sudo mount -
    v -t nfs ${server}:${directorio_servidor} ${directorio_cliente}"
23 echo -e "Creacion del archivo texto.txt en carpeta\nPrimero ls del servidor del
    cliente 1"
24 server_cmd=$(sshpass -p ${password} ssh ${user}@${server} "ls -la ${
    directorio_servidor}")
25 cliente_cmd=$(sshpass -p ${password} ssh ${user}@${client1} "ls -la ${
    directorio_cliente}")
26 msg_cliente1="esta es una prueba de NFS"
27 echo -e "Output de server\n${LB}${server_cmd}${N}\nOutput de cliente 1\n${LB}${
    cliente_cmd}${N}\n"
28 cliente_cmd=$(sshpass -p ${password} ssh ${user}@${client1} "echo ${msg_cliente1
    } > ${directorio_cliente}/texto.txt")
29 server_cmd=$(sshpass -p ${password} ssh ${user}@${server} "ls -la ${
    directorio_servidor}")
30 cliente_cmd=$(sshpass -p ${password} ssh ${user}@${client1} "ls -la ${
    directorio_cliente}")
31 echo -e "El cliente 1 escribio archivo, verificacion de existencia de archivo\
    nOutput de server\n${LB}${server_cmd}${N}\nOutput de cliente 1\n${LB}${
    cliente_cmd}${N}\nCreacion de archivo con mensaje del cliente 1 al archivo
    servidor ${LB}${msg_cliente1}${N}\nParecido al comando more se llama el
    comando cat para imprimir el texto.txt en cliente 2"
32 cliente_cmd=$(sshpass -p ${password} ssh ${user}@${client2} "cat ${
    directorio_cliente}/texto.txt")
33 echo -e "Contenido del archivo en el cliente 2:\n${LB}${cliente_cmd}${N}"
34 echo -e "Montando el contenido de los clientes y el servidor de forma automatica
    cuando realiza el boot"
35 sshpass -p ${password} ssh ${user}@${client1} "echo -e \"${server}:${
    directorio_servidor}\t${directorio_cliente}\tnfs\tdefaults\t0\t0\" | sudo
    tee -a /etc/fstab "
36 sshpass -p ${password} ssh ${user}@${client2} "echo -e \"${server}:${
    directorio_servidor}\t${directorio_cliente}\tnfs\tdefaults\t0\t0\" | sudo
    tee -a /etc/fstab "
37 cliente_cmd=$(sshpass -p ${password} ssh ${user}@${client1} "sudo cat /etc/fstab
    ")
38 echo -en "Imprimiendo contenido de fstab de cliente 1 y cliente 2\nCliente 1:\n$
    {LB}${cliente_cmd}${N}\n"
39 cliente_cmd=$(sshpass -p ${password} ssh ${user}@${client2} "sudo cat /etc/fstab
    ")
40 echo -e "Cliente 2:\n${LB}${cliente_cmd}${N}\nReboot en ambas maquinas..."
41 sshpass -p ${password} ssh ${user}@${client1} "sudo reboot"
42 sshpass -p ${password} ssh ${user}@${client2} "sudo reboot"
43 echo "Lo siguiente continua en el siguiente script"

```

2.3. Script parte 2:

```

1 #!/usr/bin/env bash
2

```

```

3 # @author Adrian Gonzalez Pardo
4 if [ $# -ne 3 ];then
5     echo "Usage script $0 <IP-Server> <IP-Client1> <IP-Client2>"
6     exit 1
7 fi
8 LB='\033[1;34m'
9 N='\033[0m'
10 server=$1
11 client1=$2
12 client2=$3
13 user="adrian"
14 password="adrianPardo_99"
15 directorio_servidor="/var/nfs/servidor"
16 directorio_cliente="/nfs/cliente"
17 msg_cliente2="estamos agregando texto al archivo"
18 cliente_cmd=$(sshpass -p ${password} ssh ${user}@${client1} "cat ${
19     directorio_cliente}/texto.txt")
20 echo -ne "Para evitar uso de more por script remoto se realiza un cat directo en
21     el archivo texto.txt por cliente 1 y cliente 2\nCliente 1:\n${LB}${
22     cliente_cmd}${N}"
23 cliente_cmd=$(sshpass -p ${password} ssh ${user}@${client2} "cat ${
24     directorio_cliente}/texto.txt")
25 echo -e "Cliente 2:\n${LB}${cliente_cmd}${N}\nAgrega nuevo mensaje desde el
26     cliente 2\n${LB}${msg_cliente2}${N}\nDesplegar el mensaje desde cliente 1 y
27     cliente 2 del archivo de texto.txt"
28 cliente_cmd=$(sshpass -p ${password} ssh ${user}@${client2} "echo ${msg_cliente2
29     } >>${directorio_cliente}/texto.txt")
30 cliente_cmd=$(sshpass -p ${password} ssh ${user}@${client1} "cat ${
31     directorio_cliente}/texto.txt")
32 echo -e "Contenido en cliente 1:\n${LB}${cliente_cmd}${N}\nEliminacion de
33     archivo desde cliente 1"
34 cliente_cmd=$(sshpass -p ${password} ssh ${user}@${client1} "rm ${
35     directorio_cliente}/texto.txt")
36 echo -e "ls en ambos clientes:"
37 cliente_cmd=$(sshpass -p ${password} ssh ${user}@${client1} "ls -la ${
38     directorio_cliente}")
39 echo -e "Cliente 1:\n${LB}${cliente_cmd}${N}\nCliente 2:\n"
40 cliente_cmd=$(sshpass -p ${password} ssh ${user}@${client2} "ls -la ${
41     directorio_cliente}")
42 echo -e "${LB}${cliente_cmd}${N}\nFin de practica"

```

3. Capturas

3.1. VM

This screenshot shows the initial steps of creating a new virtual machine in the Azure portal. The interface includes several dropdown menus and a checkbox. The first dropdown is set to 'Azure para estudiantes'. The second is 'Distribuidos'. Below it is a link 'Crear nuevo'. The third dropdown is 'azureClient2' with a green checkmark. The fourth is '(US) Centro-Sur de EE. UU.'. The fifth is 'No se requiere redundancia de la infraestructura'. The sixth is 'Ubuntu Server 18.04 LTS - Gen1' with a link 'Examinar todas las imágenes públicas y privadas'. Below this is a checkbox labeled 'Virtualizar un sistema operativo existente' which is currently unchecked. The final dropdown is 'Standard_B1s - 1 vcpu, 1 GiB de memoria (MXN 176.11/mes)' with a link 'Seleccionar tamaño'.

Figura 1: Casillas de selección para la creación de la VM

This screenshot shows the 'Cuenta de administrador' section of the VM creation process. It includes two radio buttons for 'Tipo de autenticación': 'Clave pública SSH' and 'Contraseña', with 'Contraseña' selected. Below are three input fields: 'Nombre de usuario' with the value 'adrian', 'Contraseña' (masked with dots), and 'Confirmar contraseña' (also masked with dots). Each input field has a green checkmark on the right.

Figura 2: Casillas de selección para la creación de la VM

This screenshot shows the 'Discos' tab in the VM creation process. At the top is a navigation bar with tabs: 'Datos básicos', 'Discos' (active), 'Redes', 'Administración', 'Opciones avanzadas', 'Etiquetas', and 'Revisar y crear'. Below is a paragraph explaining that VMs have a system disk and a temporary disk, and that the VM size determines the storage type. A link 'Más información' is provided. Under 'Opciones de disco', there is a dropdown for 'Tipo de disco del sistema operativo' set to 'HDD estándar', with a detailed note about disk compatibility and performance. Below that is a dropdown for 'Tipo de cifrado' set to '(Predeterminado) Cifrado en reposo con una clave administrada por la pl...'.

Figura 3: Casillas de selección para la creación de la VM

Configure las opciones de supervisión y administración de la VM.

Azure Security Center

Azure Security Center proporciona características unificadas de administración de la seguridad y protección contra amenazas en todas las cargas de trabajo de nube híbrida. [Más información](#)

✓ La suscripción está protegida por el plan básico de Azure Security Center.

Supervisión


Diagnósticos de arranque  ☐ Habilitar con la cuenta de almacenamiento administrada (recomendado) ☐ Habilitar con la cuenta de almacenamiento personalizada ☒ Deshabilitar

Figura 4: Casillas de selección para la creación de la VM

Prioridad	Nombre	Puerto	Protocolo	Origen	Destino	Acción	
300	SSH	22	TCP	Cualquiera	Cualquiera	✓ Permitir	...
310	Port_NFS	2049	TCP	Cualquiera	Cualquiera	✓ Permitir	...
65000	AllowVnetInBound	Cualquiera	Cualquiera	VirtualNetwork	VirtualNetwork	✓ Permitir	...
65001	AllowAzureLoadBalancer...	Cualquiera	Cualquiera	AzureLoadBalancer	Cualquiera	✓ Permitir	...
65500	DenyAllInBound	Cualquiera	Cualquiera	Cualquiera	Cualquiera	✗ Denegar	...

Figura 5: Configuración de puertos para funcionamiento

3.2. En ejecución

```
Warning: Permanently added '13.65.207.251' (ECDSA) to the list of known hosts.
adrian@13.65.207.251's password:
Permission denied, please try again.
adrian@13.65.207.251's password:
Permission denied, please try again.
adrian@13.65.207.251's password:
adrian@13.65.207.251: Permission denied (publickey,password).
The authenticity of host '13.65.119.124 (13.65.119.124)' can't be established.
ECDSA key fingerprint is SHA256:DRR8PAuwVHvq+4RDJ/bFZGGZwXj6LDjEjPfdP040sU8.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '13.65.119.124' (ECDSA) to the list of known hosts.
adrian@13.65.119.124's password:
Permission denied, please try again.
adrian@13.65.119.124's password:
Permission denied, please try again.
adrian@13.65.119.124's password:
adrian@13.65.119.124: Permission denied (publickey,password).
The authenticity of host '13.84.146.25 (13.84.146.25)' can't be established.
ECDSA key fingerprint is SHA256:G5BDf8PYIJqYmb9Kdiwm99+YFzHudTCVEKwim3s5sgk.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
```

Figura 6: Conexión inicial al servidor y cliente via ssh y ejecución remota de comandos

```

Reading package lists...
Building dependency tree...
Reading state information...
nfs-common is already the newest version (1:1.3.4-2.1ubuntu5.3).
The following package was automatically installed and is no longer required:
  linux-headers-4.15.0-124
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 14 not upgraded.
mount.nfs: timeout set for Wed Dec 16 21:57:06 2020
mount.nfs: trying text-based options 'vers=4.2,addr=13.65.207.251,clientaddr=10.0.0.6'
mount.nfs: mount(2): Device or resource busy
Texto obtenido del cliente 1 en archivo del server nfs:
  Texto del cliente 1 con IP: 13.65.119.124
  Esta es una prueba cliente 1
Texto obtenido del cliente 1 y cliente dos en archivo server nfs:
  Texto del cliente 1 con IP: 13.65.119.124
  Esta es una prueba cliente 1
  Texto del cliente 2 con IP: 13.84.146.25
  Esta es una prueba cliente 2

```

Figura 7: Ejecución del primer script para la instalación de datos

```

0 upgraded, 0 newly installed, 0 to remove and 14 not upgraded.
mount.nfs: mount(2): Device or resource busy
mount.nfs: timeout set for Thu Dec 17 02:26:59 2020
mount.nfs: trying text-based options 'vers=4.2,addr=13.65.207.251,clientaddr=10.0.0.6'
Creacion del archivo texto.txt en carpeta
Primero ls del servidor del cliente 1
Output de server
total 8
drwxrwxrwx 2 nobody nogroup 4096 Dec 17 02:24
drwxr-xr-x 4 root root 4096 Dec 17 02:05
Output de cliente 1
total 8
drwxrwxrwx 2 nobody nogroup 4096 Dec 17 02:24
drwxr-xr-x 4 root root 4096 Dec 17 02:05
El cliente 1 escribio archivo, verificacion de existencia de archivo
Output de server
total 12
drwxrwxrwx 2 nobody nogroup 4096 Dec 17 02:25
drwxr-xr-x 4 root root 4096 Dec 17 02:05
-rw-rw-r-- 1 adrian adrian 26 Dec 17 02:25 texto.txt
Output de cliente 1
total 12
drwxrwxrwx 2 nobody nogroup 4096 Dec 17 02:25
drwxr-xr-x 4 root root 4096 Dec 17 02:05
-rw-rw-r-- 1 adrian adrian 26 Dec 17 02:25 texto.txt
Creacion de archivo con mensaje del cliente 1 al archivo servidor esta es una prueba de NFS
Parecido al comando more se llama el comando cat para imprimir el texto.txt en cliente 2
Contenido del archivo en el cliente 2:
esta es una prueba de NFS
Montando el contenido de los clientes y el servidor de forma automatica cuando realiza el boot
13.65.207.251:/var/nfs/servidor /nfs/cliente nfs defaults 0 0
13.65.207.251:/var/nfs/servidor /nfs/cliente nfs defaults 0 0

```

Figura 8: Contenido de las siguientes instrucciones del script


```

# CLOUD_IMG: This file was created/modified by the Cloud Image build process
UUID=37a35aa8-cfae-400b-b3cf-d498145b528e
UUID=9A97-7176 /boot/efi vfat defaults,discard
/dev/disk/cloud/azure_resource-part1 /mnt auto defaults,nofail,x-systemd.requires=cloud-init.service,comment=cloudconfig
13.65.207.251:/var/nfs/servidor /nfs/cliente nfs defaults
13.65.207.251:/var/nfs/servidor /nfs/cliente nfs defaults
Cliente 2:
# CLOUD_IMG: This file was created/modified by the Cloud Image build process
UUID=37a35aa8-cfae-400b-b3cf-d498145b528e
UUID=9A97-7176 /boot/efi vfat defaults,discard
/dev/disk/cloud/azure_resource-part1 /mnt auto defaults,nofail,x-systemd.requires=cloud-init.service,comment=cloudconfig
13.65.207.251:/var/nfs/servidor /nfs/cliente nfs defaults
13.65.207.251:/var/nfs/servidor /nfs/cliente nfs defaults
Reboot en ambas maquinas...
Connection to 13.65.119.124 closed by remote host.
Connection to 13.84.146.25 closed by remote host.
Lo siguiente continua en el siguiente script

```

Figura 9: Contenido de las siguientes instrucciones del script en la parte donde añade los datos suficientes al fstab para el montado durante el boot

```

Para evitar uso de more por script remoto se realiza un cat directo en el archivo texto.txt por cliente 1 y cliente 2
Cliente 1:
esta es una prueba de NFS
esta es una prueba de NFS
Agrega nuevo mensaje desde el cliente 2
estamos agregando texto al archivo
Desplegar el mensaje desde cliente 1 y cliente 2 del archivo de texto.txt
Contenido en cliente 1:
esta es una prueba de NFS
estamos agregando texto al archivo
Eliminacion de archivo desde cliente 1
ls en ambos clientes:
Cliente 1:
total 8
drwxrwxrwx 2 nobody nogroup 4096 Dec 17 02:45
drwxr-xr-x 4 root root 4096 Dec 17 02:05
Cliente 2:
total 8
drwxrwxrwx 2 nobody nogroup 4096 Dec 17 02:45
drwxr-xr-x 4 root root 4096 Dec 17 02:05
Fin de practica

```

Figura 10: Ejecución del ultimo script para las ultimas instrucciones despues del reboot sin necesidad de volver a montar los servicios de NFS

4. Conclusiones

El hacer uso de NFS a nivel de sistemas distribuidos son muy importantes ya que nos permiten realizar una modificación directa de datos a través de la idea de cliente servidor, pero con la ayuda o el beneficio de que se pueda realizar vía distribuida para que muchos usuarios puedan acceder al recurso algunas aplicaciones que siguen esta idea para realizar trabajos de forma colaborativa es con aplicaciones como GitDuck o en plataformas de CodeShare que permiten trabajo colaborativo en el mismo espacio de trabajo.