

MPT01. Servicios en la nube

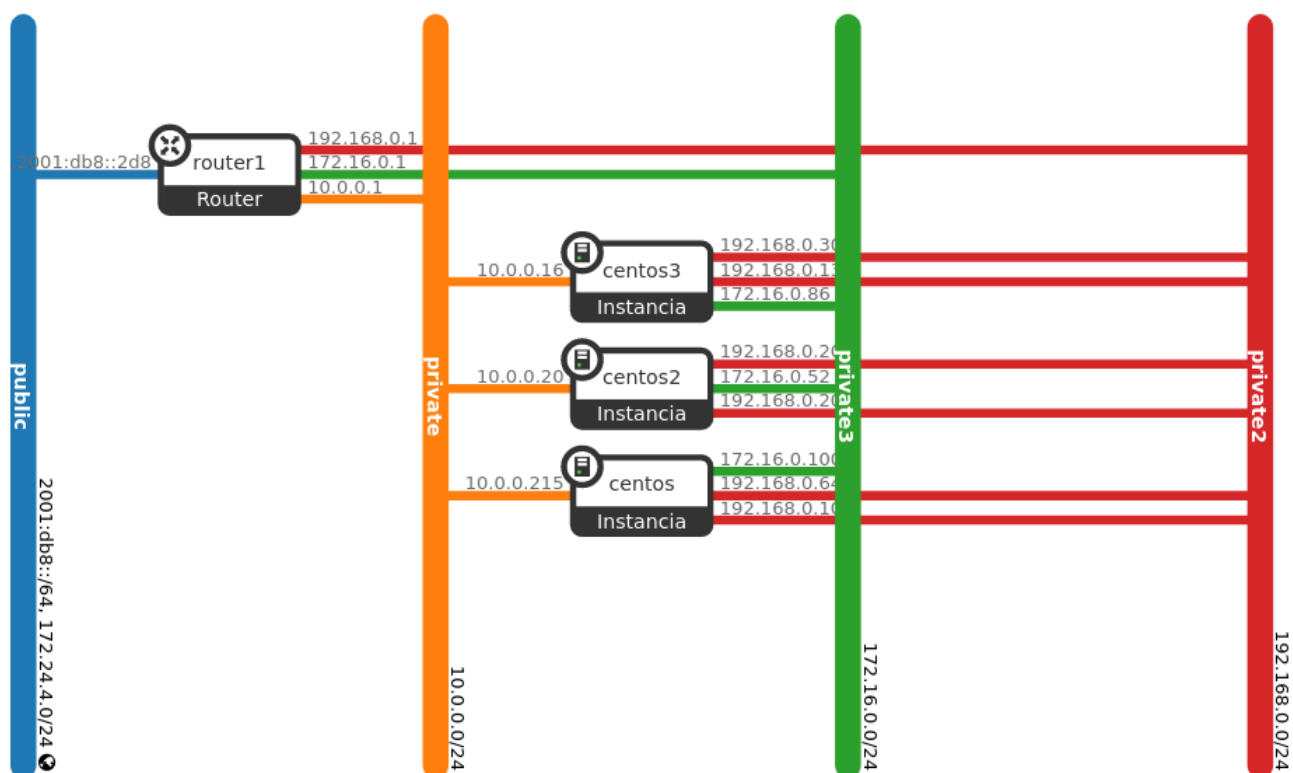
En esta practica vamos a realizar un RAC (Cluster) de una base de datos de Oracle utilizando Openstack. [Para entender mas todo lo que hacemos ver prueba de vida de openstack](#)

Utilizaremos una maquina Centos 7 con extensión .vdi exportada desde VirtualBox.

Principalmente tenemos que preparar el entorno de OpenStack para que podamos empezar a levantar instancias.

1. Preparación OpenStack

Vamos a crear las redes que nos hace falta para el cluster. La topologia de red seria esta:



La cual tendremos primero una red privada 10.0.0.0/24 con la que tendremos la salida a internet dada por el router con ip flotante 172.24.4.0/24

Nombre	private.subnet
ID	3f3c33d7-f7e3-41bb-93e8-784041fbc45b
ID del proyecto	33ed52935cdc4866b06a383d4cd70ff0
Nombre de la red	private
ID de la red	32723e6d-101e-49fa-9fb7-40f59e2cc0e6
Pool de subredes	Ninguno
Versión de IP	IPv4
CIDR	10.0.0.0/24
Pool de IPs asignadas	Arrancar 10.0.0.2 - Fin 10.0.0.254
IP de la puerta de enlace	10.0.0.1
DHCP Habilitado	Sí
Enrutadores adicionales	Ninguno
Servidores DNS	8.8.8.8

Luego tendremos una red privada para tener ip publica y publica vip estas ips nos servirán para el listerner del servicio que estamos dando con el cluster, para el tema de balanceo y para que cuando un nodo se caiga tengamos el siguiente nodo escuchando. La dirección ip seria de 192.168.0.0/24

Nombre	private2.subred
ID	ea0321a5-d9a7-4976-b618-4b2e661df8b0
ID del proyecto	33ed52935cdc4866b06a383d4cd70ff0
Nombre de la red	private2
ID de la red	a604307a-a7cb-49b5-a54c-f9b4dd6137d4
Pool de subredes	Ninguno
Versión de IP	IPv4
CIDR	192.168.0.0/24
Pool de IPs asignadas	Arrancar 192.168.0.2 - Fin 192.168.0.254
IP de la puerta de enlace	192.168.0.1
DHCP Habilitado	Sí
Enrutadores adicionales	Ninguno
Servidores DNS	8.8.8.8

Y la ultima red que creamos seria para la aplicación de control de almacenamiento ASM que tendrfa una ip 172.16.0.0/24

Nombre	private3.subnet
ID	888274cc-bb2a-43d5-94b0-dc2533922ffa
ID del proyecto	33ed52935cdc4866b06a383d4cd70ff0
Nombre de la red	private3
ID de la red	574f6dc8-1f21-45c4-ba23-21af7ec11caa
Pool de subredes	Ninguno
Versión de IP	IPv4
CIDR	172.16.0.0/24
Pool de IPs asignadas	Arrancar 172.16.0.2 - Fin 172.16.0.254
IP de la puerta de enlace	172.16.0.1
DHCP Habilitado	Sí
Enrutadores adicionales	Ninguno
Servidores DNS	8.8.8.8

En resumen cada nodo tendrá un total de 4 interfaces y una ip flotante para salida a internet.

Ahora creamos una un par de claves para poder conectarnos luego a la instancia:

<input type="checkbox"/> > clave1	ssh	d1:ab:90:4c:32:cd:cc:69:43:d9:cf:65:8f:55:bc:d9	Borrar par de claves
-----------------------------------	-----	---	--------------------------------------

También añadimos varias reglas necesarias al grupo de seguridad:

<input type="checkbox"/> Direction	Ether Type	IP Protocol	Port Range	Remote IP Prefix	Remote Security Group	Description	Actions
<input type="checkbox"/> Saliente	IPv4	Cualquier	Cualquier	0.0.0.0/0	-	-	Eliminar Regla
<input type="checkbox"/> Saliente	IPv6	Cualquier	Cualquier	:::0	-	-	Eliminar Regla
<input type="checkbox"/> Entrante	IPv4	Cualquier	Cualquier	-	default	-	Eliminar Regla
<input type="checkbox"/> Entrante	IPv4	ICMP	Cualquier	0.0.0.0/0	-	-	Eliminar Regla
<input type="checkbox"/> Entrante	IPv4	TCP	22 (SSH)	0.0.0.0/0	-	-	Eliminar Regla
<input type="checkbox"/> Entrante	IPv4	TCP	53 (DNS)	0.0.0.0/0	-	-	Eliminar Regla
<input type="checkbox"/> Entrante	IPv6	Cualquier	Cualquier	-	default	-	Eliminar Regla

También creamos un nuevo sabor personalizado para nuestras instancias centos:

<input type="checkbox"/> centos	1	2GB	0GB	0GB	0MB	1,0	1cb8eb5f-044c-4e8b-ab8a-238551f9023e	Sí	no
---------------------------------	---	-----	-----	-----	-----	-----	--------------------------------------	----	--------------------

Ahora que todo esta listo ya podemos levantar las instancias, subiendo la imagen centos 7

también elegir si se utiliza almacenamiento permanente al crear un volumen nuevo.

Seleccionar Origen de arranque

Image

Crear nuevo volumen

Sí

No

Asignados

Mostrando 1 articulo

Nombre	Actualizado	Tamaño	Tipo	Visibilidad	
> centos7	5/11/21 3:42 PM	5.26 GB	VDI	Público	↓

Mostrando 1 articulo

Los valores definen el tamaño que tendrá la instalación respecto a CPU, memoria y almacenamiento.

Asignados

Nombre	VCPUS	RAM	Total de Disco	Disco raíz	Disco efímero	Público	
> centos	1	2 GB	0 GB	0 GB	0 GB	Sí	↓

▼ Asignados 3

Seleccionar redes de las listadas abajo.

	Network	Subredes Asociadas	Compartido	Estado del Administrador	Estado	
↕ 1	> private	private.subnet	No	Arriba	Activo	↓
↕ 2	> private3	private3.subnet	No	Arriba	Activo	↓
↕ 3	> private2	private2.subred	No	Arriba	Activo	↓

▼ Asignados 1

Mostrando 1 articulo

Nombre	Descripción	
> default	Default security group	↓

Mostrando 1 articulo

Asignados

Mostrando 1 artículo

Nombre	Tipo	Fingerprint ↕	
> clave1	ssh	d1:ab:90:4c:32:cd:cc:69:43:d9:cf:65:8f:55:bc:d9	↓

Y así sucesivamente con centos2 (nodo2) y centos3 (nodo3)

Cuando estén las tres instancias levantadas le asignamos una IP flotante.

Al haber creado las redes, automáticamente también hemos configurado las interfaces de las instancias.

2. Configuración de usuarios y grupos

Principalmente decir que cada nodo es una clonación del primero, por lo tanto hay un usuario ya creado llamado grid y a este usuario pues le daremos algunos permisos. Nos conectamos como root y hacemos:

1. Creamos varios grupos poniendo **'groupadd oinstall'** **'groupadd dba'** **'groupadd asmadmin'**
2. Y modificamos el usuario grid poniendo como grupo principal oinstall y lo demas como secundarios para esto ponemos el siguiente comando **'usermod -g oinstall -G dba, asmadmin grid'**
3. Para comprobar lo anterior nos logueamos con grid y ponemos el comando **'id'**
4. Para cambiar el nombre de la maquina y poner nodo1, nodo2, etc debemos utilizar el comando **'hostnamectl set-hostname nombre'** en cada instancia

3. Configurar el DNS

El DNS vamos a configurarlo en el nodo1, esto en la realidad tendria que se con una instancia aparte, pero en nuestro caso vamos a hacerlo en el nodo1.

Entramos en el nodo1 como root por ssh `'ssh -i clave.pem root@172.24.4.73'`

Lo haremos con dnsmasq que ya viene instalado.

Lo primero que haremos es crear el fichero `'/etc/rac.dns'` y dentro pondremos las ips de cada nodo y especificando su nombre:

```
192.168.0.64      nodo1
192.168.0.10     nodo1-vip
10.0.0.215       nodo1-priv
172.16.0.100     nodo1-asm

192.168.0.20     nodo2
192.168.0.200    nodo2-vip
10.0.0.20        nodo2-priv
172.16.0.52      nodo2-asm

192.168.0.30     nodo3
192.168.0.139    nodo3-vip
10.0.0.16        nodo3-priv
172.16.0.86      nodo3-asm

192.168.0.100    scan
192.168.0.110    scan
192.168.0.120    scan
```

Ahora levantamos el servicio de dnsmasq con `'systemctl start dnsmasq'` pero si luego ponemos `'systemctl status dnsmasq'` veremos que ha habido un fallo y no se puede levantar el servicio la forma de resolverlo es matando el proceso el cual este usando el puerto 53 que es el que utiliza dnsmasq. Para ello hacemos esto:

```
[root@nodo1 grid]# lsof -i :53
COMMAND PID  USER  FD  TYPE DEVICE SIZE/OFF NODE NAME
dnsmasq 1609 nobody  5u  IPv4  25050      0t0  UDP nodo1:domain
dnsmasq 1609 nobody  6u  IPv4  25051      0t0  TCP nodo1:domain (LISTEN)
[root@nodo1 grid]# kill 1609
[root@nodo1 grid]# lsof -i :53
[root@nodo1 grid]# systemctl start dnsmasq
[root@nodo1 grid]# systemctl status dnsmasq
● dnsmasq.service - DNS caching server.
   Loaded: loaded (/usr/lib/systemd/system/dnsmasq.service; enabled; vendor pre
  set: disabled)
   Active: active (running) since mar 2021-05-11 20:20:47 CEST; 7s ago
     Main PID: 4589 (dnsmasq)
        Tasks: 1
       CGroup: /system.slice/dnsmasq.service
               └─4589 /usr/sbin/dnsmasq -k
```

Luego configuramos el fichero '/etc/resolv.conf' añadiendo la ip de la maquina:

```
# Generated by NetworkManager
search openstacklocal
nameserver 8.8.8.8
nameserver 192.168.0.10
```

Y ya tendríamos listo el DNS:

```
[root@nodo1 ~]# ping nodo2
PING nodo2 (192.168.0.200) 56(84) bytes of data.
64 bytes from nodo2 (192.168.0.200): icmp_seq=1 ttl=64 time=1.85 ms
64 bytes from nodo2 (192.168.0.200): icmp_seq=2 ttl=64 time=0.934 ms
64 bytes from nodo2 (192.168.0.200): icmp_seq=3 ttl=64 time=1.23 ms
^C
--- nodo2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.934/1.340/1.857/0.385 ms
[root@nodo1 ~]#
```

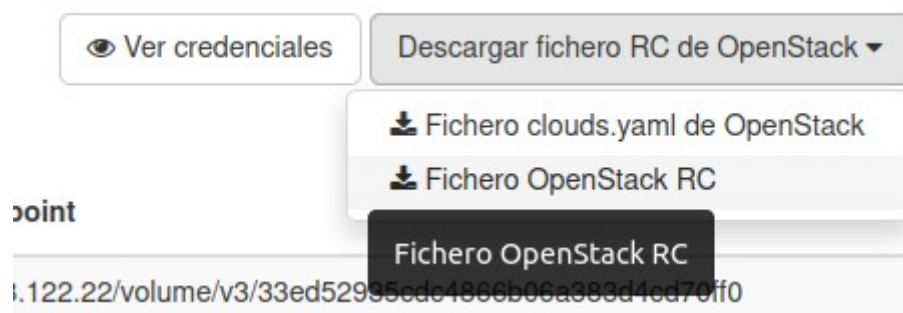
4. Creación de volúmenes compartidos

Ahora vamos a preparar los volúmenes para todos los nodos que iran compartidos. En total son 6 volúmenes; 2 volúmenes de 5GB para clusterware, 2 volúmenes de 25GB para repositorio, 1 volumen de 25 GB para base de datos y 1 volumen de 25GB para FRA.

Antes de crearlo vamos a crear un tipo de volumen que lo vamos a llamar compartido. De primeras OpenStack no hace que este tipo de volumen sea compartido hay que habilitarlo por CLI con este comando:

'cinder type-key compartido set multiattach="<is> True"' Donde compartido es el nombre del tipo

Obviamente antes de meter este comando debemos de descargar esto:



Y poner **'source nombre_fichero'** y nos pedirá la contraseña de admin de OpenStack

Ahora ya con nuestro tipo de volumen ya habilitado para que se puedan asignar a varias instancias. Vamos a crear los volúmenes y le asignamos el tipo compartido.

La creación de volúmenes es muy sencilla por lo tanto no añado captura de pantalla.

<input type="checkbox"/>	repo2	-	25GiB	En-uso	-	compartido
<input type="checkbox"/>	bdd	-	25GiB	En-uso	-	compartido
<input type="checkbox"/>	fra	-	25GiB	En-uso	-	compartido
<input type="checkbox"/>	repo1	-	25GiB	En-uso	-	compartido
<input type="checkbox"/>	clusterware2	-	5GiB	En-uso	-	compartido
<input type="checkbox"/>	clusterware1	-	5GiB	En-uso	-	compartido

Una vez creado para añadir cada volumen a las tres instancias tendremos que hacerlo también por CLI ya que por el dashboard no nos deja.

El comando que tendremos que poner seria **'nova volume-attach id_instancia id_volumen'**

<input type="checkbox"/>	repo2	-	25GiB	En-uso	-	compartido	/dev/vdg en centos, /dev/vdg en centos3, /dev/vdg en centos2
<input type="checkbox"/>	bdd	-	25GiB	En-uso	-	compartido	/dev/vdf en centos2, /dev/vdf en centos, /dev/vdf en centos3
<input type="checkbox"/>	fra	-	25GiB	En-uso	-	compartido	/dev/vde en centos3, /dev/vde en centos, /dev/vde en centos2
<input type="checkbox"/>	repo1	-	25GiB	En-uso	-	compartido	/dev/vdd en centos2, /dev/vdd en centos3, /dev/vdd en centos
<input type="checkbox"/>	clusterware2	-	5GiB	En-uso	-	compartido	/dev/vdc en centos3, /dev/vdc en centos2, /dev/vdc en centos
<input type="checkbox"/>	clusterware1	-	5GiB	En-uso	-	compartido	/dev/vdb en centos, /dev/vdb en centos2, /dev/vdb en centos3

5. Configuración de volúmenes en la instancia

Ahora con cada volumen vamos a crear una particion primaria.

Primero listamos los discos poniendo `fdisk -l` y luego pues vamos haciendo esto:

- `'fdisk /dev/vdb'`

- n (nueva particion)
- p (particion primaria)
- 1
- ENTER
- ENTER

Y así con todos los disco que van desde vdb-g

Despues de haber hecho esto, tenemos que añadir un comando que reconozca los volúmenes este comando es 'scsi_id' y para añadirlo hacemos un enlace con el comando 'ln /lib/udev/scsi_id /sbin/scsi_id' con esto luego podremos hacer que ASM pueda manejar los discos.

Bueno este comando lo que hace es que nos devuelve el id que tiene un disco. En nuestro caso este comando no nos devuelve nada se usaria con 'scsi_id /dev/vdb' y saldria el id del disco pero no sale nada

```
[root@nodo1 ~]# scsi_id
No device specified.
[root@nodo1 ~]# scsi_id /dev/vdb
[root@nodo1 ~]#
```

Entonces hemos improvisado un poco ya que podemos obtener el id de nuestros discos en listando el directorio.

```
[root@nodo1 ~]# ls -l /dev/disk/by-id/
total 0
lrwxrwxrwx. 1 root root 10 may 12 04:06 dm-name-centos-home -> ../../dm-2
lrwxrwxrwx. 1 root root 10 may 12 04:06 dm-name-centos-root -> ../../dm-0
lrwxrwxrwx. 1 root root 10 may 12 04:06 dm-name-centos-swap -> ../../dm-1
lrwxrwxrwx. 1 root root 10 may 12 04:06 dm-uuid-LVM-YZZD6lsH03X6Wg9JjYwfX0DNDYaaLiA38lyvgSU9DHCxSEBMSNvRCqVV10gM88g -> ../../dm-0
lrwxrwxrwx. 1 root root 10 may 12 04:06 dm-uuid-LVM-YZZD6lsH03X6Wg9JjYwfX0DNDYaaLiA4Xwi4tqHULfYlNkZqB5pugeggxoeRYKe -> ../../dm-1
lrwxrwxrwx. 1 root root 10 may 12 04:06 dm-uuid-LVM-YZZD6lsH03X6Wg9JjYwfX0DNDYaaLiAzc0JoJcfcAaJ3PFH0cRAuy0JGKyglis7 -> ../../dm-2
lrwxrwxrwx. 1 root root 10 may 12 04:05 lvm-pv-uuid-qx2Ncn-UxT5-L0P6-8QVM-g6fq-eMcj-P7Wqh8 -> ../../vda2
lrwxrwxrwx. 1 root root 9 may 12 04:58 virtio-118242a3-6cbd-4d5b-a -> ../../vdg
lrwxrwxrwx. 1 root root 10 may 12 05:45 virtio-118242a3-6cbd-4d5b-a-part1 -> ../../vdg1
lrwxrwxrwx. 1 root root 9 may 12 04:05 virtio-14afc202-133f-41f8-8 -> ../../vdh
lrwxrwxrwx. 1 root root 9 may 12 04:58 virtio-179e64ea-4703-48e1-9 -> ../../vde
lrwxrwxrwx. 1 root root 10 may 12 04:58 virtio-179e64ea-4703-48e1-9-part1 -> ../../vde1
lrwxrwxrwx. 1 root root 9 may 12 04:58 virtio-284db8a7-906f-4731-8 -> ../../vdf
lrwxrwxrwx. 1 root root 10 may 12 05:44 virtio-284db8a7-906f-4731-8-part1 -> ../../vdf1
lrwxrwxrwx. 1 root root 9 may 12 04:58 virtio-43868912-5146-4a98-b -> ../../vdc
lrwxrwxrwx. 1 root root 10 may 12 04:58 virtio-43868912-5146-4a98-b-part1 -> ../../vdc1
lrwxrwxrwx. 1 root root 9 may 12 04:58 virtio-b0e81cfc-e9e0-49f7-a -> ../../vdb
lrwxrwxrwx. 1 root root 10 may 12 04:58 virtio-b0e81cfc-e9e0-49f7-a-part1 -> ../../vdb1
lrwxrwxrwx. 1 root root 9 may 12 04:58 virtio-e5091f4f-41ec-4123-8 -> ../../vdd
lrwxrwxrwx. 1 root root 10 may 12 04:58 virtio-e5091f4f-41ec-4123-8-part1 -> ../../vdd1
```

¿Para que queremos el id? Esto luego lo añadiremos a un fichero con extensión .rules en el directorio '/etc/udev/rules.d' y llamaremos a ese fichero 99-oracle-asmdevices.rules aquí añadiremos las reglas para localizar los discos.

```
KERNEL == "vdb1", ENV{ID_SERIAL}=="virtio-b0e81cfc-e9e0-49f7-a-part1", SYMLINK+="DISCO1", OWNER="grid", GROUP="asmadmin", MODE="0660"  
KERNEL == "vdc1", ENV{ID_SERIAL}=="virtio-43868912-5146-4a98-b-part1", SYMLINK+="DISCO2", OWNER="grid", GROUP="asmadmin", MODE="0660"  
KERNEL == "vdd1", ENV{ID_SERIAL}=="virtio-e5091f4f-41ec-4123-8-part1", SYMLINK+="DISCO3", OWNER="grid", GROUP="asmadmin", MODE="0660"  
KERNEL == "vde1", ENV{ID_SERIAL}=="virtio-179e64ea-4703-48e1-9-part1", SYMLINK+="DISCO4", OWNER="grid", GROUP="asmadmin", MODE="0660"
```

Al fallarnos el comando 'scsi_id' hemos configurado el fichero de forma que especificamos directamente la id del disco.

Lo que hace esto es que si encuentra un disco con el nombre que especificamos y con la id que especificamos, lo llame DISCOX y como propietario grid y el grupo

Para actualizar las reglas ponemos primero 'udevadm control -reload-rules' y luego 'systemctl restart systemd-udev' y por ultimo 'udevadm trigger'

Ahora vemos que se ha creado en '/dev' los DISCOSX en nuestro caso no se crean pero como al fin y al cabo lo que realmente se crea son enlaces, si mas para delante nos da problema solo habra que crear enlaces de las particiones y llamarlas DISCOX.

Pasamos el fichero 99-oracle-asmdevices.rules al nodo2 y al nodo3 con 'scp 99-oracle-asmdevices.rules nodo3:/etc/udev/rules.d/'

```
[root@nodo1 rules.d]# scp 99_oracle_asmdevice.rules nodo3:/etc/udev/rules.d/  
root@nodo3's password:  
99_oracle_asmdevice.rules          100% 537   98.7KB/s   00:00  
[root@nodo1 rules.d]# scp 99_oracle_asmdevice.rules nodo2:/etc/udev/rules.d/  
root@nodo2's password:  
99_oracle_asmdevice.rules          100% 537  116.6KB/s   00:00  
[root@nodo1 rules.d]#
```

6. Instalación ASM

Ahora vamos a instalar ASM para ello necesitamos estos tres repositorios:

```
kmod-oracleasm-2.0.8-19.el7.x86_64.rpm  
oracleasm-lib-2.0.12-1.el7.x86_64.rpm  
oracleasm-support-2.1.8-3.el7.x86_64.rpm
```

Y instalamos cada uno de ellos con `'rpm -ivh nombre_paquete'`

Luego ya podremos configurar ASM poniendo `'oracleasm configure -i'` nos pedirá usuario en este caso `'grid'` y grupo que sera `'asmadmin'` y luego ponemos `'y'`.

Para levantar el modulo ponemos `'oracleasm init'` y creamos los discos de asm con `'oracleasm createdisk DISCO5 /dev/vdf1'` y `'oracleasm createdisk DISCO6 /dev/vdg1'`

Con nodo2 y con nodo3 hacemos la misma instalacion y al arrancar oracleasm solo tendremos que poner `'oracleasm scandisks'`