

MPT01. Servicios en la nube

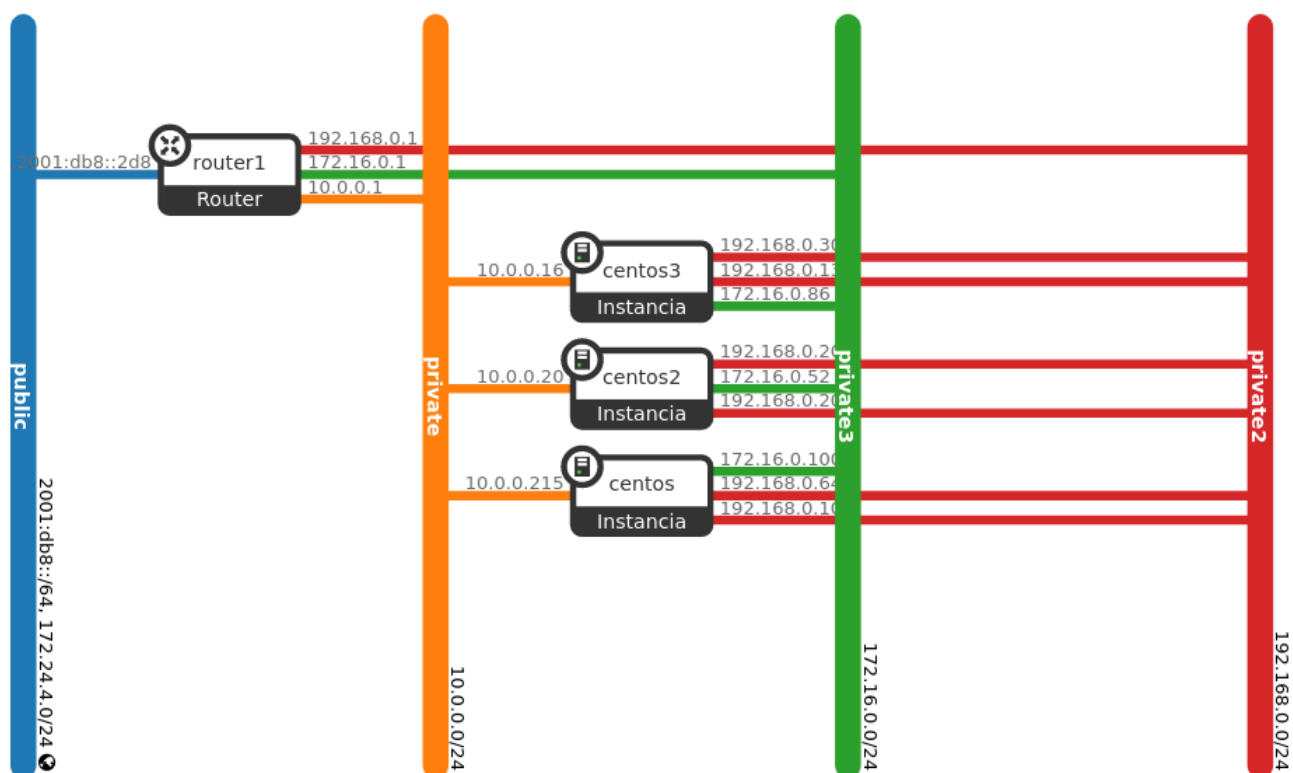
En esta practica vamos a realizar un RAC (Cluster) de una base de datos de Oracle utilizando Openstack. [Para entender mas todo lo que hacemos ver prueba de vida de openstack](#)

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 ~]#
```

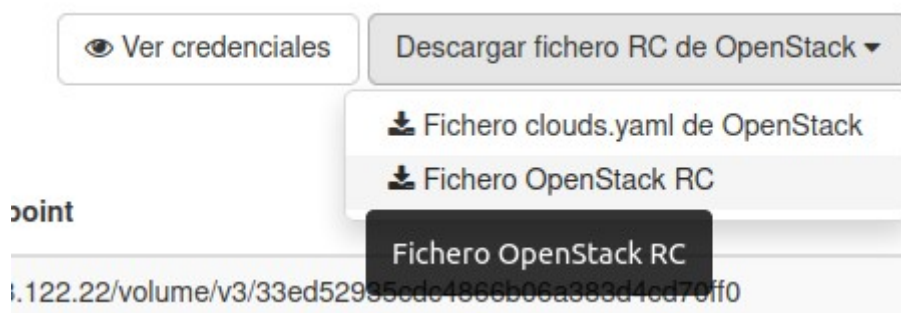
4. Creación de volúmenes compartidos

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-YZZD6lsH03X6Wg9JjjYwfX0DNDYaaLiA38lyvgSU9DHCxSEBMSNvRCqVV10gM88g -> ../../dm-0
lrwxrwxrwx. 1 root root 10 may 12 04:06 dm-uuid-LVM-YZZD6lsH03X6Wg9JjjYwfX0DNDYaaLiA4Xwi4tqHULfYlNkZqB5pugeggxoeRYKe -> ../../dm-1
lrwxrwxrwx. 1 root root 10 may 12 04:06 dm-uuid-LVM-YZZD6lsH03X6Wg9JjjYwfX0DNDYaaLiAzc0JoJcfcAaJ3PFH0cRAuy0JGKyglis7 -> ../../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+="DISC01", OWNER="grid", GROUP="asmadmin", MODE="0660"  
KERNEL == "vdc1", ENV{ID_SERIAL}=="virtio-43868912-5146-4a98-b-part1", SYMLINK+="DISC02", OWNER="grid", GROUP="asmadmin", MODE="0660"  
KERNEL == "vdd1", ENV{ID_SERIAL}=="virtio-e5091f4f-41ec-4123-8-part1", SYMLINK+="DISC03", OWNER="grid", GROUP="asmadmin", MODE="0660"  
KERNEL == "vde1", ENV{ID_SERIAL}=="virtio-179e64ea-4703-48e1-9-part1", SYMLINK+="DISC04", 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'`

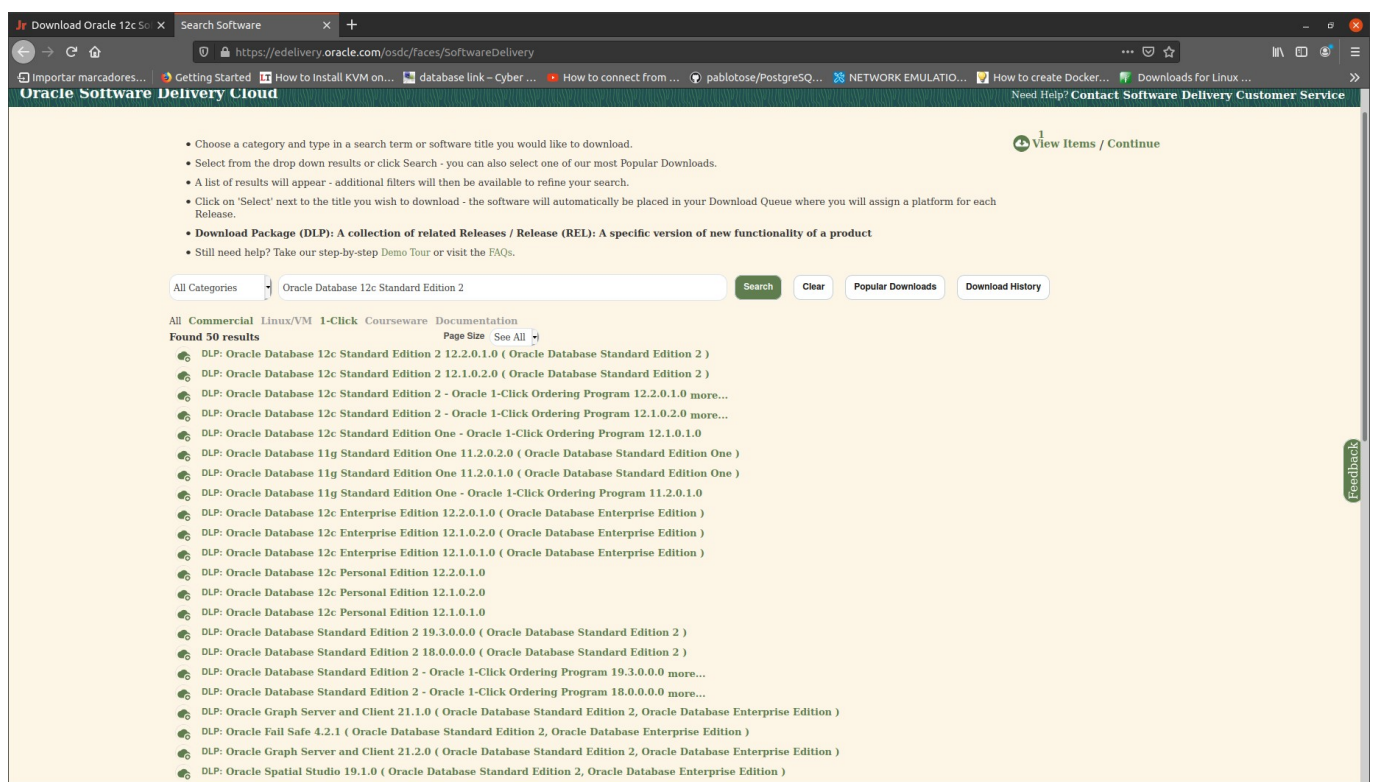
Luego ya podremos configurar ASM poniendo `'oracleasm configure -i'` nos pedirá usuario en este caso `'grid'` y grupo que sera `'asmadmin'` y luego ponemos `'y'`.

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'`

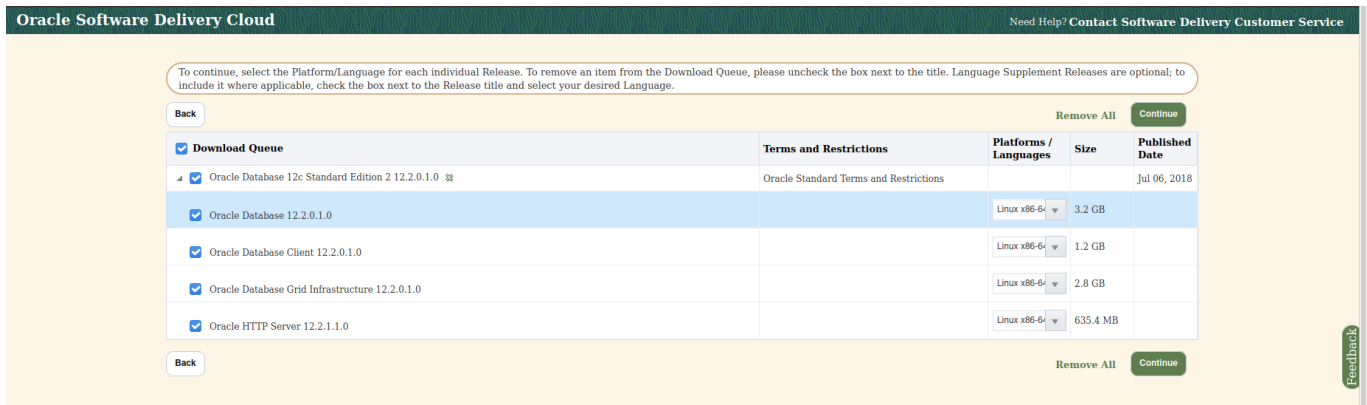
7. Instalación de Oracle 12c e Infraestructure grid

Teniendo todos los volúmenes listos vamos a proceder a instalar el software que necesitará nuestro cluster para funcionar adecuadamente. Para ello nos vamos a la página oficial de Oracle y, en el filtro, se hará una búsqueda de Oracle Database 12c Standart Edition (la segunda):



La seleccionamos y escogemos todas las funciones que queramos implantar:

Aceptamos las condiciones que se nos ofrece y podremos descargarlas.



Una vez descargado, en nuestro nodo1 creamos una carpeta donde solo vayamos a tener el software que vamos a instalar. Hacemos unzip a todos ellos y se nos generará estas dos carpetas:

```
[grid@node1 software]$ ls
database  grid
[grid@node1 software]$ ls -l
```

Dentro de la carpeta grid debemos de tener una serie de archivos. Ejecutaremos el siguiente:

```
[grid@node1 grid]$ ls
addnode      deinstall      inventory      opmn           qos            srvn
assistants   demo           javavm         oracore        racg           suptools
bin          diagnostics    jdbc          ord            rdbms         tomcat
cdata        dmdu           jdk            ordim          relnotes      ucp
cha          env.ora        jlib          ords           rhp           usm
clone        evm            ldap          oss            root.sh       utl
crs          gpn           lib           oui            rootupgrade.sh welcome.html
css          gridSetup.sh  log           owm            runcluvfy.sh  wlm
cv           has           md            perl           scheduler     wwg
dbjava       hs            network       plssql        slax          xag
dbs          install       nls           precomp       sqlpatch     xdk
dc_ocm       instantclient OPatch        QOpatch       sqlplus
```

```
[grid@node1 grid]$ ./gridSetup.sh
```

Esto nos instalará el clusterware.

Se nos iniciará una interfaz de instalación y debemos seguir los siguientes pasos:

- En primer lugar elegimos la opción de “Configurar Oracle Grid Infraestructure para un Nuevo Cluster”
- Seguidamente, seleccionamos configurar un Cluster Estándar.
- Introducimos los datos de nuestro cluster, donde el nombre debe ser el que pusimos en nuestro DNS

Single Client Access Name (SCAN) allows clients to use one name in connection strings to connect to the cluster as a whole. Client connect requests to the SCAN name can be handled by any cluster node.

Cluster Name:

SCAN Name:

SCAN Port:

- A la hora de añadir los nodos que forman parte del cluster añadimos su nombre público, rol HUB y Hostname virtual el de la dirección IP virtual VIP que atribuimos en su momento a los nodos. Además han de tener mismo usuario (grid) y contraseña:

Provide the list of nodes to be managed by Oracle Grid Infrastructure with their Public Hostname and Virtual Hostname.

Public Hostname	Role	Virtual Hostname
node1	HUB	node1-vip
node2	HUB	node2-vip

SSH connectivity... Use Cluster Configuration File... Add... Edit... Remove

OS Username: OS Password:

- Cuando terminemos le damos a Setup y se nos terminará de configurar los nodos generando un directorio llamado ‘.ssh’ con sus claves públicas y privadas. Se puede probar haciendo ssh directamente con el nombre del nodo:

```
[grid@node1 .ssh]$ ssh node2
Last login: Sun Jan 28 04:59:12 2018
[grid@node2 ~]$ pwd
/home/grid
```


- Ahora toca configurar las interfaces de red de los nodos. Alguna puede que se haya desconfigurado, así que volveremos a ponerlas como estaban:

Private interfaces are used by Oracle Grid Infrastructure for internode traffic.

Interface Name	Subnet	Use for
enp0s3	10.0.2.0	Do Not Use
enp0s8	192.168.0.0	Public
enp0s9	10.0.0.0	Private
enp0s10	172.17.0.0	ASM
virbr0	192.168.122.0	Do Not Use

- En la pestaña de Opciones de Almacenamiento elegimos la casilla de Configurar ASM usando bloques de dispositivos.

- En la pregunta de que si queremos crear un ASM separado le damos a que 'sí'.

- A continuación, en la creación de grupos de discos ASM. Nos deben salir los discos que no forman parte del ASM. Configuramos para que salgan todos los discos añadiendo el filtro correspondiente. También añadimos la opción de redundancia normal:

Configuration Option

Cluster Configuration

Grid Plug and Play

Cluster Node Information

Network Interface Usage

Storage Option

Grid Infrastructure Managem

Create ASM Disk Group

ASM Password

Operating System Groups

Installation Location

Root script execution

Prerequisite Checks

Summary

OCR and Voting disk data will be stored in the following ASM Disk group. Select disks and characteristics of this Disk group.

Disk group name: DATA

Redundancy: ☐ Flex ☐ High ☒ Normal ☐ External

Allocation Unit Size: 4 MB

Select Disks:

	Disk Path	Size (in MB)	Status	Failure Group
<input type="checkbox"/>	/dev/DISK01	5119	Candidate	
<input type="checkbox"/>	/dev/DISK02	5119	Candidate	
<input type="checkbox"/>	/dev/DISK03	25599	Candidate	
<input type="checkbox"/>	/dev/DISK04	15359	Candidate	
<input type="checkbox"/>	/dev/oracleasm/disks/DISK05	25599	Provisioned	
<input type="checkbox"/>	/dev/oracleasm/disks/DISK06	25599	Provisioned	

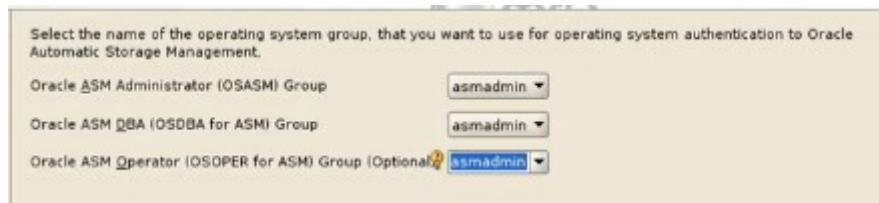
- En el grupo de fallo debemos añadirle los grupos de fallo.

- Configuramos también el grupo de discos GIMR. No hace falta tocar nada, solo darle a siguiente.

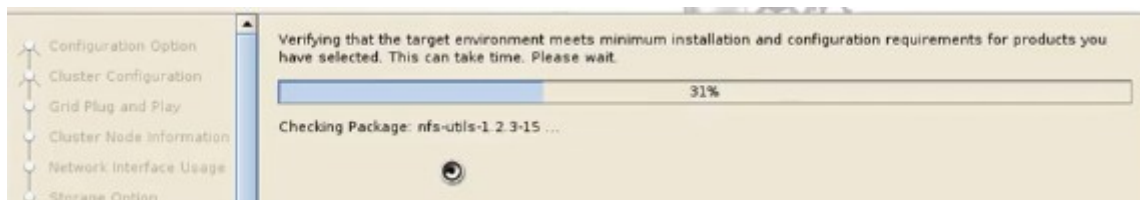
- Establecemos nuestra contraseña para el ASM.

- En los Fallos de Isolación, elegimos la opción de no usar IPMI o la Interfaz de Administración de Plataforma Inteligente, ya que en nuestro caso no nos conviene.

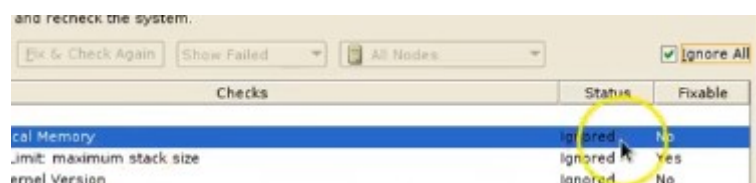
- En cuanto a Opciones de Administración, tenemos la opción de registrarnos con Enterprise Manager Cloud Control. Como nosotros no estamos registrados podemos prescindir de esta parte.
- Para la siguiente parte, elegiremos los grupos que se encargaran de gestionar el sistema. El grupo administrador para todas las opciones será 'asmadmin':



- Seguimos con la ruta de instalación de nuestro Oracle Infrastructure Grid.
- Continuamos especificando el directorio de inventario, donde se añadirán todos los recursos de Oracle instalados en nuestra máquina.
- Nosotros no poseemos un script de configuración, por lo que prescindiremos del siguiente paso. Nos encargaremos de eso más adelante.
- Antes de seguir, se hará una comprobación de que se cumplen los prerequisites:



Y nos saldrá un log con todos los errores que tendremos que revisar. Algunos se podrán arreglar, otros son prescindibles. Señalamos la casilla para ignorar a los restantes:



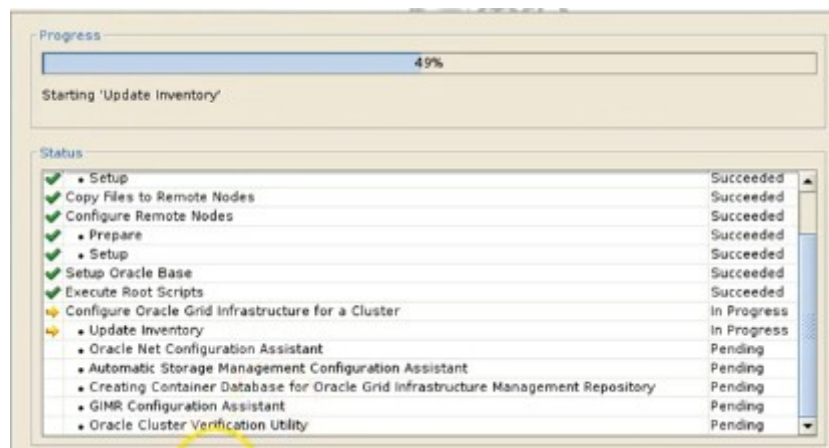
- Luego nos aparecerá un resumen donde saldrá todo lo que vamos a instalar.
- Para finalizar, le damos al botón de instalar y empezará la instalación.

Una vez llegado a cierto punto, todos los archivos de configuración pasarán también a los nodos.

Así, la infraestructura del grid ya estará así instalado.

El software nos pedirá ahora que ejecutemos un archivo llamado `./oraInstRoot.sh` que estará dentro del directorio `/app/oraInventory/` se que creó en la Instalación como root en ambos nodos.

Después de esta acción la instalación deberá terminar sola...



Para comprobar que se ha instalado exitosamente el clúster vemos que efectivamente se está ejecutando el proceso:

```
grid@node1:~  
File Edit View Search Terminal Tabs Help  
grid@node1:~  
grid@node1 ~]$ ps -ef | grep pmon  
grid 20626 20373 0 07:10 pts/0 00:00:00 grep --color=auto pmon  
grid 25826 1 0 Feb03 ? 00:00:03 mdb_pmon -MGMTDB  
grid 28341 1 0 Feb03 ? 00:00:03 asm_pmon +ASM1  
grid@node1 ~]$ crsctl
```

Este proceso existe en todos los nodos del clúster,

```

grid 27363 1 0 Feb03 ? 00:01:28 asm_lmd0 +ASM2
grid 27365 1 0 Feb03 ? 00:02:34 asm_lms0 +ASM2
grid 27369 1 0 Feb03 ? 00:01:00 asm_lmhb +ASM2
grid 27371 1 0 Feb03 ? 00:00:01 asm_lck1 +ASM2
grid 27373 1 0 Feb03 ? 00:00:01 asm_dbw0 +ASM2
grid 27376 1 0 Feb03 ? 00:00:01 asm_lgwr +ASM2
grid 27378 1 0 Feb03 ? 00:00:03 asm_ckpt +ASM2
grid 27380 1 0 Feb03 ? 00:00:01 asm_smon +ASM2
grid 27382 1 0 Feb03 ? 00:00:01 asm_lreg +ASM2
grid 27384 1 0 Feb03 ? 00:00:01 asm_pxmn +ASM2
grid 27386 1 0 Feb03 ? 00:00:22 asm_rbal +ASM2
grid 27388 1 0 Feb03 ? 00:00:05 asm_gmon +ASM2
grid 27390 1 0 Feb03 ? 00:00:05 asm_mmon +ASM2
grid 27392 1 0 Feb03 ? 00:00:24 asm_mnnl +ASM2
grid 27394 1 0 Feb03 ? 00:01:56 asm_imr0 +ASM2
grid 27397 1 0 Feb03 ? 00:01:16 asm_gcr0 +ASM2
grid 27400 1 0 Feb03 ? 00:00:15 asm_lck0 +ASM2
grid 27409 1 0 Feb03 ? 00:01:58 asm_ppa7 +ASM2
grid 27463 1 0 Feb03 ? 00:00:00 oracle+ASM2 ocr (DESCRIPTION=(LOCAL=YES)(ADDRESS=(PROTOCOL=beq)))
grid 27465 1 0 Feb03 ? 00:00:01 asm_amb +ASM2
grid 27467 1 0 Feb03 ? 00:00:05 oracle+ASM2 asmb +asm2 (DESCRIPTION=(LOCAL=YES)(ADDRESS=(PROTOCOL=beq)))
grid 27502 1 0 Feb03 ? 00:00:00 oracle+ASM2 (DESCRIPTION=(LOCAL=YES)(ADDRESS=(PROTOCOL=beq)))
grid 27504 1 0 Feb03 ? 00:00:00 oracle+ASM2 (DESCRIPTION=(LOCAL=YES)(ADDRESS=(PROTOCOL=beq)))
grid 27548 1 0 Feb03 ? 00:00:00 oracle+ASM2 (DESCRIPTION=(LOCAL=YES)(ADDRESS=(PROTOCOL=beq)))
root 28205 26833 0 Feb03 ? 00:01:25 /home/grid/software/grid/perl/bin/perl /home/grid/software/grid/bin/diag
snap.pl start
root 28316 1 0 Feb03 ? 00:00:01 /home/grid/software/grid/bin/tnslsnr LISTENER -no crs notify -inherit
root 29919 1 0 Feb03 ? 00:02:31 /home/grid/software/grid/jdk/jre/bin/java -Xms128m -Xmx512m oracle.rat.t

```

Y estos son otros de los tantos que están ejecutando los nodos...

Podemos invocar un programa Oracle para crear un entorno con el siguiente comando:

```

[grid@node1 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /home/grid/app/grid
[grid@node1 ~]$

```

Y este para ver todos los recursos arrancados y en que nodo:

```

[grid@node1 ~]$ crsctl status res -t

```

Name	Target	State	Server	State details
Local Resources				
ora.ASMNET1LSNR_ASM.1snr	ONLINE	ONLINE	node1	STABLE
	ONLINE	ONLINE	node2	STABLE
ora.CLUW.dg	ONLINE	ONLINE	node1	STABLE
	ONLINE	ONLINE	node2	STABLE
ora.LISTENER.1snr	ONLINE	ONLINE	node1	STABLE
	ONLINE	ONLINE	node2	STABLE
ora.MGMT.dg	ONLINE	ONLINE	node1	STABLE
	ONLINE	ONLINE	node2	STABLE
ora.chad	ONLINE	ONLINE	node1	STABLE
	ONLINE	ONLINE	node2	STABLE

Y con esto, la infraestructura del grid está lista.

8. FUNCIONAMIENTO DEL CLÚSTER

Con todo el clúster montado, podemos ya usarlo con normalidad (aunque luego tenemos que instalar el Oracle 12c).

El comando **crsctl** nos permite ejercer un montón de acciones sobre nuestro clúster como por ejemplo, en el apartado anterior, visualizar distintos recursos.

```
[grid@node1 ~]$ crsctl
Usage: crsctl <command> <object> [<options>]
  command: add|check|config|create|debug|delete|disable|discover|enable|eval|export|get|getperm|lsmodules|modify|pin|query
  release|relocate|replace|request|set|setperm|start|status|stop|unpin|unset
For complete usage, use:
  crsctl [-h | --help]
```

Otro comando es **ocrcheck** que nos permite ver fichero de configuración del clúster:

```
[grid@node1 ~]$ ocrcheck
Status of Oracle Cluster Registry is as follows :
  Version               : 4
  Total space (kbytes)   : 409568
  Used space (kbytes)    : 2092
  Available space (kbytes) : 407476
  ID                    : 195181840
  Device/File Name       : +CLUW1
                          Device/File integrity check succeeded
                          Device/File not configured
                          Device/File not configured
                          Device/File not configured
                          Device/File not configured
  Cluster registry integrity check succeeded
```

El comando **crsctl query css votedisk** permite averiguar donde se encuentra el archivo votedisk, el cual identifica que los nodos funcionan. Este comando se debería ejecutar como root:

```
[root@node2 ~]# crsctl query css votedisk
## STATE      File Universal Id                File Name               Disk group
  1. ONLINE    b5bcabaa37754fd8bf3d1f9cbc3d09b0 (/dev/OIS02) [CLUW]
Located 1 voting disk(s).
```

Estos comando se pueden ejecutar en todos los nodos.

Al tener instalada nuestra infraestructura también tendremos instalado nuestro cliente de Oracle 12c:

```
[grid@node1 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Sun Feb 4 11:17:38 2018

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit Production

SQL> show sga

Total System Global Area 1073741824 bytes
Fixed Size 8801008 bytes
Variable Size 620758288 bytes
Database Buffers 436207616 bytes
Redo Buffers 7974912 bytes
SQL> selef
```

Y podremos ver que se ha creado también una base de datos que es la del GIMR con 'select name from v\$datafile':

```
NAME
-----
56255
+MGMT/_MGMTDB/64588ED181826C33E0530A00A8C089C2/DATAFILE/syslogdata.276.9671562
55
+MGMT/_MGMTDB/64588ED181826C33E0530A00A8C089C2/DATAFILE/sysmgmtdata.277.96715626
3
+MGMT/_MGMTDB/64588ED181826C33E0530A00A8C089C2/DATAFILE/sysmgmtadatdb.278.967156
269
NAME
-----
+MGMT/_MGMTDB/64588ED181826C33E0530A00A8C089C2/DATAFILE/sysmgmtdataachafix.279.96
7156305
+MGMT/_MGMTDB/64588ED181826C33E0530A00A8C089C2/DATAFILE/sysmgmtdataq.280.9671563
29
16 rows selected.
```

También podemos todos los usuarios dentro de la base con 'select username from all_users':

```

-----
USERNAME
-----
DBSNMP
APPQOSSYS
GSMCATUSER
GGSYS
XDB
ANONYMOUS
WM SYS
SYSBACKUP
SYSDG
SYSKM
SYSRAC

```

9. TRABAJANDO CON ASM

Si ejecutamos el comando **asmcmd** entraremos en una CLI de administración de nuestro ASM:

```
[grid@node1 ~]$ asmcmd
ASMCMD>
```

Y si escribimos **help** obtendremos ayuda para ver las opciones que tenemos con nuestro ASM:

```

afd_configure, afd_deconfigure, afd_di, afd_dsset, afd_dsset
afd_filter, afd_label, afd_ksdsk, afd_kslsl, afd_refresh
afd_scan, afd_state, afd_unlabel

md_backup, md_restore

amdu_extract

lsattr, setattr

audcleanaudittrail, audclearproperty, audcleartimestamp
audcreatejob, auddropjob, audloadunifile, audsetdebug
audsetjobinterval, audsetjobstatus, audsetproperty, audsettimestamp
audsettraillocation, audshowtimestamp, lsaudcleanevents
lsaudcleanupjobs, lsaudconfigparams

cd, cp, du, find, help, ls, lsct, lsdg, lsdf, mkalias
mkdir, pwd, rm, rmalias, setsparseparent, showclustermode
showclusterstate, showpatches, showversion

chblk, lsblk, mapau, mapblk, mapextent

chdg, chkdg, dropdg, iostat, lsdsk, lsod, mkdg, mount
offline, online, rebal, remap, stamp, stampist, umount

chfg, lsfg, mkfg, rmfg

pwcop, pwcreate, pwdelete, pwget, pwmove, pwset

chqg, lsqg, mkqg, mvfg, rmqg

dsget, dsset, lsc, lson, mkcc, rmcc, shutdown, sbackup

```

Disponemos de una gran variedad de opciones.

Si hacemos `ls` podremos ver los dos grupos de discos que tenemos:

```

ASMCMDS> ls
CLUN/
MGMT/

```

Si investigamos un poco podemos encontrar el `ocrcheck -local`:

```

ASMCMDS> cd OCRFILE
ASMCMDS> ls
Type      Redund  Striped  Time          Sys  Name
OCRFILE   UNP3OT  COARSE   FEB 04 10:00:00 Y    REGISTRY.255.967154075
ASMCMDS>

```

A parte del `ocrcheck` podremos ver todos los archivos de configuración que tenemos del ASM. También podemos ver todos los grupos que tenemos:

```

ASMCMDS> lsdg
State Type Rebal Sector Logical_Sector Block AU Total_MB Free_MB Req_mir_free_MB Usable_file_MB Offline_d
MOUNTED EXTERN N 512 512 4096 4194304 10232 9884 0 9884
0
MOUNTED EXTERN N 512 512 4096 4194304 40952 7068 0 7068
0
ASMCMDS>

```

Y podemos ver los discos con `lsdisk`:

```

ASMCMDS> lsdsk
Path
/dev/DISC01
/dev/DISC02
/dev/DISC03
/dev/DISC04
ASMCMDS>

```

```

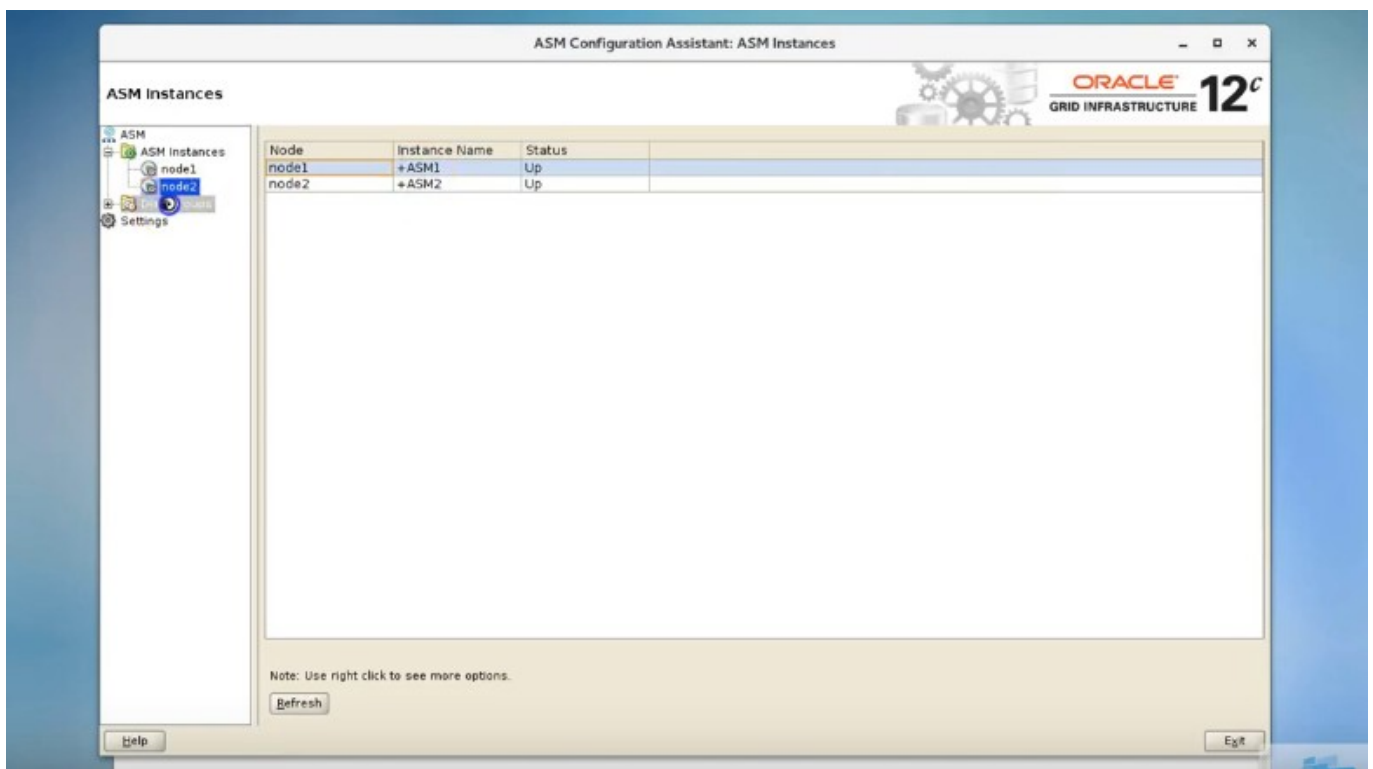
ASMCMDS [+] > lsdsk -t -G DATA
Create Date Mount Date Repair_Timer Path
13-JUL-09 13-JUL-09 0 /devices/diska1
13-JUL-09 13-JUL-09 0 /devices/diska2
13-JUL-09 13-JUL-09 0 /devices/diska3
13-JUL-09 13-JUL-09 0 /devices/diskb1
13-JUL-09 13-JUL-09 0 /devices/diskb2
13-JUL-09 13-JUL-09 0 /devices/diskb3
13-JUL-09 13-JUL-09 0 /devices/diskc1
13-JUL-09 13-JUL-09 0 /devices/diskc2
...

ASMCMDS [+] > lsdsk -p -G DATA /devices/diska*
Group_Num Disk_Num Incarn Mount Stat Header_Stat Mode Stat State Path
1 0 2105454210 CACHED MEMBER ONLINE NORMAL /devices/diska1
1 1 2105454199 CACHED MEMBER ONLINE NORMAL /devices/diska2
1 2 2105454205 CACHED MEMBER ONLINE NORMAL /devices/diska3

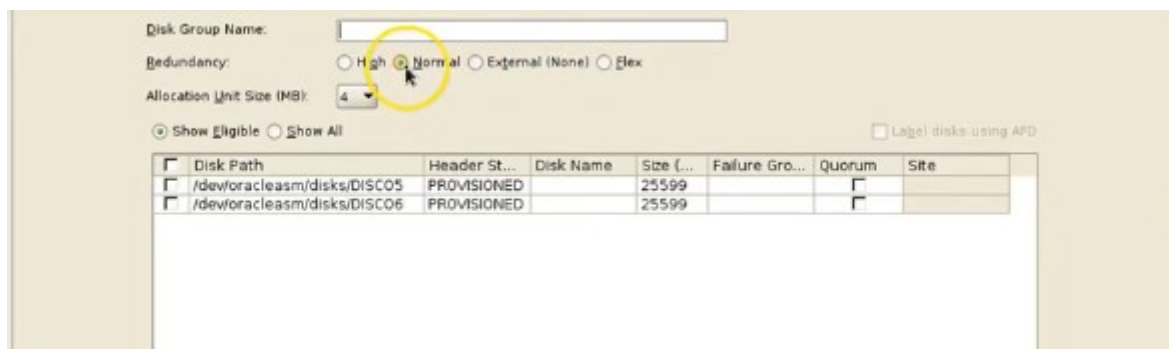
ASMCMDS [+] > lsdsk --candidate -p
Group_Num Disk_Num Incarn Mount Stat Header_Stat Mode Stat State Path
0 5 2105454171 CLOSED CANDIDATE ONLINE NORMAL /devices/diske1
0 25 2105454191 CLOSED CANDIDATE ONLINE NORMAL /devices/diske2
0 18 2105454184 CLOSED CANDIDATE ONLINE NORMAL /devices/diske3
0 31 2105454197 CLOSED CANDIDATE ONLINE NORMAL /devices/diskk1
0 21 2105454187 CLOSED CANDIDATE ONLINE NORMAL /devices/diskk2
0 26 2105454192 CLOSED CANDIDATE ONLINE NORMAL /devices/diskk3
0 14 2105454180 CLOSED CANDIDATE ONLINE NORMAL /devices/diskl1
...

```


También podemos configurar las instancias desde el Asistente de Instancias de ASM:

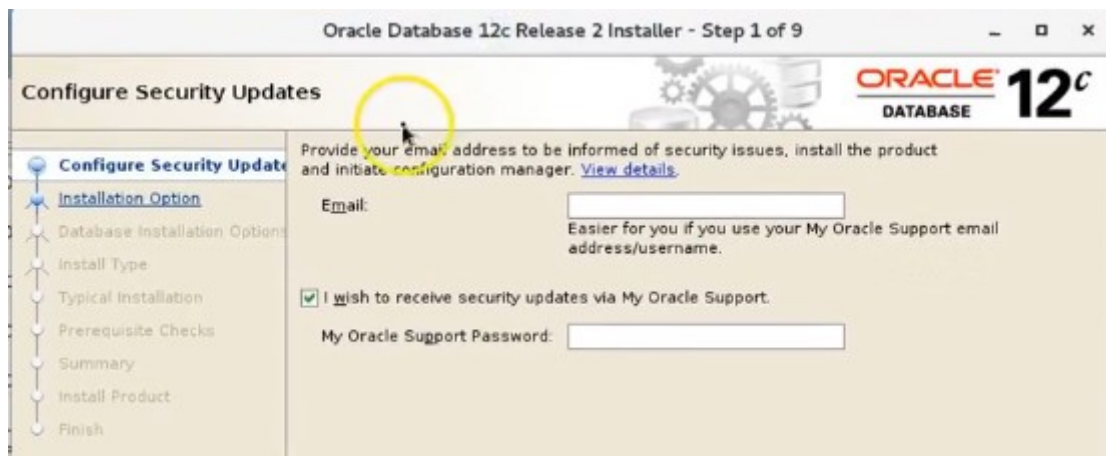


Y podemos hacer cosas como añadir nuevos grupos de discos:



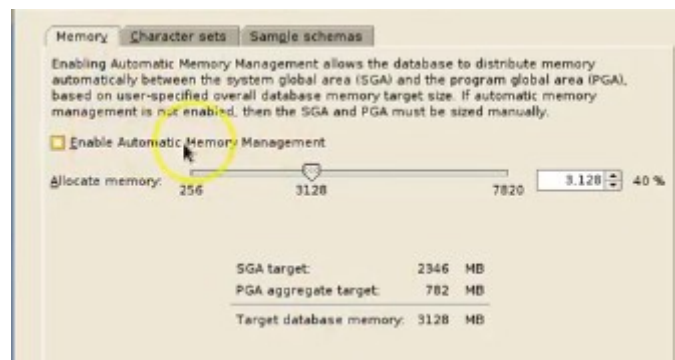
10. INSTALACIÓN ORACLE RAC 12C

Para instalar la interfaz gráfica de Oracle 12 debemos ejecutar el archivo 'runInstaller' que está dentro del comprimido que descargamos en su momento. Así nos saltará la ventana del instalador de Oracle Database 12:

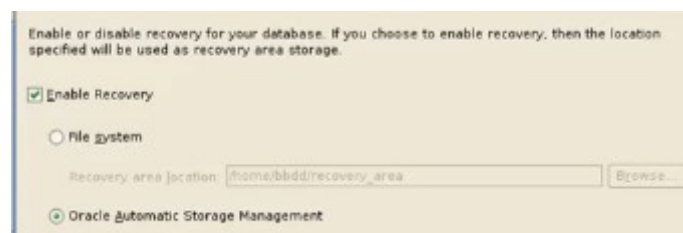


- No es necesario proveer el email. Además, podemos desactivar la casilla para recibir actualizaciones de seguridad si queremos.
- A continuación elegimos la opción de crear y configurar una base de datos.
- En el siguiente paso elegiremos el tipo servidor o '**Server class**'.
- Y luego elegimos '**Oracle Real Application Clusters database installation**'.
- Seleccionamos todos los nodos que van a poseer esta base de datos (1-3)
- Elegimos la instalación avanzada (Advanced Install) ya que vamos a especificar ciertas opciones.
- Escogemos la edición **Enterprise Edition**.
- Y especificamos los directorios de instalación de la base de datos y del software.
- La base que usaremos será de '**General purpose**' o Propósito general.

- Ahora rellenaremos los datos de la base de datos, como el identificador y el SID. Desactivaremos la creación del contenedor de la base de datos.
- Seleccionamos la memoria que usaremos para la base de datos, la cual será elegido de manera automática al seleccionar la casilla correspondiente.

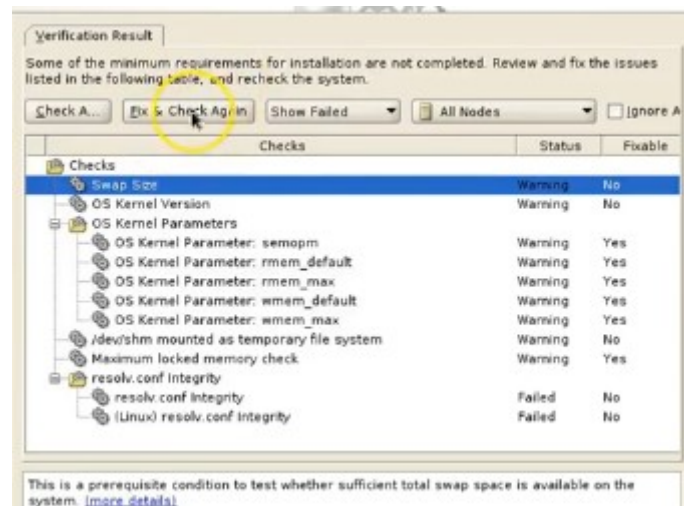


- En el mismo paso, en las otra pestañas seleccionaremos el juego de caracteres (UTF-8).
- El almacenamiento se gestionará de manera automática seleccionando '**Oracle Automatic Storage Management**'.
- Muy importante, en el siguiente paso elegiremos la opción de '**Enable Recovery**' para activar la recuperación en caso de caída:



- Finalmente, elegiremos el disco que se encargará de almacenar la base de datos (**BBDD**).
- Al igual que en el Infraestructure Grid, tendremos que escoger los grupos para todas las funciones, el cual será '**dba**'.

- Le damos a siguiente y empezará la instalación.
- Este instalador también devolverá un log con advertencias que vamos simplemente a ignorar porque no influyen en nuestro clúster. Arreglaremos solo los importantes:



Y después de esto finalizará la instalación...

Para terminar, si comprobamos los procesos con este comando, veremos que están arrancados tanto el ASM como la base de datos 'orcl1':

```
[grid@node1 bin]$ ps -ef | grep pmon
grid      5873      1   0 13:47 ?        00:00:01 asm_pmon_+ASM1
grid      21225     1   0 18:24 ?        00:00:00 ora_pmon_orcl1
grid      38569 23545   0 18:47 pts/3    00:00:00 grep --color=auto pmon
[grid@node1 bin]$
```