Como poner en funcionamiento un SQL SERVER.

Asignatura: Administración de sistemas gestores de datos.

Alumno: Ignacio Govantes Ojeda.

índice.

Introducción	.3
Configuración de las máquinas virtuales	.3
Configuración de la red de los equipos	. 4
Configuración de la red en Ubuntu	. 4
Configuración de la red en Windows	.5
Instalación y configuración de Docker en Ubuntu	.7
Instalación y configuración de SQL en Ubuntu1	1
Instalación y configuración de SSMS en Windows1	2

Introducción

En esta práctica se va a descargar y configurar un SQL server a partir de dos máquinas virtuales conectadas a través de red interna. En la primera, Un ubuntu-20.04.2.0-desktop, ira alojada el servidor SQL y en la segunda máquina, un Windows 10, tendremos un entorno llamado SSMS (SQL SERVER MANAGAMENT STUDIO) que se encargará de administrar la base de datos de forma gráfica.

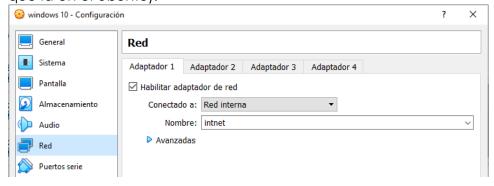
Configuración de las máquinas virtuales

Utilizaremos dos máquinas virtuales, como dijimos, un Ubuntu que servirá como servidor SQL y un Windows como cliente. Para poder conectarlas entre sí en la configuración de la máquina virtual Ubuntu tendremos dos adaptadores de red. Uno será un adaptador puente que permitirá tener internet y otro configurado con red interna, para poder conectarse al Windows.





La configuración del adaptador de red de la maquina Windows se recomienda dejarlo con la configuración predeterminado y hacerlo al final de la practica ya que dejaremos al Windows (por ahora) sin acceso a internet cuando lo pongamos en red interna. Pero la explicación para la conexión de estos equipos será en este y en el siguiente apartado. En la parte de red de la configuración de la maquina solamente tendremos un adaptador de red conectado a una red interna (la misma que la en el Ubuntu).



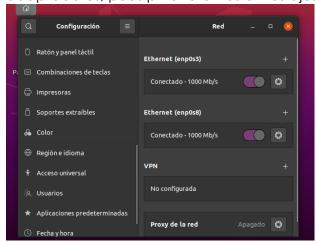
Configuración de la red de los equipos

Algo importante para un servidor SQL es poder conectar los distintos equipos.

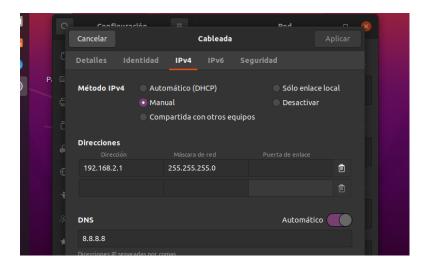
Configuración de la red en Ubuntu.

Después de la instalación de la maquina Ubuntu vamos a hacer la configuración de la red. El primer adaptador de red (El adaptador puente) lo dejamos como esta ya que solo nos da conexión a internet. Sin embargo, la red interna, al ser un servidor, lo más recomendable es que su ip sea fija. Esta configuración se puede hacer a partir del terminal o a partir de la interfaz, lo haremos de la segunda forma ya que es más sencilla.

Entramos en configuración en el apartado Red. Como podéis ver nos indica los dos adaptadores, pues pincharemos en los ajustes del segundo.



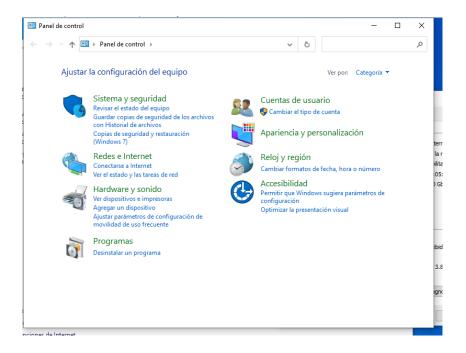
Nos encontramos con una ventana con varios apartados. Nos vamos a IPv4 y ahí, en Método IPv4 le damos a manual y ponemos la ip que queramos. En mi caso 192.168.2.1 y mascara de red 255.255.255.0



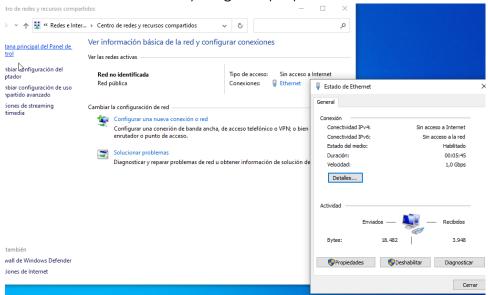
Configuración de la red en Windows.

En Windows, como dije antes, esta configuración (incluido la configuración del adaptador de red) sería lo más lógico hacerlo después de la descarga y configuración de SSMS ya que no tendremos acceso de internet.

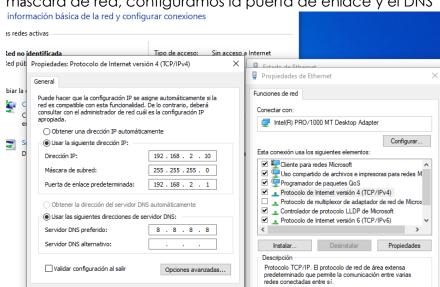
Escribimos en el buscador panel de control y pinchamos en él. A continuación en Redes e Internet pinchamos en ver estado y las tareas de red.



Después, veremos que en red no identificada se encuentra nuestra conexión "Ethernet" Pinchamos ahí y luego en propiedades.



A continuación, buscamos el elemento Protocolo de internet versión 4. Y ponemos la dirección fija, en realidad no haría falta ya que no necesitamos una ip fija, pero para no tener problemas de conexión nos aseguramos poniendo la ip que queramos. En concreto pondré 192.168.2.10 y como pone la imagen además de la



Aceptar Cancelar

máscara de red, configuramos la puerta de enlace y el DNS

Comprobamos que hacen ping los equipos y Ubuntu con Google (Para hacer ping en Windows es necesario desactivar el firewall)

Aceptar Cancelar

```
abc@abc-VirtualBox:~$ ping 8.8.8.8
 PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=13.2 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=13.2 ms
 64 bytes from 8.8.8.8: icmp_seq=3 ttl=115 time=13.7 ms
  ·-- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 13.208/13.370/13.672/0.213 ms
abc@abc-VirtualBox:~$ ping 192.168.2.10
PING 192.168.2.10 (192.168.2.10) 56(84) bytes of data.
 64 bytes from 192.168.2.10: icmp_seq=1 ttl=128 time=0.394 ms
64 bytes from 192.168.2.10: icmp_seq=1 ttl=128 time=0.404 ms
64 bytes from 192.168.2.10: icmp_seq=3 ttl=128 time=0.394 ms
64 bytes from 192.168.2.10: icmp_seq=4 ttl=128 time=0.744 ms
```

Instalación y configuración de Docker en Ubuntu

Para instalar Docker tenemos que seguir una serie de pasos en el terminal.

Lo primero es un sudo apt update y sudo apt upgrade:

```
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
          Archivo Editar Ver Buscar Terminal Ayuda
Archivo Editar Ver Buscar Terminal Ayuda

abc@abc-VirtualBox:~$ sudo apt update

0bj:1 http://es.archive.ubuntu.com/ubuntu bionic InRel

Des:2 http://es.archive.ubuntu.com/ubuntu bionic-updat

Des:3 https://download.docker.com/linux/ubuntu focal ISe

1 innux-headers-4.15.0-161 linux-headers-4.15.0-161-

Des:5 http://es.archive.ubuntu.com/ubuntu bionic-updat

Des:6 http://es.archive.ubuntu.com/ubuntu bionic-updat

Des:7 http://es.archive.ubuntu.com/ubuntu bionic-updat

Des:8 http://es.archive.ubuntu.com/ubuntu bionic-updat

Des:9 http://es.archive.ubuntu.com/ubuntu bionic-updat
```

-Descargar paquetes de requisitos previos que permitan a apt usar paquetes a través de https.

sudo apt install apt-transport-https ca-certificates curl software-properties-common

```
abc@abc-VirtualBox:~$ sudo apt install apt-transport-https ca-certificates curl software-properties-common
```

-Añadir la clave de GPG para el repositorio oficial de Docker:

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -

```
abc@abc-VirtualBox:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
```

-Añadir el repositorio Docker a las fuentes de APT:

sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"

```
abc@abc-VirtualBox:-$ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"
Obj:: https://download.docker.com/linux/ubuntu focal InRelease
Obj:2 http://security.ubuntu.com/ubuntu bionic-security InRelease
Obj:3 http://es.archive.ubuntu.com/ubuntu bionic InRelease
Obj:4 http://es.archive.ubuntu.com/ubuntu bionic-updates InRelease
Obj:5 http://es.archive.ubuntu.com/ubuntu bionic-backports InRelease
```

Volvemos a Actualizar:

```
Archivo Editar Ver Buscar Terminal Ayuda

Archivo Editar Ver Buscar Terminal Ayuda

abc@abc-VirtualBox:-$ sudo apt upgrade
Leyendo lista de paquetes... Hecho

Creando árbol de dependencias

Leyendo la información de estado... Hecho

Des:2 http://es.archive.ubuntu.com/ubuntu bionic-updat

Des:3 https://download.docker.com/linxv/ubuntu focal I

Des:4 http://es.archive.ubuntu.com/ubuntu bionic-backp

Des:5 http://es.archive.ubuntu.com/ubuntu bionic-securit

Des:6 http://es.archive.ubuntu.com/ubuntu bionic-updat

Des:8 http://es.archive.ubuntu.com/ubuntu bionic-updat

Des:8 http://es.archive.ubuntu.com/ubuntu bionic-updat

Des:8 http://es.archive.ubuntu.com/ubuntu bionic-updat

Des:9 http://es.archive.ubuntu.com/ubuntu bionic-updat
```

-Comprobamos el estado del docker:

apt-cache policy docker-ce

```
abc@abc-VirtualBox:~$ apt-cache policy docker-ce
docker-ce:
Instalados: (ninguno)
Candidato: 5:20.10.9~3-0~ubuntu-focal
Tabla de versión:
5:20.10.9~3-0~ubuntu-focal 500
500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
```

-Instalar docker

sudo apt install docker-ce

```
abc@abc-VirtualBox:~$ sudo apt install docker-ce
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
```

-Estado de Docker

sudo systematl status docker

```
abc@abc-VirtualBox: ~ Q ≡ - Ø ⊗

systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.socket.
Configurando git (1:2.25.1-1ubuntu3.2) ...
Procesando disparadores para man-db (2.9.1-1) ...
Procesando disparadores para systemd (245.4-4ubuntu3.13) ...
abc@abc-VirtualBox:-$ sudo systemctl status docker

docker.service - Docker Application Container Engine
Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: Active: active (running) since Wed 2021-10-20 15:18:09 CEST; 54s ago

TriggeredBy: @ docker.socket
Docs: https://docs.docker.com
Main PID: 39658 (dockerd)
Tasks: 7
Memory: 27.4M
CGroup: /system.slice/docker.service
—39658 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd oct 20 15:18:09 abc-VirtualBox dockerd[39658]: time="2021-10-20T15:18:09.35263516 oct 20 15:18:09 abc-VirtualBox dockerd[39658]: time="2021-10-20T15:18:09.35294940 oct 20 15:18:09 abc-VirtualBox dockerd[39658]: time="2021-10-20T15:18:09.35294940 oct 20 15:18:09 abc-VirtualBox dockerd[39658]: time="2021-10-20T15:18:09.35294940 oct 20 15:18:09 abc-VirtualBox dockerd[39658]: time="2021-10-20T15:18:09.35800683 oct 20 15:18:09 abc-VirtualBox dockerd[39658]: time="2021-10-20T15:18:09.58600683 oct 20 15:18:09 abc-VirtualBox dockerd[39658]: time="2021-10-20T15:18:09.65671159 oct 20 15:18:09 abc-
```

Como vemos en la imagen, si te sale algo parecido a eso, Docker ha sido instalado correctamente y a continuación nos toca la configuración.

Lo primero será darle todos los permisos a nuestro usuario para que no tengamos que estar todo el tiempo poniendo sudo a la hora de escribir los comandos.

- -para dejar de escribir sudo siempre que quieras usar docker:

sudo usermod -aG docker \${USER}

---->sudo usermod -aG docker username (para otro usuario)

-Reinicia y escriba lo siguiente:

su - \${USER}

-Para comprobar que nuestro usuario esta agregado en el grupo Docker.

id -nG

Comandos básicos para Docker.

Aunque no necesitaremos usar muchos ya que para la instalación del SQL usaremos un Docker compose (ya veremos más adelante que es) es recomendable tener unos conocimientos básicos.

-para ver comandos de docker: docker docker-subcommand --help -para saber info de docker: docker info -descargar imagenes oficiales: docker pull nombre_imagen -para ver las imagenes descargadas docker images -Ejecutar un contenedor y obtener acceso interactivo del shell del contenedor docker run -it nombre_imagen -Para ver todos los contenedores docker ps -a -Para iniciar un contenedor determinado docker start nombreasignado --> (viene en docker ps -a) -Para parar un contenedor en funcionamiento docker stop nombreasignado --> (viene en docker ps -a) -Para borrar un contenedor asignado

docker rm nombreasignado --> (viene en docker ps -a)

Estos últimos será los que más utilizaremos

Instalación y configuración de SQL con Docker compose en Ubuntu

Docker compose es una herramienta que permite definir y ejecutar contenedores Docker. Es un script, donde pondremos toda la configuración, y así nos ahorramos de poner los comandos. El código se encuentra en un código .yml y se puede encontrar en internet. En nuestro caso el Docker compose de SQL está en el teams en archivos de clase (aunque puedes encontrar otro a través de este enlace https://blog.logrocket.com/docker-sql-server/. Este documento .yml tiene esta forma:

En el código ya viene la contraseña de nuestro administrador, se encuentra en la línea 10.

Para poder usarlo en linux tendremos que descargarlo.

Con sudo apt install docker compose

```
abc@abc-VirtualBox:~/docker/Docker/docker-compose/sql$ sudo apt install docker-compose
[sudo] contraseña para abc:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
linux-headers-5.8.0-43-generic linux-hwe-5.8-headers-5.8.0-43
linux-inage-5.8.0-43-generic linux-modules-5.8.0-43-generic
linux-modules-extra-5.8.0-43-generic
utulice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes adicionales:
python3-attr python3-cached-property python3-distutils python3-docker
python3-dockerpty python3-docopt python3-insortilb-metadata
python3-jsonschema python3-libzto3 python3-mere-itertools python3-pyrsistent
python3-setuptools python3-texttable python3-websocket python3-zipp
```

A continuación, en la carpeta que hemos que se encuentra el archivo descargado de SQL ejecutamos el comando:

Docker-compose -d up

```
bc@abc-VirtualBox:-/docker/Docker/docker-compose/sql$ docker-compose up
Creating network "sql_default" with the default driver
Pulling sql-server-db (mcr.microsoft.com/mssql/server:2017-latest)...
2017-latest: Pulling from mssql/server
528184910841: Pull complete
5394f81d603d: Pull complete
536d9303bf66: Pull complete
572b5bdcef61: Pull complete
3bf3sea46ae83: Pull complete
179860c1c2f: Pull complete
179860c1c2f: Pull complete
20179860c1c2f: Pull complete
228.1MB/355.9MBwmload complete
26.46MB/26.94MB
```

Comprobamos con un docker ps –a para ver si nos sale en la lista de contenedores

```
abc@abc_VirtualBox:~/docker/Docker/docker-compose/sql$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS
761be006ea23 mcr.mlcrosoft.com/mssql/server:2017-latest "/opt/mssql/bln/nonr..." 7 mlnutes ago Exited (0) 8 seconds ago
-db
```

Y si sale como la imagen se ha instalado correctamente.

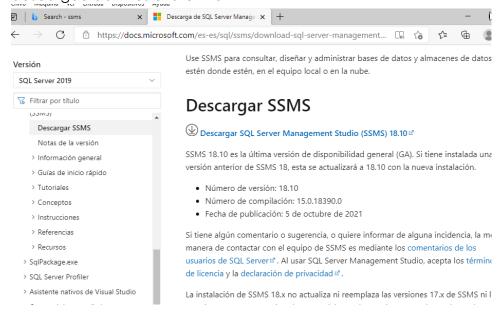
Para arrancarlo usaremos: docker start nombreasignado

```
bhedbic-virtualBox:-/docker/Bocker/docker-docker-compose/sql$ docker start sql-server-db
sql-server-db
shcgabe-virtualBox:-/docker/Bocker/Cocker-compose/sql$ docker ps -8
contraines in JamaGE
contraines in JamaGE
rollenBoxea3
rollenBoxea4
rollenBo
```

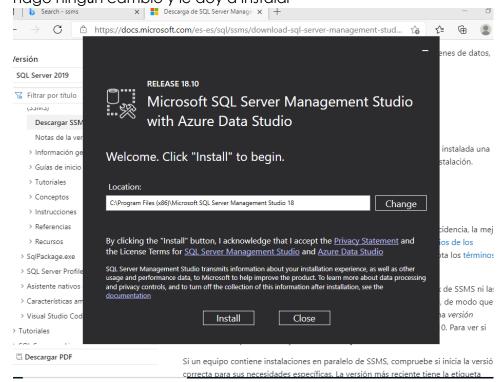
Instalación y configuración de SSMS en Windows

Ahora toca la instalación de SSMS.

Lo primero será buscarlo en el buscador y entrar en la página oficial de Microsoft, pinchamos en el enlace de descarga que sale en la imagen y cuando se nos descargue abrimos el archivo



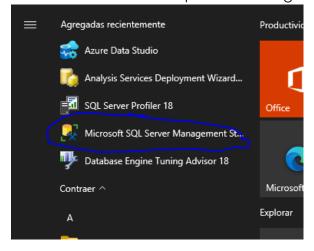
En el archivo nos da la opcion de escoger el lugar de la instalación, en mi caso no hago ningún cambio y le doy a instalar



Cuando termine la instalación le damos a cerrar y ya tenemos instalado el programa.

Conexión con la base de datos y primera tabla

Entramos en microsoft sql server management studio y nos saldrá esta pantalla.

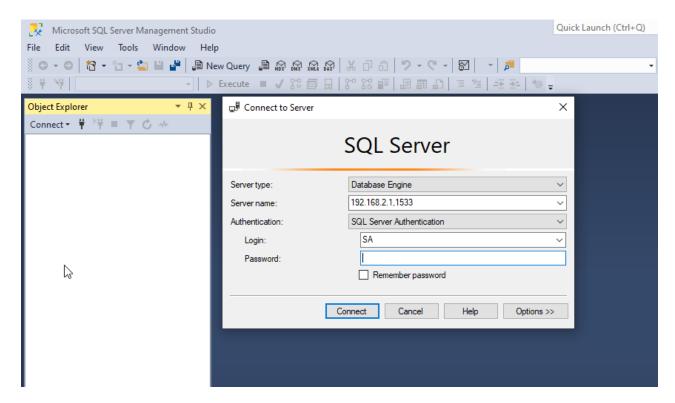


El nombre del servidor, al ser en remoto, tenemos que poner primero la ip y luego el puerto en el que se encuentra separadolos por una coma

En Authentication pondremos SQL Server authentication

En login: SA (Usuario del administrador)

Password: (La contraseña que venía en el docker-compose)



A continuación, crearemos una tabla.

Cuando estemos dentro pincharemos en New query y usaremos el comando CREATE TABLE nombre(

Y dentro pondremos los campos como pone en la imagen

);

