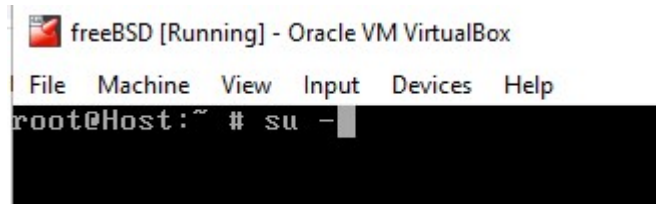
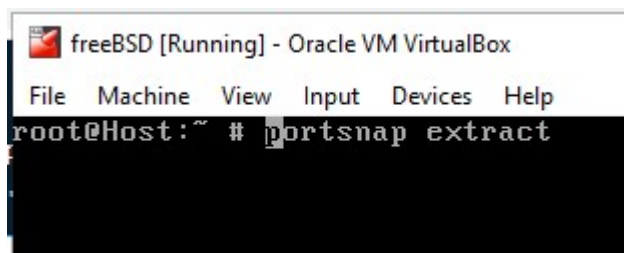


- a. Instale el motor de base de datos MariaDB sobre una máquina virtual con sistema operativo FreeBSD en VirtualBox
- Se habilitan los permisos restantes del super usuario




```
freeBSD [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@Host:~ # su -
```

- Extraemos las funciones portsnap



```
freeBSD [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@Host:~ # portsnap extract
```

- Sabemos que fueron bien extraidas cuando sale la línea “Building new INDEX files... done.” Y realizamos el update a las funciones del portsnap



```
freeBSD [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/usr/ports/x11/xttoolwait/
/usr/ports/x11/xtrans/
/usr/ports/x11/xtrlock/
/usr/ports/x11/xtset/
/usr/ports/x11/xvidtune/
/usr/ports/x11/xvinfo/
/usr/ports/x11/xvkbd/
/usr/ports/x11/xvncinfo/
/usr/ports/x11/xvt/
/usr/ports/x11/xwatchwin/
/usr/ports/x11/xwd/
/usr/ports/x11/xwininfo/
/usr/ports/x11/xwinwrap/
/usr/ports/x11/xwit/
/usr/ports/x11/xwud/
/usr/ports/x11/xxkb/
/usr/ports/x11/xzoom/
/usr/ports/x11/yad/
/usr/ports/x11/yakuake/
/usr/ports/x11/yalias/
/usr/ports/x11/yeahconsole/
/usr/ports/x11/yelp/
/usr/ports/x11/zenity/
Building new INDEX files... done.
root@Host:~ # portsnap fetch update
```

- Despues actualizamos el pkg

```
root@Host:~ # pkg update
Updating FreeBSD repository catalogue...
Fetching packagesite.txz: 100% 6 MiB 1.3MB/s
Processing entries: 0%
Newer FreeBSD version for package glew:
To ignore this error set IGNORE_OSVERSION=yes
- package: 1202000
- running kernel: 1201000
Ignore the mismatch and continue? [y/N]: y
Processing entries: 100%
```

- Y ahora revisamos como encontramos los mariadb de la versión 10.3

```
freeBSD [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@Host:~ # pkg search mariadb | grep 10.3
mariadb103-client-10.3.27_1      Multithreaded SQL database (client)
mariadb103-server-10.3.27      Multithreaded SQL database (server)
root@Host:~ #
```

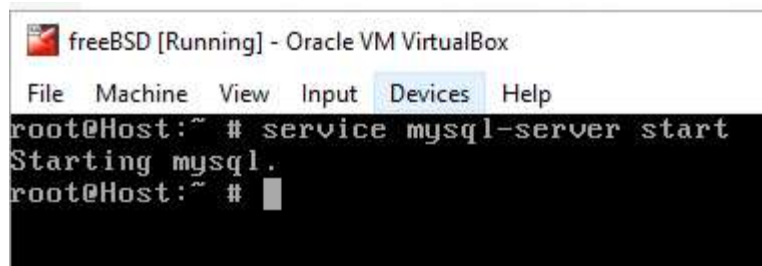
- Instalamos los paquetes de mariadb 10.3 tanto del server como del client

```
freeBSD [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@Host:~ # pkg install mariadb103-server mariadb103-client
```

- Habilitamos el mysql_enable, para poder manipular los estados del servicio

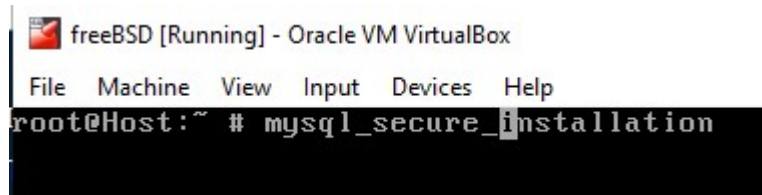
```
freeBSD [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@Host:~ # sysrc mysql_enable="YES"
```

- Ahora inicializamos el servicio de mysql-server



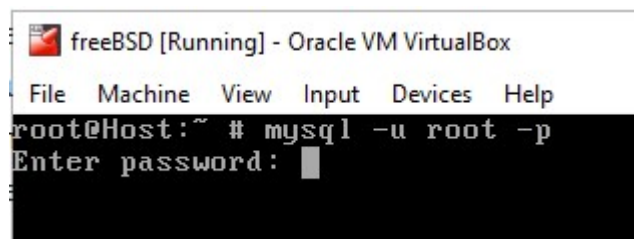
```
freeBSD [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@Host:~ # service mysql-server start
Starting mysql.
root@Host:~ #
```

- Después enviamos a realizar la instalación segura del mysql



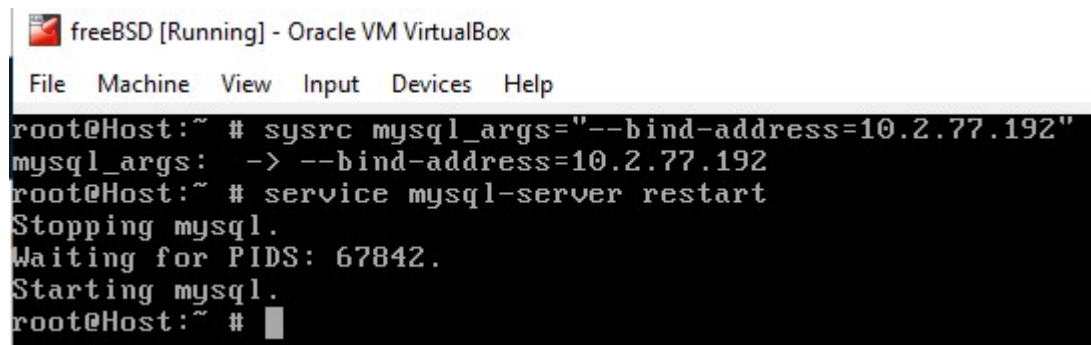
```
freeBSD [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@Host:~ # mysql_secure_installation
```

- Y ahora accedemos al super usuario root del nuestro mariadb para verificar que haya quedado correctamente instalado



```
freeBSD [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@Host:~ # mysql -u root -p
Enter password:
```

- Ahora volvemos al root de nuestra maquina y cambiamos la configuración del bind-address por la de nuestra dirección ip



```
freeBSD [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@Host:~ # sysrc mysql_args="--bind-address=10.2.77.192"
mysql_args:  -> --bind-address=10.2.77.192
root@Host:~ # service mysql-server restart
Stopping mysql.
Waiting for PIDS: 67842.
Starting mysql.
root@Host:~ #
```

- b. Cree un usuario por cada estudiante del grupo. Use los apellidos de los estudiantes como nombre de usuario

- Volvemos a entrar al usuario root del mariadb y creamos nuestros 3 usuarios

```
root@localhost [(none)]> CREATE USER 'cubillos' IDENTIFIED BY 'Clave1+*';
Query OK, 0 rows affected (0.00 sec)

root@localhost [(none)]> CREATE USER 'pachon' IDENTIFIED BY 'Clave1+*';
Query OK, 0 rows affected (0.00 sec)

root@localhost [(none)]> CREATE USER 'urrea' IDENTIFIED BY 'Clave1+*';
Query OK, 0 rows affected (0.00 sec)
```

- c. Cree una base de datos para manejar los destinos turísticos que desea visitar. Para cada destino debe tener información básica del destino (ej: nombre, lugar, país, descripción, etc), condiciones especiales del lugar (Ej. Clima, tipo de viaje, con quién quisiera ir, vestuario recomendado, etc) y fecha estimada de viaje (mes y año). La base de datos debe tener al menos 3 tablas. Cada estudiante debe tener acceso sólo a la base de datos de cada uno.

- Ahora creamos las 3 bases de datos para cada usuario que posteriormente crearemos

```
root@localhost [(none)]> CREATE DATABASE viajescubillos;
Query OK, 1 row affected (0.00 sec)

root@localhost [(none)]> CREATE DATABASE viajespachon;
Query OK, 1 row affected (0.00 sec)

root@localhost [(none)]> CREATE DATABASE viajesurrea;
Query OK, 1 row affected (0.00 sec)
```

- Y posteriormente le asignamos los permisos a cada usuario sobre su base de datos

```
root@localhost [(none)]> GRANT ALL PRIVILEGES ON viajescubillos.* TO 'cubillos';
Query OK, 0 rows affected (0.00 sec)

root@localhost [(none)]> GRANT ALL PRIVILEGES ON viajespachon.* TO 'pachon';
Query OK, 0 rows affected (0.00 sec)

root@localhost [(none)]> GRANT ALL PRIVILEGES ON viajesurrea.* TO 'urrea';
Query OK, 0 rows affected (0.00 sec)
```

- Ahora para acceder a la base de datos usamos el comando “use” y después el nombre de la base de datos a la que nos queremos conectar, a continuación vemos las tablas creadas, estas mismas fueron creadas para las otras 2 bases de datos

```

freeBSD [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
cubillos@localhost [(none)]> use viajescubillos
Database changed
cubillos@localhost [viajescubillos]> CREATE TABLE DestinosPlaneados(codPlaneado
NUMERIC(5) NOT NULL, codDestino NUMERIC(10) NOT NULL, fechaViaje DATE NOT NULL);
Query OK, 0 rows affected (0.01 sec)

cubillos@localhost [viajescubillos]> CREATE TABLE Conocidos(telefono NUMERIC(15)
NOT NULL, PrimerNombre VARCHAR(15) NOT NULL, SegundoNombre VARCHAR(15), Primera
pellido VARCHAR(15) NOT NULL, SegundoApellido VARCHAR(15));
Query OK, 0 rows affected (0.01 sec)

cubillos@localhost [viajescubillos]> CREATE TABLE Acompañante(codPlaneado NUMERI
C(5) NOT NULL, telefono NUMERIC(15) NOT NULL);
Query OK, 0 rows affected (0.01 sec)

cubillos@localhost [viajescubillos]> CREATE TABLE Paises(codPais NUMERIC(4) NOT
NULL, nombrePais VARCHAR(30) NOT NULL);
Query OK, 0 rows affected (0.01 sec)

cubillos@localhost [viajescubillos]> CREATE TABLE Ciudades(codCiudad NUMERIC(10)
NOT NULL, codPais NUMERIC(4) NOT NULL, nombreCiudad VARCHAR(30) NOT NULL);
Query OK, 0 rows affected (0.00 sec)

cubillos@localhost [viajescubillos]> CREATE TABLE destinosTuristicos(codDestino
NUMERIC(10) NOT NULL, codCiudad NUMERIC(10) NOT NULL, nombreDestino VARCHAR(30)
NOT NULL, descripcion VARCHAR(500) NOT NULL);
Query OK, 0 rows affected (0.00 sec)

cubillos@localhost [viajescubillos]>

```

- Ahora asignamos llaves primarias a nuestras tablas

```

freeBSD [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
stinosPlaneados_PK PRIMARY KEY (codPlaneado);
Query OK, 0 rows affected (0.01 sec)
Records: 0 Duplicates: 0 Warnings: 0

root@localhost [viajescubillos]> ALTER TABLE Conocidos ADD CONSTRAINT Conocidos_
PK PRIMARY KEY (telefono);
Query OK, 0 rows affected (0.01 sec)
Records: 0 Duplicates: 0 Warnings: 0

root@localhost [viajescubillos]> ALTER TABLE Acompañante ADD CONSTRAINT Acompañ
nte_PK PRIMARY KEY (codPlaneado,telefono);
Query OK, 0 rows affected (0.01 sec)
Records: 0 Duplicates: 0 Warnings: 0

root@localhost [viajescubillos]> ALTER TABLE Paises ADD CONSTRAINT Paises_PK PRI
MARY KEY (codPais);
Query OK, 0 rows affected (0.01 sec)
Records: 0 Duplicates: 0 Warnings: 0

root@localhost [viajescubillos]> ALTER TABLE Ciudades ADD CONSTRAINT Ciudades_PK
PRIMARY KEY (codCiudad);
Query OK, 0 rows affected (0.01 sec)
Records: 0 Duplicates: 0 Warnings: 0

root@localhost [viajescubillos]>

root@localhost [viajescubillos]> ALTER TABLE destinosTuristicos ADD CONSTRAINT d
estinosTuristicos_PK PRIMARY KEY (codDestino);
Query OK, 0 rows affected (0.01 sec)
Records: 0 Duplicates: 0 Warnings: 0

root@localhost [viajescubillos]>

```

- Ahora le colocamos llaves foráneas a nuestras tablas

```

freeBSD [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Records: 0 Duplicates: 0 Warnings: 0

root@localhost [viajesurreal]> use viajescubillos
Database changed
root@localhost [viajescubillos]> ALTER TABLE DestinosPlaneados ADD CONSTRAINT P
aneadoTuristicos_FK FOREIGN KEY(codDestino) REFERENCES destinosTuristicos(codDes
tino);
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0

root@localhost [viajescubillos]> ALTER TABLE Acompañante ADD CONSTRAINT Destino
acompañante_FK FOREIGN KEY(codPlaneado) REFERENCES DestinosPlaneados(codPlaneado
);
Query OK, 0 rows affected (0.01 sec)
Records: 0 Duplicates: 0 Warnings: 0

root@localhost [viajescubillos]> ALTER TABLE Acompañante ADD CONSTRAINT DestinoC
onocido_FK FOREIGN KEY(telefono) REFERENCES Conocidos(telefono);
ERROR 1215 (HY000): Cannot add foreign key constraint
root@localhost [viajescubillos]> ALTER TABLE Acompañante ADD CONSTRAINT DestinoC
onocido_FK FOREIGN KEY(telefono) REFERENCES Conocidos(telefono);
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0

root@localhost [viajescubillos]>

root@localhost [viajescubillos]> ALTER TABLE Acompañante ADD CONSTRAINT DestinoC
onocido_FK FOREIGN KEY(telefono) REFERENCES Conocidos(telefono);
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0

root@localhost [viajescubillos]> ALTER TABLE Ciudades ADD CONSTRAINT CiudadPais_
FK FOREIGN KEY(codPais) REFERENCES Paises(codPais);
Query OK, 0 rows affected (0.01 sec)
Records: 0 Duplicates: 0 Warnings: 0

root@localhost [viajescubillos]> ALTER TABLE destinosTuristicos ADD CONSTRAINT D
estinoCiudad_FK FOREIGN KEY(codCiudad) REFERENCES Ciudades(codCiudad);
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0

root@localhost [viajescubillos]>

```

d. Inserte datos en las bases de datos

- Poblamos las bases de datos

```

urrea@localhost [viajesurreal]> INSERT INTO DestinosPlaneados VALUES(1,2,'2021-08
10');
Query OK, 1 row affected (0.00 sec)

urrea@localhost [viajesurreal]> INSERT INTO DestinosPlaneados VALUES(2,1,'2021-09
-30');
Query OK, 1 row affected (0.00 sec)

urrea@localhost [viajesurreal]> INSERT INTO Acompañante VALUES(1,1);
Query OK, 1 row affected (0.00 sec)

urrea@localhost [viajesurreal]> INSERT INTO Acompañante VALUES(1,2);
Query OK, 1 row affected (0.01 sec)

urrea@localhost [viajesurreal]> INSERT INTO Acompañante VALUES(2,2);
Query OK, 1 row affected (0.00 sec)

urrea@localhost [viajesurreal]>

```