Con la carpeta creada en C:\DockerMysqlSSL tal y como muestra la imagen:

|  |
| --- |
|  |

Llamamos a una consola en Windows si es nuestro caso:

|  |
| --- |
| Texto  Descripción generada automáticamente |

De las carpetas que aparecen, se crean en el momento del despliegue y son:

-certs, db, db\_data

Si quiéramos vaciar todo y empezar de nuevo, sólo nos quedaríamos con la carpeta script y el fichero mysql\_ssl.yml.

En el caso de solo tener estos dos ficheros, ejecutamos en la consola lo siguiente:

docker-compose -f mysql\_ssl.yml up

Con esto se levanta el servicio de la base de datos para que funcione la base de datos de MySql con encriptación de los datos.

La carpeta “certs” es necesaria para que tengamos los certificados que hacen falta para conectarse con un cliente de base de datos.

La conexión tendríamos que hacerla por SSL. Ej:

|  |
| --- |
|  |

Debemos de fijarnos que en este caso como van los ficheros “pem”. En este caso:

SSL Key File: cliente-key.pem

SSL CERT File: client-cert.pem

SSL CA File: root-ca.pem

En otros programas como el Toad Edge:

|  |
| --- |
|  |

En este caso ponemos la siguiente configuración:

Certificate authority file: root-ca.pem

Certification file: client-cert.pem

Key file: client-key.pem

En caso de querer parar el servidor de Mysql lo que hacemos es:

docker-compose -f mysql\_ssl.yml down

Para poder acceder a la máquina virtual de Docker hacemos el siguiente comando en otra consola:

|  |
| --- |
| Interfaz de usuario gráfica, Texto |

O sea:

docker ps

Con esto identificamos las conexiones y los servidores arrancados en Docker. Ahora que ya sabemos el identificador que es en nuestro caso “dockermysqlssl-mysql-1” que nos va a hacer falta para conectarnos a la máquina virtual en cuestión.

Para entrar dentro, escribimos:

docker exec -it dockermysqlssl-mysql-1 /bin/bash

y nos aparecerá la siguiente imagen:

|  |
| --- |
|  |

Ahora que ya sabemos arrancar el servicio de Mysql con SSL y accedemos a la máquina, vamos a analizar el código del fichero “mysql\_ssl.yml”:

Texto

Descripción generada automáticamente

Miramos el contenido de dicho fichero:

|  |
| --- |
|  |

En “volumes:” nos aseguramos de que el servidor tenga persistencia en la máquina anfitriona, de esta forma, si pasara algo con la máquina, siempre quedan los ficheros en la máquina anfitrionas.

En “environment:” ponemos las variables de partida como son la contraseña del root y un usuario normal.

En “command:” ponemos parámetros para indicar como queremos que se comporte el servidor. En nuestro caso, ponemos la codificación de los caracteres, los certificados que se han generado para encriptar.

En “ports:” ponemos en el lado izquierdo tenemos 3307 que es la máquina anfitriona y 3306 internamente en la máquina virtual.

Por otro lado mas abajo hay un servicio que llamamos “cert-gen” que es necesario para que genere los certificados y lo haremos con la imagen “alpine” que es pequeña y nos va a servir muy bien.

En “volumes:” ponemos que tenga acceso a la carpeta “certs” en la máquina anfitriona y “/etc/certs” en la máquina virtual. Del mismo modo, tenemos acceso a “script” en los dos lados, maquina anfitriona y maquina virtual.

En “entrypoint:” ejecutamos las ordenes que queremos que funcionen inmediatamente antes de desplegar el servidor. En nuestro caso, para que ejecutemos los códigos en la máquina Linux debemos poner estas líneas:

|  |
| --- |
|  |

Estas líneas son importantes, son las que indican que lo que venga después se ejecuta tal cual en la máquina virtual. Que en nuestro caso es:

|  |
| --- |
|  |

Y lo que hacemos es llamar a que añada en nuestro sistema “openssl” que es necesario para generar nuestros certificados.

Y es muy importante llamar a nuestro “script” gencerts.sh, pero no solo eso, tenemos que indicarle los permisos a dichos certificados porque si no no los va a cargar.

En la carpeta de Script está todo lo necesario para generar nuestros scripts, con uno principal que llama al resto. Así:

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |