

Running SQL scripts with Windocks

Put SQL scripts in files, in a directory and create a dockerfile in the same directory.

Build an image with database(s) in it ready to be cloned. (See the dockerfile below)

docker build -t myimage path/to/directory/with/dockerfile

Create containers with DB clones from this image. DB clones can be attached to the Windocks SQL container or any other SQL instance or a docker SQL Linux container

docker create myimage

dockerfile

```
FROM mssql-2012

# Creates the image "myimage" with customers database ready to be cloned
SETUPCLONING FULL customers C:\dbbackups\customerfullbackup.bak

# Copies the script from your local directory to image when myimage is built
# Required for encrypted databases. Extension must be sqlsys
COPY build-time-tde-encryption-script.sqlsys .

# Runs script once during myimage build before customers DB is ready
# Required for encrypted databases. Extension must be sqlsys
RUN build-time-tde-encryption-script.sqlsys

# Copies script from your directory to image when myimage is built
COPY build-time-database-script.sql .

# Runs script once during myimage build after customers DB is ready
RUN build-time-database-script.sql

# Copies script from your directory to image when myimage is built
# Required for encrypted databases. Extension must be sqlsys
COPY container-create-time-tde-encryption-script.sqlsys .

# Copies the script from your directory to image when myimage is built
COPY container-create-time-script2.sql .

# RUN scripts below this ENV runs for each container created from myimage
# Put all COPY commands above this ENV, don't put COPY commands below
ENV USE_DOCKER_FILE_TO_CREATE_CONTAINER=1

# Runs script each time container created from myimage before DB is ready
# Required for encrypted databases. Extension must be sqlsys
RUN container-create-time-tde-encryption-script.sqlsys

# Runs script each time container created from myimage after DB is ready
RUN container-create-time-script2.sql
```

Scripts

Each SQL command in the script file must be on one line and the commands must be separated by a semicolon (;) Do not use GO

build-time-tde-encryption-script.sqlsys

Use this script only if you want to use TDE encryption. In the above dockerfile, this script runs once at myimage build time. (See TDE doc on windocks.com for details)

```
USE master;
ALTER SERVICE MASTER KEY FORCE REGENERATE;
OPEN MASTER KEY DECRYPTION BY PASSWORD='StrongPassword';
ALTER MASTER KEY ADD ENCRYPTION BY SERVICE MASTER KEY;
CLOSE MASTER KEY;
```

build-time-database-script.sql

Use this script to do things on the database like masking data. In the above dockerfile, this script runs once at myimage build time and myimage will have the results of this script.

```
USE customers;
ALTER TABLE personalinfo DROP email;
```

container-create-time-tde-encryption-script.sqlsys

Use this script only if you want to use TDE encryption. In the above dockerfile, this script runs each time a container is created from myimage.

```
USE MASTER;
ALTER SERVICE MASTER KEY FORCE REGENERATE;
OPEN MASTER KEY DECRYPTION BY PASSWORD='StrongPassword';
ALTER MASTER KEY ADD ENCRYPTION BY SERVICE MASTER KEY;
CLOSE MASTER KEY;
```

container-create-time-script2.sql

Use this script to do things specific to each SQL server environment in a container and each database in a container. In the above dockerfile, this script will result in each container created from myimage having a SQL login called testsys. Also the user named testsysuser is created for the database customers for the Login testsys

```
CREATE LOGIN testsys WITH PASSWORD = 'asr5567@23345';
USE customers;
CREATE USER testsysuser For LOGIN testsys;
```