

ENCRYPTED COLUMNS AND SQL SERVER CONTAINERS

ENCRYPT COLUMN IN PRODUCTION DATABASE BEFORE BACKUP

```
-- Verify that there is a service master key
USE master;
SELECT * FROM sys.symmetric_keys WHERE name = '##MS_ServiceMasterKey##';
-- Create Database level Master Key, Certificate, and symmetric key
USE YourDatabaseName;
CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'YourPassword';
CREATE CERTIFICATE YourDatabaseCertificate WITH SUBJECT = 'Column Protect';
CREATE SYMMETRIC KEY YourSymmetricKey WITH ALGORITHM = AES_128 ENCRYPTION
BY CERTIFICATE YourCertificate;
-- Create column to hold encrypted data and encrypt the sensitive data with symmetric key
ALTER TABLE YourTable ADD SensitiveColumnEncrypted varbinary(MAX) NULL;
OPEN SYMMETRIC KEY YourSymmetricKey DECRYPTION BY CERTIFICATE YourCertificate;
UPDATE YourTable SET SensitiveColumnEncrypted =
EncryptByKey(Key_GUID('YourSymmetricKey'), SensitiveColumn) FROM YourTable;
CLOSE SYMMETRIC KEY YourSymmetricKey;
-- Remove SensitiveColumn since data is encrypted and now in SensitiveColumnEncrypted
ALTER TABLE Customer_data DROP COLUMN SensitiveColumn
```

BUILD IMAGE AND CREATE CONTAINERS

dockerfile to build image

```
FROM mssql-2014
SETUPCLONING FULL YourDatabaseName C:\windocks\dbbackups\YourBackup.bak
COPY setupdecryption.sql .
#Run the setup decryption script each time a container is created from image
ENV USE_DOCKER_FILE_TO_CREATE_CONTAINER=1
RUN setupdecryption.sql
```

setupdecryption.sql

```
USE YourDatabaseName;
ALTER SERVICE MASTER KEY REGENERATE;
OPEN MASTER KEY DECRYPTION BY PASSWORD = 'YourPassword';
ALTER MASTER KEY ADD ENCRYPTION BY SERVICE MASTER KEY;
```

```
docker build -t yourimagename path\to\dockerfile
docker run -d yourimagename
```

DECRYPT THE COLUMN

After container is created, in SQL Management Studio, connect to 127.0.0.1,containerport

```
USE YourDatabaseName;
OPEN SYMMETRIC KEY YourSymmetricKey DECRYPTION BY CERTIFICATE YourCertificate;
SELECT CONVERT(varchar, DecryptByKey(SensitiveColumnEncrypted)) FROM YourTable;
CLOSE SYMMETRIC KEY YourSymmetricKey;
```