# Lab: Data Encryption

|  |
| --- |
| * This is worth 2 points. * The due date is tomorrow midnight. * Use the following naming convention: homework, underscore, last name, first initial, and extension (e.g., Lab\_Encrypt\_ImG.docx). |

## 1. Preparation

First, if your SQL Server does not have Oldhouse database, create it using this script: **Oldhouse-Table-Create (Lab).sql**.

Next, perform the lab using this script: **Encryption-Cert (Lab).sql**.

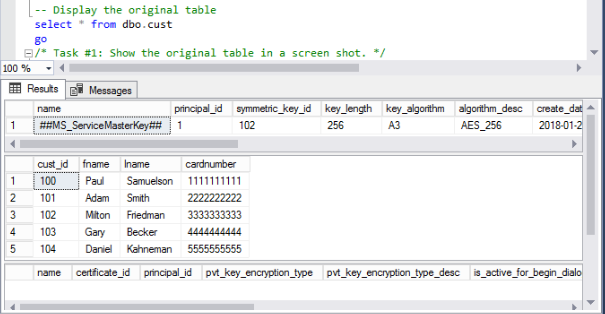
## 2. Deliverables

-- Display the original table

select \* from dbo.cust

go

/\* Task #1: Show the original table in a screen shot. \*/

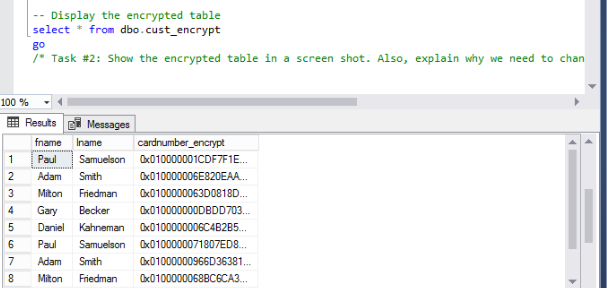


-- Display the encrypted table

select \* from dbo.cust\_encrypt

go



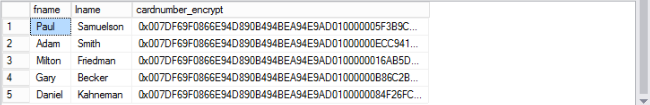
/\* Task #2: Show the encrypted table in a screen shot. Also, explain why we need to change the data type for encryption. \*/

The reason why we need to change the data type for encryption is because the result of the encryption is not a string. It is an array of 8-bit bytes and is not valid character in any character coding.

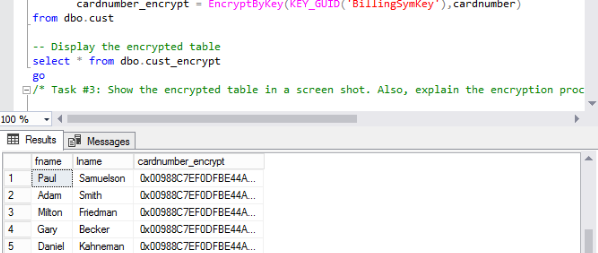
-- Display the encrypted table

select \* from dbo.cust\_encrypt

go



/\* Task #3: Show the encrypted table in a screen shot. Also, explain the encryption process after Task #2. \*/



After task 2 a certificate was created. The certificate is a digitally signed security object. After that a symmetric key was created and was encrypted with the certificate.

Then, the table is emptied out by **Truncating** it. The key is then decrypted using **BillingCert** certificate. The rows are inserted using the symmetric key and are encrypted by the certificate.

-- Display the decrypted table

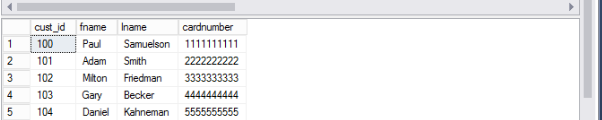
select fname,

lname,

cardnumber = convert(nvarchar(25), DecryptByKey(cardnumber\_encrypt))

from dbo.cust\_encrypt

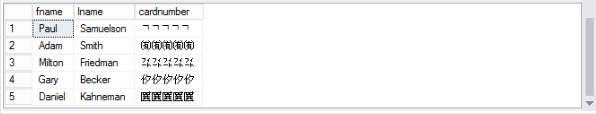
go



/\* Task #4: Show the encrypted table in a screen shot. Also, explain the decryption process after Task #3. \*/

/\* Did you get the original data back? If not, what's wrong? \*/

/\* Hint: Check out the current data type of cardnumber with the original one \*/



The authorized user is granted access to the data after task 3. Then the decrypted table is displayed by changing the data type. I didn’t get the original data back. Because the data type **nvarchar** can store Unicode, and the **varbinary** data type cannot store Unicode. That is how the **cardnumber** column returned the results.