Configure Email Masking in Azure SQL Database

We are tasked with implementing a masking solution that meets these requirements:

- Provide a masking scheme that masks all but the first letter of clients' last names
- Provide an email mask for customer email addresses
- Provide a complete mask for customer phone numbers

To do so, we will need to complete the following:

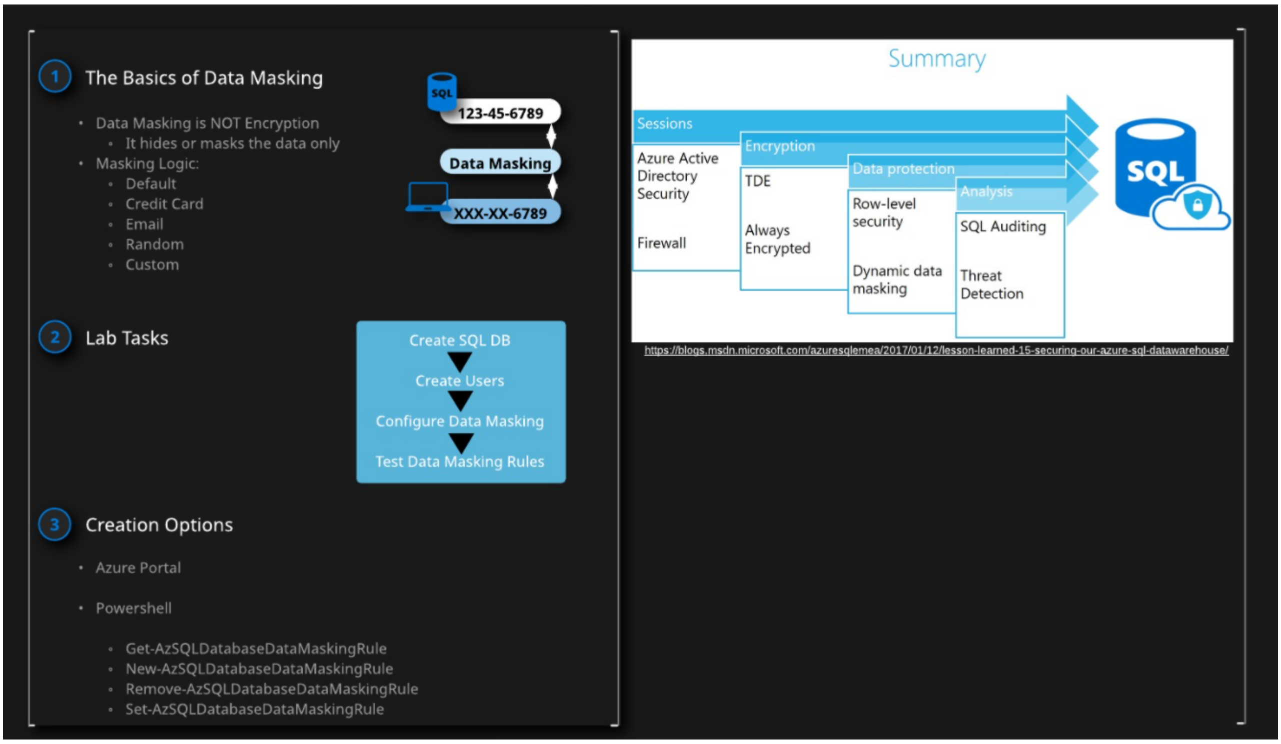
1. Log in to the live environment with the provided Azure Labs credentials.

2. Create a SQL Database.

3. Create a user to test the data mask in Azure Data Studio.

4. Configure masks as described in scenario requirements.

5. Test the data mask in Azure Data Studio using the created user.



1 The Basics of Data Masking

- Data Masking is NOT Encryption
  - It hides or masks the data only
- Masking Logic:
  - Default
  - Credit Card
  - Email
  - Random
  - Custom

2 Lab Tasks

Create SQL DB → Create Users → Configure Data Masking → Test Data Masking Rules

3 Creation Options

- Azure Portal
- Powershell
  - Get-AzSQLDatabaseDataMaskingRule
  - New-AzSQLDatabaseDataMaskingRule
  - Remove-AzSQLDatabaseDataMaskingRule
  - Set-AzSQLDatabaseDataMaskingRule

https://blogs.msdn.microsoft.com/azuresqlemea/2017/01/12/lesson-learned-15-securing-our-azure-sql-datawarehouse/

Before We Begin

Before we begin, you need to install Azure Data Studio. Here is the download from Microsoft:
https://docs.microsoft.com/en-us/sql/azure-data-studio/download?view=sql-server-ver15

After you've downloaded Azure Data Studio, log in to the Azure portal using the lab credentials. As soon as you're logged in, we can begin.

Create and Configure a SQL Database

Create a Standard edition SQL Database with 10 DTUs. While doing this, we'll also create a SQL Server as well. To create the database, first search for SQL DB in the search bar in your Azure portal. From it, select SQL databases. From there, select **Create SQL Database**.

Here, leave Subscription as its default and select the only option for the Resource group. If it does not appear immediately, we may need to wait a moment or reload the page.

For the Database details section, enter a unique name into the **Database name** field. Then, for Server, select **Create new**. Here, enter the following:

- Server name: Create a unique name
- Server admin login: admint
- Password: 12345LinuxAcademy
- Password:12345LinuxAcademy
- Location: East US

Once we've filled in the server information, select OK.
Next, for Compute + storage, select Configure database. Here, select Standard and make sure that it is set to 10 DTUs. Once selected, click Apply.
Now, on the Create SQL Database page, select the Additional settings. Here, select Sample next to Use existing data, then select Review + create at the bottom of the page, and finally click Create.

Set Server Firewall

With our server created, it is time to set our firewall. To do so, select Overview and then Set server firewall. On this page, we want to set Allow Azure services and resources to access this server to ON, and then click Add client IP. Once we've done this, select Save. When prompted, click OK.

View the Server

Now, to enter our server, select Query editor (preview) from the sidebar. Here, we are prompted for the Login and Password we used when we created the server. Enter them, and then click OK. Once we do, we can see all of our data.

Apply Data Mask Rules

To start the masking process, we need to access the database. For this lab, we are using Azure Data Studio, which is the program that is linked at the beginning of this lab.
First, we need to connect to our server using the server button. To do so, we will need the server name, which is located in the Azure portal on the Overview page. Copy this and place it into the Server field. Next, we'll need the same user name and password as before. With those entered, select Connect.
Now we have access to our database.

Create a Test User

Now, to make sure that our masking is working correctly later on, we need to create a test user. To do so, select New Query and enter in the following:

```
CREATE USER TestUser WITHOUT LOGIN;
GRANT SELECT ON SalesLT.Customer TO TestUser;
```

The SalesLT.Customer references the table with the same name.

Data Mask Rules

Back on the Azure portal, select the Query editor (preview), and then find the SalesLT.Customer table. Click on the ellipsis and select Select Top 1000 Rows. Data appears under Results.
Great, we can see our information and apply the mask. To do so, find and click on Dynamic Data Masking under the Security section. If prompted, we can agree to discard any edits. On the next page, select +Add. Here, under Add masking rule, enter the following:

- Schema: SalesLT
- Table: Customer
- Column: LastName (nvarchar)

Under Select how to mask set the Masking field format to Custom string (prefix padding suffix) and enter int the following data:

- Exposed Prefix: 1
- Padding String: (Leave blank)
- Suffix: 0

With that information entered, select Add from the top of the page.

That is our first mask. Now, let's make the one for email. Again, select +Add and enter the following into the Add masking rule fields:

- Schema: SalesLT
- Table: Customer
- Column: EmailAddress (nvarchar)

Under Select how to mask set the Masking field format to Email (aXXX@XXXX.com). Select Add.
Now we need to do the same for customer phone numbers. Once again, select Add from the top of the page, the +Add, and enter the following into the Add masking rule fields:

- Schema: SalesLT
- Table: Customer

- Column: Phone (nvarchar)

Under Select how to mask set the Masking field format to Default value (0, xxxx, 01-01-1900. Select Add.

All of our new masking rules appear.

Test Masking Rules

To test these masks, go back into Azure Data Studio. Here, select New Query and enter in the following information:

```
EXECUTE AS USER = 'TestUser';
SELECT * FROM SalesLT.Customer;
REVERT;
```

Once all the information is entered, select Run. Under Results, we will see all of our customer data. This time, we will see the information has been masked.