SQL Moderation Hack – Database Migration Lab

V2.4

PROBLEM STATEMENT

You have 3 SQL Server 2008r2 Database(s) on a single virtual server, used by an Application "Online Transaction Monitor". The Databases and Application need to be migrated from SQL Server 2008r2 to latest versions of SQL Server. The business would like to minimize patching and maintenance for the future but maintain full functionality. However, the Application source code is lost, the only configuration change you can make is the Connection String.

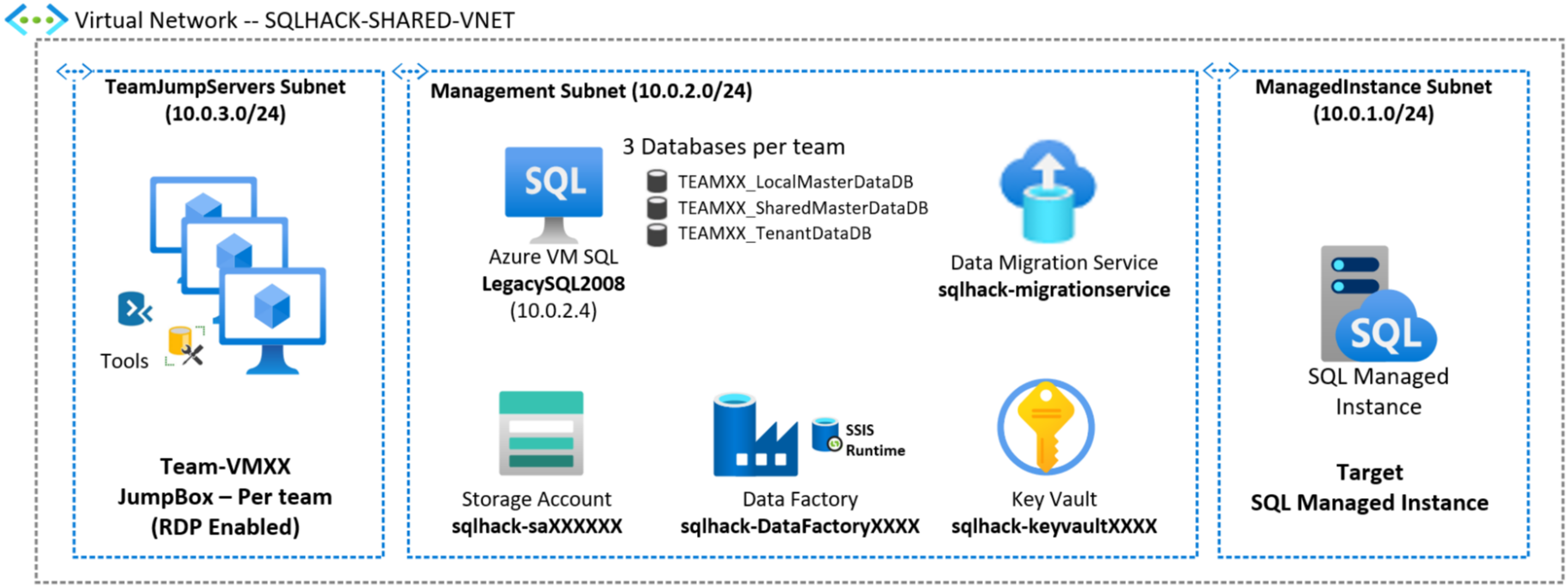
Task: Migrate Databases form SQL Server 2008r2 to suitable environment, with a successful test of the application, meeting all business objectives.

LAB INSTRUCTIONS

Time: 1 Hour

For Connection Strings and Passwords see LAB ENVIROMENT and APPENDIX

1. Test the Online Transaction Monitor with the databases held on SQL Server 2008R2 Legacy server (IP: 10.0.2.4) using your TEAM assigned databases and Login
2. Plan your 3 databases for migration, using the Database Migration Assistant. Are the Legacy Databases best suited for Azure SQL Database Single or Azure SQL Managed Instance?
3. Use the Database Migration Service to Migrate your 3 Databases and Login from the Legacy SQL Server 2008R2 to the Azure SQL Database
    a. See Appendix for connection strings
    b. SAS URI Key is available in C:_SQLHACK_\LABS\01-Data_Migration\SASKey.txt
    c. Managed instance FQDN is in C:_SQLHACK_\LABS\01-Data_Migration\ ManagedInstanceFDQN.txt
4. Test your Migration by using the Setting screen to update the connection string (Instructions below)
5. Note any errors and work through SQL fixes held within C:_SQLHACK_\LABS\01-Data_Migration\Migration Helper Script.sql



SQLHACK-SHARED-VNET Single Virtual Network containing all workshop resources				
TeamJumpServers Subnet Each team is assigned a Win10 VM that mimics their company desktop		**Management Subnet** Several machines and services are already deployed within a dedicated subnet within the Virtual Network		**ManagedInstance Subnet** The Azure SQL Managed Instance has been deployed into a dedicated Subnet

NOTE: There are 20 workshop environments using a SHARED source SQL Server and target Azure SQL Database Managed Instance. Please be respectful of only migrating your teams Databases and Logins.

A STEP-BY-STEP SOLUTION GUIDE FOR THIS LAB IS AVAILIBLE ON THE Win10 VMs IN:

C:_SQLHACK_\LABS\01-Data_Migration\DB Migration Lab Step-by-step.pdf

LAB ENVIROMENT

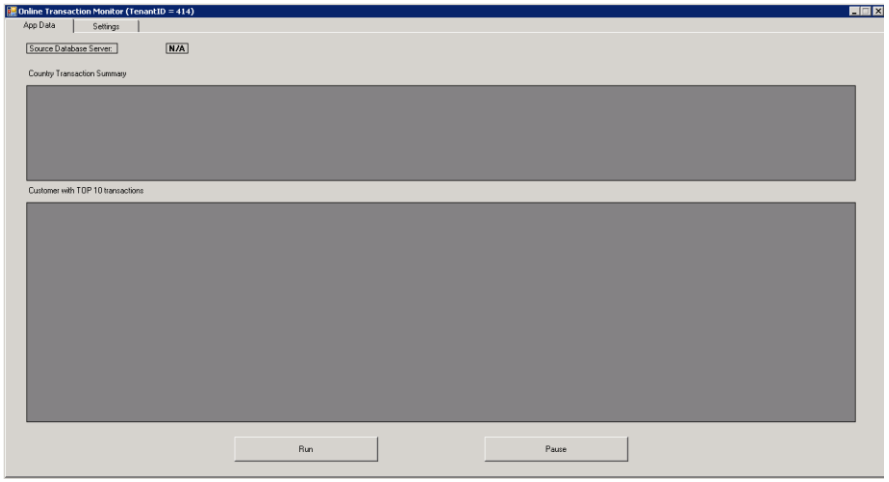
The lab environment has been setup for you to become familiar with an offline (restore from backup) database migration from SQL Server 2008 to Azure SQL Database Managed Instance. Offline migrations are the most common migration approach supported by Azure Database Migration Service (DMS).

During the workshop you will need to refer to a set of parameters that are specific to that workshop.

For this reason please use the parameters shown in Appendix of this document which are specific to your workshop.

APPLICATION – Online Transaction Monitor

The Online Transaction Monitor is a simple application that lists the number of transactions for a given country. The data for this application is held in 3 databases on a SQL Server 2008 r2 Server.



These databases are named (where XX is your team number/name):

- TEAMxx_LocalMasterDataDB
- TEAMxx_SharedMasterDataDB
- TEAMxx_TenantDataDb

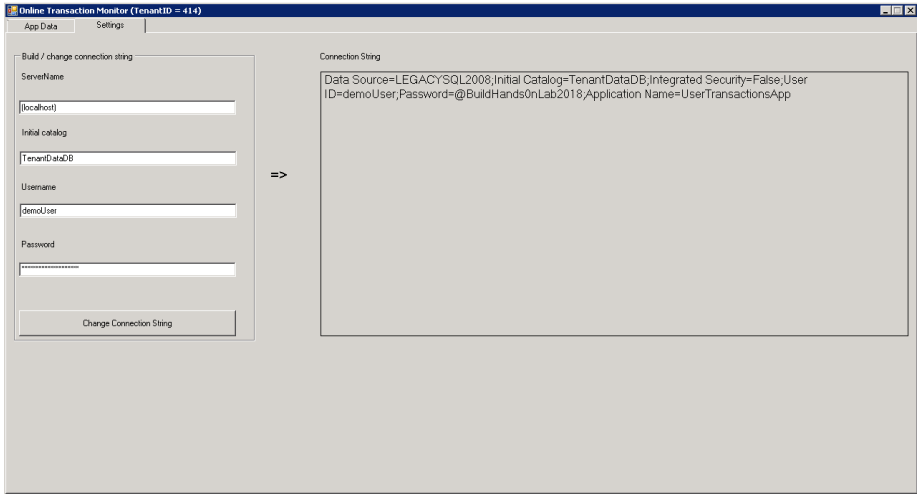
The SQL Databases use CLR with an assembly embedded in the TEAMxx_TenantDataDb.

Additionally, the Application currently uses a SQL LOGIN which must also be considered. The Login is DB_Owner of each of the 3 TEAM databases only. The SQL Login for your team is:

SQL Account Login	**TEAMxx**
Password	**TEAMxx**

Changing the connection string

The Application can be configured through its setting screen:



The Settings that can be changed include:

Server name	**LEGACYSQL2008**
Initial Catalog	**TEAMxx_TenantDataDb**
SQL Account login	**TEAMxx**
Password	**TEAMxx**

APPENDIX

Summary of Logins and Accounts Used

There are several different environments that you need to login/connect to during the labs. Sometimes you will need to login into the same environment with different accounts depending on what you are doing e.g. logging into SQL Server with a standard or sysadmin privileged account.

Here is a summary of all the accounts used in the labs and there uses:

Used for	Username	Password	Machine Name/IP Address/Address/Instance Name
Win10 VM windows login for RDP	**vm-TEAMxx\DemoUser**	**Demo@pass1234567**	**Vm IP address see your Teams channel**
Azure portal domain account	***See your Teams channel***	***See your Teams channel***	**https://portal.azure.com**
Legacy SQL2008 SQL login used by the dummy application	**TEAMxx**	**TEAMxx**	**LEGACYSQL2008** (default SQL instance)
Legacy SQL2008 SQL login sysadmin	**DemoUser**	**Demo@pass1234567**	**LEGACYSQL2008** (default SQL instance)
Azure SQL Managed Instance sysadmin	**DemoUser**	**Demo@pass1234567**	*(see **C:_SQLHACK_\LABS\01-Data_Migration\ManagedInstanceFDQN.txt**)*

Win10 Management VM RDP details

Machine IP address (use for RDP connection)	**Vm IP address see your Teams channel**
Machine Name (Replace XX with Team number)	**vm-TEAMxx**
Win10 Username: (Use for RDP connection)	**Demouser**
Win10 Password: (Use for RDP connection)	**Demo@pass1234567**
Resource Group	**SQLHACK-TEAM-VMs**

Azure Portal Access (from anywhere)

Azure Portal URL	http://portal.azure.com
Azure Username	*(See **C:_SQLHACK_\Azure Portal Domain Accounts.pdf**)*
Azure User Password	*(See **C:_SQLHACK_\Azure Portal Domain Accounts.pdf**)*
Azure Resource Groups	**SQLHACK-SHARED** **SQLHACK-TEAM-VMs**

Database Migration Service Settings

Resource Group	**SQLHACK-SHARED**
Service Name	**sqlhack-migrationservice**
Migration Project Name *(Replace XX with Team number)*	**TEAMxx_Migration**
Source server type	**SQL Server**
Target server type	**Azure SQL Managed Instance**
Migration activity type	**Offline data migration**
Select Source blade	
Source SQL Server instance name	**LEGACYSQL2008**
Authentication type	**SQL Authentication**
User Name	**demouser**
Password	**Demo@pass1234567**
Encrypt connections *(tick boxes)*	*(either tick both boxes or none)*
Select Target blade	
Target server name	*(See **C:_SQLHACK_\LABS\01-Data_Migration\ ManagedInstanceFQDN.txt**)*
Authentication type	**SQL Authentication**
User Name	**demouser**
Password	**Demo@pass1234567**
Select Databases blade	
Source Databases *(Chose only your teams 3 databases)*	**TEAMxx_LocalMasterDataDB** **TEAMxx_SharedMasterDataDB** **TEAMxx_TenantDataDb**

Select Logins blade	
Select Logins *(Chose only your teams login)*	**TEAMxx**
Configure Migration Settings blade	
Chose source backup option	**I will Let Azure Database Migration Service create Backup files**
Backup Settings – Network Share Location	**\\LEGACYSQL2008\FILESHARE**
Backup Settings – Windows User to impersonate	**legacysql2008\demouser**
Backup Settings – Windows Password	**Demo@pass1234567**
Storage Account Settings – SAS URI	*(See File C:_SQLHACK_\LABS\01-Data_Migration\SASKey.txt)*
Summary blade	
Activity name	**TEAMxx_migration_activity**

Source SQL Server (SQL 2008 VM Server)

Server Name	**LEGACYSQL2008**
Resource Group	**SHARED**
Demo application SQL Login username: (Use for Application connection) (Replace xx with Team number)	**TEAMxx**
Demo application SQL Login password: (Use for Application connection) (Replace xx with Team number)	**TEAMxx**
Databases: (Replace xx with Team number)	**TEAMxx_LocalMasterDataDB** **TEAMxx_SharedMasterDataDB** **TEAMxx_TenantDataDb**

Target SQL Server (Azure SQL Managed Instance)

Server Name	***C:_SQLHACK_\LABS\01-Data_Migration\ManagedInstanceFQDN.txt***
Resource Group	**SQLHACK-SHARED**
Sysadmin Login Name: (Use for Migrations)	**DemoUser**
Admin Login Password:	**Demo@pass1234567**