Online Lab - Integrating SaaS Services Available on the Azure Platform

Topic: Securing Secrets in Azure

Before we start

1. Ensure that you are logged in to your Windows 10 lab virtual machine using the following credentials:
   - Username: **Admin**
   - Password: **Pa55w.rd**
2. Review Taskbar located at the bottom of your Windows 10 desktop. The Taskbar contains the icons for the common applications you will use in the labs:
   - Microsoft Edge
   - File Explorer
   - Visual Studio Code
   - Microsoft Azure Storage Explorer
   - Bash on Ubuntu on Windows
   - Windows PowerShell

   Note: You can also find shortcuts to these applications in the **Start Menu**.

Exercise 1: Deploy Key Vault resources

Task 1: Open the Azure Portal

1. On the Taskbar, click the **Microsoft Edge** icon.
2. In the open browser window, navigate to the **Azure Portal** (https://portal.azure.com).
3. When prompted, authenticate with the user account account that has the owner role in the Azure subscription you will be using in this lab.

Task 2: Deploy a key vault

1. In the upper left corner of the Azure portal, click **Create a resource**.
2. At the top of the **New** blade, in the **Search the Marketplace** text box, type **Key Vault** and press **Enter**.
3. On the **Everything** blade, in the search results, click **Key Vault**.

4. On the **Key Vault** blade, click the **Create** button.
5. On the **Create key vault** blade, perform the following tasks:
    - In the **Name** text box, type a globally unique value.
    - Leave the **Subscription** drop-down list entry set to its default value.
    - In the **Resource group** section, ensure that the **Create new** option is selected and then, in the text box, type **AADesignLab0901-RG**.
    - In the **Location** drop-down list, select the Azure region to which you intend to deploy resources in this lab.
    - Click **Pricing tier**, on the **Pricing tier** blade, click **A1 Standard**, and then click **Select**.
    - Leave all remaining settings with their default values.
    - Click the **Create** button.
6. Wait for the provisioning to complete before you proceed to the next task.

Task 3: Add a secret to a key vault by using the Azure portal

1. In the hub menu in the Azure portal, click **Resource groups**.
2. On the **Resource groups** blade, click **AADesignLab0901-RG**.
3. On the **AADesignLab0901-RG** blade, click the entry representing the newly created key vault.
4. On the key vault blade, click **Secrets**.
5. On the key vault secrets blade, click the **Generate/Import** button at the top of the pane.
6. On the **Create a secret** blade, perform the following tasks:
    - In the **Upload options** drop-down list, ensure that the **Manual** entry is selected.
    - In the **Name** text-box, type **thirdPartyKey**.
    - In the **Value** text box, enter the value **56d95961e597ed0f04b76e58**.
    - Leave all remaining settings with their default values.
    - Click the **Create** button.

Task 4: Open Cloud Shell

1. At the top of the portal, click the **Cloud Shell** icon to open a new shell instance.

    > **Note**: The **Cloud Shell** icon is a symbol that is constructed of the combination of the *greater than* and *underscore* characters.

2. If this is your first time opening the **Cloud Shell** using your subscription, you will see a wizard to configure **Cloud Shell** for first-time usage. When prompted, in the **Welcome to Azure Cloud Shell** pane, click **Bash (Linux)**.

> **Note**: If you do not see the configuration options for **Cloud Shell**, this is most likely because you are using an existing subscription with this course's labs. If so, proceed directly to the next task.

3. In the **You have no storage mounted** pane, click **Show advanced settings**, perform the following tasks:
   - Leave the **Subscription** drop-down list entry set to its default value.
   - In the **Cloud Shell region** drop-down list, select the Azure region matching or near the location where you intend to deploy resources in this lab
   - In the **Resource group** section, select the **Use existing** option and then, in the drop-down list, select **AADesignLab0901-RG**.
   - In the **Storage account** section, ensure that the **Create new** option is selected and then, in the text box below, type a unique name consisting of a combination of between 3 and 24 characters and digits.
   - In the **File share** section, ensure that the **Create new** option is selected and then, in the text box below, type **cloudshell**.
   - Click the **Create storage** button.
4. Wait for the **Cloud Shell** to finish its first-time setup procedures before you proceed to the next task.

Task 5: Add a secret to a key vault using the CLI

1. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to create a variable which value designates the name of the resource group that contains the Azure key vault you deployed earlier in this exercise:

   ```
   RESOURCE_GROUP='AADesignLab0901-RG'
   ```

2. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to retrieve the name of the Azure key vault you created earlier in this exercise:

   ```
   KEY_VAULT_NAME=$(az keyvault list --resource-group $RESOURCE_GROUP --
   query "[0].name" --output tsv)
   ```

3. At the **Cloud Shell** command prompt, type in the following command, and press **Enter** to list secrets in the key vault:

   ```
   az keyvault secret list --vault-name $KEY_VAULT_NAME
   ```

4. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to display the value of the **thirdPartyKey** secret:

   ```
   az keyvault secret show --vault-name $KEY_VAULT_NAME --name
   thirdPartyKey --query value --output tsv
   ```

5. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to add a new secret to your key vault:

   ```
   az keyvault secret set --vault-name $KEY_VAULT_NAME --name
   firstPartyKey --value 56f8a55119845511c81de488
   ```

6. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to list secrets in the key vault:

   ```
   az keyvault secret list --vault-name $KEY_VAULT_NAME --query
   "[*].{Id:id,Created:attributes.created}" --out table
   ```

7. Close the **Cloud Shell** pane.

Task 6: Add secrets to a key vault by using Azure Resource Manager templates

1. In the upper left corner of the Azure portal, click **Create a resource**.
2. At the top of the **New** blade, in the **Search the Marketplace** text box, type **Template Deployment** and press **Enter**.
3. On the **Everything** blade, in the search results, click **Template Deployment**.
4. On the **Template deployment** blade, click the **Create** button.
5. On the **Custom deployment** blade, click the **Build your own template in the editor** link.
6. On the **Edit template** blade, click **Load file**.
7. In the **Choose File to Upload** dialog box, navigate to the **F:\Labfiles\Mod09\Starter\** folder, select the **secret-template.json** file, and click **Open**. This will load the following content into the template editor pane:

   ```
   { "$schema": "http://schema.management.azure.com/schemas/2015-01-
   01/deploymentTemplate.json#", "contentVersion": "1.0.0.0",
   "parameters": { "vaultName": { "type": "string" } }, "variables": {
   "secretName": "vmPassword" }, "resources": [ { "apiVersion": "2016-10-
   01", "type": "Microsoft.KeyVault/vaults/secrets", "name":
   "[concat(parameters('vaultName'), '/', variables('secretName'))]",
   "properties": { "contentType": "text/plain", "value": "StudentPa$$w.rd"
   } } ] }
   ```

8. Click the **Save** button to persist the template.
9. Back on the **Custom deployment** blade, perform the following tasks:

   - Leave the **Subscription** drop-down list entry set to its default value.
   - In the **Resource group** section, select the **Use existing** option and then, in the drop-down list, select **AADesignLab0901-RG**.
   - In the **Vault Name** text box, type the name of the key vault you created earlier in this exercise.
   - In the **Terms and Conditions** section, select the **I agree to the terms and conditions stated above** checkbox.
   - Click the **Purchase** button.

10. Do not wait for the deployment to complete but proceed to the next step.
11. In the upper left corner of the Azure portal, click **Create a resource**.
12. At the top of the **New** blade, in the **Search the Marketplace** text box, type **Template Deployment** and press **Enter**.
13. On the **Everything** blade, in the search results, click **Template Deployment**.
14. On the **Template deployment** blade, click the **Create** button.
15. On the **Custom deployment** blade, click the **Build your own template in the editor** link.
16. On the **Edit template** blade, click **Load file**.
17. In the **Choose File to Upload** dialog box, navigate to the **F:\Labfiles\Mod09\Starter\** folder, select the **storage-template.json** file, and click **Open**. This will load the following content into the template editor pane:

```
{ "$schema": "http://schema.management.azure.com/schemas/2015-01-
01/deploymentTemplate.json#", "contentVersion": "1.0.0.0",
"parameters": { "vaultName": { "type": "string" } }, "variables": {
"secretName": "storageConnectionString", "storageName":
"[concat('stor', uniqueString(resourceGroup().id))]" }, "resources": [
{ "apiVersion": "2017-10-01", "type":
"Microsoft.Storage/storageAccounts", "name":
"[variables('storageName')]", "location": "[resourceGroup().location]",
"kind": "Storage", "sku": { "name": "Standard_LRS" }, "properties": { }
}, { "apiVersion": "2016-10-01", "type":
"Microsoft.KeyVault/vaults/secrets", "name":
"[concat(parameters('vaultName'), '/', variables('secretName'))]",
"dependsOn": [ "[resourceId('Microsoft.Storage/storageAccounts',
variables('storageName'))]" ], "properties": { "contentType":
"text/plain", "value":
"[concat('DefaultEndpointsProtocol=https;AccountName=',
variables('storageName'), ';', 'AccountKey=',
listKeys(resourceId('Microsoft.Storage/storageAccounts',
variables('storageName')), providers('Microsoft.Storage',
'storageAccounts').apiVersions[0]).keys[0].value, ';')]" } } ] }
```

18. Click the **Save** button to persist the template.
19. Back on the **Custom deployment** blade, perform the following tasks:
    - Leave the **Subscription** drop-down list entry set to its default value.

- In the **Resource group** section, select the **Use existing** option and then, in the drop-down list, select **AADesignLab0901-RG**.
- In the **Vault Name** field, type the name of the key vault you created earlier in this exercise.
- In the **Terms and Conditions** section, select the **I agree to the terms and conditions stated above** checkbox.
- Click the **Purchase** button.

20. Wait for the deployment to complete before you proceed to the next task.

Task 7: View key vault secrets

1. In the hub menu of the Azure portal, click **Resource groups**.
2. On the **Resource groups** blade, click **AADesignLab0901-RG**.
3. On the **AADesignLab0901-RG** blade, click the entry representing the key vault you created earlier in this exercise.
4. On the key vault blade, click **Secrets**.
5. On the key vault secrets blade, review the list of secrets created during this lab.
6. Click the entry representing the **vmPassword** secret.
7. On the **vmPassword** blade, click the entry representing the current version of the secret.
8. On the Secret Version blade, click the **Show secret value** button.
9. Verify that the value of the secret matches the one included in the template you deployed in the previous task.

Review: In this exercise, you created a **Key Vault** instance and used several different methods to add secrets to the key vault.

Exercise 2: Deploy Azure VM using Key Vault secret

Task 1: Retrive the value of the key vault Resource Id parameter

1. At the top of the portal, click the **Cloud Shell** icon to open a new Clould Shell instance.
2. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to create a variable which value designates the name of the resource group that will contain the hub virtual network:

```
RESOURCE_GROUP='AADesignLab0901-RG'
```

3. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to retrieve the resource id of the Azure key vault you created earlier in this exercise:

   ```
   KEY_VAULT_ID=$(az keyvault list --resource-group $RESOURCE_GROUP --
   query "[0].id" --output tsv)
   ```

4. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to create a variable which value designates the name of the Azure key vault resource id and which takes into account any special character the resource id might include:

   ```
   KEY_VAULT_ID_REGEX="$(echo $KEY_VAULT_ID | sed -e 's/\\/\\\\/g;
   s/\//\\\//g; s/&/\\\&/g')"
   ```

Task 2: Prepare the Azure Resource Manager deployment template and parameters files

1. In the **Cloud Shell** pane, click the **Upload/Download files** icon and, in the drop-down menu, click **Upload**.
2. In the **Open** dialog box, navigate to the **F:\Labfiles\Mod09\Starter\** folder, select the **vm-template.json** file, and click **Open**.
3. In the **Cloud Shell** pane, click the **Upload/Download files** icon and, in the drop-down menu, click **Upload**.
4. In the **Open** dialog box, navigate to the **F:\Labfiles\Mod09\Starter\** folder, select the **vm-template.parameters.json** file, and click **Open**.
5. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to replace the placeholder for the **$KEY*VAULT*ID** parameter in the **vm-template.parameters.json** parameters file with the value of the **$KEY*VAULT*ID** variable:

   ```
   sed -i.bak1 's/"$KEY_VAULT_ID"/"'"$KEY_VAULT_ID_REGEX"'"/' ~/vm-
   template.parameters.json
   ```

6. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to verify that the placeholder was successfully replaced in the parameters file:

   ```
   cat ~/vm-template.parameters.json
   ```

Task 3: Configure a key vault for deployment of Azure Resource Manager templates

1. In the hub menu in the Azure portal, click **Resource groups**.
2. On the **Resource groups** blade, click **AADesignLab0901-RG**.
3. On the **AADesignLab0901-RG** blade, click the entry representing the key vault you created in the previous exercise.
4. On the key vault blade, click **Access policies**.
5. On the **Access policies** blade, click the **Click to show advanced access policies** link.
6. Select the **Enable access to Azure Resource Manager for template deployment** checkbox.
7. Click the **Save** button at the top of the pane.

Task 4: Deploy a Linux VM with the password paramter set by using a key vault secret.

1. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to deploy the Azure Resource Manager template with the specified parameters file:

   ```
   az group deployment create --resource-group $RESOURCE_GROUP --template-file ~/vm-template.json --parameters @~/vm-template.parameters.json
   ```

2. Wait for the deployment to complete before you proceed to the next task.

Task 5: Verify the outcome of the deployment

1. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to create a variable which value designates the name of the resource group that contains the newly deployed Azure VM:

   ```
   RESOURCE_GROUP='AADesignLab0901-RG'
   ```

2. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to retrieve the name of the Azure key vault containing the secret that stores the value of the password of the local Administrator account:

   ```
   KEY_VAULT_NAME=$(az keyvault list --resource-group $RESOURCE_GROUP --query "[0].name" --output tsv)
   ```

3. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to retrieve the value of the secret:

   ```
   az keyvault secret show --vault-name $KEY_VAULT_NAME --name vmPassword --query value --output tsv
   ```

4. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to retrieve the public IP address of the Azure VM you deployed in the previous task:

   ```
   PUBLIC_IP=$(az network public-ip list --resource-group $RESOURCE_GROUP
   --query "[0].ipAddress" --output tsv)
   ```

5. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to connect to the Azure VM via SSH:

   ```
   ssh Student@$PUBLIC_IP
   ```

6. At the **Cloud Shell** command prompt, when prompted whether you want to continue connecting, type `yes` and press **Enter**.
7. At the **Cloud Shell** command prompt, when prompted for password, type the value of the secret you retrieved earlier in this task and press **Enter**.
8. Verify that you successfully authenticated.
9. At the **Cloud Shell** command prompt, type `exit` to log out from the Azure VM.

Review: In this exercise, you deployed a Linux VM using a password stored as a key vault secret.

Exercise 3: Remove lab resources

Task 1: Open Cloud Shell

1. At the top of the portal, click the **Cloud Shell** icon to open the Cloud Shell pane.
2. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to list all resource groups you created in this lab:

   ```
   az group list --query "[?starts_with(name,'AADesignLab09')]".name --
   output tsv
   ```

3. Verify that the output contains only the resource groups you created in this lab. These groups will be deleted in the next task.

Task 2: Delete resource groups

1. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to delete the resource groups you created in this lab

   ```
   az group list --query "[?starts_with(name,'AADesignLab09')]".name --
   output tsv | xargs -L1 bash -c 'az group delete --name $0 --no-wait --
   yes'
   ```

2. Close the **Cloud Shell** prompt at the bottom of the portal.

Review: In this exercise, you removed the resources used in this lab.