

Online Lab - Monitoring and Automating Azure Solutions

Topic: Deploying Configuration Management solutions to 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 compute 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: 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 **Create New** option and then, in the text box, type **AADesignLab1201-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 3: Deploy a Linux VM

1. At the top of the portal, click the **Cloud Shell** icon to open a new Cloud Shell instance.
2. In the **Cloud Shell** pane, click the **Upload/Download files** icon and, in the drop-down menu, click **Upload**.
3. In the **Open** dialog box, navigate to the **F:\Labfiles\Mod12\Starter** folder, select the **linux-template.json** file, and click **Open**. The file contains the following template:

```
{ "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#", "contentVersion": "1.0.0.0",
  "parameters": { "userName": { "type": "string", "defaultValue": "Student" }, "password": { "type": "securestring" } }, "variables": {
    "vmName": "[concat('lvm', uniqueString(resourceGroup().id))]",
    "nicName": "[concat('nic', uniqueString(resourceGroup().id))]",
    "publicIPAddressName": "[concat('pip', uniqueString(resourceGroup().id))]", "virtualNetworkName":
```

```
"[concat('vnt', uniqueString(resourceGroup().id))]", "subnetName":
"Linux", "imageReference": { "publisher": "suse", "offer": "opensuse-
leap", "sku": "42.3", "version": "latest" } }, "resources": [ {
"apiVersion": "2017-06-01", "type":
"Microsoft.Network/publicIPAddresses", "name":
"[variables('publicIPAddressName')]", "location":
"[resourceGroup().location]", "properties": {
"publicIPAllocationMethod": "Dynamic" } }, { "apiVersion": "2017-06-
01", "type": "Microsoft.Network/virtualNetworks", "name":
"[variables('virtualNetworkName')]", "location":
"[resourceGroup().location]", "properties": { "addressSpace": {
"addressPrefixes": [ "10.0.0.0/16" ] }, "subnets": [ { "name":
"[variables('subnetName')]", "properties": { "addressPrefix":
"10.0.0.0/24" } } ] } }, { "apiVersion": "2017-10-01", "type":
"Microsoft.Network/networkInterfaces", "name":
"[variables('nicName')]", "location": "[resourceGroup().location]",
"dependsOn": [ "[resourceId('Microsoft.Network/publicIPAddresses/',
variables('publicIPAddressName'))]",
"[resourceId('Microsoft.Network/virtualNetworks/',
variables('virtualNetworkName'))]" ], "properties": {
"ipConfigurations": [ { "name": "ipconfig1", "properties": {
"privateIPAllocationMethod": "Dynamic", "publicIPAddress": { "id":
"[resourceId('Microsoft.Network/publicIPAddresses',
variables('publicIPAddressName'))]" }, "subnet": { "id":
"[concat(resourceId('Microsoft.Network/virtualNetworks', variables('virt
ualNetworkName')), '/subnets/', variables('subnetName'))]" } } } ] } },
{ "apiVersion": "2017-03-30", "type":
"Microsoft.Compute/virtualMachines", "name": "[variables('vmName')]",
"location": "[resourceGroup().location]", "dependsOn": [
"[resourceId('Microsoft.Network/networkInterfaces/',
variables('nicName'))]" ], "properties": { "hardwareProfile": {
"vmSize": "Standard_A1_v2" }, "osProfile": { "computerName":
"[variables('vmName')]", "adminUsername": "[parameters('username')]",
"adminPassword": "[parameters('password')]" }, "storageProfile": {
"imageReference": "[variables('imageReference')]", "osDisk": {
"createOption": "FromImage" } }, "networkProfile": {
"networkInterfaces": [ { "id":
"[resourceId('Microsoft.Network/networkInterfaces',
variables('nicName'))]" } ] } } } ] }
```

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 resource group that will contain the hub virtual network:

```
RESOURCE_GROUP='AADesignLab1202-RG'
```

5. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to create a variable which value designates the Azure region you will use for the deployment (replace the placeholder `<Azure region>` with the name of the Azure region to which you intend to deploy resources in this lab):

```
LOCATION='<Azure region>'
```

6. At the **Cloud Shell** command prompt, type in the following command and press **Enter** to create a new resource group:

```
az group create --name $RESOURCE_GROUP --location $LOCATION
```

7. 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 ~/linux-template.json --parameters password=Pa55w.rd1234
```

8. Do not wait for the deployment to complete before you proceed to the next task.

Task 4: Deploy an Azure Automation account

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 **Automation** and press **Enter**.
3. On the **Everything** blade, in the search results, click **Automation**.
4. On the **Automation** blade, click **Create**.
5. On the **Add Automation Account** blade, perform the following tasks:
 - In the **Name** text box, type **LinuxAutomation**.
 - Leave the **Subscription** drop-down list entry set to its default value.
 - In the **Resource group** section, select the **Create new** option and then, in the text box, type **AADesignLab1203-RG**.
 - In the **Location** drop-down list, select the Azure region matching or near the location where you deployed the Azure VM in the previous task.
 - In the **Create Azure Run As account** section, ensure that **Yes** option is selected.
 - Click the **Create** button.
6. Wait for the provisioning to complete before you proceed to the next task.

Review: In this exercise, you created a Linux VM using an Azure Resource Manager template and provisioned an Azure Automation account from the Azure portal.

Exercise 2: Configure Azure Automation DSC

Task 1: Import Linux PowerShell DSC modules

1. In the hub menu of the Azure portal, click **Resource groups**.
2. On the **Resource groups** blade, click **AADesignLab1203-RG**.

3. On the **AADesignLab1203-RG** blade, click the newly created Azure Automation account.
4. On the **LinuxAutomation** blade, in the **SHARED RESOURCES** section on the left side of the blade, click **Modules gallery**.
5. On the **LinuxAutomation - Modules gallery** blade, perform the following tasks:
 - In the **Search** text box, type **nx** and press **Enter**.
 - In the search results, click the **nx** module.
6. On the **nx** blade, click the **Import** button at the top of the blade.
7. On the **Import** blade, click the **OK** button.
8. Wait for the import process to finish before you proceed to the next task. A status message on the **nx Module** blade will indicate that the module was successfully imported.

Note: This process should take about 2 minutes.

Task 2: Create Linux DSC Configuration

1. Navigate back to the **LinuxAutomation** blade.
2. Back on the **LinuxAutomation** blade, in the **CONFIGURATION MANAGEMENT** section, click **State configuration (DSC)**.
3. On the **LinuxAutomation - State configuration (DSC)** blade, click the **Configurations** link.
4. On the **LinuxAutomation - State configuration (DSC)** blade, click the **+ Add** button at the top of the pane.
5. On the **Import** blade, perform the following tasks:
 - Next to the **Configuration file** field, click the blue button with a folder icon.
 - In the **Choose File to Upload** dialog box, navigate to the **F:\Mod12\Labfiles\Starter** folder.
 - Select the **lampserver.ps1** file.
 - Click the **Open** button to close the dialog and return to the **Import** blade.
 - In the **Name** text box, accept the default entry **lampserver**.
 - In the **Description** text box, type **LAMP Server configuration using PHP and MySQL**.
 - Click the **OK** button.
6. Back in the **DSC configurations** pane, click **Refresh** and then click the newly created **lampserver** configuration.
7. On the **lampserver Configuration** blade, click the **Compile** button at the top of the blade. In the confirmation dialog box, click **Yes** to proceed with compiling the configuration.

8. Wait for the compilation task to finish. To determine the status of the compilation task, review the **STATUS** column of the **Compilation jobs** section of the **lampserver Configuration** blade.

Note: You may need to close and re-open the blade to see the latest compilation status. This blade does not refresh automatically.

Task 3: Onboard Linux VM

1. Navigate back to the **LinuxAutomation - State Configuration (DSC)** blade.
2. Back on the **LinuxAutomation - State Configuration (DSC)** blade, click the **Nodes** link.
3. On the **LinuxAutomation - State configuration (DSC)** blade, click the **+ Add** button at the top of the pane.
4. On the **Virtual Machines** blade, click the entry representing the Linux virtual machine you deployed in the previous exercise.
5. On the virtual machine blade, click **+ Connect**.
6. On the **Registration** blade, perform the following tasks:
 - o Leave the **Registration key** setting with its default value.
 - o In the **Node configuration name** drop-down list, select the **lampserver.localhost** entry.
 - o Leave all remaining settings with their default values.
 - o Click the **OK** button.
7. Wait for the connection process to complete before you proceed to the next step.
8. Navigate back to the **LinuxAutomation - State Configuration (DSC)** blade.
9. Back on the **LinuxAutomation - State Configuration (DSC)** blade, click the **Refresh** button.
10. In the list of DSC nodes, verify that the Linux virtual machine has the **Compliant** status.

Review: In this exercise, you created a PowerShell DSC configuration and applied the configuration to a Linux virtual machine.

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 at the bottom of the portal, 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,'AADesignLab12')].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,'AADesignLab12')].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.