**Create a Windows VM using an ARM template**

**Lab Steps**

**Task 1: Sign in to Azure Portal**

1. Go to the Azure portal by clicking on the **Open Console** button or by using URL [https://portal.azure.com](https://portal.azure.com/).

**Task 2: Explore the ARM template**

1. Download the ARM template which we will use to deploy Azure Windows virtual machine from [**Azure ARM Template**](https://github.com/ssudarshanrajan/az104_projects/raw/refs/heads/main/AZ%20Win%20ARM%20Template.zip) or [Click Here](#ARM) to View the JSON file contents and copy it to a Notepad file and save as ***xxxx.json*** on the local computer.

Before Proceeding Further, understanding the ARM script is vital.

Let's Break It Down into Different Groups And Understand Each Part Separately.

Schema and Metadata:

* The $schema field specifies the version of the deployment template schema being used.
* contentVersion indicates the version of the template.
* metadata provides information about the template generator, including its name, version, and a template hash.

Parameters:

* Defines input parameters that can be provided when deploying the template. Parameters include the administrator username, administrator password, DNS label prefix, public IP details, OS version, VM size, location, and VM name.

Variables:

* Defines variables to store values that are reused in the template.
* Variables include the storage account name, NIC name, address prefix, subnet name, subnet prefix, virtual network name, and network security group name.

Resources:

* Describes the Azure resources that the template will deploy.
  + Storage Account: Creates a storage account to be used for boot diagnostics.
  + Public IP Address: Configures a public IP address with specified settings, including allocation method, SKU, and DNS label prefix.
  + Network Security Group (NSG): Defines an NSG with an inbound security rule allowing RDP traffic on port 3389.
  + Virtual Network: Specifies the creation of a virtual network with a defined address space and subnet configuration, associating it with the NSG.
  + Network Interface (NIC): Creates a NIC with specific IP configurations, associating it with the public IP and subnet.
  + Virtual Machine (VM): Deploys a Windows VM with specified hardware, OS, storage, and network configurations. Boot diagnostics are enabled, and a data disk is added.

Outputs:

* Defines an output to retrieve the fully qualified domain name (FQDN) of the public IP address associated with the VM.

Dependencies:

* The depends on property within each resource specifies the resources that must be deployed before the current resource can be created, ensuring proper dependencies are met.

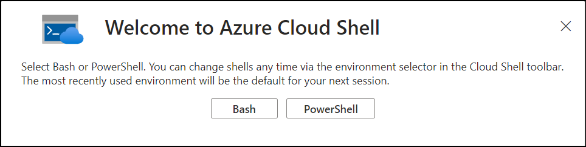
Overall, the template creates a Windows VM with associated networking components, including a public IP address, NSG, virtual network, NIC, and storage account for boot diagnostics. The output provides the hostname (FQDN) of the VM for easy access

**Task 3: Deploy the ARM template**

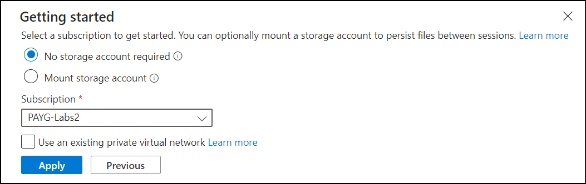
1. In the Azure Portal, open the **Bash** session within the **Cloud Shell** pane by clicking on the **Cloud Shell** icon.



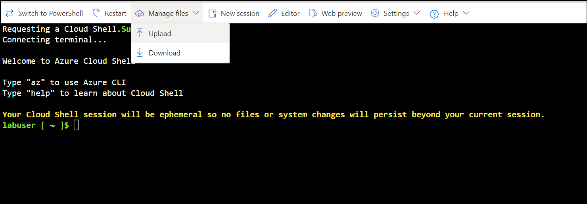
1. In the pop up box, Click on **Bash**.



1. In the getting started box, enter the following details and click on **Apply**.
   * Select **No storage account required.**
   * Subscription: Select **Payg-lab2**



1. In the toolbar of Cloud Shell pane, select the **Manage file** option and then click on **upload** to upload the **template.json** file into the Cloud Shell home directory.

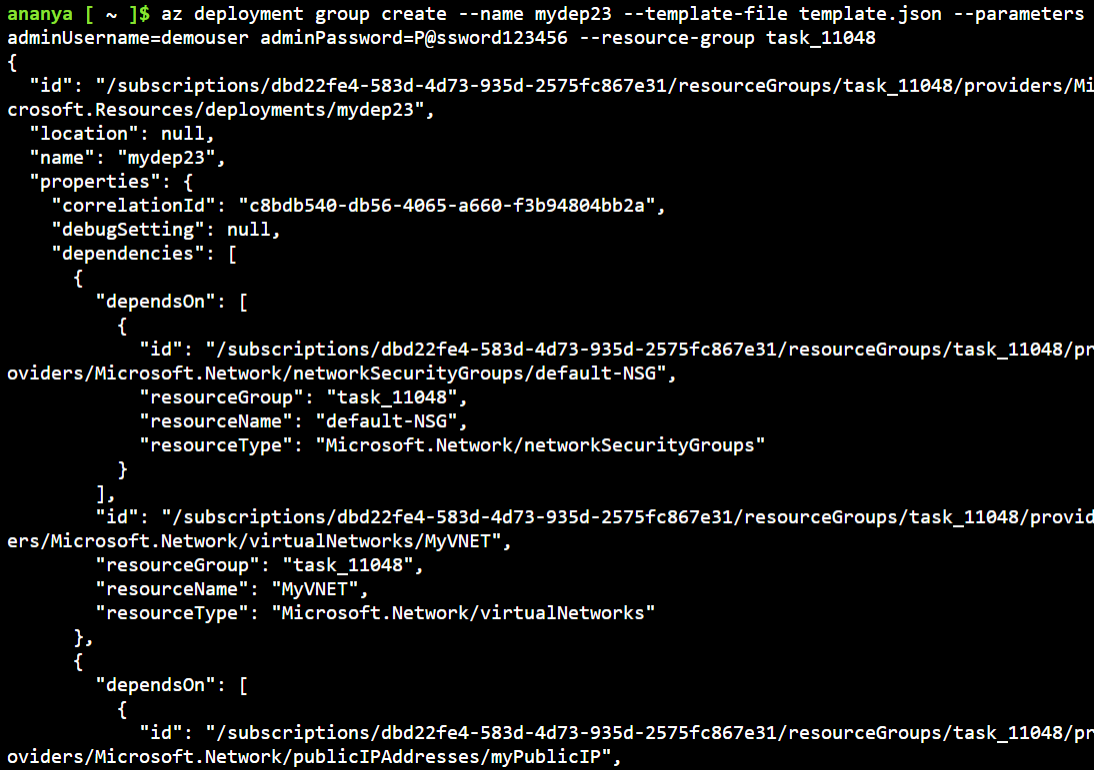


1. Now deploy the resources using the given command. This step may take a few minutes in order for the resources to get deployed successfully.

az deployment group create --name <deployment-name> --template-file <path-to-template-file> --parameters '{"adminUsername": {"value": "<admin-username>"}, "adminPassword": {"value": "<admin-password>"}}' --resource-group <resource-group-name>

**NOTE:** Copy this command to a text editor and replace:

* ‘**<deployment-name>’** with the name of your deployment
* **`<resource-group-name>`** with the name of your resource group which you can get from whizlabs page of the current lab session
* **`<path-to-template-file>`** with the path to file containing the ARM template which in this case would be the file name
* **`<admin-username>`** with the username
* **`<admin-password>`** with the password of length **greater than or equal to 12**.



**Task 4: Verify your deployments**

1. After the successful execution of the above command, you will get a similar output on the CLI.
2. Go to your resource group in Azure portal to check your deployments.
3. Now, from the resources, you can see the Windows VM created.

ARM Template:

{

"$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",

"contentVersion": "1.0.0.0",

"metadata": {

"\_generator": {

"name": "bicep",

"version": "0.8.9.13224",

"templateHash": "15495738823141086515"

}

},

"parameters": {

"adminUsername": {

"type": "string",

"metadata": {

"description": "Username for the Virtual Machine."

}

},

"adminPassword": {

"type": "secureString",

"minLength": 12,

"metadata": {

"description": "Password for the Virtual Machine."

}

},

"dnsLabelPrefix": {

"type": "string",

"defaultValue": "[toLower(format('{0}-{1}', parameters('vmName'), uniqueString(resourceGroup().id, parameters('vmName'))))]",

"metadata": {

"description": "Unique DNS Name for the Public IP used to access the Virtual Machine."

}

},

"publicIpName": {

"type": "string",

"defaultValue": "myPublicIP",

"metadata": {

"description": "Name for the Public IP used to access the Virtual Machine."

}

},

"publicIPAllocationMethod": {

"type": "string",

"defaultValue": "Dynamic",

"allowedValues": [

"Dynamic",

"Static"

],

"metadata": {

"description": "Allocation method for the Public IP used to access the Virtual Machine."

}

},

"publicIpSku": {

"type": "string",

"defaultValue": "Basic",

"allowedValues": [

"Basic",

"Standard"

],

"metadata": {

"description": "SKU for the Public IP used to access the Virtual Machine."

}

},

"OSVersion": {

"type": "string",

"defaultValue": "2022-datacenter-azure-edition-core",

"allowedValues": [

"2008-R2-SP1",

"2008-R2-SP1-smalldisk",

"2012-Datacenter",

"2012-datacenter-gensecond",

"2012-Datacenter-smalldisk",

"2012-datacenter-smalldisk-g2",

"2012-Datacenter-zhcn",

"2012-datacenter-zhcn-g2",

"2012-R2-Datacenter",

"2012-r2-datacenter-gensecond",

"2012-R2-Datacenter-smalldisk",

"2012-r2-datacenter-smalldisk-g2",

"2012-R2-Datacenter-zhcn",

"2012-r2-datacenter-zhcn-g2",

"2016-Datacenter",

"2016-datacenter-gensecond",

"2016-datacenter-gs",

"2016-Datacenter-Server-Core",

"2016-datacenter-server-core-g2",

"2016-Datacenter-Server-Core-smalldisk",

"2016-datacenter-server-core-smalldisk-g2",

"2016-Datacenter-smalldisk",

"2016-datacenter-smalldisk-g2",

"2016-Datacenter-with-Containers",

"2016-datacenter-with-containers-g2",

"2016-datacenter-with-containers-gs",

"2016-Datacenter-zhcn",

"2016-datacenter-zhcn-g2",

"2019-Datacenter",

"2019-Datacenter-Core",

"2019-datacenter-core-g2",

"2019-Datacenter-Core-smalldisk",

"2019-datacenter-core-smalldisk-g2",

"2019-Datacenter-Core-with-Containers",

"2019-datacenter-core-with-containers-g2",

"2019-Datacenter-Core-with-Containers-smalldisk",

"2019-datacenter-core-with-containers-smalldisk-g2",

"2019-datacenter-gensecond",

"2019-datacenter-gs",

"2019-Datacenter-smalldisk",

"2019-datacenter-smalldisk-g2",

"2019-Datacenter-with-Containers",

"2019-datacenter-with-containers-g2",

"2019-datacenter-with-containers-gs",

"2019-Datacenter-with-Containers-smalldisk",

"2019-datacenter-with-containers-smalldisk-g2",

"2019-Datacenter-zhcn",

"2019-datacenter-zhcn-g2",

"2022-datacenter",

"2022-datacenter-azure-edition",

"2022-datacenter-azure-edition-core",

"2022-datacenter-azure-edition-core-smalldisk",

"2022-datacenter-azure-edition-smalldisk",

"2022-datacenter-core",

"2022-datacenter-core-g2",

"2022-datacenter-core-smalldisk",

"2022-datacenter-core-smalldisk-g2",

"2022-datacenter-g2",

"2022-datacenter-smalldisk",

"2022-datacenter-smalldisk-g2"

],

"metadata": {

"description": "The Windows version for the VM. This will pick a fully patched image of this given Windows version."

}

},

"vmSize": {

"type": "string",

"defaultValue": "standard\_b2s",

"metadata": {

"description": "Size of the virtual machine."

}

},

"location": {

"type": "string",

"defaultValue": "[resourceGroup().location]",

"metadata": {

"description": "Location for all resources."

}

},

"vmName": {

"type": "string",

"defaultValue": "simple-vm",

"metadata": {

"description": "Name of the virtual machine."

}

}

},

"variables": {

"storageAccountName": "[format('bootdiags{0}', uniqueString(resourceGroup().id))]",

"nicName": "myVMNic",

"addressPrefix": "10.0.0.0/16",

"subnetName": "Subnet",

"subnetPrefix": "10.0.0.0/24",

"virtualNetworkName": "MyVNET",

"networkSecurityGroupName": "default-NSG"

},

"resources": [

{

"type": "Microsoft.Storage/storageAccounts",

"apiVersion": "2021-04-01",

"name": "[variables('storageAccountName')]",

"location": "[parameters('location')]",

"sku": {

"name": "Standard\_LRS"

},

"kind": "Storage"

},

{

"type": "Microsoft.Network/publicIPAddresses",

"apiVersion": "2021-02-01",

"name": "[parameters('publicIpName')]",

"location": "[parameters('location')]",

"sku": {

"name": "[parameters('publicIpSku')]"

},

"properties": {

"publicIPAllocationMethod": "[parameters('publicIPAllocationMethod')]",

"dnsSettings": {

"domainNameLabel": "[parameters('dnsLabelPrefix')]"

}

}

},

{

"type": "Microsoft.Network/networkSecurityGroups",

"apiVersion": "2021-02-01",

"name": "[variables('networkSecurityGroupName')]",

"location": "[parameters('location')]",

"properties": {

"securityRules": [

{

"name": "default-allow-3389",

"properties": {

"priority": 1000,

"access": "Allow",

"direction": "Inbound",

"destinationPortRange": "3389",

"protocol": "Tcp",

"sourcePortRange": "\*",

"sourceAddressPrefix": "\*",

"destinationAddressPrefix": "\*"

}

}

]

}

},

{

"type": "Microsoft.Network/virtualNetworks",

"apiVersion": "2021-02-01",

"name": "[variables('virtualNetworkName')]",

"location": "[parameters('location')]",

"properties": {

"addressSpace": {

"addressPrefixes": [

"[variables('addressPrefix')]"

]

},

"subnets": [

{

"name": "[variables('subnetName')]",

"properties": {

"addressPrefix": "[variables('subnetPrefix')]",

"networkSecurityGroup": {

"id": "[resourceId('Microsoft.Network/networkSecurityGroups', variables('networkSecurityGroupName'))]"

}

}

}

]

},

"dependsOn": [

"[resourceId('Microsoft.Network/networkSecurityGroups', variables('networkSecurityGroupName'))]"

]

},

{

"type": "Microsoft.Network/networkInterfaces",

"apiVersion": "2021-02-01",

"name": "[variables('nicName')]",

"location": "[parameters('location')]",

"properties": {

"ipConfigurations": [

{

"name": "ipconfig1",

"properties": {

"privateIPAllocationMethod": "Dynamic",

"publicIPAddress": {

"id": "[resourceId('Microsoft.Network/publicIPAddresses', parameters('publicIpName'))]"

},

"subnet": {

"id": "[resourceId('Microsoft.Network/virtualNetworks/subnets', variables('virtualNetworkName'), variables('subnetName'))]"

}

}

}

]

},

"dependsOn": [

"[resourceId('Microsoft.Network/publicIPAddresses', parameters('publicIpName'))]",

"[resourceId('Microsoft.Network/virtualNetworks', variables('virtualNetworkName'))]"

]

},

{

"type": "Microsoft.Compute/virtualMachines",

"apiVersion": "2021-03-01",

"name": "[parameters('vmName')]",

"location": "[parameters('location')]",

"properties": {

"hardwareProfile": {

"vmSize": "[parameters('vmSize')]"

},

"osProfile": {

"computerName": "[parameters('vmName')]",

"adminUsername": "[parameters('adminUsername')]",

"adminPassword": "[parameters('adminPassword')]"

},

"storageProfile": {

"imageReference": {

"publisher": "MicrosoftWindowsServer",

"offer": "WindowsServer",

"sku": "[parameters('OSVersion')]",

"version": "latest"

},

"osDisk": {

"createOption": "FromImage",

"managedDisk": {

"storageAccountType": "StandardSSD\_LRS"

}

},

"dataDisks": [

{

"diskSizeGB": 1023,

"lun": 0,

"createOption": "Empty"

}

]

},

"networkProfile": {

"networkInterfaces": [

{

"id": "[resourceId('Microsoft.Network/networkInterfaces', variables('nicName'))]"

}

]

},

"diagnosticsProfile": {

"bootDiagnostics": {

"enabled": true,

"storageUri": "[reference(resourceId('Microsoft.Storage/storageAccounts', variables('storageAccountName'))).primaryEndpoints.blob]"

}

}

},

"dependsOn": [

"[resourceId('Microsoft.Network/networkInterfaces', variables('nicName'))]",

"[resourceId('Microsoft.Storage/storageAccounts', variables('storageAccountName'))]"

]

}

],

"outputs": {

"hostname": {

"type": "string",

"value": "[reference(resourceId('Microsoft.Network/publicIPAddresses', parameters('publicIpName'))).dnsSettings.fqdn]"

}

}

}