**Automated Web Hosting Solution with single click replication to UAT And Prod Environment**

To build an automated web hosting solution using Azure Resource Manager (ARM) templates, we'll create a setup that includes virtual machine-based web servers, an Azure Load Balancer, and a method for deploying this setup to Development, UAT (test), and Production environments using environment-specific parameters.

Objective:

Create an automated web hosting solution with single-click deployment to multiple environments (Development, UAT, Production) using ARM templates. The solution includes:

* Virtual machines for hosting web servers.
* An Azure Load Balancer to distribute traffic.
* Environment-specific parameterization for easy replication and configuration management.

Prerequisites:

* Azure subscription.
* Azure CLI installed and authenticated.
* PowerShell installed.
* Resource groups for each environment (dev, uat, prod).
* Virtual Network and subnets (or create them using ARM templates).
* Service principal (optional, for automated deployments).
* Azure Key Vault (optional, for storing sensitive information like passwords).

**Main Template**

This template links to individual templates for the web server and load balancer.

templates/mainTemplate.json

|  |
| --- |
| {  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",  "contentVersion": "1.0.0.0",  "resources": [  {  "type": "Microsoft.Resources/deployments",  "apiVersion": "2021-04-01",  "name": "webServerDeployment",  "properties": {  "mode": "Incremental",  "templateLink": {  "uri": "[concat(parameters('\_artifactsLocation'), 'webServerTemplate.json', parameters('\_artifactsLocationSasToken'))]",  "contentVersion": "1.0.0.0"  },  "parameters": {  "vmNamePrefix": { "value": "[parameters('vmNamePrefix')]" },  "vmSize": { "value": "[parameters('vmSize')]" },  "adminUsername": { "value": "[parameters('adminUsername')]" },  "adminPassword": { "value": "[parameters('adminPassword')]" }  }  }  },  {  "type": "Microsoft.Resources/deployments",  "apiVersion": "2021-04-01",  "name": "loadBalancerDeployment",  "properties": {  "mode": "Incremental",  "templateLink": {  "uri": "[concat(parameters('\_artifactsLocation'), 'loadBalancerTemplate.json', parameters('\_artifactsLocationSasToken'))]",  "contentVersion": "1.0.0.0"  },  "parameters": {  "vmNamePrefix": { "value": "[parameters('vmNamePrefix')]" }  }  }  }  ],  "parameters": {  "\_artifactsLocation": {  "type": "string",  "metadata": {  "description": "The base URI where artifacts are located."  }  },  "\_artifactsLocationSasToken": {  "type": "string",  "metadata": {  "description": "The SAS token for the artifacts."  },  "defaultValue": ""  },  "vmNamePrefix": {  "type": "string",  "metadata": {  "description": "Prefix for the virtual machine names."  }  },  "vmSize": {  "type": "string",  "defaultValue": "Standard\_B1s",  "metadata": {  "description": "Size of the virtual machines."  }  },  "adminUsername": {  "type": "string",  "metadata": {  "description": "Admin username for the virtual machines."  }  },  "adminPassword": {  "type": "securestring",  "metadata": {  "description": "Admin password for the virtual machines." } } } } |

**Web Server Template**

This template deploys the virtual machines.

templates/webServerTemplate.json

|  |
| --- |
| {  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",  "contentVersion": "1.0.0.0",  "resources": [  {  "type": "Microsoft.Network/networkInterfaces",  "apiVersion": "2021-02-01",  "name": "[concat(parameters('vmNamePrefix'), '-nic')]",  "location": "[resourceGroup().location]",  "properties": {  "ipConfigurations": [  {  "name": "ipconfig1",  "properties": {  "subnet": {  "id": "[variables('subnetRef')]"  },  "privateIPAllocationMethod": "Dynamic"  }  }  ]  }  },  {  "type": "Microsoft.Compute/virtualMachines",  "apiVersion": "2021-03-01",  "name": "[concat(parameters('vmNamePrefix'), '-vm')]",  "location": "[resourceGroup().location]",  "properties": {  "hardwareProfile": {  "vmSize": "[parameters('vmSize')]"  },  "osProfile": {  "computerName": "[concat(parameters('vmNamePrefix'), '-vm')]",  "adminUsername": "[parameters('adminUsername')]",  "adminPassword": "[parameters('adminPassword')]",  "customData": "[base64(concat('#!/bin/bash\nsudo apt-get update\nsudo apt-get install -y apache2\nsudo systemctl start apache2\nsudo systemctl enable apache2\n'))]"  },  "networkProfile": {  "networkInterfaces": [  {  "id": "[resourceId('Microsoft.Network/networkInterfaces', concat(parameters('vmNamePrefix'), '-nic'))]"  }  ]  },  "storageProfile": {  "imageReference": {  "publisher": "Canonical",  "offer": "UbuntuServer",  "sku": "18.04-LTS",  "version": "latest"  },  "osDisk": {  "createOption": "FromImage"  }  }  }  }  ],  "parameters": {  "vmNamePrefix": {  "type": "string",  "metadata": {  "description": "Prefix for the virtual machine names."  }  },  "vmSize": {  "type": "string",  "defaultValue": "Standard\_B1s",  "metadata": {  "description": "Size of the virtual machines."  }  },  "adminUsername": {  "type": "string",  "metadata": {  "description": "Admin username for the virtual machines."  }  },  "adminPassword": {  "type": "securestring",  "metadata": {  "description": "Admin password for the virtual machines."  }  }  }  } |

**Load Balancer Template**

This template deploys the Azure Load Balancer.

templates/loadBalancerTemplate.json

|  |
| --- |
| {  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",  "contentVersion": "1.0.0.0",  "resources": [  {  "type": "Microsoft.Network/loadBalancers",  "apiVersion": "2021-02-01",  "name": "[concat(parameters('vmNamePrefix'), '-lb')]",  "location": "[resourceGroup().location]",  "properties": {  "frontendIPConfigurations": [  {  "name": "LoadBalancerFrontEnd",  "properties": {  "subnet": {  "id": "[variables('subnetRef')]"  },  "privateIPAllocationMethod": "Dynamic"  }  }  ],  "backendAddressPools": [  {  "name": "LoadBalancerBackEndPool"  }  ],  "loadBalancingRules": [  {  "name": "HTTP",  "properties": {  "frontendIPConfiguration": {  "id": "[concat(variables('lbID'), '/frontendIPConfigurations/LoadBalancerFrontEnd')]"  },  "backendAddressPool": {  "id": "[concat(variables('lbID'), '/backendAddressPools/LoadBalancerBackEndPool')]"  },  "probe": {  "id": "[concat(variables('lbID'), '/probes/HttpProbe')]"  },  "protocol": "Tcp",  "frontendPort": 80,  "backendPort": 80,  "enableFloatingIP": false,  "idleTimeoutInMinutes": 4,  "loadDistribution": "Default"  }  }  ],  "probes": [  {  "name": "HttpProbe",  "properties": {  "protocol": "Http",  "port": 80,  "requestPath": "/",  "intervalInSeconds": 5,  "numberOfProbes": 2  }  }  ]  }  },  {  "type": "Microsoft.Network/networkInterfaces",  "apiVersion": "2021-02-01",  "name": "[concat(parameters('vmNamePrefix'), '-nic')]",  "location": "[resourceGroup().location]",  "properties": {  "ipConfigurations": [  {  "name": "ipconfig1",  "properties": {  "subnet": {  "id": "[variables('subnetRef')]"  },  "loadBalancerBackendAddressPools": [  {  "id": "[concat(variables('lbID'), '/backendAddressPools/LoadBalancerBackEndPool')]"  }  ],  "privateIPAllocationMethod": "Dynamic"  }  }  ]  }  }  ],  "parameters": {  "vmNamePrefix": {  "type": "string",  "metadata": {  "description": "Prefix for the virtual machine names."  }  }  }  } |

**Environment Parameter Files**

environments/dev.parameters.json

|  |
| --- |
| {  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentParameters.json#",  "contentVersion": "1.0.0.0",  "parameters": {  "vmNamePrefix": {  "value": "dev-web"  },  "vmSize": {  "value": "Standard\_B1s"  },  "adminUsername": {  "value": "adminUser"  },  "adminPassword": {  "value": "P@ssw0rd123"  },  "\_artifactsLocation": {  "value": "https://yourstorageaccount.blob.core.windows.net/templates/"  },  "\_artifactsLocationSasToken": {  "value": "?sasToken"  }  }  } |

environments/uat.parameters.json

|  |
| --- |
| {  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentParameters.json#",  "contentVersion": "1.0.0.0",  "parameters": {  "vmNamePrefix": {  "value": "uat-web"  },  "vmSize": {  "value": "Standard\_B2s"  },  "adminUsername": {  "value": "uatAdminUser"  },  "adminPassword": {  "value": "U@tP@ssw0rd123"  },  "\_artifactsLocation": {  "value": "https://yourstorageaccount.blob.core.windows.net/templates/"  },  "\_artifactsLocationSasToken": {  "value": "?sasToken"  }  }  } |

environments/prod.parameters.json

|  |
| --- |
| {  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentParameters.json#",  "contentVersion": "1.0.0.0",  "parameters": {  "vmNamePrefix": {  "value": "prod-web"  },  "vmSize": {  "value": "Standard\_B2ms"  },  "adminUsername": {  "value": "prodAdminUser"  },  "adminPassword": {  "value": "Pr0dP@ssw0rd123"  },  "\_artifactsLocation": {  "value": "https://yourstorageaccount.blob.core.windows.net/templates/"  },  "\_artifactsLocationSasToken": {  "value": "?sasToken"  }  }  } |

**Deployment Script(PowerShell)**

deploy.ps1

|  |
| --- |
| param (  [string]$environment  )  $ErrorActionPreference = "Stop"  $parameterFile = ".\environments\$environment.parameters.json"  $templateFile = ".\templates\mainTemplate.json"  $resourceGroupName = "rg-$environment"  $location = "East US"  # Create Resource Group  Write-Host "Creating resource group..."  az group create --name $resourceGroupName --location $location  # Deploy ARM Template  Write-Host "Deploying ARM template..."  az deployment group create --resource-group $resourceGroupName --template-file $templateFile --parameters $parameterFile  Write-Host "Deployment complete." |

**Deploy the Solution**

To deploy the solution, run the PowerShell script with the appropriate environment parameter.

PowerShell:

|  |
| --- |
| .\deploy.ps1 -environment "dev"  .\deploy.ps1 -environment "uat"  .\deploy.ps1 -environment "prod" |

This setup allows you to dynamically deploy your web hosting solution to different environments (Development, UAT, Production) with a single click by using ARM templates and parameter files. This approach ensures consistency across environments and makes it easy to manage infrastructure as code in Azure.