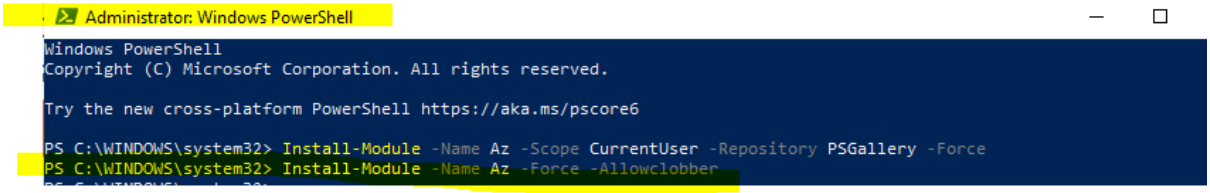
Contents

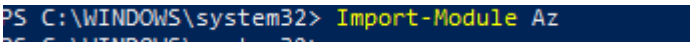
Azure Resource Manager Templates provide simple and unified way to create and manage your Azure environments for both developers and administrators. With simple JSON files you can deploy your environment in consistent fashion and at great speed.

Pre Requiste Setup :

1) Setting up Powershell :

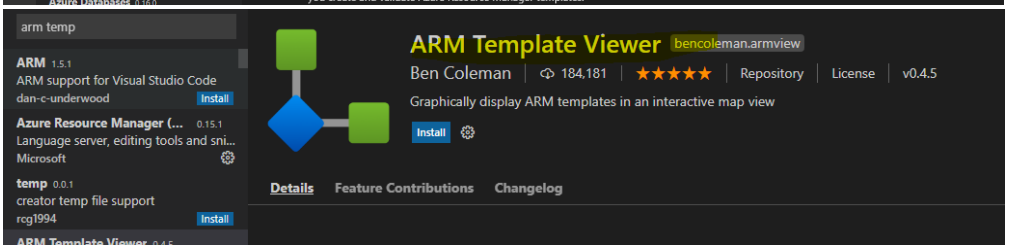


Run

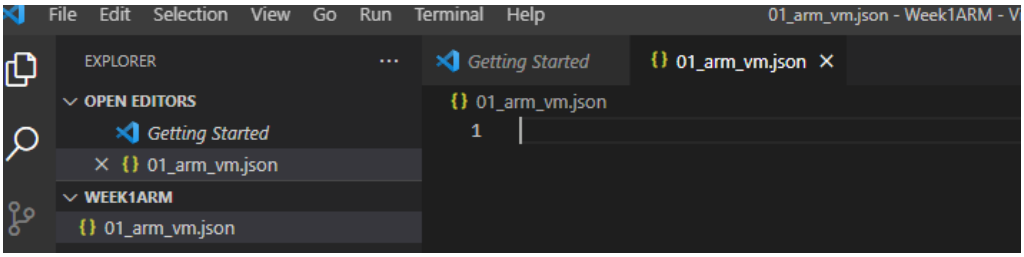


https://docs.microsoft.com/en-us/powershell/azure/install-az-ps?view=azps-6.0.0

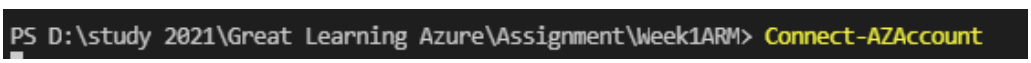
2) Open Visiual Studio Code and Install ARM tools and ARM Template Viewer

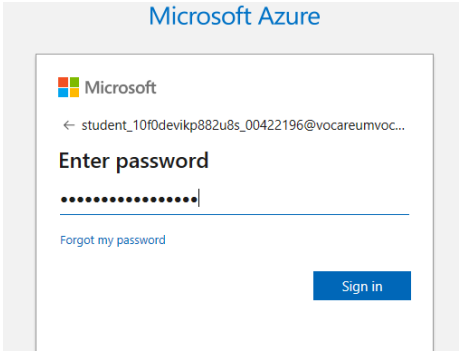


3) Create json file json file



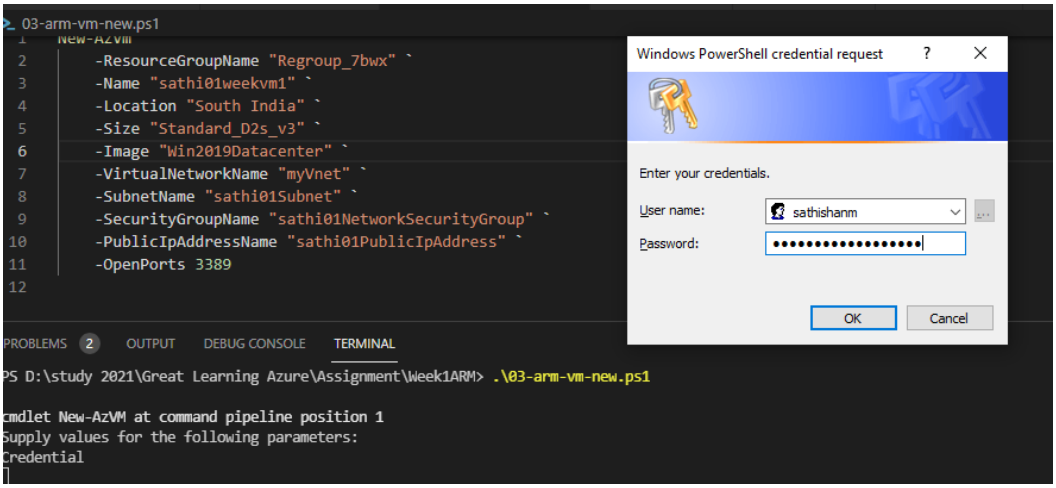
4) Test azure connection : connect-azaccount



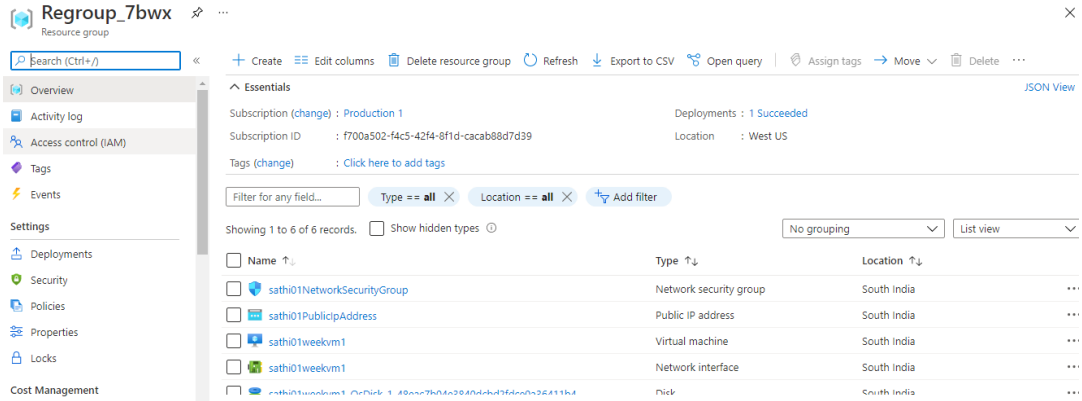



```
Account                                                    SubscriptionName TenantId
-------                                                    ---------------- --------
student_10f0devikp882u8s_00422196@vocareumvocareum.onmicrosoft.com Production 1     9b7cbd77-6d6b-4879-8aba-...
```

Create VM using powershell




```
ResourceGroupName       : Regroup_7bwx
Id                      : /subscriptions/f700a502-f4c5-42f4-8f1d-cacab88d7d39/resourceGroups/Regroup_7bwx/providers/Microsoft.Comp
ute/virtualMachines/sathi01weekvm1
VmId                    : 3674767e-dbb3-4e99-a2de-92329c7439d6
Name                    : sathi01weekvm1
Type                    : Microsoft.Compute/virtualMachines
Location                : southindia
Tags                    : {}
HardwareProfile         : {VmSize}
NetworkProfile          : {NetworkInterfaces}
OSProfile               : {ComputerName, AdminUsername, WindowsConfiguration, Secrets, AllowExtensionOperations,
RequireGuestProvisionSignal}
ProvisioningState       : Succeeded
StorageProfile          : {ImageReference, OsDisk, DataDisks}
FullyQualifiedDomainName : sathi01weekvm1-b1be95.South India.cloudapp.azure.com
```



Create VM Using ARM Template

01_arm_vm.json

```
{
  "$schema": "https://schema.management.azure.com/schemas/2019-04-
01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "adminUsername": {
      "type": "string",
      "defaultValue": "sathishanm",
      "metadata": {
        "description": "Username for the Virtual Machine."
      }
    },
    "adminPassword": {
      "type": "securestring",
      "minLength": 12,
      "metadata": {
        "description": "Password for the Virtual Machine."
      }
    },
    "dnsLabelPrefix": {
      "type": "string",
      "defaultValue": "[toLower(concat(parameters('vmName'),'-
', uniqueString(resourceGroup().id, parameters('vmName'))))]",
      "metadata": {
        "description": "Unique DNS Name for the Public IP used to access the V
irtual Machine."
      }
    },
    "publicIpName": {
      "type": "string",
      "defaultValue": "myPublicIP",
      "metadata": {
        "description": "Name for the Public IP used to access the Virtual Mach
ine."
      }
    },
    "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 Machi
ne."
      }
    },

    "OSVersion": {
      "type": "string",
      "defaultValue": "2019-Datacenter",
      "allowedValues": [
        "2008-R2-SP1",
        "2012-Datacenter",
        "2012-R2-Datacenter",
        "2016-Nano-Server",
        "2016-Datacenter-with-Containers",
        "2016-Datacenter",
        "2019-Datacenter",
        "2019-Datacenter-Core",
        "2019-Datacenter-Core-smalldisk",
        "2019-Datacenter-Core-with-Containers",
        "2019-Datacenter-Core-with-Containers-smalldisk",
        "2019-Datacenter-smalldisk",
        "2019-Datacenter-with-Containers",
        "2019-Datacenter-with-Containers-smalldisk"
      ],
      "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_DS1_v2",
      "metadata": {
        "description": "Size of the virtual machine."
      }
    },
    "location": {
      "type": "string",
      "defaultValue": "[resourceGroup().location]",
      "metadata": {
        "description": "Location for all resources."
      }
    },
    "vmName": {
      "type": "string",
      "defaultValue": "saweekdemovm",
      "metadata": {
        "description": "Name of the virtual machine."
      }
    }
  },
  "variables": {
    "storageAccountName": "[concat('bootdiags', uniquestring(resourceGroup().i
d))]",
```

```
    "nicName": "savnnickname",
    "addressPrefix": "10.0.0.0/16",
    "subnetName": "Subnet",
    "subnetPrefix": "10.0.0.0/24",
    "virtualNetworkName": "savirnetdemo",
    "subnetRef": "[resourceId('Microsoft.Network/virtualNetworks/subnets', var
iables('virtualNetworkName'), variables('subnetName'))]",
    "networkSecurityGroupName": "default-NSG"
  },
  "resources": [
    {
      "type": "Microsoft.Storage/storageAccounts",
      "apiVersion": "2019-06-01",
      "name": "[variables('storageAccountName')]",
      "location": "[parameters('location')]",
      "sku": {
        "name": "Standard_LRS"
      },
      "kind": "Storage",
      "properties": {}
    },
    {
      "type": "Microsoft.Network/publicIPAddresses",
      "apiVersion": "2020-06-01",
      "name": "[parameters('publicIPName')]",
      "location": "[parameters('location')]",
      "sku": {
        "name": "[parameters('publicIpSku')]"
      },
      "properties": {
        "publicIPAllocationMethod": "[parameters('publicIPAllocationMethod')]"
,
        "dnsSettings": {
          "domainNameLabel": "[parameters('dnsLabelPrefix')]"
        }
      }
    },
    {
      "type": "Microsoft.Network/networkSecurityGroups",
      "apiVersion": "2020-06-01",
      "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": "2020-06-01",
      "name": "[variables('virtualNetworkName')]",
      "location": "[parameters('location')]",
      "dependsOn": [
        "[resourceId('Microsoft.Network/networkSecurityGroups', variables('net
workSecurityGroupName'))]"
      ],
      "properties": {
        "addressSpace": {
          "addressPrefixes": [
            "[variables('addressPrefix')]"
          ]
        },
        "subnets": [
          {
            "name": "[variables('subnetName')]",
            "properties": {
              "addressPrefix": "[variables('subnetPrefix')]",
              "networkSecurityGroup": {
                "id": "[resourceId('Microsoft.Network/networkSecurityGroups',
variables('networkSecurityGroupName'))]"
              }
            }
          }
        ]
      }
    },
    {
      "type": "Microsoft.Network/networkInterfaces",
      "apiVersion": "2020-06-01",
      "name": "[variables('nicName')]",
      "location": "[parameters('location')]",
      "dependsOn": [
        "[resourceId('Microsoft.Network/publicIPAddresses', parameters('public
IPName'))]",
        "[resourceId('Microsoft.Network/virtualNetworks', variables('virtualNe
tworkName'))]"
      ],
      "properties": {
        "ipConfigurations": [
          {
            "name": "ipconfig1",
            "properties": {
              "privateIPAllocationMethod": "Dynamic",
              "publicIPAddress": {
                "id": "[resourceId('Microsoft.Network/publicIPAddresses', para
meters('publicIPName'))]"
              },
              "subnet": {
                "id": "[variables('subnetRef')]"
              }
            }
          }
        ]
      }
    },
```

```
    {
      "type": "Microsoft.Compute/virtualMachines",
      "apiVersion": "2020-06-01",
      "name": "[parameters('vmName')]",
      "location": "[parameters('location')]",
      "dependsOn": [
        "[resourceId('Microsoft.Storage/storageAccounts', variables('storageAc
countName'))]",
        "[resourceId('Microsoft.Network/networkInterfaces', variables('nicName
'))]"
      ],
      "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', variab
les('nicName'))]"
            }
          ]
        },
        "diagnosticsProfile": {
          "bootDiagnostics": {
            "enabled": true,
            "storageUri": "[reference(resourceId('Microsoft.Storage/storageAcc
ounts', variables('storageAccountName'))).primaryEndpoints.blob]"
          }
        }
      }
    }
  ],
```

```
  "outputs": {
    "hostname": {
      "type": "string",
      "value": "[reference(parameters('publicIPName')).dnsSettings.fqdn]"
    }
  }
}
```

01_arm_vm_run.ps1

```
$resourceGroupName = "Regroup_1tbb"
$location = "westus"
$OSVersion = "2019-Datacenter"
$adminUsername = "sathishanm"
$adminPassword = Read-Host -Prompt "Enter the administrator password" -
AsSecureString
$dnsLabelPrefix = "week1demoip"

New-AzResourceGroup -Name $resourceGroupName -Location "$location"
New-AzResourceGroupDeployment `
    -ResourceGroupName $resourceGroupName `
    -TemplateFile "01_arm_vm.json" `
    -adminUsername $adminUsername `
    -adminPassword $adminPassword `
    -dnsLabelPrefix $dnsLabelPrefix
```

```
PS D:\study 2021\Great Learning Azure\Assignment\Week1ARM> .\01_arm_vm_run.ps1
Enter the administrator password: ******************

Confirm
Provided resource group already exists. Are you sure you want to update it?
[Y] Yes  [N] No  [S] Suspend  [?] Help (default is "Y"): Y


ResourceGroupName : Regroup_1tbb
Location          : westus
ProvisioningState : Succeeded
Tags              :
ResourceId        : /subscriptions/f700a502-f4c5-42f4-8f1d-cacab88d7d39/resourceGroups/Regroup_1tbb

```

Deployment name	Status	Last modified	Duration	Related events
01_arm_vm	Deploying	6/10/2021, 10:19:31 PM	21 seconds	Related events

Showing 1 to 5 of 5 records. Show hidden types ⓘ — No grouping — List view

Name	Type	Location
savirnetdemo	Virtual network	West US
savnnickname	Network interface	West US
saweekdemovm	Virtual machine	West US
saweekdemovm_disk2_f4ddb46a1ce941818d1e65d4f4ad4927	Disk	West US
saweekdemovm_OsDisk_1_792084798945463f83865f5e41054dac	Disk	West US

a) OS: Windows 2019 datacenter

```
"OSVersion": {
  "type": "string",
  "defaultValue": "2019-Datacenter",
```

```
"storageProfile": {
  "imageReference": {
    "publisher": "MicrosoftWindowsServer",
    "offer": "WindowsServer",
    "sku": "[parameters('OSVersion')]",
    "version": "latest"
  },
},
```

```
"vmSize": {
  "type": "string",
  "defaultValue": "Standard_DS1_v2",
  "metadata": {
    "description": "Size of the virtual machine."
  }
},
```

b) VM size: DS1_v2

```
"hardwareProfile": {
  "vmSize": "[parameters('vmSize')]"
},
```

c) Open ports: 3389

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

d) Storage option: Standard SSD

```
"createOption": "FromImage",
"managedDisk": {
  "storageAccountType": "StandardSSD_LRS"
}
},
```

Azure Virtual Network (VNet)

Azure Virtual Network (VNet) is the fundamental building block for your private network in Azure. VNet enables many types of Azure resources, such as Azure Virtual Machines (VM), to securely communicate with each other, the internet, and on-premises networks.There is no charge for using Azure VNet, it is free of cost. Standard charges are applicable for resources, such as Virtual Machines (VMs)

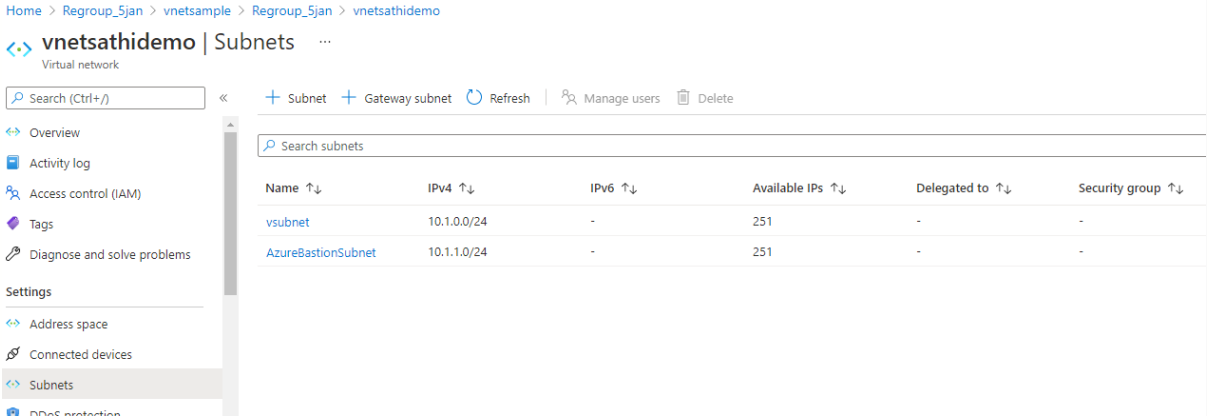
ARM for creating Vnet

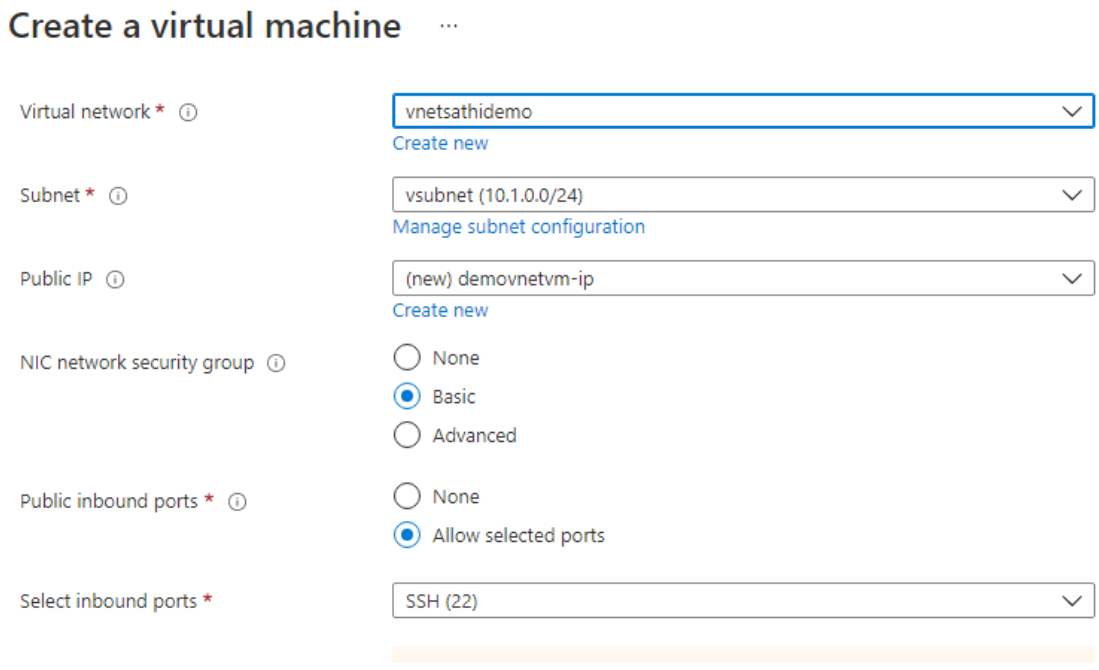
```
{
    "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
    "contentVersion": "1.0.0.0",
    "parameters": {
```

```
        "virtualNetworks_vnetsathidemo_name": {
            "defaultValue": "vnetsathidemo",
            "type": "String"
        }
    },
    "variables": {},
    "resources": [
        {
            "type": "Microsoft.Network/virtualNetworks",
            "apiVersion": "2020-11-01",
            "name": "[parameters('virtualNetworks_vnetsathidemo_name')]",
            "location": "westus",
            "properties": {
                "addressSpace": {
                    "addressPrefixes": [
                        "10.1.0.0/16"
                    ]
                },
                "subnets": [
                    {
                        "name": "vsubnet",
                        "properties": {
                            "addressPrefix": "10.1.0.0/24",
                            "delegations": [],
                            "privateEndpointNetworkPolicies": "Enabled",
                            "privateLinkServiceNetworkPolicies": "Enabled"
                        }
                    },
                    {
                        "name": "AzureBastionSubnet",
                        "properties": {
                            "addressPrefix": "10.1.1.0/24",
                            "delegations": [],
                            "privateEndpointNetworkPolicies": "Enabled",
                            "privateLinkServiceNetworkPolicies": "Enabled"
                        }
                    }
                ],
                "virtualNetworkPeerings": [],
                "enableDdosProtection": false
            }
        },
        {
            "type": "Microsoft.Network/virtualNetworks/subnets",
            "apiVersion": "2020-11-01",
            "name": "[concat(parameters('virtualNetworks_vnetsathidemo_name'), '/AzureBastionSubnet')]",
            "dependsOn": [
                "[resourceId('Microsoft.Network/virtualNetworks', parameters('virtualNetworks_vnetsathidemo_name'))]"
            ],
            "properties": {
                "addressPrefix": "10.1.1.0/24",
                "delegations": [],
                "privateEndpointNetworkPolicies": "Enabled",
                "privateLinkServiceNetworkPolicies": "Enabled"
            }
        },
        {
            "type": "Microsoft.Network/virtualNetworks/subnets",
            "apiVersion": "2020-11-01",
            "name": "[concat(parameters('virtualNetworks_vnetsathidemo_name'), '/vsubnet')]",
            "dependsOn": [
                "[resourceId('Microsoft.Network/virtualNetworks', parameters('virtualNetworks_vnetsathidemo_name'))]"
            ],
            "properties": {
                "addressPrefix": "10.1.0.0/24",
                "delegations": [],
                "privateEndpointNetworkPolicies": "Enabled",
                "privateLinkServiceNetworkPolicies": "Enabled"
            }
        }
    ]
}
```

Deploy ARM

```powershell
vnetdeploy.ps1 > ...
1   $resourceGroupName = "Regroup_5jan"
2   $location = "westus"
3
4   New-AzResourceGroup -Name $resourceGroupName -Location "$location"
5   New-AzResourceGroupDeployment `
6       -ResourceGroupName $resourceGroupName `
7       -TemplateFile "vnetsample.json" `
8
```





NETWORK SECURITY GROUPS

Sample template for allwing RDP connection on 3389 port

```
"securityRules": [
  {
    "name": "default-allow-3389",
    "properties": {
      "priority": 1000,
      "access": "Allow",
      "direction": "Inbound",
      "destinationPortRange": "3389",
      "protocol": "Tcp",
      "sourcePortRange": "*",
      "sourceAddressPrefix": "*",
      "destinationAddressPrefix": "*"
    }
  }
```

ARM template

```
{
  "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
   "securityGroupName": {
      "type": "string",
      "metadata": {
        "description": "Name of the network security group"
      }
    }
  },
  "variables": {
    "location": "[resourceGroup().location]"
  },
  "resources": [
    {
      "apiVersion": "2015-05-01-preview",
      "type": "Microsoft.Network/networkSecurityGroups",
      "name": "[parameters('securityGroupName')]",
      "location": "[variables('location')]",
      "properties": {
        "securityRules": [
          {
            "name": "allow_http",
            "properties": {
              "description": "Permit access to HTTP",
              "protocol": "Tcp",
              "sourcePortRange": "*",
              "destinationPortRange": "80",
              "sourceAddressPrefix": "*",
              "destinationAddressPrefix": "*",
              "access": "Allow",
              "priority": 105,
              "direction": "Inbound"
            }
          },
          {
            "name": "allow_https",
            "properties": {
              "description": "Permit access to HTTPS",
              "protocol": "Tcp",
              "sourcePortRange": "*",
              "destinationPortRange": "443",
              "sourceAddressPrefix": "*",
              "destinationAddressPrefix": "*",
              "access": "Allow",
              "priority": 110,
```

```
      "direction": "Inbound"
    }
  },
  {
    "name": "allow_h323cs",
    "properties": {
      "description": "Permit access to H.323 CS",
      "protocol": "Tcp",
      "sourcePortRange": "*",
      "destinationPortRange": "1720",
      "sourceAddressPrefix": "*",
      "destinationAddressPrefix": "*",
      "access": "Allow",
      "priority": 115,
      "direction": "Inbound"
    }
  },
  {
    "name": "allow_sip_tcp",
    "properties": {
      "description": "Permit access to SIP/TCP",
      "protocol": "Tcp",
      "sourcePortRange": "*",
      "destinationPortRange": "5060",
      "sourceAddressPrefix": "*",
      "destinationAddressPrefix": "*",
      "access": "Allow",
      "priority": 120,
      "direction": "Inbound"
    }
  },
  {
    "name": "allow_sip_tls",
    "properties": {
      "description": "Permit access to SIP/TLS",
      "protocol": "Tcp",
      "sourcePortRange": "*",
      "destinationPortRange": "5061",
      "sourceAddressPrefix": "*",
      "destinationAddressPrefix": "*",
      "access": "Allow",
      "priority": 125,
      "direction": "Inbound"
    }
  },
  {
    "name": "allow_signalling_tcp",
    "properties": {
      "description": "Permit access to ephemeral TCP call signalling ports",
      "protocol": "Tcp",
      "sourcePortRange": "*",
      "destinationPortRange": "33000-39999",
      "sourceAddressPrefix": "*",
      "destinationAddressPrefix": "*",
      "access": "Allow",
      "priority": 130,
      "direction": "Inbound"
    }
  },
  {
    "name": "allow_media_tcp",
    "properties": {
      "description": "Permit access to ephemeral TCP media ports",
      "protocol": "Tcp",
      "sourcePortRange": "*",
      "destinationPortRange": "40000-49999",
      "sourceAddressPrefix": "*",
      "destinationAddressPrefix": "*",
      "access": "Allow",
      "priority": 135,
      "direction": "Inbound"
    }
  },
```

```
        {
          "name": "allow_h323ls",
          "properties": {
            "description": "Permit access to H.323 LS",
            "protocol": "Udp",
            "sourcePortRange": "*",
            "destinationPortRange": "1719",
            "sourceAddressPrefix": "*",
            "destinationAddressPrefix": "*",
            "access": "Allow",
            "priority": 140,
            "direction": "Inbound"
          }
        },
        {
          "name": "allow_signalling_udp",
          "properties": {
            "description": "Permit access to ephemeral UDP call signalling ports",
            "protocol": "Udp",
            "sourcePortRange": "*",
            "destinationPortRange": "33000-39999",
            "sourceAddressPrefix": "*",
            "destinationAddressPrefix": "*",
            "access": "Allow",
            "priority": 150,
            "direction": "Inbound"
          }
        },
        {
          "name": "allow_media_udp",
          "properties": {
            "description": "Permit access to ephemeral UDP media ports",
            "protocol": "Udp",
            "sourcePortRange": "*",
            "destinationPortRange": "40000-49999",
            "sourceAddressPrefix": "*",
            "destinationAddressPrefix": "*",
            "access": "Allow",
            "priority": 155,
            "direction": "Inbound"
          }
        }
      ]
    }
  }
 ]
}
```

Deploy

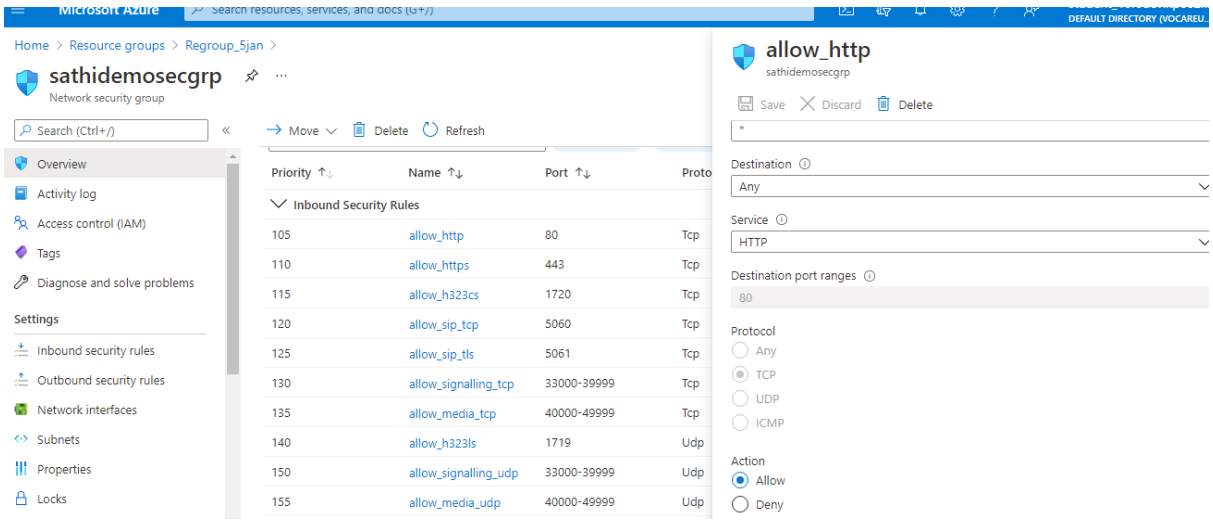
securitygroupdeploy.ps1 | 05-arm-vm-deploy.json | 05-arm-vm-param.json | 05-arm-vm-run.ps1 | 02-arm-vm

securitygroupdeploy.ps1 > ...

```
1  $resourceGroupName = "Regroup_5jan"
2  $location = "westus"
3  $securityGroupName =Read-Host -Prompt "Name of the network security group"
4
5  New-AzResourceGroup -Name $resourceGroupName -Location "$location"
6  New-AzResourceGroupDeployment `
7      -ResourceGroupName $resourceGroupName `
8      -TemplateFile "securitygroup.json" `
9      -securityGroupName $securityGroupName `
10
```

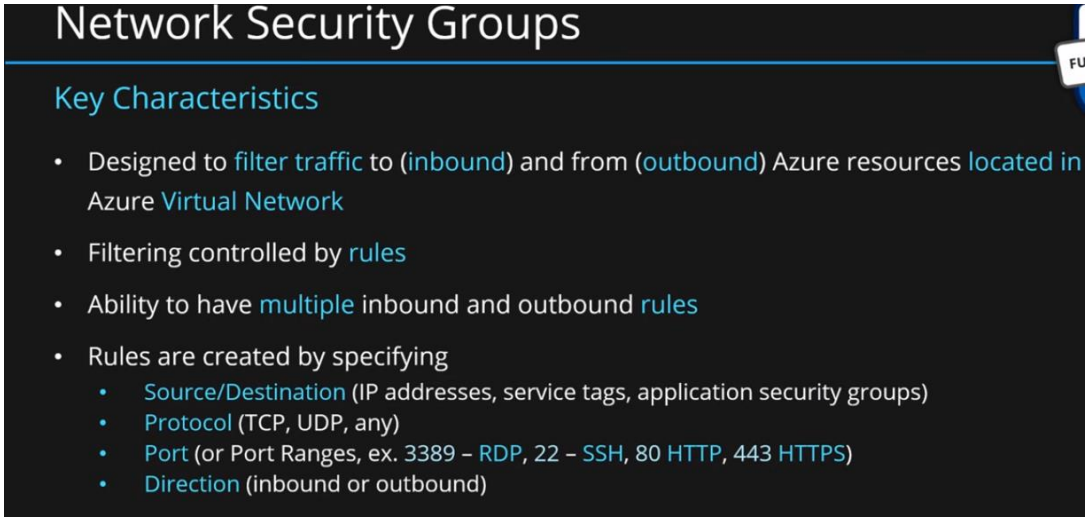
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS D:\study 2021\Great Learning Azure\Assignment\Week1ARM> .\securitygroupdeploy.ps1
Name of the network security group: sathidemosecgrp

Confirm
Provided resource group already exists. Are you sure you want to update it?
[Y] Yes  [N] No  [S] Suspend  [?] Help (default is "Y"): Y


ResourceGroupName : Regroup_5jan
Location          : westus
ProvisioningState : Succeeded
Tags              :
ResourceId        : /subscriptions/f700a502-f4c5-42f4-8f1d-cacab88d7d39/resourceGroups/Regroup_5jan


ResourceGroupName       : Regroup_5jan
OnErrorDeployment       :
DeploymentName          : securitygroup
CorrelationId           : f3941ce5-d720-48b8-a50c-a4ba8aceadc4
ProvisioningState       : Succeeded
```

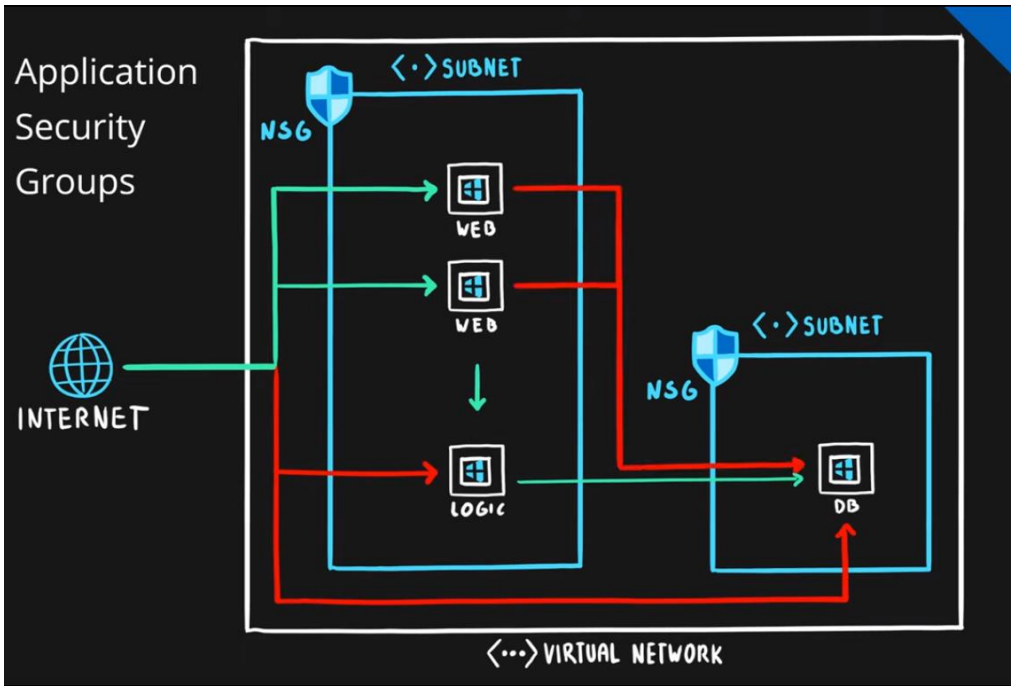


Additional Information

Network Security Groups



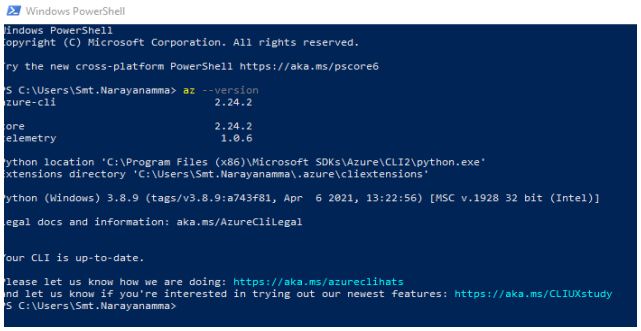
Green – allow red – do not allow



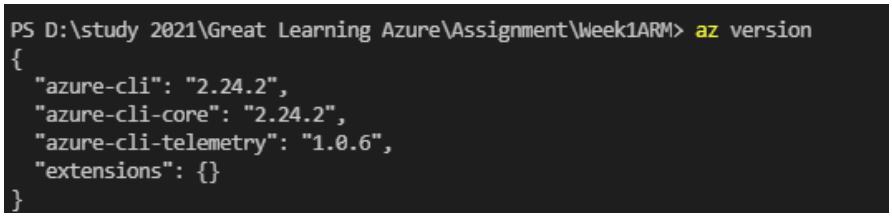
Setup azure CLI

https://docs.microsoft.com/en-us/cli/azure/install-azure-cli-windows?view=azure-cli-latest&tabs=azure-powershell
https://aka.ms/installazurecliwindows
or

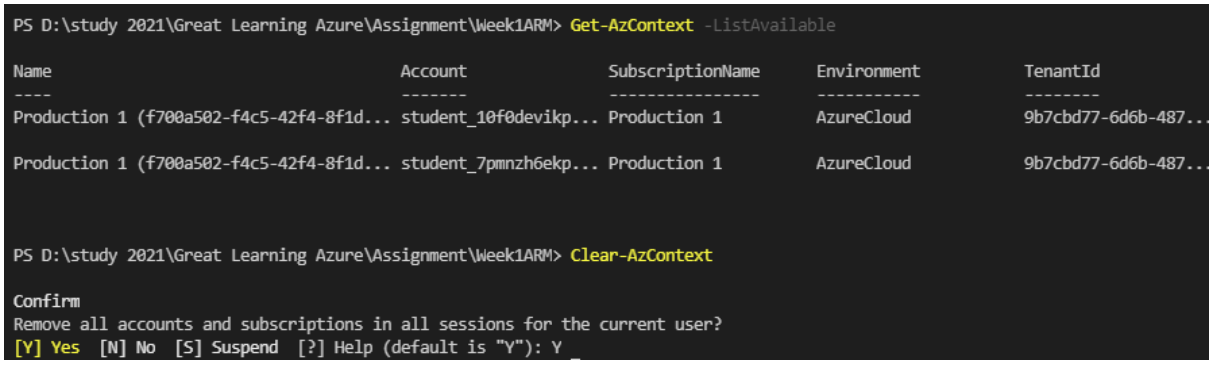
run in powershell
Invoke-WebRequest -Uri https://aka.ms/installazurecliwindows -OutFile .\AzureCLI.msi; Start-Process msiexec.exe -Wait -ArgumentList '/I AzureCLI.msi /quiet'; rm .\AzureCLI.msi
After Installation , Verify version in power shell



And in vs code



List and clear contexts



Refrences :

https://docs.microsoft.com/en-us/azure/azure-resource-manager/templates/deploy-powershell
https://docs.microsoft.com/en-us/azure/virtual-machines/windows/ps-template
https://docs.microsoft.com/en-us/azure/templates/microsoft.network/virtualnetworks?tabs=json
https://docs.microsoft.com/en-us/azure/templates/microsoft.network/networksecuritygroups?tabs=json