PowerShell Script Documentation: Azure Resource Deployment

Overview:

This PowerShell script automates the deployment of Azure resources for a virtual machine (VM) along with associated networking resources. It creates or checks the existence of a resource group, network security group (NSG), storage account, public IP address, virtual network (VNet), subnet, network interface (NIC), and finally provisions a VM.

Prerequisites

- Azure PowerShell module should be installed (`Az` module).

- Azure account with appropriate permissions to create resources.

Script Flow

1. Resource Group Creation/Validation:

   - Checks if the specified resource group exists. If not, creates a new one in the specified location (`$resource_group_Location`).

2. Network Security Group Creation:

   - Defines two inbound security rules for RDP and HTTP traffic.

   - Creates a new NSG (`MYNSG`) in the specified resource group.

3. Storage Account Creation:

   - Creates a new storage account in the specified resource group and location.

4. Public IP Address Creation:

   - Creates a static public IP address in the specified resource group and location.

5. Virtual Network and Subnet Creation:

   - Creates a new virtual network (`MyVNet`) with a subnet (`MySubnet`) in the specified resource group and location.

6. Network Interface Creation:

  - Creates a new network interface (`MyNIC`) associated with the previously created resources (NSG, public IP, subnet).

7. Virtual Machine Deployment:

  - Prompts the user for credentials.

  - Defines parameters for VM creation including resource group, name, location, VNet, subnet, NSG, credentials, size, and image.

  - Creates the VM in the specified resource group.

8. Wait for VM Creation:

  - Monitors the VM creation process until completion.

Code:

```
$resource_group_name = "MyResourceGroup"
$resource_group_Location = "centralus"

# Check if the resource group exists, if not create it
$existingResourceGroup = Get-AzResourceGroup -Name $resource_group_name -
ErrorAction SilentlyContinue
if (-not $existingResourceGroup) {
   Write-Host "Creating resource group $resource_group_name..."
   New-AzResourceGroup -Name $resource_group_name -Location
$resource_group_Location
   # Wait for the resource group creation
   Write-Host "Waiting for the resource group to be created..."
   do {
      $existingResourceGroup = Get-AzResourceGroup -Name $resource_group_name
-ErrorAction SilentlyContinue
      Start-Sleep -Seconds 10
   } while (-not $existingResourceGroup)
}
else {
   Write-Host "Resource group $resource_group_name already exists. Proceeding..."
}

# Create a detailed network security group
```

```
$rule1 = New-AzNetworkSecurityRuleConfig -Name rdp-rule -Description "Allow
RDP" `
   -Access Allow -Protocol Tcp -Direction Inbound -Priority 300 -SourceAddressPrefix `
   Internet -SourcePortRange * -DestinationAddressPrefix * -DestinationPortRange
3389

$rule2 = New-AzNetworkSecurityRuleConfig -Name web-rule -Description "Allow
Http" `
   -Access Allow -Protocol Tcp -Direction Inbound -Priority 400 -SourceAddressPrefix `
   Internet -SourcePortRange * -DestinationAddressPrefix * -DestinationPortRange 80

$NSG = New-AzNetworkSecurityGroup -ResourceGroupName
$resource_group_name -Location $resource_group_Location -Name "MYNSG" -
SecurityRules $rule1,$rule2

Write-Host "Waiting for NSG to be created..."
do {
   $NSG = Get-AzNetworkSecurityGroup -Name 'MYNSG' -ResourceGroupName
$resource_group_name -ErrorAction SilentlyContinue
   Start-Sleep -Seconds 10
} while (-not $NSG)

# Check if NSG was successfully created
if (-not $NSG) {
   Write-Error "Failed to create NSG. Exiting..."
   exit
}

# Create a storage account
$storage_acc_name = "jobansstorageacc"
$storage_acc_location = "centralus"
$storageacc = New-AzStorageAccount -ResourceGroupName
$resource_group_name -Name $storage_acc_name -Location $storage_acc_location
-SkuName "Standard_LRS" -Kind "StorageV2"

Write-Host "Waiting for storage account to be created..."
do {
   $storageacc = Get-AzStorageAccount -ResourceGroupName
$resource_group_name -Name $storage_acc_name -ErrorAction SilentlyContinue
   Start-Sleep -Seconds 10
} while (-not $storageacc)

# Check if storage account was successfully created
if (-not $storageacc) {
   Write-Error "Failed to create storage account. Exiting..."
   exit
}
```

```
# Create a Public IP address
$publicIp = New-AzPublicIpAddress -ResourceGroupName $resource_group_name -
Name "MyPublicIP" -AllocationMethod Static -Location $resource_group_Location

# Wait for Public IP creation
Write-Host "Waiting for Public IP to be created..."
do {
   $publicIp = Get-AzPublicIpAddress -ResourceGroupName $resource_group_name -
Name "MyPublicIP" -ErrorAction SilentlyContinue
   Start-Sleep -Seconds 10
} while (-not $publicIp)

# Check if Public IP was successfully created
if (-not $publicIp) {
   Write-Error "Failed to create Public IP. Exiting..."
   exit
}

# Create a subnet configuration
$subnetConfig = New-AzVirtualNetworkSubnetConfig -Name "MySubnet" -
AddressPrefix "10.0.0.0/24"

# Create a virtual network
$vnet = New-AzVirtualNetwork -ResourceGroupName $resource_group_name -
Location $resource_group_Location -Name "MyVNet" -AddressPrefix "10.0.0.0/16" -
Subnet $subnetConfig

# Wait for Virtual Network creation
Write-Host "Waiting for Virtual Network to be created..."
do {
   $vnet = Get-AzVirtualNetwork -ResourceGroupName $resource_group_name -
Name "MyVNet" -ErrorAction SilentlyContinue
   Start-Sleep -Seconds 10
} while (-not $vnet)

# Check if Virtual Network was successfully created
if (-not $vnet) {
   Write-Error "Failed to create Virtual Network. Exiting..."
   exit
}

# Create a network interface and associate it with NSG, public IP, and subnet
$nic = New-AzNetworkInterface -Name "MyNIC" -ResourceGroupName
$resource_group_name -Location $resource_group_Location -SubnetId
$vnet.Subnets[0].Id -PublicIpAddressId $publicIp.Id -NetworkSecurityGroupId
$NSG.Id
```

```
# Wait for NIC creation
Write-Host "Waiting for NIC to be created..."
do {
  $nic = Get-AzNetworkInterface -Name "MyNIC" -ResourceGroupName
$resource_group_name -ErrorAction SilentlyContinue
  Start-Sleep -Seconds 10
} while (-not $nic)

# Check if NIC was successfully created
if (-not $nic) {
  Write-Error "Failed to create NIC. Exiting..."
  exit
}

# Create the VM configuration
$VM_name = "jobans-vm"
$cred = Get-Credential -Message "Enter a username and password for the virtual
machine."
$VM = New-AzVMConfig -VMName $VM_name -VMSize 'Standard_DS1_v2'
$VM = Set-AzVMOperatingSystem -VM $VM -Windows -ComputerName $VM_name -
Credential $cred -ProvisionVMAgent -EnableAutoUpdate
$VM = Add-AzVMNetworkInterface -VM $VM -Id $nic.Id

# Create the OS disk
$VM = Set-AzVMOSDisk -VM $VM -Name "osdisk1" -CreateOption FromImage -
Windows

# Create the VM
$GETVM=New-AzVM -ResourceGroupName $resource_group_name -Location
$resource_group_Location -VM $VirtualMachine
# Wait for NIC creation
Write-Host "Waiting for VM to be created..."
do {
  $GETVM = Get-AzVM -Name $VM_name -ResourceGroupName
$resource_group_name -ErrorAction SilentlyContinue
  Start-Sleep -Seconds 10
} while (-not $GETVM)

# Check if NIC was successfully created
if (-not $GETVM) {
  Write-Error "Failed to create VM. Exiting..."
  exit
}
Write-Host "All resources created Successfully"
```

Usage

1. Ensure Azure PowerShell module is installed.

2. Copy the script into a PowerShell environment or editor.

3. Update variables `$resource_group_name` and `$resource_group_Location` with desired values.

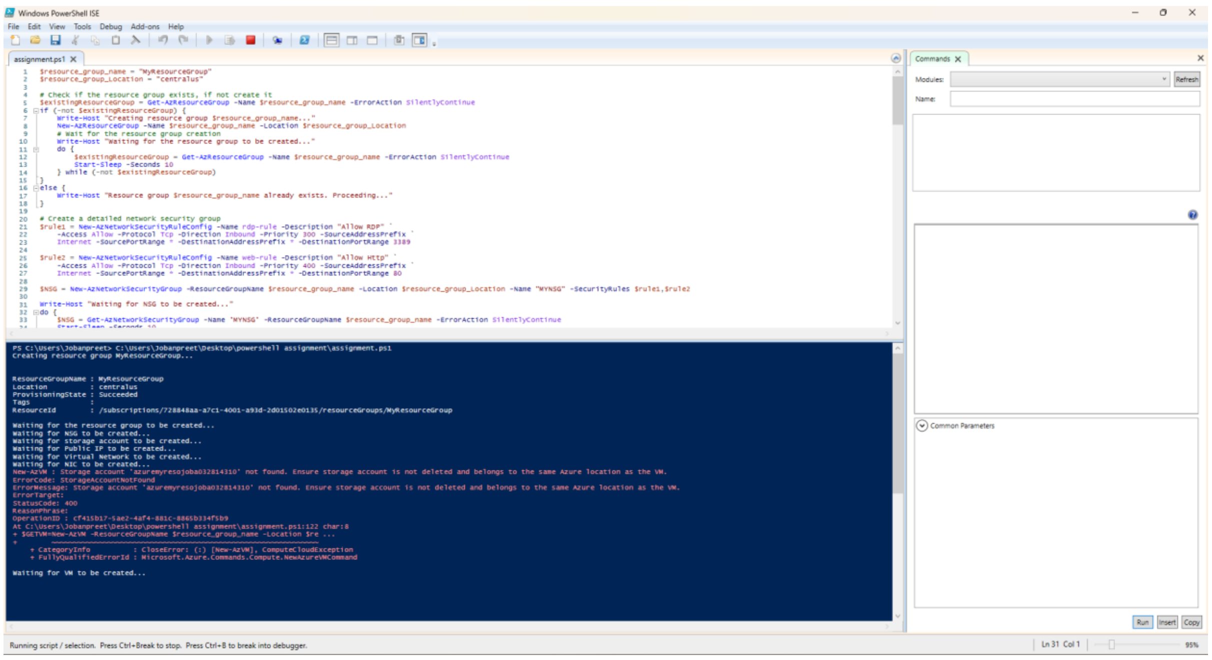
4. Execute the script.

5. Follow the prompts to enter credentials for the VM.

6. Monitor the script execution for any errors or warnings.

Demo:

Run the script

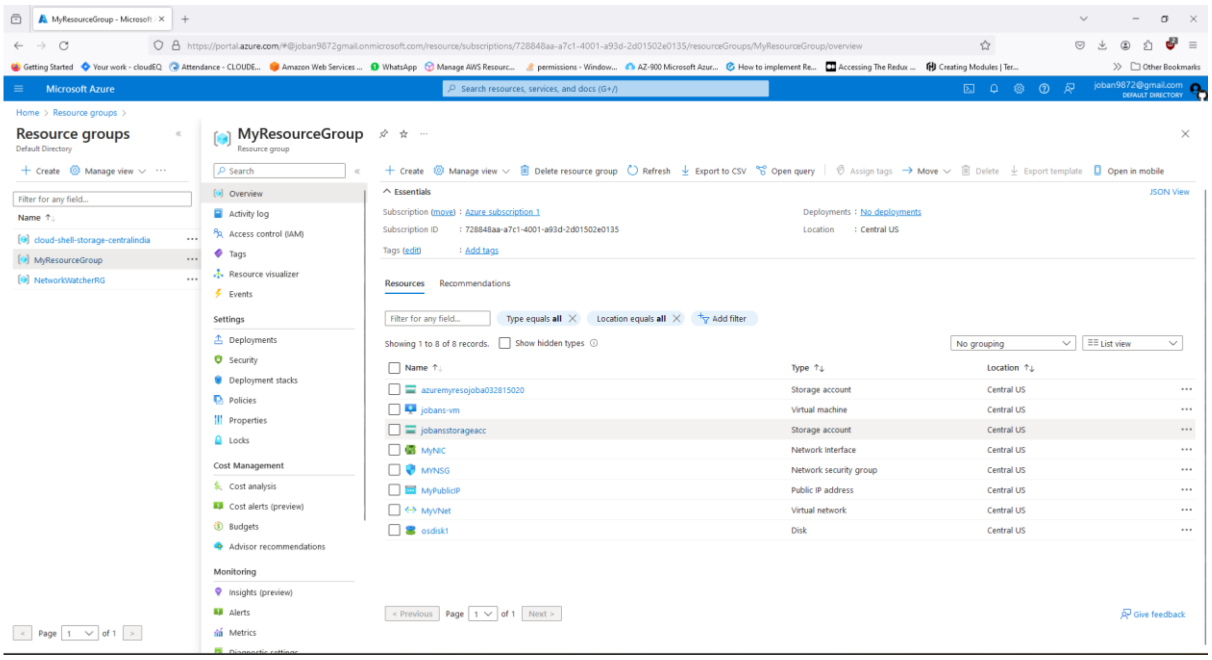


```
$resource_group_name = "MyResourceGroup"
$resource_group_Location = "centralus"

# Check if the resource group exists, if not create it
$existingResourceGroup = Get-AzResourceGroup -Name $resource_group_name -ErrorAction SilentlyContinue
if (-not $existingResourceGroup) {
    Write-Host "Creating resource group $resource_group_name..."
    New-AzResourceGroup -Name $resource_group_name -Location $resource_group_Location
    # Wait for the resource group creation
    Write-Host "Waiting for the resource group to be created..."
    do {
        $existingResourceGroup = Get-AzResourceGroup -Name $resource_group_name -ErrorAction SilentlyContinue
        Start-Sleep -Seconds 10
    } while (-not $existingResourceGroup)
}
else {
    Write-Host "Resource group $resource_group_name already exists. Proceeding..."
}

# Create a detailed network security group
$rule1 = New-AzNetworkSecurityRuleConfig -Name rdp-rule -Description "Allow RDP" `
    -Access Allow -Protocol Tcp -Direction Inbound -Priority 300 -SourceAddressPrefix `
    Internet -SourcePortRange * -DestinationAddressPrefix * -DestinationPortRange 3389

$rule2 = New-AzNetworkSecurityRuleConfig -Name web-rule -Description "Allow Http" `
    -Access Allow -Protocol Tcp -Direction Inbound -Priority 400 -SourceAddressPrefix `
    Internet -SourcePortRange * -DestinationAddressPrefix * -DestinationPortRange 80

$NSG = New-AzNetworkSecurityGroup -ResourceGroupName $resource_group_name -Location $resource_group_Location -Name "MYNSG" -SecurityRules $rule1,$rule2

Write-Host "Waiting for NSG to be created..."
do {
    $NSG = Get-AzNetworkSecurityGroup -Name 'MYNSG' -ResourceGroupName $resource_group_name -ErrorAction SilentlyContinue
```

```
PS C:\Users\Jobanpreet> C:\Users\Jobanpreet\Desktop\powershell assignment\assignment.ps1
Creating resource group MyResourceGroup...


ResourceGroupName : MyResourceGroup
Location          : centralus
ProvisioningState : Succeeded
Tags              :
ResourceId        : /subscriptions/728848aa-a7c1-4001-a93d-2d01502e0135/resourceGroups/MyResourceGroup

Waiting for the resource group to be created...
Waiting for NSG to be created...
Waiting for storage account to be created...
Waiting for Public IP to be created...
Waiting for Virtual Network to be created...
Waiting for NIC to be created...
New-AzVM : Storage account 'azuremyresojoba032814310' not found. Ensure storage account is not deleted and belongs to the same Azure location as the VM.
ErrorCode: StorageAccountNotFound
ErrorMessage: Storage account 'azuremyresojoba032814310' not found. Ensure storage account is not deleted and belongs to the same Azure location as the VM.
ErrorTarget:
StatusCode: 400
ReasonPhrase:
OperationID : cf415b17-5ae2-4af4-881c-8865b334f5b9
At C:\Users\Jobanpreet\Desktop\powershell assignment\assignment.ps1:122 char:8
+ $GETVM=New-AzVM -ResourceGroupName $resource_group_name -Location $re ...
+        ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : CloseError: (:) [New-AzVM], ComputeCloudException
    + FullyQualifiedErrorId : Microsoft.Azure.Commands.Compute.NewAzureVMCommand

Waiting for VM to be created...
```

Running script / selection. Press Ctrl+Break to stop. Press Ctrl+B to break into debugger.

Verify that resources have been created



MyResourceGroup
Resource group

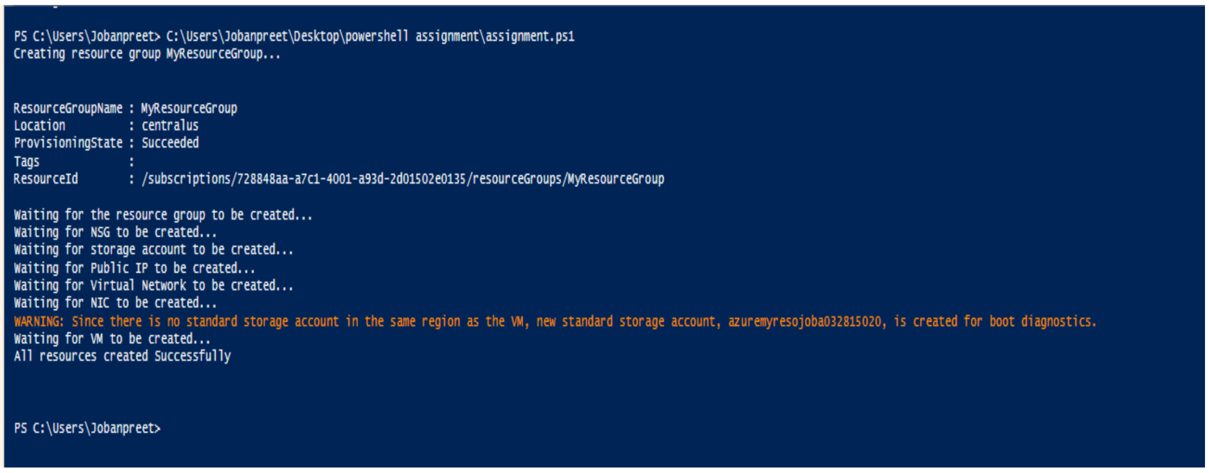
Essentials

Subscription (move) : Azure subscription 1
Subscription ID : 728848aa-a7c1-4001-a93d-2d01502e0135
Tags (edit) : Add tags
Deployments : No deployments
Location : Central US

Showing 1 to 8 of 8 records.

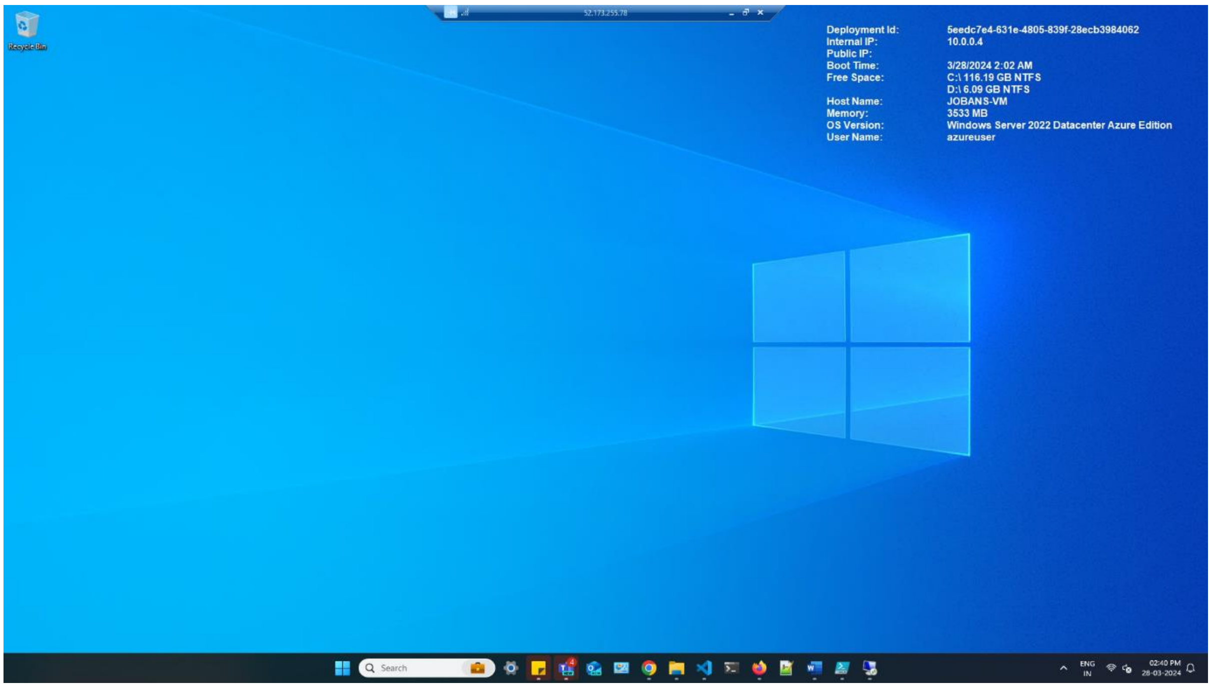
Name	Type	Location
azuremyresojoba032815020	Storage account	Central US
jobans-vm	Virtual machine	Central US
jobansstorageacc	Storage account	Central US
MyNIC	Network Interface	Central US
MYNSG	Network security group	Central US
MyPublicIP	Public IP address	Central US
MyVNet	Virtual network	Central US
osdisk1	Disk	Central US

Check the result



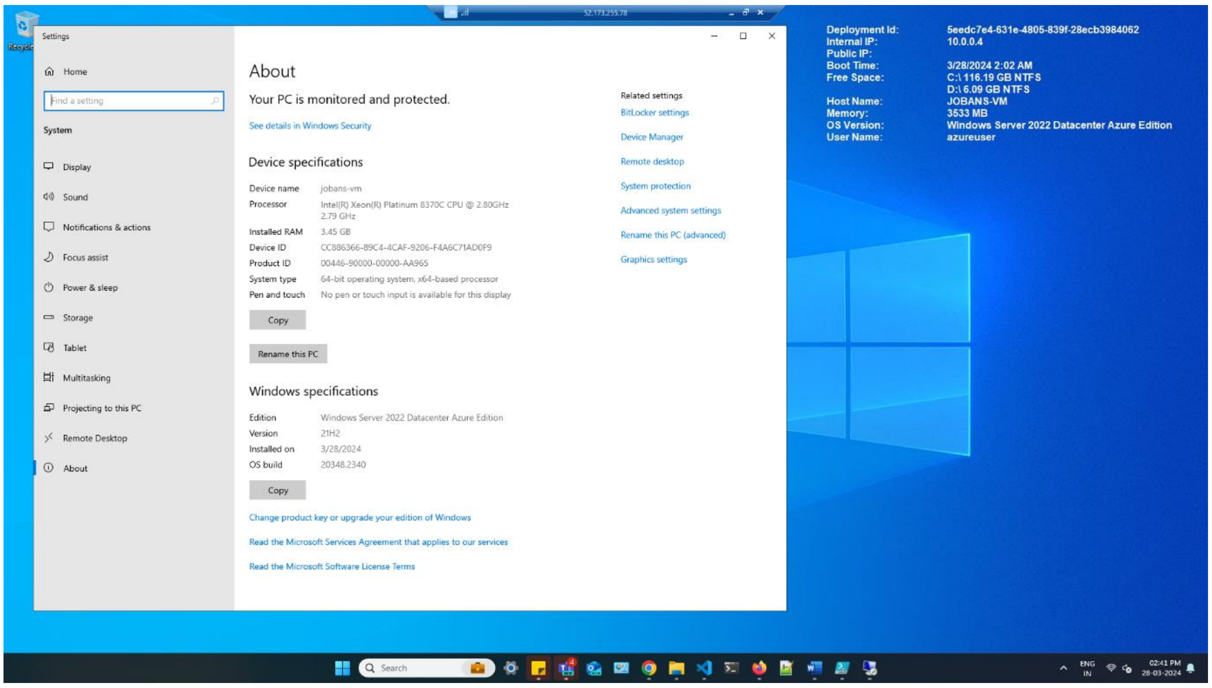
```
PS C:\Users\Jobanpreet> C:\Users\Jobanpreet\Desktop\powershell assignment\assignment.ps1
Creating resource group MyResourceGroup...


ResourceGroupName : MyResourceGroup
Location          : centralus
ProvisioningState : Succeeded
Tags              :
ResourceId        : /subscriptions/728848aa-a7c1-4001-a93d-2d01502e0135/resourceGroups/MyResourceGroup

Waiting for the resource group to be created...
Waiting for NSG to be created...
Waiting for storage account to be created...
Waiting for Public IP to be created...
Waiting for Virtual Network to be created...
Waiting for NIC to be created...
WARNING: Since there is no standard storage account in the same region as the VM, new standard storage account, azuremyresojoba032815020, is created for boot diagnostics.
Waiting for VM to be created...
All resources created Successfully


PS C:\Users\Jobanpreet>
```

Connect to the vm using rdp



Deployment Id:	5eedc7e4-631e-4805-839f-28ecb3984062
Internal IP:	10.0.0.4
Public IP:	
Boot Time:	3/28/2024 2:02 AM
Free Space:	C:\ 116.19 GB NTFS
	D:\ 6.09 GB NTFS
Host Name:	JOBANS-VM
Memory:	3533 MB
OS Version:	Windows Server 2022 Datacenter Azure Edition
User Name:	azureuser

check the details of vm



Deployment Id:	5eedc7e4-631e-4805-839f-28ecb3984062
Internal IP:	10.0.0.4
Public IP:	
Boot Time:	3/28/2024 2:02 AM
Free Space:	C:\ 116.19 GB NTFS
	D:\ 6.09 GB NTFS
Host Name:	JOBANS-VM
Memory:	3533 MB
OS Version:	Windows Server 2022 Datacenter Azure Edition
User Name:	azureuser

About

Your PC is monitored and protected.

Device specifications

Device name	jobans-vm
Processor	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz 2.79 GHz
Installed RAM	3.45 GB
Device ID	CC886366-89C4-4CAF-9206-F4A6C71AD0F9
Product ID	00446-90000-00000-AA965
System type	64-bit operating system, x64-based processor
Pen and touch	No pen or touch input is available for this display

Windows specifications

Edition	Windows Server 2022 Datacenter Azure Edition
Version	21H2
Installed on	3/28/2024
OS build	20348.2340

Notes

- The script contains wait loops to ensure the completion of resource creation operations.

- Error handling is implemented to exit the script in case of failures during resource creation.

- Users may customize the script by modifying variables or adding additional resource creation steps as needed.