**Vnet peering**

#Variables

$rgname1 = "EUS-RG"

$rgname2 = "SEA-RG"

$vnet1 = "vnet-eus"

$vnet2 = "vnet-SEA"

#Getting Vnet info

$seavnet= Get-AzVirtualNetwork -name $vnet1 -ResourceGroupName $rgname1

$euvnet = Get-AzVirtualNetwork -name $vnet2 -ResourceGroupName $rgname2

Add-AzVirtualNetworkPeering `

-Name vnetsea-vneteus `

-VirtualNetwork $seavnet `

-RemoteVirtualNetworkId $euvnet.Id

Add-AzVirtualNetworkPeering `

-Name vneteus-vnetsea `

-VirtualNetworkn $euvnet `

-RemoteVirtualNetworkId $seavnet.Id

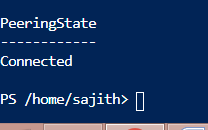
Get-AzVirtualNetworkPeering `

-ResourceGroupName $rgname1 `

-VirtualNetworkName $seavnet.Name `

| Select PeeringState

Output:



**2. backup1**

$rgname = "SEA-RG"

$loc = "southeastasia"

$VMname = "HA-webserver1"

$RSV = "SEA-webservers-backup"

Register-AzResourceProvider -ProviderNamespace "Microsoft.RecoveryServices"

New-AzRecoveryServicesVault `

-ResourceGroupName $rgname `

-Name $RSV `

-Location $loc

Get-AzRecoveryServicesVault `

-Name $RSV | Set-AzRecoveryServicesVaultContext

Get-AzRecoveryServicesVault `

-Name $RSV | Set-AzRecoveryServicesBackupProperty -BackupStorageRedundancy LocallyRedundant

$policy = Get-AzRecoveryServicesBackupProtectionPolicy -Name "DefaultPolicy"

Enable-AzRecoveryServicesBackupProtection `

-ResourceGroupName $rgname `

-Name $VMname `

-Policy $policy

$backupcontainer = Get-AzRecoveryServicesBackupContainer `

-ContainerType "AzureVM" `

-FriendlyName $VMname

$item = Get-AzRecoveryServicesBackupItem `

-Container $backupcontainer `

-WorkloadType "AzureVM"

Backup-AzRecoveryServicesBackupItem -Item $item

Get-AzRecoveryservicesBackupJob

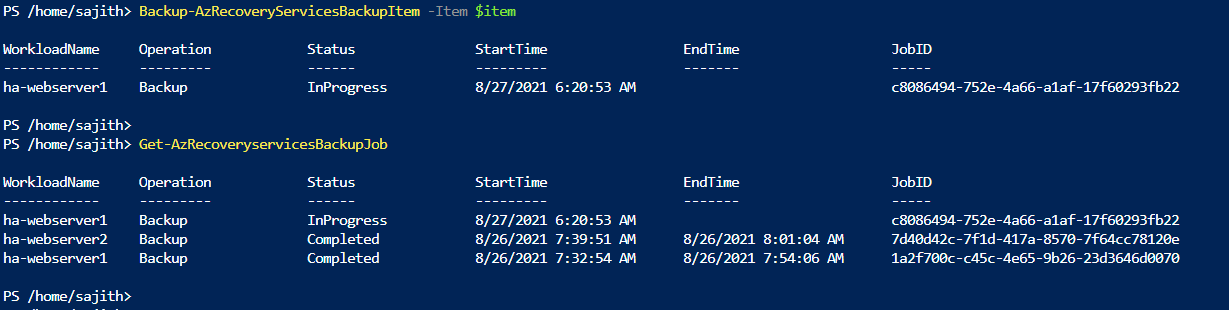
##Disable-AzRecoveryServicesBackupProtection -Item $item -RemoveRecoveryPoints

#$vault = Get-AzRecoveryServicesVault -Name $RSV

#Remove-AzRecoveryServicesVault -Vault $vault

#Remove-AzResourceGroup -Name $rgname

Output:



**Backup2:**

$rgname = "SEA-RG"

$loc = "southeastasia"

$VMname = "HA-webserver2"

$RSV = "sea-vmbackupvault"

Register-AzResourceProvider -ProviderNamespace "Microsoft.RecoveryServices"

New-AzRecoveryServicesVault `

-ResourceGroupName $rgname `

-Name $RSV `

-Location $loc

Get-AzRecoveryServicesVault `

-Name $RSV | Set-AzRecoveryServicesVaultContext

Get-AzRecoveryServicesVault `

-Name $RSV | Set-AzRecoveryServicesBackupProperty -BackupStorageRedundancy LocallyRedundant

$policy = Get-AzRecoveryServicesBackupProtectionPolicy -Name "DefaultPolicy"

Enable-AzRecoveryServicesBackupProtection `

-ResourceGroupName $rgname `

-Name $VMname `

-Policy $policy

$backupcontainer = Get-AzRecoveryServicesBackupContainer `

-ContainerType "AzureVM" `

-FriendlyName $VMname

$item = Get-AzRecoveryServicesBackupItem `

-Container $backupcontainer `

-WorkloadType "AzureVM"

Backup-AzRecoveryServicesBackupItem -Item $item

Get-AzRecoveryservicesBackupJob

##Disable-AzRecoveryServicesBackupProtection -Item $item -RemoveRecoveryPoints

#$vault = Get-AzRecoveryServicesVault -Name $RSV

#Remove-AzRecoveryServicesVault -Vault $vault

#Remove-AzResourceGroup -Name $rgname

**Storage accounts**

#**EUS Storage account using ZRS**

$rgname = "EUS-RG"

$loc = "eastus2"

$sname = "nileusstorage"

$storageacc1 = New-AzStorageAccount -ResourceGroupName $rgname `

-Name $sname `

-Location $loc `

-SkuName Standard\_ZRS `

-Kind StorageV2 `

-EnableHttpsTrafficOnly $true `

-AllowSharedKeyAccess $true `

**#SEA Storage account using GRS**

$rgname1 = "SEA-RG"

$loc1 = "southeastasia"

$sname1 = "seastorageact"

$storageacc2 = New-AzStorageAccount -ResourceGroupName $rgname1 `

-Name $sname1 `

-Location $loc1 `

-SkuName Standard\_GRS `

-Kind StorageV2

**#Get Access Key**

$storageKey = Get-AzStorageAccountKey -ResourceGroupName $rgname -Name $sname

$storageKey

$sKey = (Get-AzstorageAccountKey -ResourceGroupName $rgname -Name $sname | select -first 1).Value

**#Creating File Share**

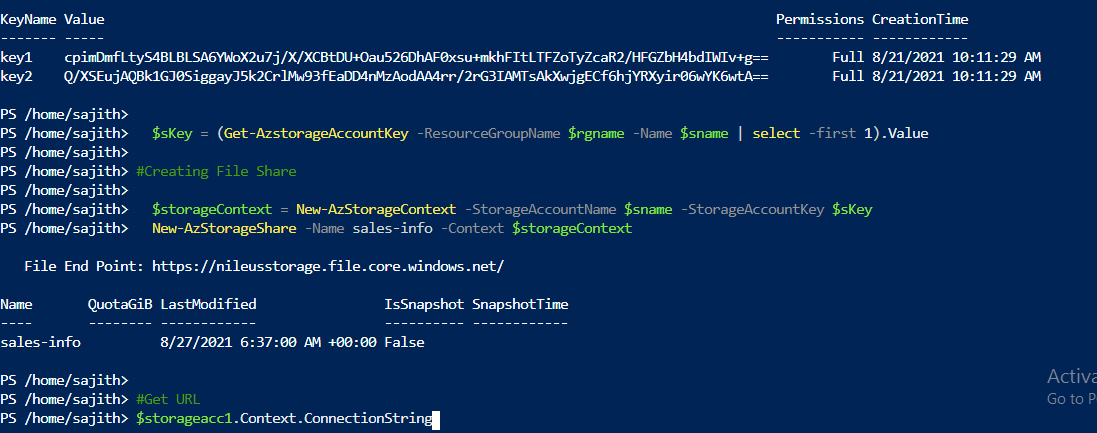
$storageContext = New-AzStorageContext -StorageAccountName $sname -StorageAccountKey $sKey

New-AzStorageShare -Name sales-info -Context $storageContext

#Get URL

$storageacc1.Context.ConnectionString

Output:



**EUvnet:**

#Variables

$rgname = "EUS-RG"

$loc = "eastus2"

$vnet = "vnet-eus"

$subnet1 = "EUSsubnet"

$nsgname1 = "EUS-WEB-NSG"

#Resource Group Location

New-AzResourceGroup $rgname $loc

#Virtual Network Creation

$vnetwork = @{

Name = $vnet

ResourceGroupName = $rgname

Location = $loc

AddressPrefix = '10.1.0.0/16'

}

$virtualNetwork = New-AzVirtualNetwork @vnetwork

**#NSG Creation**

$rule1 = New-AzNetworkSecurityRuleConfig -Name allow-access -Description "Allow Access" `

-Access Allow -Protocol Tcp -Direction Inbound -Priority 1000 -SourceAddressPrefix `

Internet -SourcePortRange \* -DestinationAddressPrefix \* -DestinationPortRange 22,80,443

$eusnsg = New-AzNetworkSecurityGroup -ResourceGroupName $rgname -Location $loc -Name `

$nsgname1 -SecurityRules $rule1

#Subnet creation

$subnet = @{

Name = $subnet1

VirtualNetwork = $virtualNetwork

AddressPrefix = '10.1.1.0/24'

NetworkSecurityGroup = $eusnsg

}

Add-AzVirtualNetworkSubnetConfig @subnet

$virtualNetwork | Set-AzVirtualNetwork

**LB**

# Variables for common values

$rgname = "SEA-RG"

$location = "southeastasia"

$lbname = "sea-lb1"

$lbpip = "sea-lb-pip"

$vnetName = "vnet-SEA"

$vmname1 = "HA-webserver1"

$vmname2 = "HA-webserver2"

$bpname = "bp1"

$hpname = "Hpb1"

$feipname = "fip"

$lbrulename = "Loadbalancingrule1"

$vm1 = Get-AzResource -ResourceGroupName $rgname -Name $vmname1

$vm2 = Get-AzResource -ResourceGroupName $rgname -Name $vmname2

# Creating public IP address

$publicIpLB = New-AzPublicIpAddress -ResourceGroupName $rgName -Name $lbpip -Location $location -AllocationMethod "Static" -Sku 'Standard'

# Creating back-end & front-end address pools.

$bppool1 = New-AzLoadBalancerBackendAddressPoolConfig -Name $bpname

$frontendip = New-AzLoadBalancerFrontendIpConfig -Name $feipname -PublicIpAddress $publicIpLB

# Creating health probe on port 80.

$probe = New-AzLoadBalancerProbeConfig -Name $hpname -Protocol Http -Port 80 `

-RequestPath / -IntervalInSeconds 360 -ProbeCount 5

# Creating load balancing rules.

$lbrule1 = New-AzLoadBalancerRuleConfig -Name $lbrulename -Protocol Tcp -DisableOutboundSNAT `

-Probe $probe -FrontendPort 80 -BackendPort 80 `

-FrontendIpConfiguration $frontendip -BackendAddressPool $bppool1 -LoadDistribution "SourceIP"

#$bp2 = New-AzLoadBalancerRuleConfig -Name 'lb-bp2' -Protocol Tcp `

# -Probe $probe -FrontendPort 80 -BackendPort 22 `

# -FrontendIpConfiguration $frontendip -BackendAddressPool $bppool2

# Creating the load balancer.

$lb = New-AzLoadBalancer -ResourceGroupName $rgName -Sku "Standard" -Name $lbname -Location $location `

-FrontendIpConfiguration $frontendip -BackendAddressPool $bppool1 `

-Probe $probe -LoadBalancingRule $lbrule1

#Adding VNet and Required VM's to the backend pool

$virtualNetwork = Get-AzVirtualNetwork -Name $vnetName -ResourceGroupName $rgname

#Getting IP:

$nic1 = Get-AzNetworkInterface -Name "$VMname1-nic" -ResourceGroupName $rgname

$nic2 = Get-AzNetworkInterface -Name "$VMname2-nic" -ResourceGroupName $rgname

$nicip1 =Get-AzNetworkInterfaceIpConfig -Name ipconfig1 -NetworkInterface $nic1

$nicip2 =Get-AzNetworkInterfaceIpConfig -Name ipconfig1 -NetworkInterface $nic2

$ip1 = New-AzLoadBalancerBackendAddressConfig -IpAddress $nicip1.PrivateIpAddress -Name "lp-bpaddress1" -VirtualNetworkId $virtualNetwork.Id

$ip2= New-AzLoadBalancerBackendAddressConfig -IpAddress $nicip2.PrivateIpAddress -Name "lp-bpaddress2" -VirtualNetworkId $virtualNetwork.Id

$ips = @($ip1, $ip2)

$b2 = Get-AzLoadBalancerBackendAddressPool -ResourceGroupName $rgname -LoadBalancerName $lbname -Name $bpname

Set-AzLoadBalancerBackendAddressPool -ResourceGroupName $rgname -LoadBalancerName $lb.Name -Name $b2.Name -LoadBalancerBackendAddress $ips

#Adding INbound NAT rules for RDP to the backend servers

#$rgname = "SEA-RG"

#$location = "southeastasia"

#$lbname = "sea-lb1"

#$lb1 = Get-AzLoadBalancer -ResourceGroupName $rgname -Name $lbname

#$lb1 | Add-AzLoadBalancerInboundNatRuleConfig -Name "RDPRule1" -FrontendIPConfiguration $lb1.FrontendIpConfigurations[0] -Protocol "SSH" -FrontendPort 3000 -BackendPort 22

#$lb1 | Add-AzLoadBalancerInboundNatRuleConfig -Name "RDPRule2" -FrontendIPConfiguration $lb1.FrontendIpConfigurations[0] -Protocol "SSH" -FrontendPort 3001 -BackendPort 22

#$natnic1 = Get-AzLoadBalancerInboundNatRuleConfig -LoadBalancer $lb1 -Name "RDPRule1"

#$natnic2 = Get-AzLoadBalancerInboundNatRuleConfig -LoadBalancer $lb1 -Name "RDPRule2"

#Set-AzNetworkInterfaceIpConfig -Name $nic1.IpConfigurations[0].Name -LoadBalancerInboundNatRule $natnic1.inboundnatrules.id -NetworkInterface $nic1

#Set-AzNetworkInterfaceIpConfig -Name $nic1.IpConfigurations[0].Name -LoadBalancerInboundNatRule $natnic2.inboundnatrules.id -NetworkInterface $nic2

**Create Vmadmin user who can manage all VM in the subscription**

Connect-AzureAD

$PasswordProfile = New-Object -TypeName Microsoft.Open.AzureAD.Model.PasswordProfile

$PasswordProfile.Password = "Xendesktop@123"

## For managing VMs in Subscription

New-AzureADUser -DisplayName "VMadmin" `

-PasswordProfile $PasswordProfile `

-UserPrincipalName "vmadmin@bsajith47outlook.onmicrosoft.com" `

-AccountEnabled $true `

-MailNickName "vmadmin"

New-AzRoleAssignment -SignInName "vmadmin@bsajith47outlook.onmicrosoft.com" `

-RoleDefinitionName "Virtual Machine Contributor" `

-Scope "/subscriptions/c16b15e2-e9f9-4269-9770-ed6e1be548a1/"

# Creating User Account

Connect-AzureAD

$PasswordProfile = New-Object -TypeName Microsoft.Open.AzureAD.Model.PasswordProfile

$PasswordProfile.Password = "Xendesktop@123"

## For managing VMs in Subscription

New-AzureADUser -DisplayName "Vmadmin" `

-PasswordProfile $PasswordProfile `

-UserPrincipalName "vmadmin@bsajith47outlook.onmicrosoft.com" `

-AccountEnabled $true `

-MailNickName "vmadmin"

New-AzRoleAssignment -SignInName "vmadmin@bsajith47outlook.onmicrosoft.com" `

-RoleDefinitionName "Virtual Machine Contributor" `

-Scope "/subscriptions/c16b15e2-e9f9-4269-9770-ed6e1be548a1/"

## For Managing backups in EUS Region

New-AzureADUser -DisplayName "Backup" `

-PasswordProfile $PasswordProfile `

-UserPrincipalName "backupadmin@bsajith47outlook.onmicrosoft.com" `

-AccountEnabled $true `

-MailNickName "backup"

New-AzRoleAssignment -SignInName "backupadmin@bsajith47outlook.onmicrosoft.com" `

-RoleDefinitionName "Backup Contributor" `

-Scope "/subscriptions/c16b15e2-e9f9-4269-9770-ed6e1be548a1/resourceGroups/EUS-RG/providers/Microsoft.Compute/virtualMachines/server11"