Instructions

Lab 05 - Implement Intersite Connectivity Student lab manual

Lab scenario

Contoso has its datacenters in Boston, New York, and Seattle offices connected via a mesh wide-area network links, with full connectivity between them. You need to implement a lab environment that will reflect the topology of the Contoso's on-premises networks and verify its functionality.

Objectives

In this lab, you will:

- Task 1: Provision the lab environment
- Task 2: Configure local and global virtual network peering
- Task 3: Test intersite connectivity

Estimated timing: 30 minutes

Instructions

Task 1: Provision the lab environment

In this task, you will deploy three virtual machines, each into a separate virtual network, with two of them in the same Azure region and the third one in another Azure region.

1. Sign in to the Azure portal.
2. In the Azure portal, open the **Azure Cloud Shell** by clicking on the icon in the top right of the Azure Portal.
3. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.

   > **Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

4. In the toolbar of the Cloud Shell pane, click the **Upload/Download files** icon, in the drop-down menu, click **Upload** and upload the files **\Allfiles\Labs\05\az104-05-vnetvm-loop-template.json** and **\Allfiles\Labs\05\az104-05-vnetvm-loop-parameters.json** into the Cloud Shell home directory.
5. From the Cloud Shell pane, run the following to create the resource group that will be hosting the lab environment. The first two virtual networks and a pair of virtual machines will be deployed in `[Azure_region_1]`. The third virtual network and the third virtual machine will be deployed in the same resource group but another `[Azure_region_2]`. (replace the `[Azure_region_1]` and `[Azure_region_2]` placeholder with the names of two different Azure regions where you intend to deploy these Azure virtual machines):

   Code | Copy

   ```
   $location1 = '[Azure_region_1]'

   $location2 = '[Azure_region_2]'

   $rgName = 'az104-05-rg1'

   New-AzResourceGroup -Name $rgName -Location $location1
   ```

> **Note**: In order to identify Azure regions, from a PowerShell session in Cloud Shell, run **(Get-AzLocation).Location**

6. From the Cloud Shell pane, run the following to create the three virtual networks and deploy virtual machines into them by using the template and parameter files you uploaded:

```
New-AzResourceGroupDeployment `
   -ResourceGroupName $rgName `
   -TemplateFile $HOME/az104-05-vnetvm-loop-template.json `
   -TemplateParameterFile $HOME/az104-05-vnetvm-loop-parameters.json `
   -location1 $location1 `
   -location2 $location2 `
   -AsJob
```

> **Note**: Wait for the deployment to complete before proceeding to the next task. This should take about 2 minutes.

> **Note**: To verify the status of the deployment, you can examine the properties of the resource group you created in this task.

7. Close the Cloud Shell pane.

Task 2: Configure local and global virtual network peering

In this task, you will configure local and global peering between the virtual networks you deployed in the previous tasks.

1. In the Azure portal, search for and select **Virtual networks**.
2. Review the virtual networks you created in the previous task and verify that the first two are located in the same Azure region and the third one in a different Azure region.

> **Note**: The template you used for deployment of the three virtual networks ensures that the IP address ranges of the three virtual networks do not overlap.

3. In the list of virtual networks, click **az104-05-vnet0**.
4. On the **az104-05-vnet0** virtual network blade, in the **Settings** section, click **Peerings** and then click **+ Add**.
5. Add a peering with the following settings (leave others with their default values) and click **Add**:

Setting	Value
This virtual network: Peering link name	**az104-05-vnet0_to_az104-05-vnet1**
This virtual network: Traffic to remote virtual network	**Allow (default)**
This virtual network: Traffic forwarded from remote virtual network	**Block traffic that originates from outside this virtual network**
Virtual network gateway	**None**
Remote virtual network: Peering link name	**az104-05-vnet1_to_az104-05-vnet0**
Virtual network deployment model	**Resource manager**
I know my resource ID	unselected

Setting	Value
Subscription	the name of the Azure subscription you are using in this lab
Virtual network	**az104-05-vnet1**
Traffic to remote virtual network	**Allow (default)**
Traffic forwarded from remote virtual network	**Block traffic that originates from outside this virtual network**
Virtual network gateway	**None**

> **Note**: This step establishes two local peerings - one from az104-05-vnet0 to az104-05-vnet1 and the other from az104-05-vnet1 to az104-05-vnet0.

6. On the **az104-05-vnet0** virtual network blade, in the **Settings** section, click **Peerings** and then click **+ Add**.
7. Add a peering with the following settings (leave others with their default values) and click **Add**:

Setting	Value
This virtual network: Peering link name	**az104-05-vnet0_to_az104-05-vnet2**
This virtual network: Traffic to remote virtual network	**Allow (default)**
This virtual network: Traffic forwarded from remote virtual network	**Block traffic that originates from outside this virtual network**
Virtual network gateway	**None**
Remote virtual network: Peering link name	**az104-05-vnet2_to_az104-05-vnet0**
Virtual network deployment model	**Resource manager**
I know my resource ID	unselected
Subscription	the name of the Azure subscription you are using in this lab
Virtual network	**az104-05-vnet2**
Traffic to remote virtual network	**Allow (default)**
Traffic forwarded from remote virtual network	**Block traffic that originates from outside this virtual network**
Virtual network gateway	**None**

> **Note**: This step establishes two global peerings - one from az104-05-vnet0 to az104-05-vnet2 and the other from az104-05-vnet2 to az104-05-vnet0.

8. Navigate back to the **Virtual networks** blade and, in the list of virtual networks, click **az104-05-vnet1**.
9. On the **az104-05-vnet1** virtual network blade, in the **Settings** section, click **Peerings** and then click **+ Add**.
10. Add a peering with the following settings (leave others with their default values) and click **Add**:

Setting	Value
This virtual network: Peering link name	**az104-05-vnet1_to_az104-05-vnet2**

Setting	Value
This virtual network: Traffic to remote virtual network	**Allow (default)**
This virtual network: Traffic forwarded from remote virtual network	**Block traffic that originates from outside this virtual network**
Virtual network gateway	**None**
Remote virtual network: Peering link name	**az104-05-vnet2_to_az104-05-vnet1**
Virtual network deployment model	**Resource manager**
I know my resource ID	unselected
Subscription	the name of the Azure subscription you are using in this lab
Virtual network	**az104-05-vnet2**
Traffic to remote virtual network	**Allow (default)**
Traffic forwarded from remote virtual network	**Block traffic that originates from outside this virtual network**
Virtual network gateway	**None**

> **Note**: This step establishes two global peerings - one from az104-05-vnet1 to az104-05-vnet2 and the other from az104-05-vnet2 to az104-05-vnet1.

Task 3: Test intersite connectivity

In this task, you will test connectivity between virtual machines on the three virtual networks that you connected via local and global peering in the previous task.

1. In the Azure portal, search for and select **Virtual machines**.
2. In the list of virtual machines, click **az104-05-vm0**.
3. On the **az104-05-vm0** blade, click **Connect**, in the drop-down menu, click **RDP**, on the **Connect with RDP** blade, click **Download RDP File** and follow the prompts to start the Remote Desktop session.

   > **Note**: This step refers to connecting via Remote Desktop from a Windows computer. On a Mac, you can use Remote Desktop Client from the Mac App Store and on Linux computers you can use an open source RDP client software.

   > **Note**: You can ignore any warning prompts when connecting to the target virtual machines.

4. When prompted, sign in by using the **Student** username and **Pa55w.rd1234** password.
5. Within the Remote Desktop session to **az104-05-vm0**, right-click the **Start** button and, in the right-click menu, click **Windows PowerShell (Admin)**.
6. In the Windows PowerShell console window, run the following to test connectivity to **az104-05-vm1** (which has the private IP address of **10.51.0.4**) over TCP port 3389:

   Code Copy

   ```
   Test-NetConnection -ComputerName 10.51.0.4 -Port 3389 -InformationLevel 'Detailed'
   ```

> **Note**: The test uses TCP 3389 since this is this port is allowed by default by operating system firewall.

7. Examine the output of the command and verify that the connection was successful.
8. In the Windows PowerShell console window, run the following to test connectivity to **az104-05-vm2** (which has the private IP address of **10.52.0.4**):

```
Test-NetConnection -ComputerName 10.52.0.4 -Port 3389 -InformationLevel 'Detailed'
```

9. Switch back to the Azure portal on your lab computer and navigate back to the **Virtual machines** blade.
10. In the list of virtual machines, click **az104-05-vm1**.
11. On the **az104-05-vm1** blade, click **Connect**, in the drop-down menu, click **RDP**, on the **Connect with RDP** blade, click **Download RDP File** and follow the prompts to start the Remote Desktop session.

> **Note**: This step refers to connecting via Remote Desktop from a Windows computer. On a Mac, you can use Remote Desktop Client from the Mac App Store and on Linux computers you can use an open source RDP client software.

> **Note**: You can ignore any warning prompts when connecting to the target virtual machines.

12. When prompted, sign in by using the **Student** username and **Pa55w.rd1234** password.
13. Within the Remote Desktop session to **az104-05-vm1**, right-click the **Start** button and, in the right-click menu, click **Windows PowerShell (Admin)**.
14. In the Windows PowerShell console window, run the following to test connectivity to **az104-05-vm2** (which has the private IP address of **10.52.0.4**) over TCP port 3389:

```
Test-NetConnection -ComputerName 10.52.0.4 -Port 3389 -InformationLevel 'Detailed'
```

> **Note**: The test uses TCP 3389 since this is this port is allowed by default by operating system firewall.

15. Examine the output of the command and verify that the connection was successful.

Clean up resources

> **Note**: Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not see unexpected charges.

1. In the Azure portal, open the **PowerShell** session within the **Cloud Shell** pane.
2. List all resource groups created throughout the labs of this module by running the following command:

```
Get-AzResourceGroup -Name 'az104-05*'
```

3. Delete all resource groups you created throughout the labs of this module by running the following command:

```
Get-AzResourceGroup -Name 'az104-05*' | Remove-AzResourceGroup -Force -AsJob
```

> **Note**: The command executes asynchronously (as determined by the -AsJob parameter), so while you will be able to run another PowerShell command immediately afterwards within the same PowerShell session, it will take a few minutes before the resource groups are actually removed.

Review

In this lab, you have:

- Provisioned the lab environment
- Configured local and global virtual network peering
- Tested intersite connectivity