User Defined Routes

This guide demonstrates how User Defined Routes (UDR) can be used to control the routing of packets through a virtual appliance. In this demonstration you will

* Show how a User Defined Route is configured in a virtual network
* Observe how a UDR routes traffic (the next hop) for packets flowing to a specific subnet to go to through a virtual appliance.

Contents

[Pre-Requisites 1](#_Toc444703514)

[Setup 1](#_Toc444703515)

[Demo Steps 2](#_Toc444703516)

[Clean Up 5](#_Toc444703517)

## Pre-Requisites

This section lists the pre-requisites required for this demonstration.

* Azure subscription
* Azure PowerShell 1.2 or newer ([download](http://aka.ms/webpi-azps))
* PowerShell Remote Execution Policy is RemoteSigned (if not already)  
  Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Force
* (Optional) Visual Studio 2015 w/Azure Tools SDK 2.8.2 or newer

## Setup

Estimated time: 20 mins (mostly waiting for ARM template to deploy)

Note: You can deploy the ARM template using Visual Studio 2015. However, there is a bug in Visual Studio at this time that *intermittently* results in an error (authentication token expiration) when deploying from Visual Studio. The deployment will still work but Visual Studio will report an error. This is expected to be fixed by summer of 2016.

|  |  |
| --- | --- |
| 1. Open Windows Explorer and navigate to **.\Networking\Demos\3\_UDR\UDR-Demo-Solution**. 2. Right-click on **PS-Deploy.ps1** and select edit. It should open PowerShell ISE. If it does not, then manually open PowerShell ISE and then open PS-Deploy.ps1 in the editor. |  |
| 1. Update variables to your preferences:    1. Set the **$subscriptionId** to your Azure Subscription Id. You can get this from the Azure Portal in the Subscriptions blade.    2. Set the **$resourceGrpLoc** to the location/region you will be doing the demo in (ie: West US, East US, North Europe, etc.)    3. Optionally, change the **$resourceGrpName**. You may want to do this if you are creating multiple deployments. 2. Click the green play button in the toolbar (or press **F5**).   The deployment will take about 20 mins to complete. Wait for it to finish before continuing.  When the deployment is complete you will see output as shown here with the **ProvisioningState** of **Succeeded**. |  |
| 1. Sign-in/Connect to **vm-1** with the following credentials:    1. Username: **adminuser**    2. Password: **P@ssword1** 2. Open Windows File Explorer and navigate to **C:\Install**. |  |
| 1. Double-click on **WinPcap\_4\_1\_3.exe** to start the WinPcap installation. Note: It is not possible to automate the installation of WinPcap. WinPcap installs the drivers used by WireShark to monitor network traffic. Hence the reason for this manual step.    1. Accept the defaults for all the dialogs to complete the installation. 2. Close Windows File Explorer.   Keep your RDP connection open to this virtual machine. You will come back to it in the demo. |  |
| 1. Sign-in/Connect to **vm-0** with the following credentials:    1. Username: **adminuser**    2. Password: **P@ssword1**   Keep your RDP connection open to this virtual machine. You will come back to it in the demo. |  |
| 1. Sign-in/Connect to **vm-2** with the following credentials:    1. Username: **adminuser**    2. Password: **P@ssword1**   Keep your RDP connection open to this virtual machine. You will come back to it in the demo. |  |

## Demo Steps

Estimated time: 10 mins

|  |  |
| --- | --- |
| 1. Sign-in to the Azure Portal. 2. Show the **Resources** blade for the UDR-Demo-Solution resource group. Identify the resources as follows:    1. The 3 virtual machines.    2. The 3 virtual NIC’s that bind each of the virtual machines to the virtual network.    3. The 3 public IP addresses which are there only so we can RDP into the VM’s    4. The virtual network.    5. The user defined route / route-table. |  |
| 1. Show the 3 **subnets** and their address range in the virtual network. |  |
| 1. Show the 3 virtual NIC’s in the **virtual network** blade. Explain that these are the internal IP addresses assigned to the NIC’s and therefore the virtual machine they are bound to. |  |
| 1. Show that the **FrontendSubnet** has a Route Table assigned to it while the other two subnets do not. We’ll see the route table details shortly. |  |
| 1. Show the **Route Table blade** for the UDR. Point out the one route defined is associated with the **FrontendSubnet** and routes traffic addressed to the **BackendSubnet** to a device (our VM appliance) in the **NVASubnet**. 2. Click on the **VirtualApplianceRoutToSubnet3** to view the details of the route. |  |
| 1. In the Edit route blade, show the **Next hop type** is set to Virtual Appliance. Also, how the drop down list and discuss other options that are available. |  |
| 1. Open the RDP connection to **vm-1** (the VM Appliance). 2. Double-click on Wireshark on the desktop to monitoring traffic. |  |
| 1. Double-click on the Ethernet graph as shown here. |  |
| 1. In the filter box under the toolbar, set the filter to **ip.addr==10.1.2.4**. There won’t be any traffic shown until you invoke some traffic, which will be done shortly. Explain that what we’re doing here is listening for any traffic that is addressed to/from **vm-2** that is in the BackendSubnet. |  |
| 1. Open the RDP connection to **vm-0**. 2. Open a **Command Prompt**. 3. Type **tracert 10.1.2.4** and press **ENTER**. Point out that the trace to 10.1.2.4 went through 10.1.1.4 (the VM Appliance). 4. Type ping **10.1.2.4** and press **ENTER**. |  |
| 1. Open the RDP connection to **vm-1**. 2. Show that Wireshark is receiving the traffic between 10.1.0.4 (vm-0) and 10.1.2.4 (vm-2).   Explain that for this to work this virtual machine (vm-1) had to have ipForwarding enabled and have the Routing and Remote Access feature installed and configured for Lan Routing. Everything else was done in the IaaS environment by configuring the route table with a next hop to a Virtual Appliance, which you showed earlier in the Azure portal. |  |

## Clean Up

To clean up this environment perform the following steps:

1. Go to the Azure portal and delete the following resource groups:
   1. UDR-Demo-Solution
   2. UDR-Demo-Solution-Deploy