How to Configure Azure Route Tables (UDR) using PowerShell and ARM

https://campus.barracuda.com/doc/48203337/

Azure Route Tables, or User Defined Routing, allow you to create network routes so that your F-Series Firewall VM can handle the traffic both between your subnets and to the Internet. For the network interfaces to be allowed to receive and forward traffic, IP forwarding must be enabled. When different route types are present in a UDR route table, user defined routes are preferred over the default system routes. When multiple routes match the destination, the more specific route is used. The default system routes always present in an Azure route table allow the following:

- Traffic within the virtual network
- Traffic to the Internet
- Traffic between different virtual networks using the Azure VPN gateway
- Traffic from the virtual network to networks connected via the Azure VPN gateway

Limitations

- Multiple network interfaces are not supported for High availability clusters
- Multiple network interfaces in one subnet are not supported for stand-alone firewall VMs.

Example deployment script

You can combine the PowerShell commandlets to create an Azure route table and then assign it to your backend subnet(s). See below for an example script. This script assumes that you already have a running F-Series Firewall deployment and are logged in to your Azure account.

Modify this script to create the routes or route table as required.

```
########################################## # Azure Route Table Settings
$routeTableName = 'NAME_ROUTE_TABLE' $location = 'West Europe' # Name of
resource group the virtual network is in. The route table will also be
created in this resource group. $ResourceGroupName = 'RESOURCE_GROUP_NAME'
$VNETName = 'NAME_OF_VIRTUAL_NETWORK' # Subnet name # add additional subnets
if needed $SubnetName = 'NAME_OF_SUBNET' $SubnetAddressPrefix = 'X.X.X.X/X' #
Create the route table. Add the routes separated by a comma to the -Route
option $routeTable = New-AzureRmRouteTable -ResourceGroupName
$ResourceGroupName -Location $location -Name $routeTableName # Create routes.
Add additional routes as needed Add-AzureRmRouteConfig -Name 'DefaultRoute' -
AddressPrefix 0.0.0.0/0 -RouteTable $routeTable -NextHopType VirtualAppliance
-NextHopIpAddress 10.8.2.10 Add-AzureRmRouteConfig -Name 'NGCCVIPNetwork' -
```

```
AddressPrefix 10.8.100.0/24 -RouteTable $routeTable -NextHopType
VirtualAppliance -NextHopIpAddress 10.8.10.10 # Assign to subnets $vnet =
Get-AzureRmVirtualNetwork -ResourceGroupName $ResourceGroupName -Name
$VNETName # Assign the route table a subnet. Repeat for each backend subnet
$newsubnet = Set-AzureRmVirtualNetworkSubnetConfig -VirtualNetwork $vnet -
Name $SubnetName -AddressPrefix $SubnetAddressPrefix -RouteTable $routeTable
Set-AzureRmVirtualNetwork -VirtualNetwork $newsubnet
```

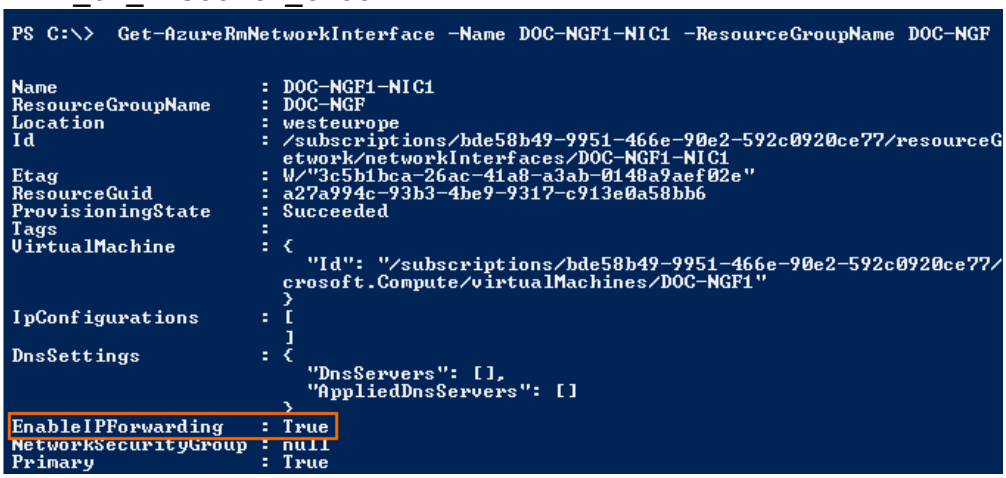
Before you begin

- Deploy a Barracuda NextGen Firewall F-Series. For more information, see How to Deploy an F-Series Firewall in Microsoft Azure using PowerShell and ARM or How to Deploy an F-Series Firewall in Microsoft Azure using Azure Portal and ARM.
- Install Azure PowerShell version 1.1.0 or higher.
- Log in to your Azure Account with `Login-AzureRmAccount`.

Step 1. Verify that IP forwarding is enabled for each network interface

To forward traffic, you must enable IP forwarding for each network interface on the NextGen Firewall F-Series VM.

1. Open Azure PowerShell.
2. Verify IP forwarding is enabled (true) for each network interface:
   `Get-AzureRmNetworkInterface -Name NAME_OF_NIC -ResourceGroup NAME_OF_RESOURCE_GROUP`



3. To enable **IP forwarding** for the primary network interface, enter:
   `$nic = Get-AzureRmNetworkInterface -Name NAME_OF_NIC -ResourceGroup NAME_OF_RESOURCE_GROUP $nic.EnableIPForwarding = 1 Set-AzureRmNetworkInterface -NetworkInterface $nic`

4. (optional) If you are using more than one network interface, repeat for the other NICs.

Your NextGen Firewall F-Series VM is now allowed to forward IP packets with a different destination address as the IP address of the VM. See the troubleshooting section below on how to check if IP forwarding is enabled for your interfaces.

Step 2. Create routes and an Azure route table

Create a routing table in Azure and apply it the backend subnets of the VNET. Add a user defined route to the routing table to change the default route for all VMs in the backend subnets to the NextGen Firewall F-Series VM. The routing table can be applied to multiple backend subnets.

1. Open Azure PowerShell.
2. Create the Azure route table.
   ```
   $routeTable = New-AzureRmRouteTable -ResourceGroupName
   YOUR_RESOURCE_GROUP_NAME -Location YOUR_LOCATION -Name ROUTE_TABLE_NAME
   ```
   
   ```
   PS C:\>
   PS C:\> $routeTable = New-AzureRmRouteTable -Name "DOCRoutetable" -ResourceGroupName DOC-Networking -Location "West Euro
   pe"
   PS C:\>
   ```
3. Create routes and add them to the route table:
   ```
   Add-AzureRmRouteConfig -Name NAME_OF_FIRST_ROUTE -AddressPrefix
   X.X.X.X/X -RouteTable $routeTable -NextHopType VirtualAppliance -
   NextHopIpAddress PRIVATE_IP_VM Add-AzureRmRouteConfig -Name
   NAME_OF_SECOND_ROUTE -AddressPrefix X.X.X.X/X -RouteTable $routeTable -
   NextHopType VirtualAppliance -NextHopIpAddress PRIVATE_IP_VM
   ```

```
PS C:\Users\mzoller\Desktop> Add-AzureRmRouteConfig -Name NGFDefaultRoute -RouteTable $routeTable -AddressPrefix 0.0.0.0
/0 -NextHopType VirtualAppliance -NextHopIpAddress 10.8.1.10


Name                 : RouteTable2
ResourceGroupName    : DOC-Networking
Location             : westeurope
Id                   : /subscriptions/bde58b49-9951-466e-90e2-592c0920ce77/resourceGroups/DOC-Networking/providers/Microso
                       ft.Network/routeTables/RouteTable2
Etag                 : W/"c89ed840-7af7-40fa-a575-f346878e43a4"
ProvisioningState    : Succeeded
Tags                 :
Routes               : [
                         {
                           "Name": "NGFDefaultRoute",
                           "AddressPrefix": "0.0.0.0/0",
                           "NextHopType": "VirtualAppliance",
                           "NextHopIpAddress": "10.8.1.10"
                         }
                       ]
Subnets              : []


PS C:\Users\mzoller\Desktop> Add-AzureRmRouteConfig -Name NGCCVIPNetwork -RouteTable $routeTable -AddressPrefix 10.8.100
.0/24 -NextHopType VirtualAppliance -NextHopIpAddress 10.8.10.10


Name                 : RouteTable2
ResourceGroupName    : DOC-Networking
Location             : westeurope
Id                   : /subscriptions/bde58b49-9951-466e-90e2-592c0920ce77/resourceGroups/DOC-Networking/providers/Microso
                       ft.Network/routeTables/RouteTable2
Etag                 : W/"c89ed840-7af7-40fa-a575-f346878e43a4"
ProvisioningState    : Succeeded
Tags                 :
Routes               : [
                         {
                           "Name": "NGFDefaultRoute",
                           "AddressPrefix": "0.0.0.0/0",
                           "NextHopType": "VirtualAppliance",
                           "NextHopIpAddress": "10.8.1.10"
                         },
                         {
                           "Name": "NGCCVIPNetwork",
                           "AddressPrefix": "10.8.100.0/24",
                           "NextHopType": "VirtualAppliance",
                           "NextHopIpAddress": "10.8.10.10"
                         }
                       ]
Subnets              : []


PS C:\Users\mzoller\Desktop>
```

Step 3. Associate the route table with subnets

Create the Azure route table and add the routes created in step 1. The route table is then applied to the backend subnets of your virtual network.

1. Open Azure PowerShell.
2. Store the virtual network in a variable:
   `$vnet = Get-AzureRmVirtualNetwork -ResourceGroupName VNET_RESOURCE_GROUP_NAME -Name VIRTUAL_NETWORK_NAME`

   ```
   PS C:\> $vnet = Get-AzureRmVirtualNetwork -ResourceName DOC-VNET -ResourceGroupName DOC-Networking
   PS C:\>
   ```
3. For each subnet, set the route table in the subnet configuration:
   `$newsubnet = Set-AzureRmVirtualNetworkSubnetConfig -VirtualNetwork $vnet -Name NAME_BACKEND_SUBNET -AddressPrefix X.X.X.X/X -RouteTable $routeTable Set-AzureRmVirtualNetwork -VirtualNetwork $newsubnet`

All traffic from the backend subnets is now routed through the NextGen Firewall F-Series VM. Propagating the routing table changes to the VMs in the subnets can take a couple of minutes. See the Troubleshooting section below on how to query Azure for the actual (effective) routing table used by the VM.

Next steps

- Create access rule to allow traffic from the backend VMs to the Internet. For more information, see Access Rules.
- Configure UDR route rewriting and IP forward protection to display the Azure route table in NextGen Admin. For more information, see How to Configure Azure Cloud Integration using ARM.

Figures

1. UDR_PS_01.png
2. UDR_PS_03.png
3. UDR_PS_02.png
4. UDR_PS_04.png