Cloud NGFW for Azure

Deployment guide

Version: 1.0
Authors: Ravisankar Pegada

Table of Contents

About This Guide

Cloud NGFW is the industry's only machine learning (ML)-powered NGFW delivered as a cloud-native service on Azure. With Cloud NGFW, you can run more apps securely at cloud speed and cloud-scale with an actual cloud-native experience. There is no trade-off between cloud agility and sophisticated, multi-layered security. You get to experience the best of both worlds with natively integrated network security delivered as a service on Azure.

This guide explains how to configure and integrate Cloud NGFW into Azure Virtual Network(VNet) and Azure Virtual WAN(VWAN) infrastructure, enabling the users to utilize the benefits of Palo Alto Networks next-generation firewall as a service. The sections in the document provide details about the architecture and various components of this service. This document also provides guidance on how to set up and configure Cloud NGFW using a simplified configuration workflow and explains how to route your application/spoke traffic through Cloud NGFW.

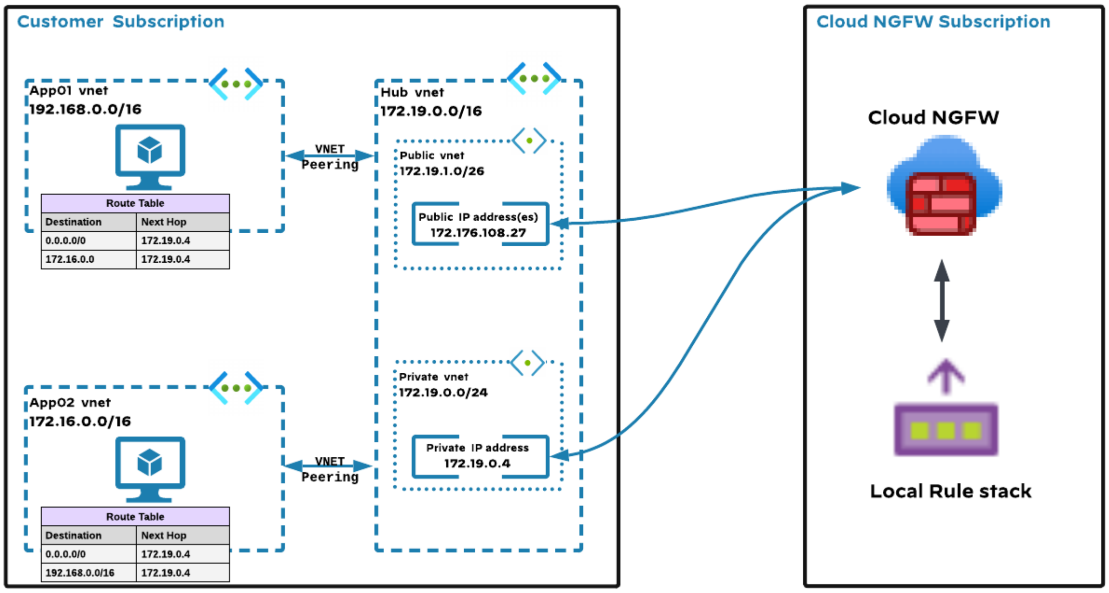
Integrate Cloud NGFW into Azure Virtual Network(VNet) Infrastructure

Pre-deployment of Cloud NGFW - setting up the VNet environment

Topology

A hub-spoke topology is used as an example to route traffic through Cloud NGFW. Cloud NGFW supports all topologies.

Integrate Cloud NGFW into Azure Virtual Network(VNet)



Route Table

Destination	Next Hop
0.0.0.0/0	172.19.0.4
172.16.0.0	172.19.0.4

Route Table

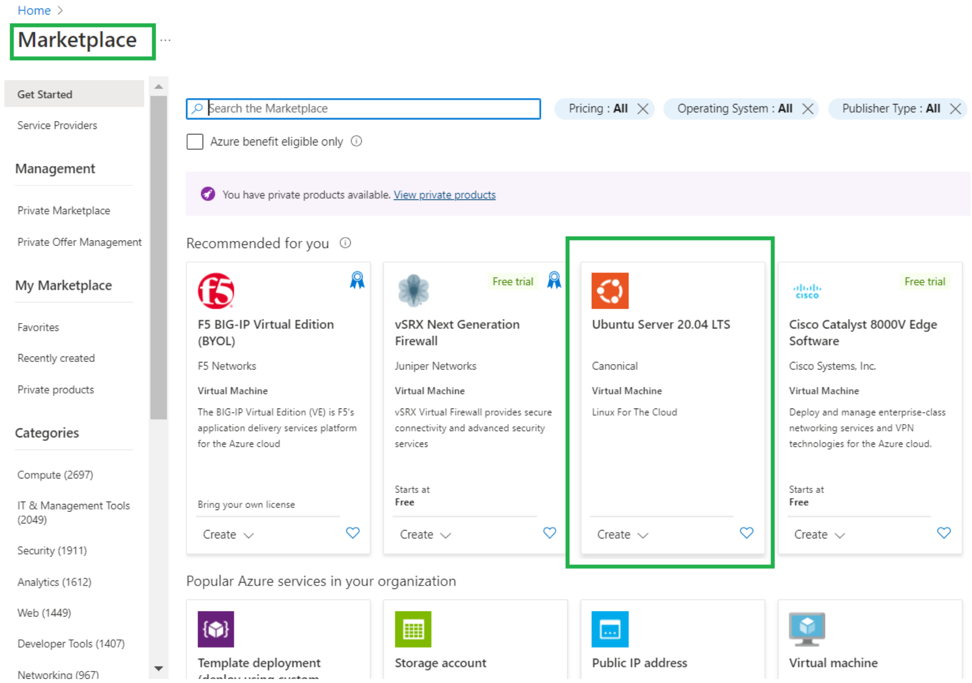
Destination	Next Hop
0.0.0.0/0	172.19.0.4
192.168.0.0/16	172.19.0.4

As per the above topology, in order to set up a lab environment, there should be a hub VNet, 2 spoke VNets and a virtual machine on one of those spoke VNets that's running a web server (apache2). Create this environment before creating and deploying the Cloud NGFW resources.

Create Spoke VNets with a virtual machine on each of them

Create Spoke App01 VNet with Ubuntu Server

1. Go to Azure Marketplace and search for "*Ubuntu*" Server as shown below:



Home >

Marketplace ...

Get Started
Service Providers
Management
Private Marketplace
Private Offer Management
My Marketplace
Favorites
Recently created
Private products
Categories
Compute (2697)
IT & Management Tools (2049)
Security (1911)
Analytics (1612)
Web (1449)
Developer Tools (1407)
Networking (967)

Search the Marketplace | Pricing : All ✕ | Operating System : All ✕ | Publisher Type : All ✕

□ Azure benefit eligible only

You have private products available. View private products

Recommended for you

F5 BIG-IP Virtual Edition (BYOL)
F5 Networks
Virtual Machine
The BIG-IP Virtual Edition (VE) is F5's application delivery services platform for the Azure cloud
Bring your own license
Create

vSRX Next Generation Firewall
Free trial
Juniper Networks
Virtual Machine
vSRX Virtual Firewall provides secure connectivity and advanced security services
Starts at
Free
Create

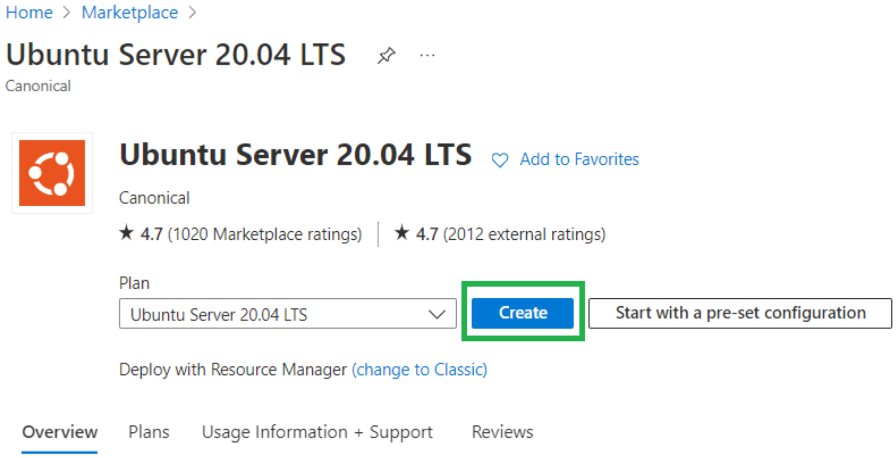
Ubuntu Server 20.04 LTS
Canonical
Virtual Machine
Linux For The Cloud
Create

Cisco Catalyst 8000V Edge Software
Free trial
Cisco Systems, Inc.
Virtual Machine
Deploy and manage enterprise-class networking services and VPN technologies for the Azure cloud.
Starts at
Free
Create

Popular Azure services in your organization

Template deployment
Storage account
Public IP address
Virtual machine

2. Choose this Ubuntu server and click 'Create' to start the creation of the Ubuntu server:



Home > Marketplace >

Ubuntu Server 20.04 LTS

Canonical

Ubuntu Server 20.04 LTS ♡ Add to Favorites

Canonical

★ 4.7 (1020 Marketplace ratings) | ★ 4.7 (2012 external ratings)

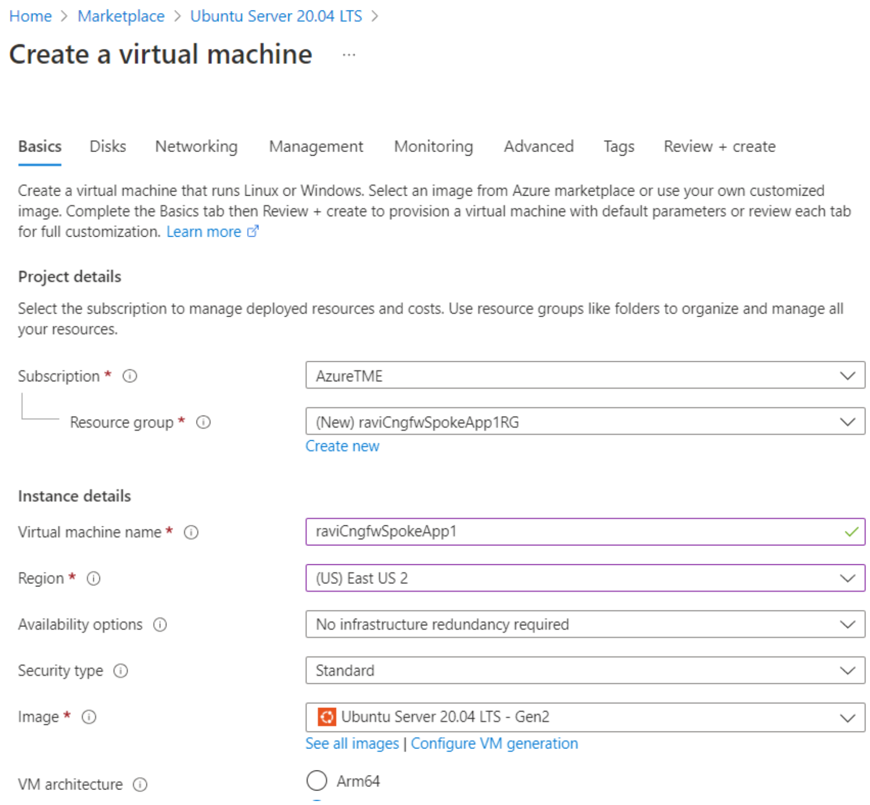
Plan

Ubuntu Server 20.04 LTS | Create | Start with a pre-set configuration

Deploy with Resource Manager (change to Classic)

Overview | Plans | Usage Information + Support | Reviews

3. Fill in the details (Resource Group, VM Name, Region and the type of image while leaving other fields to default.) to complete the creation of the Ubuntu server. Make sure to choose the region appropriately based on Cloud NGFW service.



Home > Marketplace > Ubuntu Server 20.04 LTS >

Create a virtual machine

Basics Disks Networking Management Monitoring Advanced Tags Review + create

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. Learn more

Project details

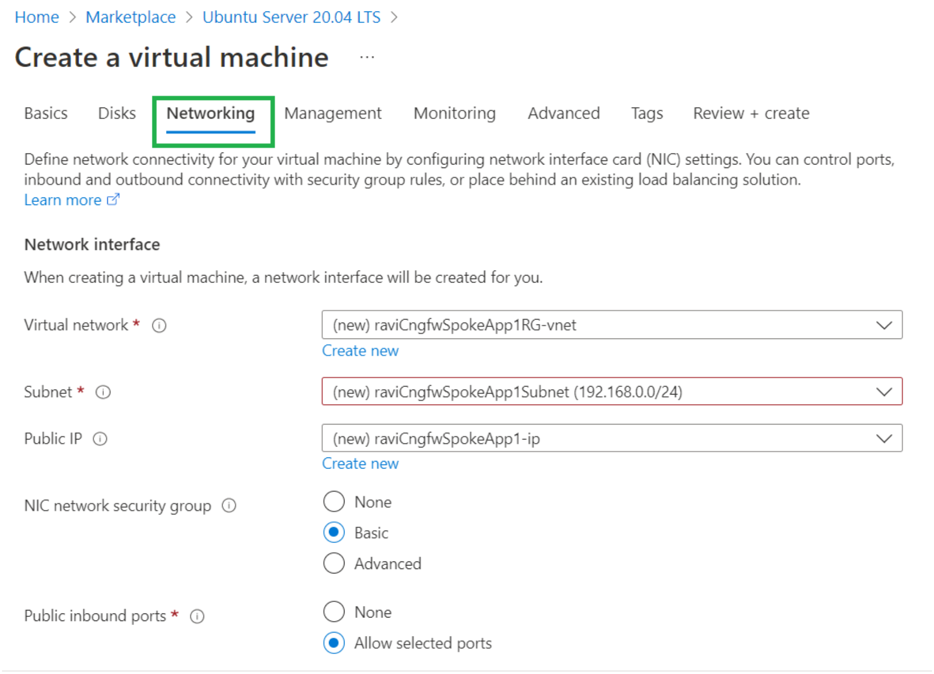
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	AzureTME
Resource group *	(New) raviCngfwSpokeApp1RG Create new

Instance details

Virtual machine name *	raviCngfwSpokeApp1
Region *	(US) East US 2
Availability options	No infrastructure redundancy required
Security type	Standard
Image *	Ubuntu Server 20.04 LTS - Gen2 See all images | Configure VM generation
VM architecture	○ Arm64

4. In the networking section, select an existing VNet or create a new one in which this Ubuntu server will be installed:



Create a virtual machine ...

Define network connectivity for your virtual machine by configuring network interface card (NIC) settings. You can control ports, inbound and outbound connectivity with security group rules, or place behind an existing load balancing solution.

Network interface

When creating a virtual machine, a network interface will be created for you.

Virtual network * : (new) raviCngfwSpokeApp1RG-vnet

Subnet * : (new) raviCngfwSpokeApp1Subnet (192.168.0.0/24)

Public IP : (new) raviCngfwSpokeApp1-ip

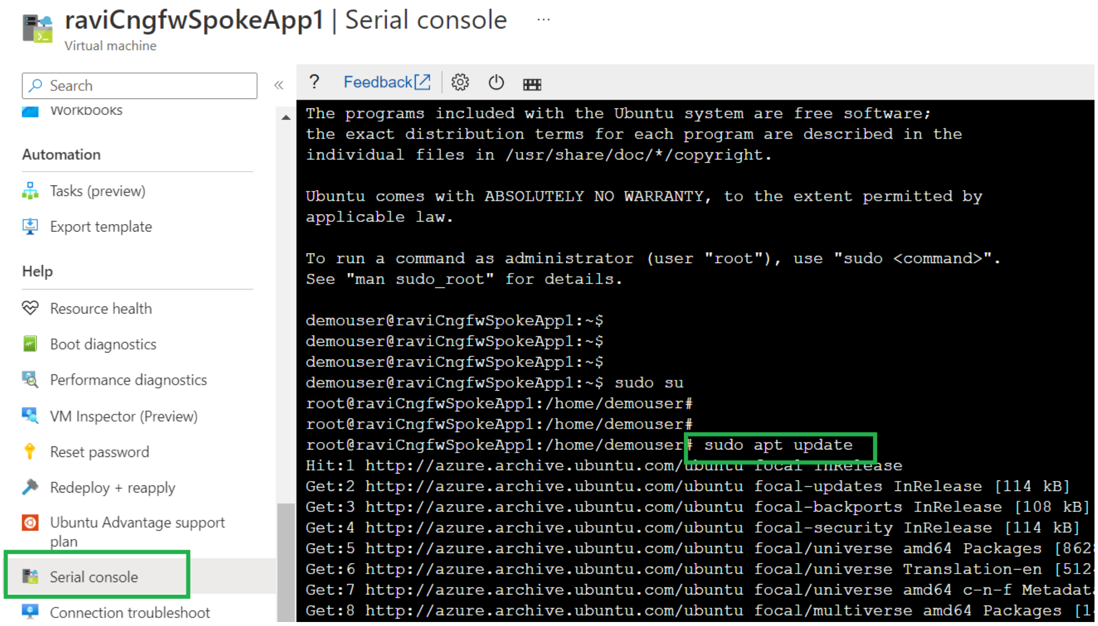
NIC network security group : ○ None ● Basic ○ Advanced

Public inbound ports * : ○ None ● Allow selected ports

5. Review the configuration and create the server.

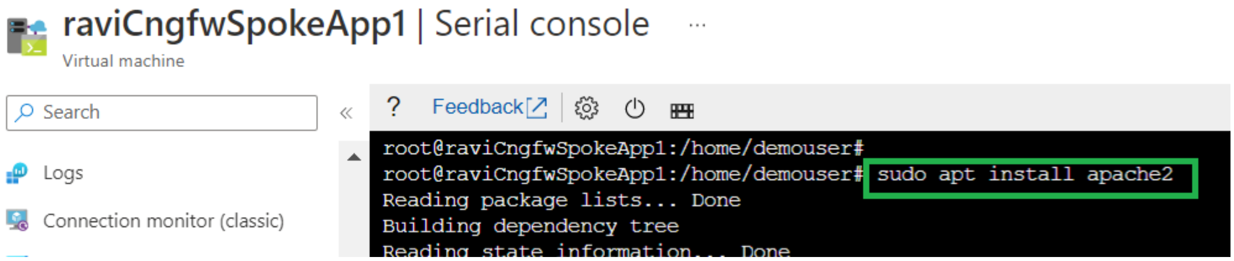
6. Once the Ubuntu server deployment is complete, install an apache server on it. To do this, go to the serial console of the created Ubuntu server and execute the commands below to install an apache server:

   sudo apt update



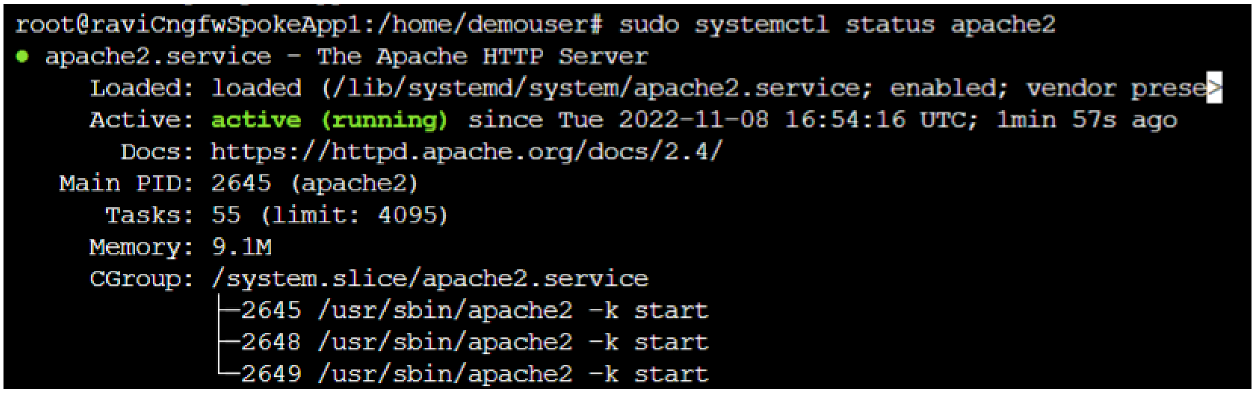
```
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

demouser@raviCngfwSpokeApp1:~$
demouser@raviCngfwSpokeApp1:~$
demouser@raviCngfwSpokeApp1:~$
demouser@raviCngfwSpokeApp1:~$ sudo su
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser# sudo apt update
Hit:1 http://azure.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://azure.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:3 http://azure.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:4 http://azure.archive.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:5 http://azure.archive.ubuntu.com/ubuntu focal/universe amd64 Packages [862
Get:6 http://azure.archive.ubuntu.com/ubuntu focal/universe Translation-en [512
Get:7 http://azure.archive.ubuntu.com/ubuntu focal/universe amd64 c-n-f Metadat
Get:8 http://azure.archive.ubuntu.com/ubuntu focal/multiverse amd64 Packages [1
```

sudo apt install apache2


raviCngfwSpokeApp1 | Serial console
Virtual machine
Search
Logs
Connection monitor (classic)
?
Feedback

```
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser# sudo apt install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

Confirm that the apache server installed successfully using the following command:
sudo systemctl status apache2



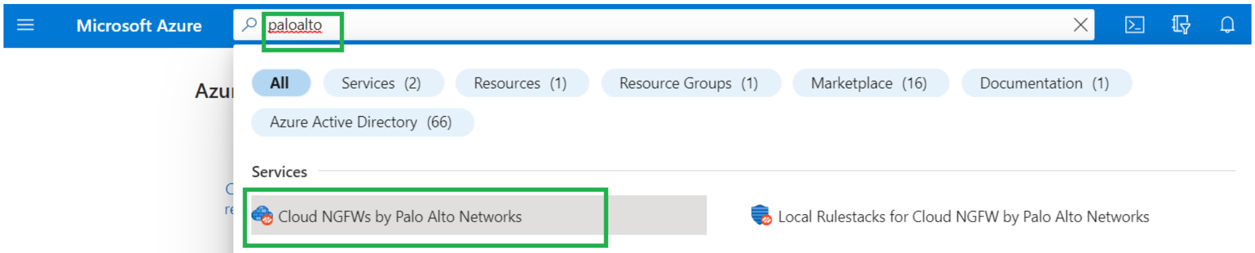
```
root@raviCngfwSpokeApp1:/home/demouser# sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor prese>
     Active: active (running) since Tue 2022-11-08 16:54:16 UTC; 1min 57s ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2645 (apache2)
      Tasks: 55 (limit: 4095)
     Memory: 9.1M
     CGroup: /system.slice/apache2.service
             ├─2645 /usr/sbin/apache2 -k start
             ├─2648 /usr/sbin/apache2 -k start
             └─2649 /usr/sbin/apache2 -k start
```

Create Spoke App02 VNet with Ubuntu Server (For EW-traffic)

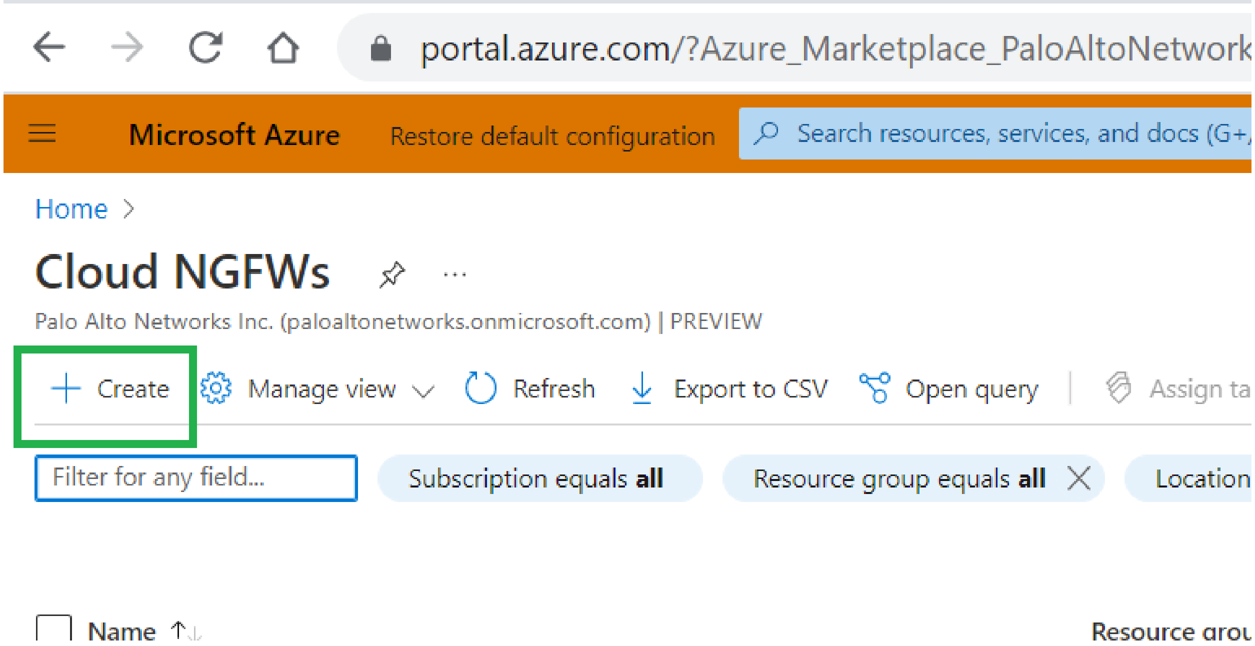
Repeat the above steps to create Spoke VNet2 with Ubuntu server.

Deployment of Cloud NGFW into VNet Infrastructure

1. Now that the environment is set up, deploy the Cloud NGFW to protect the traffic coming from that hub VNet.
2. Login to Azure portal[1] (*use this link only*) and search for “paloalto”. This search displays the Cloud NGFW service by Palo Alto Networks:


Microsoft Azure
paloalto
All
Services (2)
Resources (1)
Resource Groups (1)
Marketplace (16)
Documentation (1)
Azure Active Directory (66)
Services
Cloud NGFWs by Palo Alto Networks
Local Rulestacks for Cloud NGFW by Palo Alto Networks

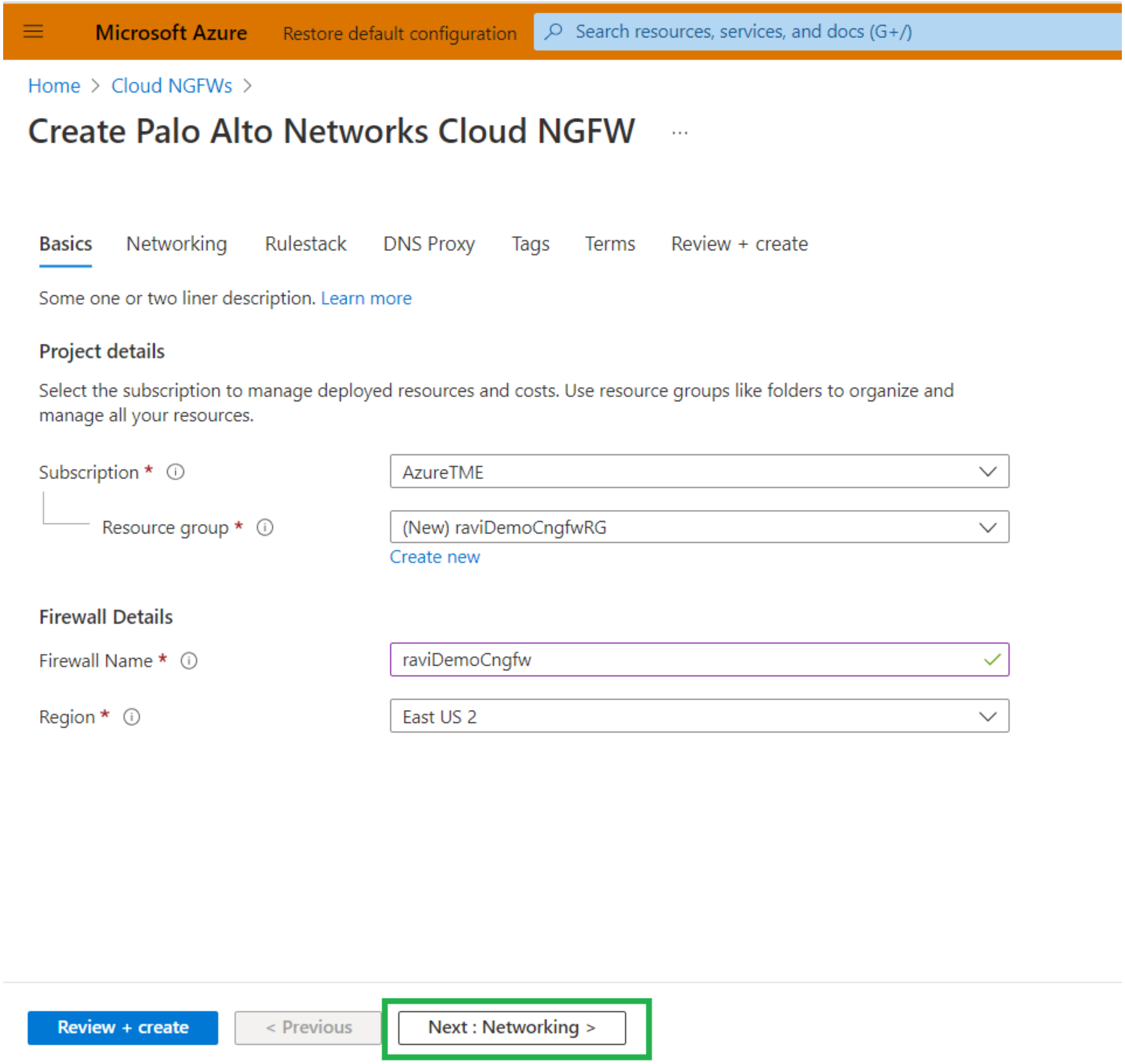
3. Click “Cloud NGFWs” to start the creation of the Palo Alto Networks Cloud NGFW service for Azure.
4. The following screen is the landing page for the Cloud NGFW resource. This screen populates all the available, pre-created Cloud NGFW instances (if not a first-time user). Click **Create** to start the creation of a Cloud NGFW resource:


portal.azure.com/?Azure_Marketplace_PaloAltoNetwork
Microsoft Azure
Restore default configuration
Search resources, services, and docs (G+/
Home >
Cloud NGFWs
Palo Alto Networks Inc. (paloaltonetworks.onmicrosoft.com) | PREVIEW
Create
Manage view
Refresh
Export to CSV
Open query
Assign ta
Filter for any field...
Subscription equals all
Resource group equals all
Location
Name
Resource grou

[1] Deployment of Cloud NGFW can only be done through this private link. This portal has orange colored on the top unlike the typical blue color.

5. After clicking **Create**, the Create Palo Alto Networks Cloud NGFW screen appears. Use the information in the table below to populate basic information for your Cloud NGFW resource:

Subscription	Automatically selected based on the subscription used while logged in.
Resource Group	Use one of the existing resource groups or create a new resource group (by clicking the **Create New** option) in which the Cloud NGFW resource is created.
Firewall Name	Name of the Cloud NGFW Firewall resource.
Region	Region in which Cloud NGFW is provisioned. For this Private Preview, only US East-2 and US Central regions are supported.



Microsoft Azure
Restore default configuration
Search resources, services, and docs (G+/)
Home > Cloud NGFWs >

Create Palo Alto Networks Cloud NGFW

Some one or two liner description. Learn more

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * : AzureTME

Resource group * : (New) raviDemoCngfwRG

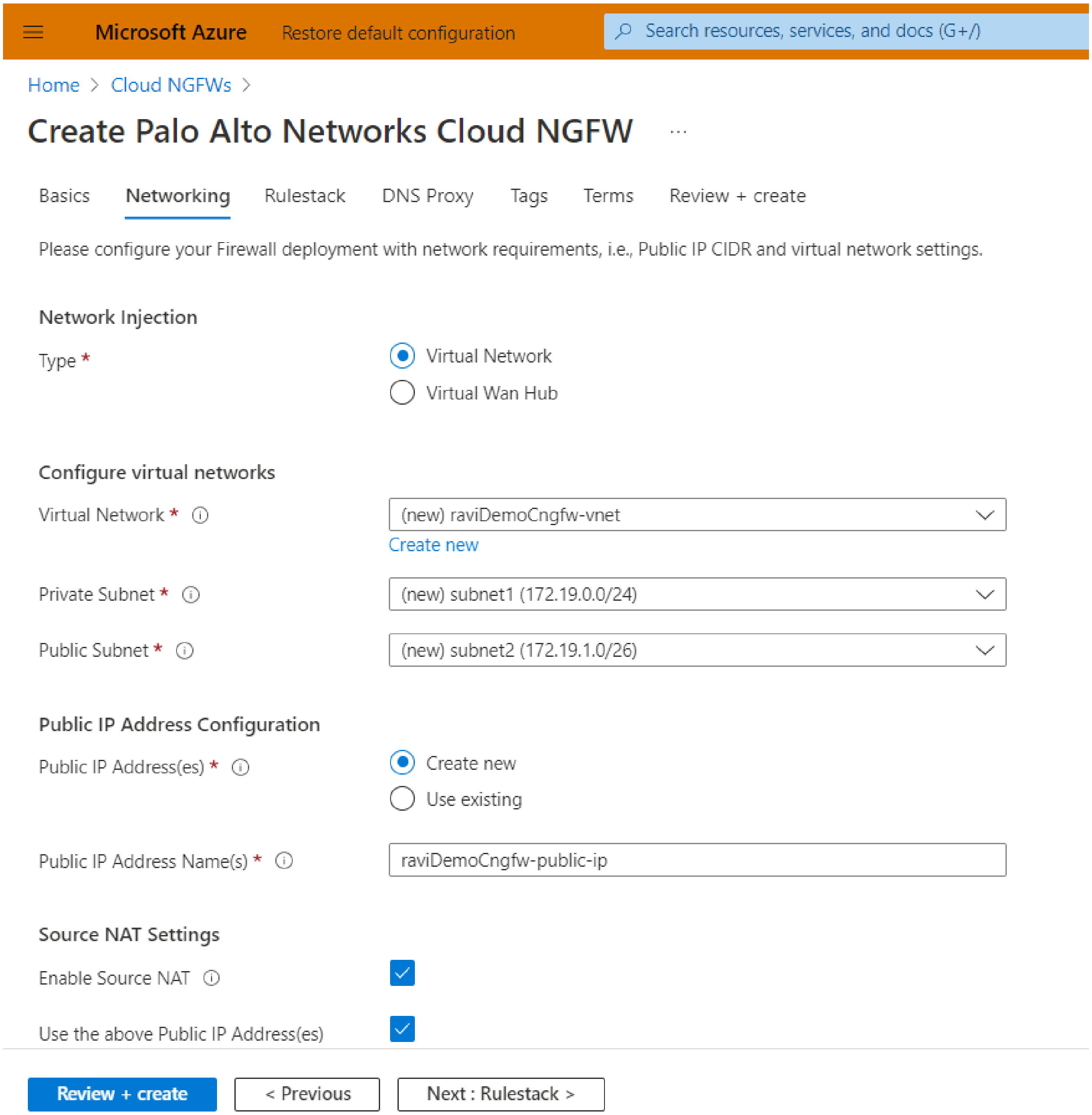
Create new

Firewall Details

Firewall Name * : raviDemoCngfw

Region * : East US 2

6. Once the details are filled in, click **Next: Networking >** and provide information for your networking environment. Choose the Network Injection type from **Virtual network** and **Virtual Wan Hub**. (*Virtual Wan Hub will be available from the end of November*). Create a new virtual network or select an existing virtual network. You can also specify IP addresses. Specify the **Source NAT** option if Network Address Translation (NAT) is used on the traffic going out to the Internet:


Microsoft Azure
Restore default configuration
Search resources, services, and docs (G+/)
Home > Cloud NGFWs >

Create Palo Alto Networks Cloud NGFW ...

Please configure your Firewall deployment with network requirements, i.e., Public IP CIDR and virtual network settings.

Network Injection

Type * ● Virtual Network ○ Virtual Wan Hub

Configure virtual networks

Virtual Network * : (new) raviDemoCngfw-vnet

Create new

Private Subnet * : (new) subnet1 (172.19.0.0/24)

Public Subnet * : (new) subnet2 (172.19.1.0/26)

Public IP Address Configuration

Public IP Address(es) * ● Create new ○ Use existing

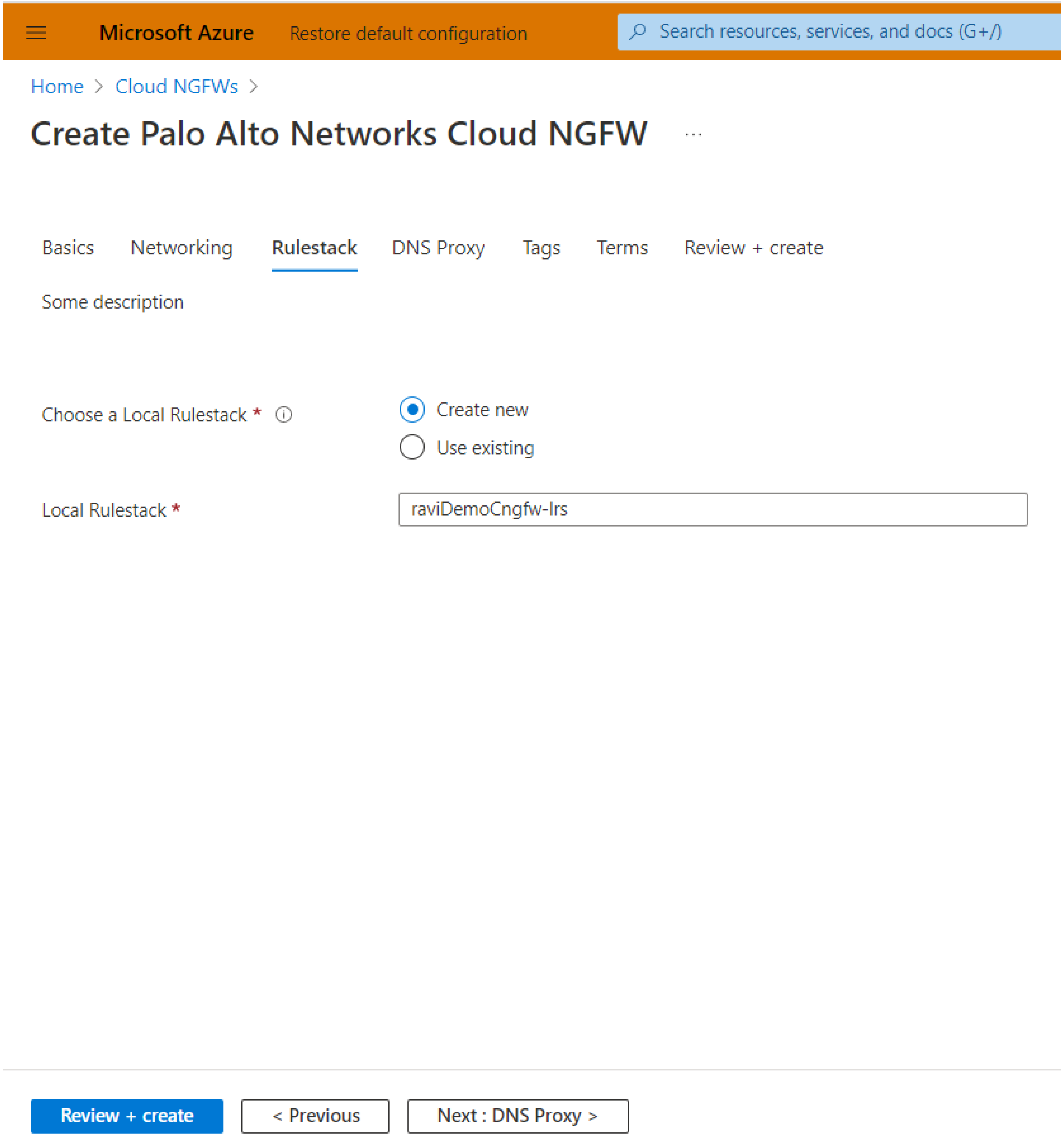
Public IP Address Name(s) * : raviDemoCngfw-public-ip

Source NAT Settings

Enable Source NAT ☑

Use the above Public IP Address(es) ☑

7. Click **Next: Rulestack >** to create a Local Rule stack where rules can be defined. This is a placeholder for the local rule stack. After the creation of Cloud NGFW resource, this rulestack can be modified to add/edit rules, FQDN, and prefix list. If there is a Local Rule Stack that's already created, select it from the drop-down menu:



Microsoft Azure
Restore default configuration
Search resources, services, and docs (G+/)
Home > Cloud NGFWs >

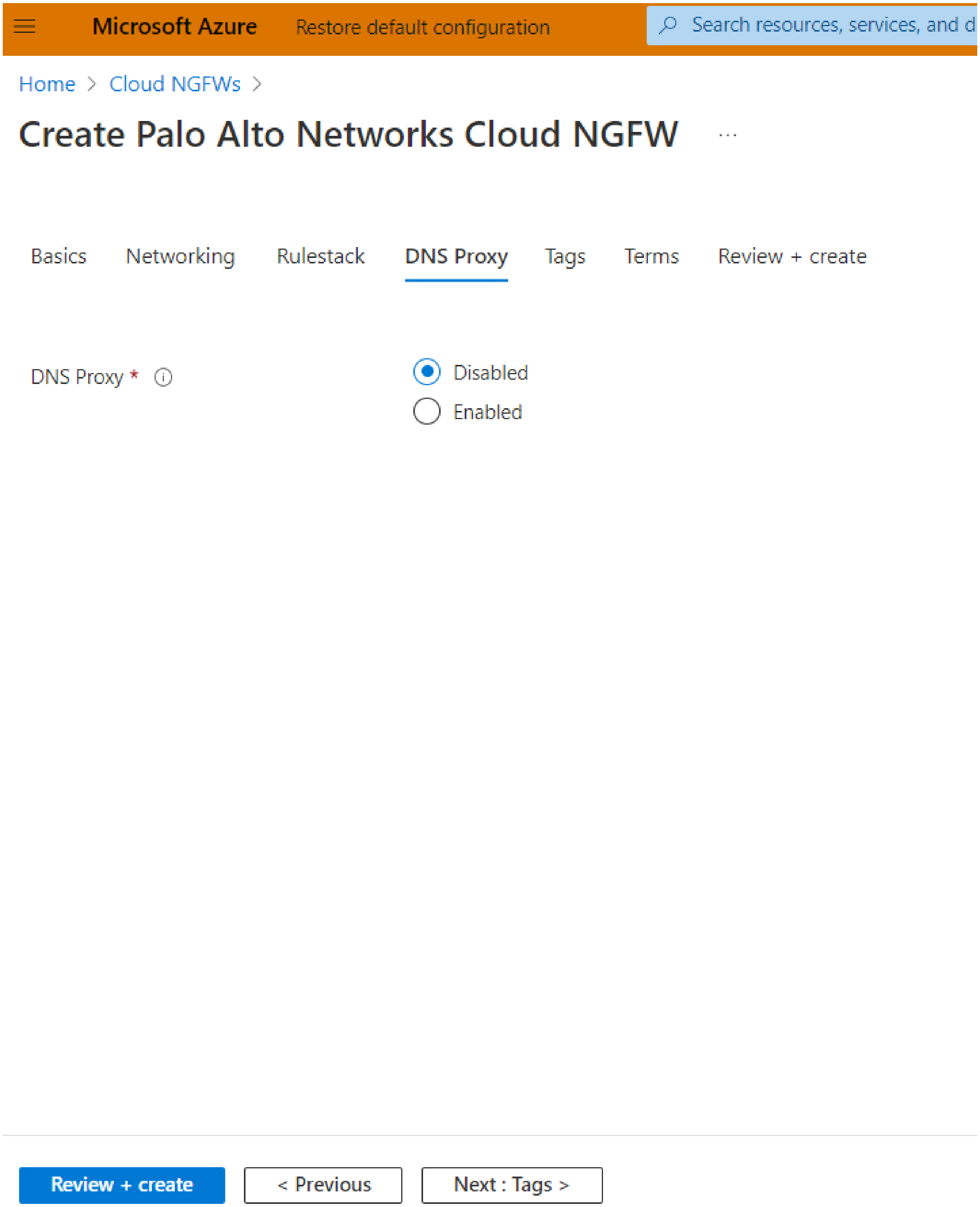
Create Palo Alto Networks Cloud NGFW

Some description

Choose a Local Rulestack * ● Create new ○ Use existing

Local Rulestack * raviDemoCngfw-lrs

8. Click **Next: DNS Proxy >** to configure Cloud NGFW as a DNS Proxy. It is disabled by default:



≡ Microsoft Azure
Restore default configuration
Search resources, services, and d
Home > Cloud NGFWs >

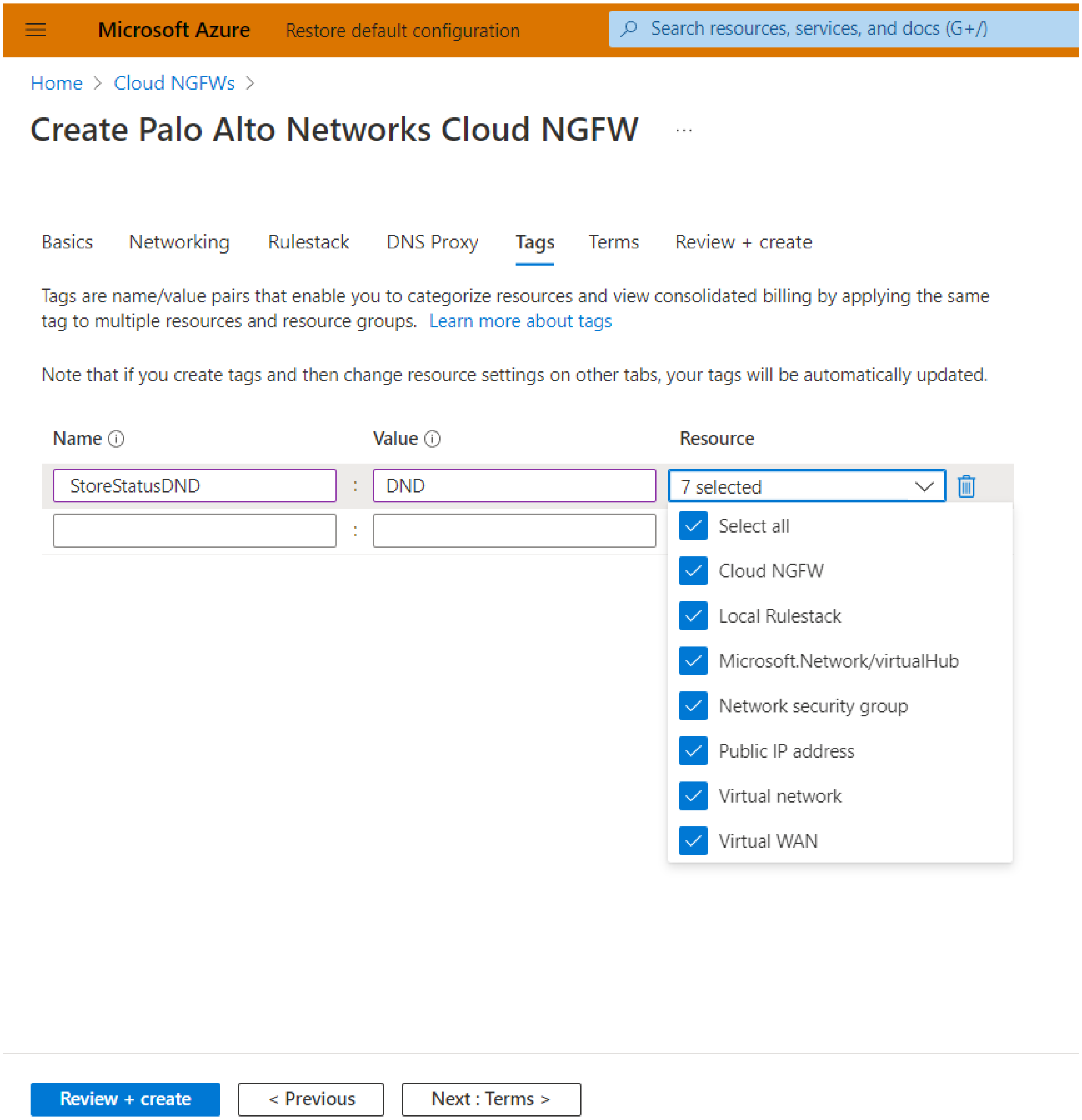
Create Palo Alto Networks Cloud NGFW ...

DNS Proxy * ⓘ

● Disabled

○ Enabled

9. Click **Next: Tags >** to specify tags as per your Azure requirements:



Microsoft Azure
Restore default configuration
Search resources, services, and docs (G+/)
Home > Cloud NGFWs >

Create Palo Alto Networks Cloud NGFW ...

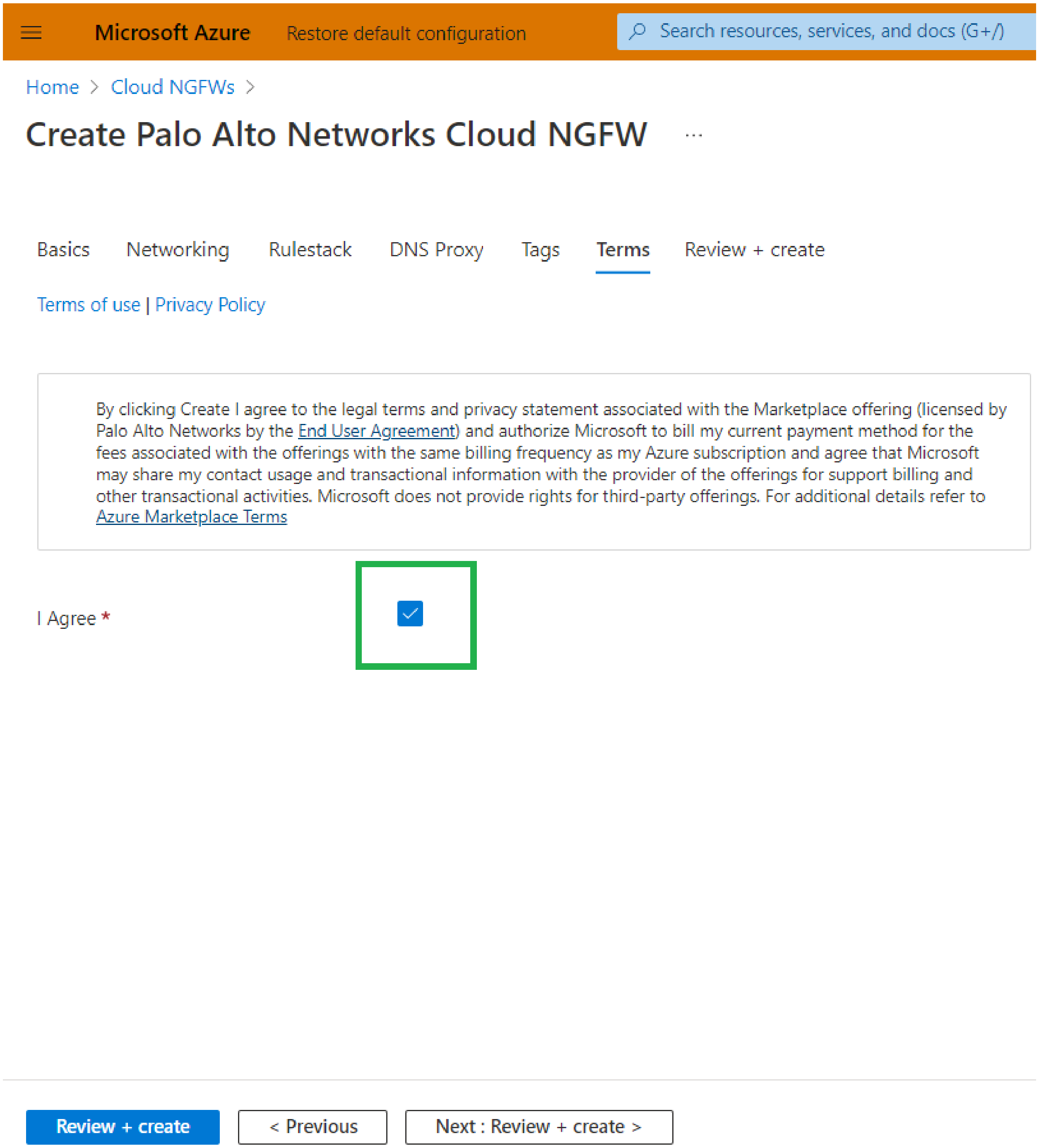
Tags are name/value pairs that enable you to categorize resources and view consolidated billing by applying the same tag to multiple resources and resource groups. Learn more about tags

Note that if you create tags and then change resource settings on other tabs, your tags will be automatically updated.

Name		Value	Resource
StoreStatusDND	:	DND	7 selected
	:		

☑ Select all
☑ Cloud NGFW
☑ Local Rulestack
☑ Microsoft.Network/virtualHub
☑ Network security group
☑ Public IP address
☑ Virtual network
☑ Virtual WAN

10. Click **Next: Terms >** and accept the terms as shown below:


Microsoft Azure
Restore default configuration
Search resources, services, and docs (G+/)
Home > Cloud NGFWs >

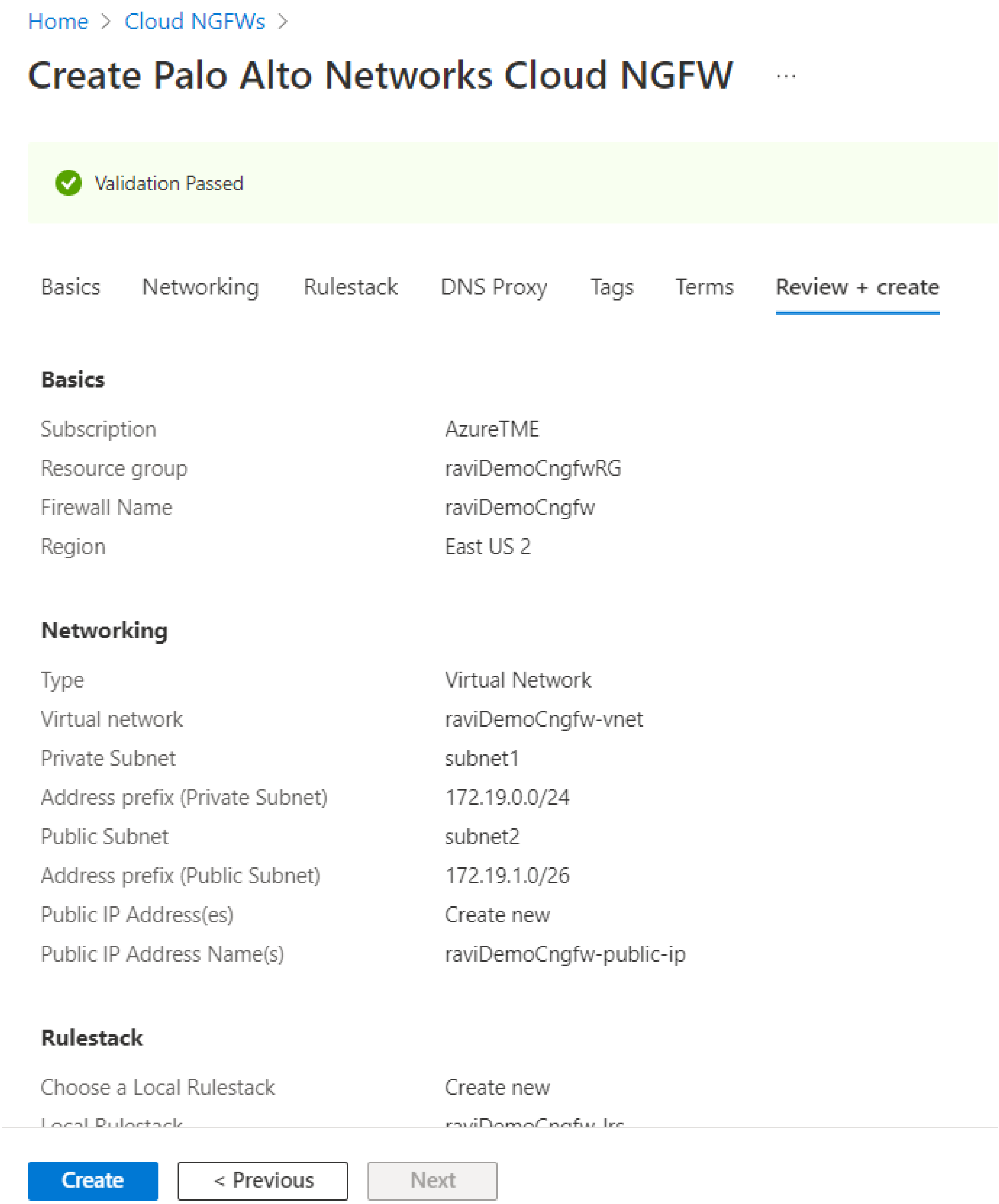
Create Palo Alto Networks Cloud NGFW

Basics
Networking
Rulestack
DNS Proxy
Tags
Terms
Review + create
Terms of use | Privacy Policy

By clicking Create I agree to the legal terms and privacy statement associated with the Marketplace offering (licensed by Palo Alto Networks by the End User Agreement) and authorize Microsoft to bill my current payment method for the fees associated with the offerings with the same billing frequency as my Azure subscription and agree that Microsoft may share my contact usage and transactional information with the provider of the offerings for support billing and other transactional activities. Microsoft does not provide rights for third-party offerings. For additional details refer to Azure Marketplace Terms

I Agree * ☑

11. Click **Next: Review + Create >** and create a Cloud NGFW service. Like any other Azure native service, the resource is validated first and then created. Once the screen shows V**alidation Passed**, click **Create** to deploy the Cloud NGFW service.



Create Palo Alto Networks Cloud NGFW

Validation Passed

Basics

Subscription	AzureTME
Resource group	raviDemoCngfwRG
Firewall Name	raviDemoCngfw
Region	East US 2

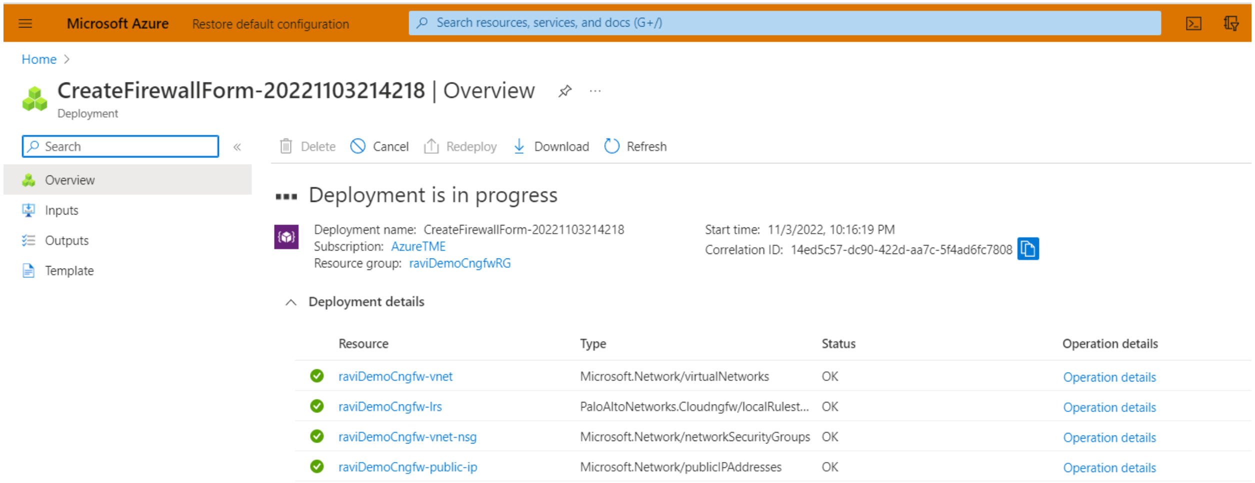
Networking

Type	Virtual Network
Virtual network	raviDemoCngfw-vnet
Private Subnet	subnet1
Address prefix (Private Subnet)	172.19.0.0/24
Public Subnet	subnet2
Address prefix (Public Subnet)	172.19.1.0/26
Public IP Address(es)	Create new
Public IP Address Name(s)	raviDemoCngfw-public-ip

Rulestack

Choose a Local Rulestack	Create new
Local Rulestack	raviDemoCngfw-lrs

After creating the Cloud NGFW service the deployment progress is displayed:



CreateFirewallForm-20221103214218 | Overview

Deployment

Deployment is in progress

Deployment name: CreateFirewallForm-20221103214218
Subscription: AzureTME
Resource group: raviDemoCngfwRG

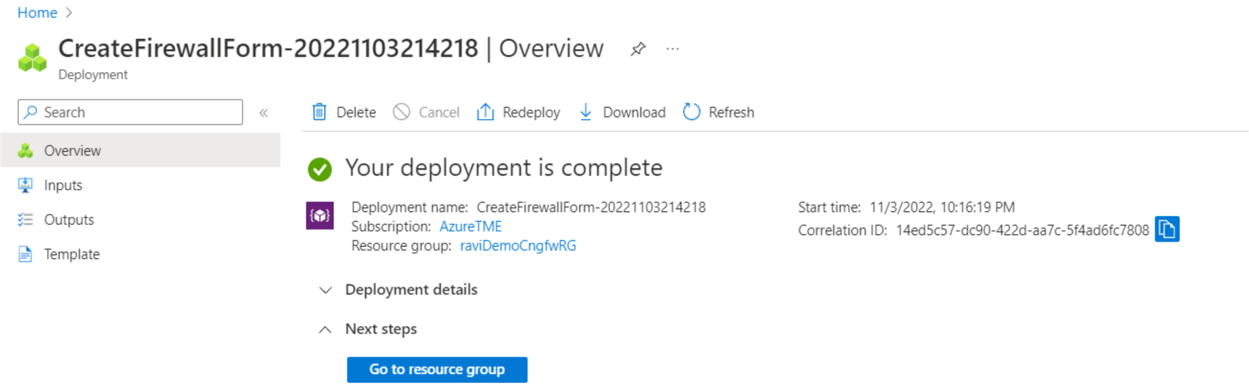
Start time: 11/3/2022, 10:16:19 PM
Correlation ID: 14ed5c57-dc90-422d-aa7c-5f4ad6fc7808

Deployment details

Resource	Type	Status	Operation details
raviDemoCngfw-vnet	Microsoft.Network/virtualNetworks	OK	Operation details
raviDemoCngfw-lrs	PaloAltoNetworks.Cloudngfw/localRulest...	OK	Operation details
raviDemoCngfw-vnet-nsg	Microsoft.Network/networkSecurityGroups	OK	Operation details
raviDemoCngfw-public-ip	Microsoft.Network/publicIPAddresses	OK	Operation details

The deployment of a Cloud NGFW takes approximately 30 minutes.

On a successful deployment, the screen below appears. Click Go to resource group to verify the resources created for this deployment:



CreateFirewallForm-20221103214218 | Overview

Deployment

Your deployment is complete

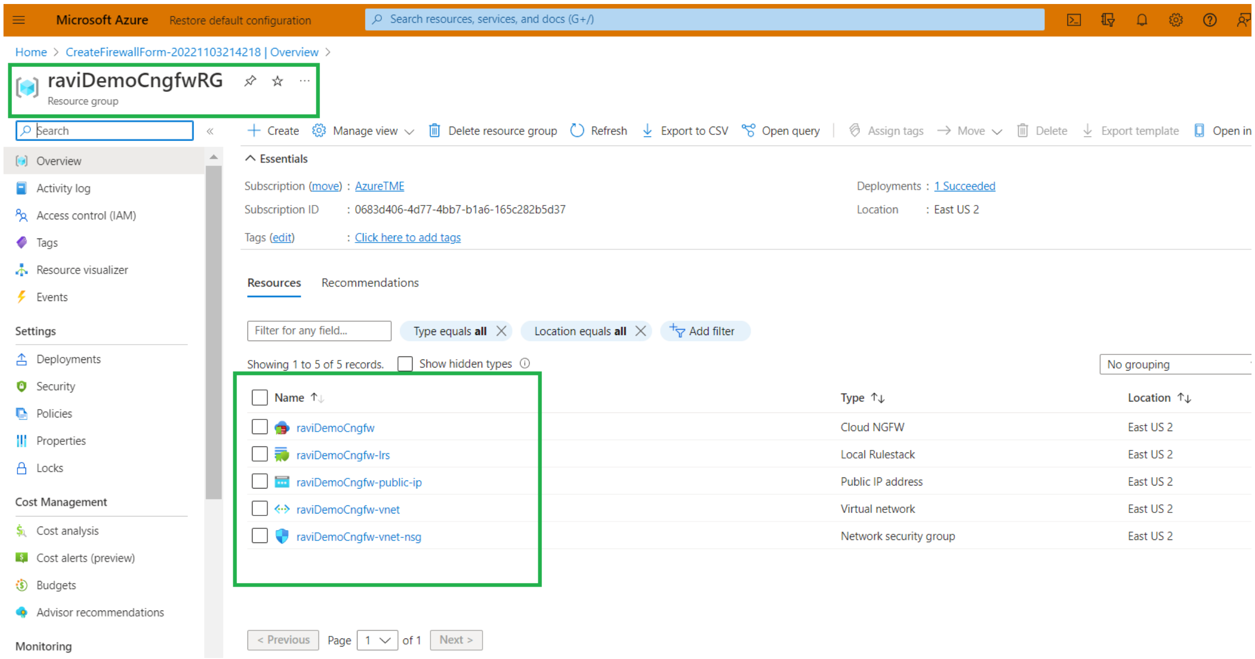
Deployment name: CreateFirewallForm-20221103214218
Subscription: AzureTME
Resource group: raviDemoCngfwRG

Start time: 11/3/2022, 10:16:19 PM
Correlation ID: 14ed5c57-dc90-422d-aa7c-5f4ad6fc7808

Deployment details

Next steps

12. There are five resources created, which include Cloud NGFW, Local Rule stack, Public IP address, Virtual Network, and security group:

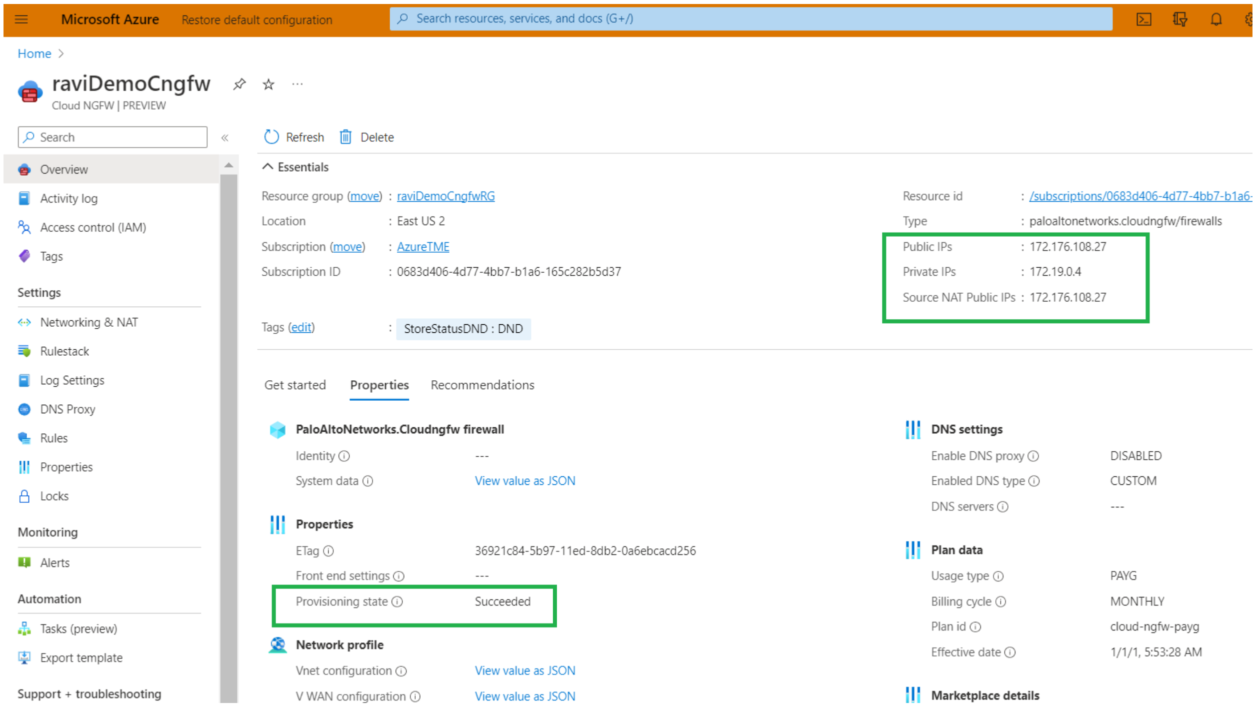


Subscription (move) : AzureTME
Subscription ID : 0683d406-4d77-4bb7-b1a6-165c282b5d37
Tags (edit) : Click here to add tags
Deployments : 1 Succeeded
Location : East US 2

Showing 1 to 5 of 5 records. □ Show hidden types

Name	Type	Location
raviDemoCngfw	Cloud NGFW	East US 2
raviDemoCngfw-lrs	Local Rulestack	East US 2
raviDemoCngfw-public-ip	Public IP address	East US 2
raviDemoCngfw-vnet	Virtual network	East US 2
raviDemoCngfw-vnet-nsg	Network security group	East US 2

13. Once the Cloud NGFW resource is created, click on it to verify that the Provisioning state shows **Succeeded**. This screen also displays Public and Private IP addresses that are associated with the Cloud NGFW service. Use this information in further steps of this document to route traffic through the Cloud NGFW service:



Resource group (move) : raviDemoCngfwRG
Location : East US 2
Subscription (move) : AzureTME
Subscription ID : 0683d406-4d77-4bb7-b1a6-165c282b5d37
Tags (edit) : StoreStatusDND : DND
Resource id : /subscriptions/0683d406-4d77-4bb7-b1a6-
Type : paloaltonetworks.cloudngfw/firewalls
Public IPs : 172.176.108.27
Private IPs : 172.19.0.4
Source NAT Public IPs : 172.176.108.27

PaloAltoNetworks.Cloudngfw firewall
Identity ---
System data View value as JSON

Properties
ETag 36921c84-5b97-11ed-8db2-0a6ebcacd256
Front end settings ---
Provisioning state Succeeded

Network profile
Vnet configuration View value as JSON
V WAN configuration View value as JSON

DNS settings
Enable DNS proxy DISABLED
Enabled DNS type CUSTOM
DNS servers ---

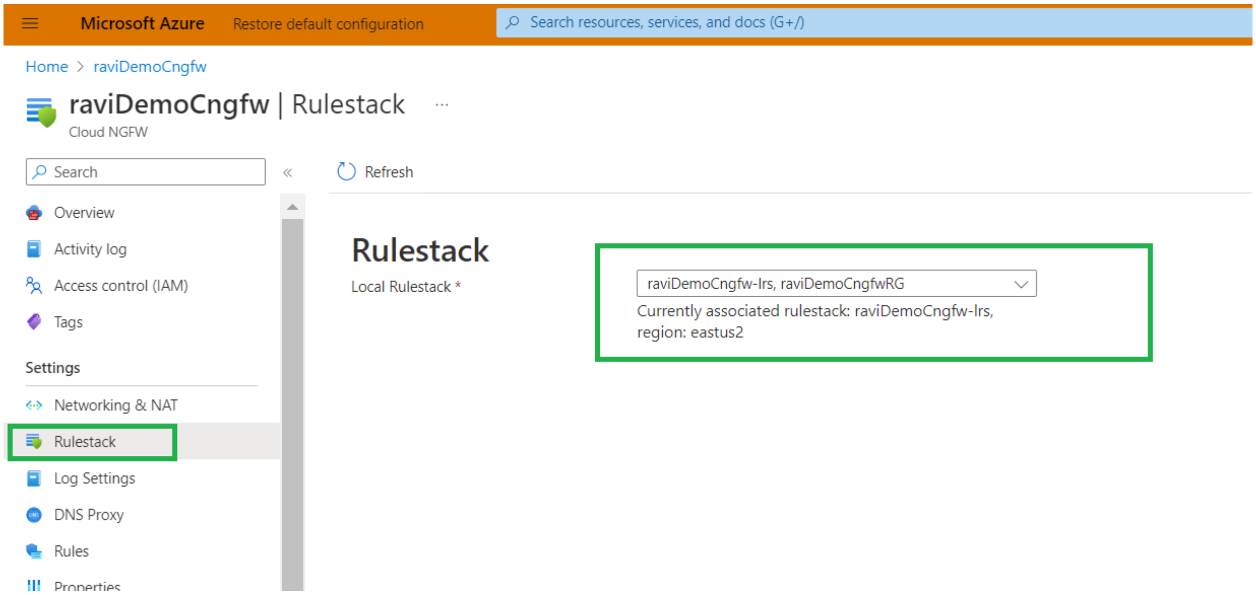
Plan data
Usage type PAYG
Billing cycle MONTHLY
Plan id cloud-ngfw-payg
Effective date 1/1/1, 5:53:28 AM

Marketplace details

Post Deployment of Cloud NGFW

Create/Update Rule stack

1. To update/edit the rulestack, click the **Rulestack** option available in the Cloud NGFW resource. As shown below, this displays the rulestack associated with the cloud NGFW service along with the resource group:

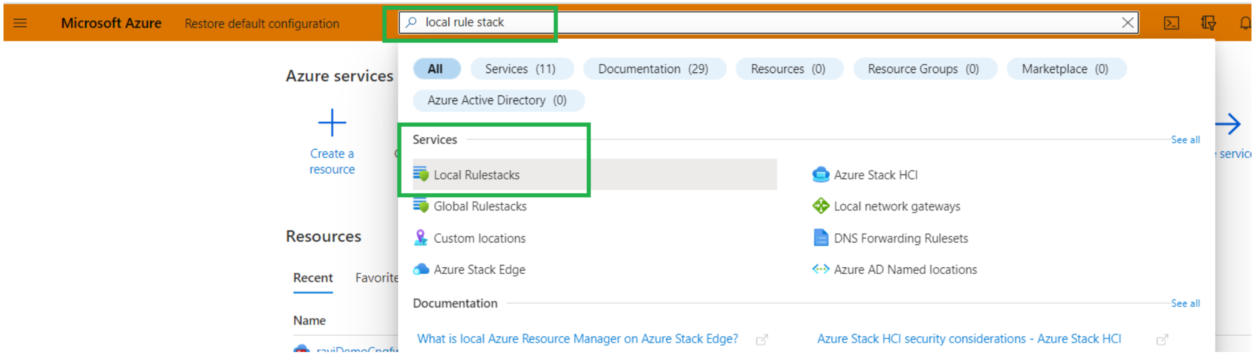


Local Rulestack * raviDemoCngfw-lrs, raviDemoCngfwRG

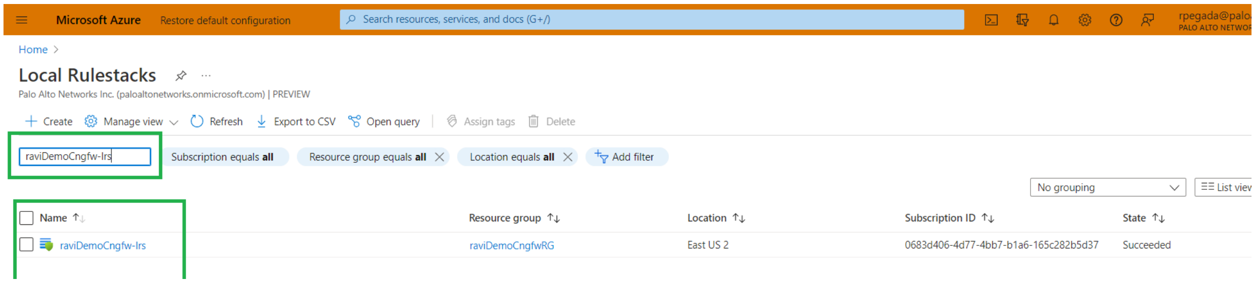
Currently associated rulestack: raviDemoCngfw-lrs, region: eastus2

The Local Rulestack is now associated with **raviDemoCngfw-lrs**.
Next, modify this rulestack to add firewall rules to allow some traffic and block specific traffic.
By default Cloud NGFW blocks all traffic.

2. Search for Local Rulestack service in the global search of the Azure portal:



Click the Local Rulestacks service to navigate to the list of local rulestacks associate with your Cloud NGFW subscription.
Within this page search for **raviDemoCngfw-lrs**, the local rulestack associated with the Cloud NGFW service created in the previous step:

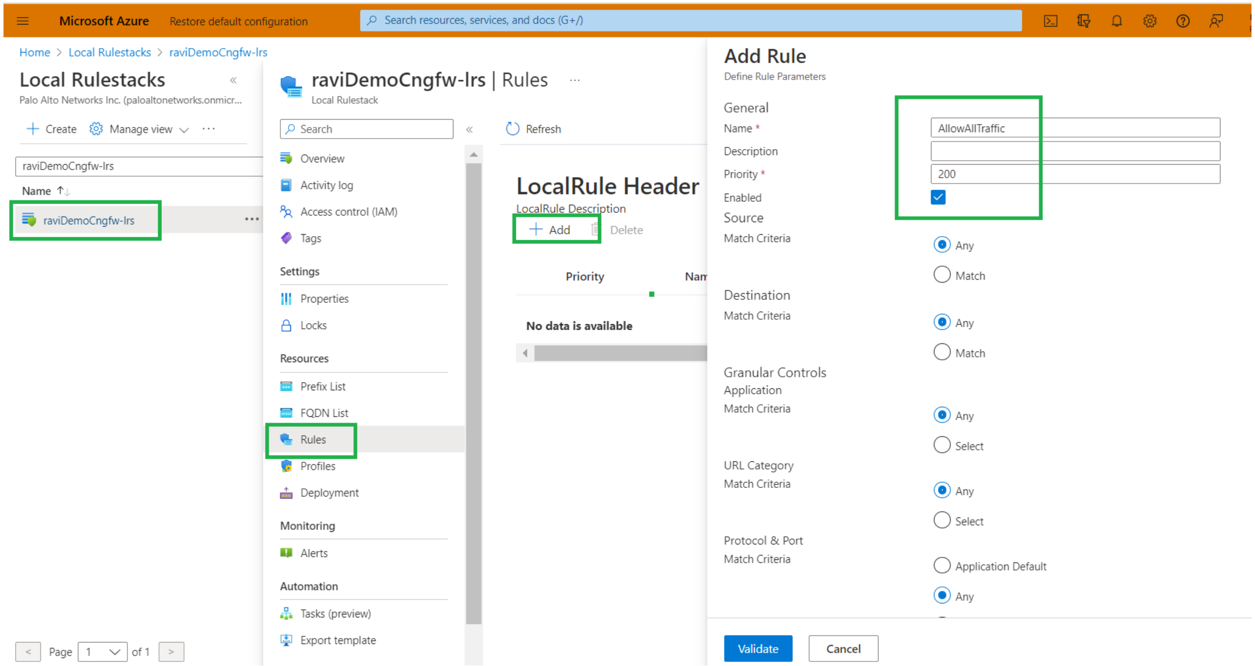


Search: raviDemoCngfw-lrs
Subscription equals **all** | Resource group equals **all** | Location equals **all**

Name	Resource group	Location	Subscription ID	State
raviDemoCngfw-lrs	raviDemoCngfwRG	East US 2	0683d406-4d77-4bb7-b1a6-165c282b5d37	Succeeded

Verify the state of this Local rulestack **Succeeded**.

3. Click your rulestack(raviDemoCngfw-lrs) to add rules as shown below. Modify the rules as per your use cases and functionality. Add a rule to allow traffic. Fill in the mandatory fields and use the default settings for the remaining fields:



Add Rule
Define Rule Parameters

General
Name * AllowAllTraffic
Description
Priority * 200
Enabled ☑

Source
Match Criteria ● Any ○ Match

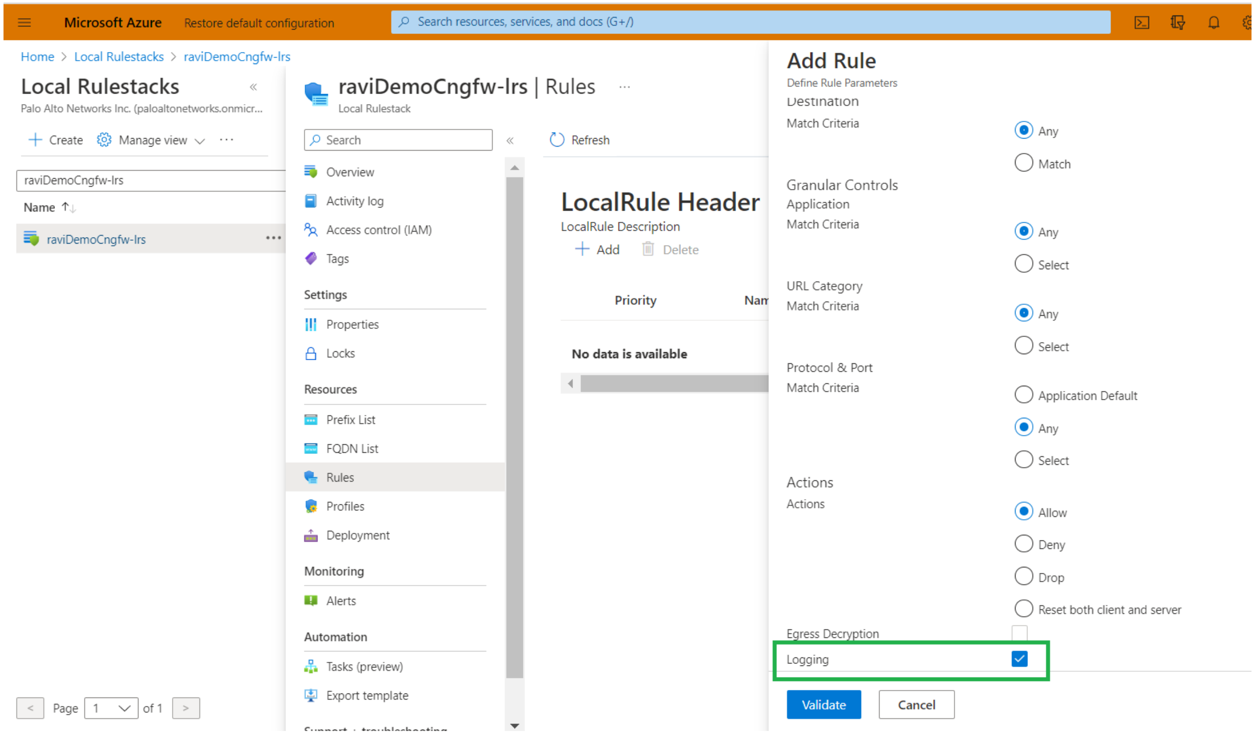
Destination
Match Criteria ● Any ○ Match

Granular Controls
Application
Match Criteria ● Any ○ Select

URL Category
Match Criteria ● Any ○ Select

Protocol & Port
Match Criteria ○ Application Default ● Any

Enable logging as part of the rule configuration, as shown below:



Add Rule

Define Rule Parameters

Destination

Match Criteria ● Any ○ Match

Granular Controls

Application

Match Criteria ● Any ○ Select

URL Category

Match Criteria ● Any ○ Select

Protocol & Port

Match Criteria ○ Application Default ● Any ○ Select

Actions

Actions ● Allow ○ Deny ○ Drop ○ Reset both client and server

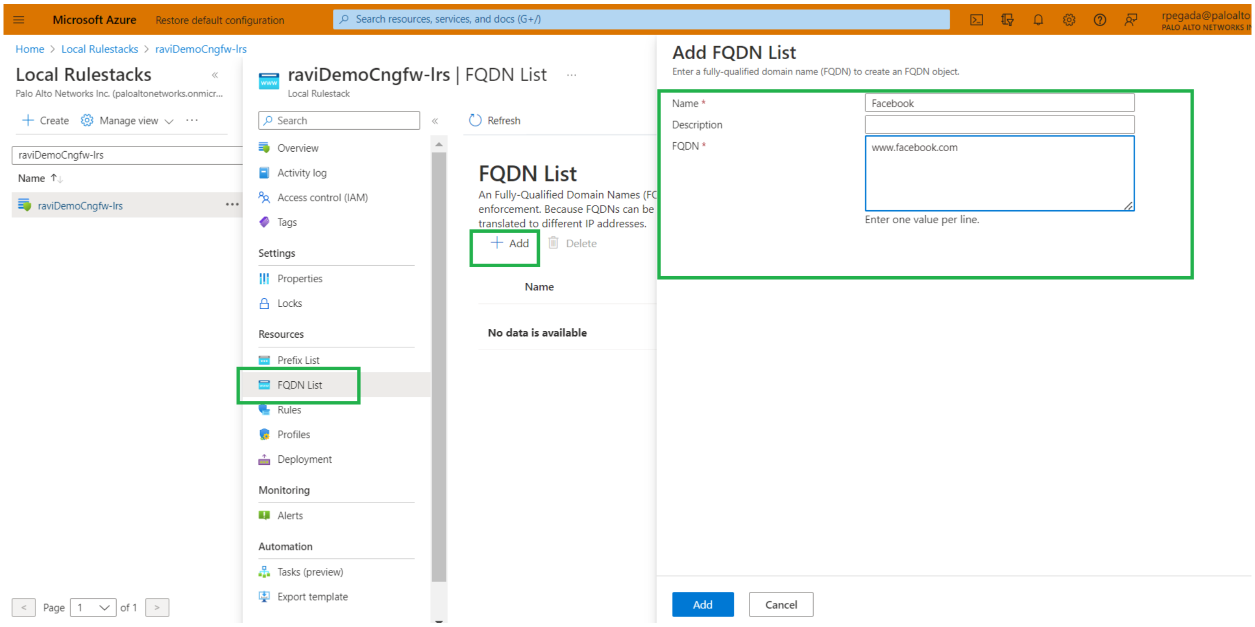
Egress Decryption □

Logging ☑

Validate Cancel

Click **Validate** and then **Add** to incorporate the rule.

4. Add an FQDN list that includes Facebook, and use this list to add a rule to block facebook.com:



Add FQDN List

Enter a fully-qualified domain name (FQDN) to create an FQDN object.

Name * Facebook

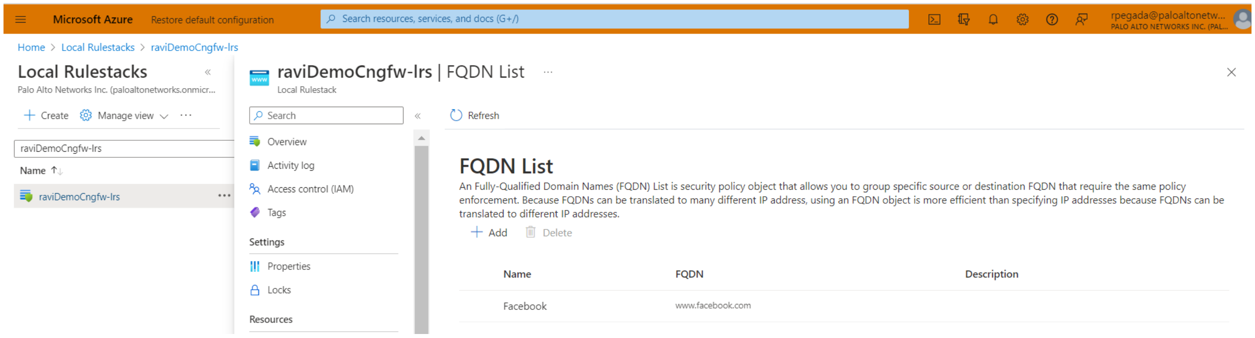
Description

FQDN * www.facebook.com

Enter one value per line.

Add Cancel

Facebook now appears in the **FQDN List**:



Home > Local Rulestacks > raviDemoCngfw-lrs

raviDemoCngfw-lrs | FQDN List

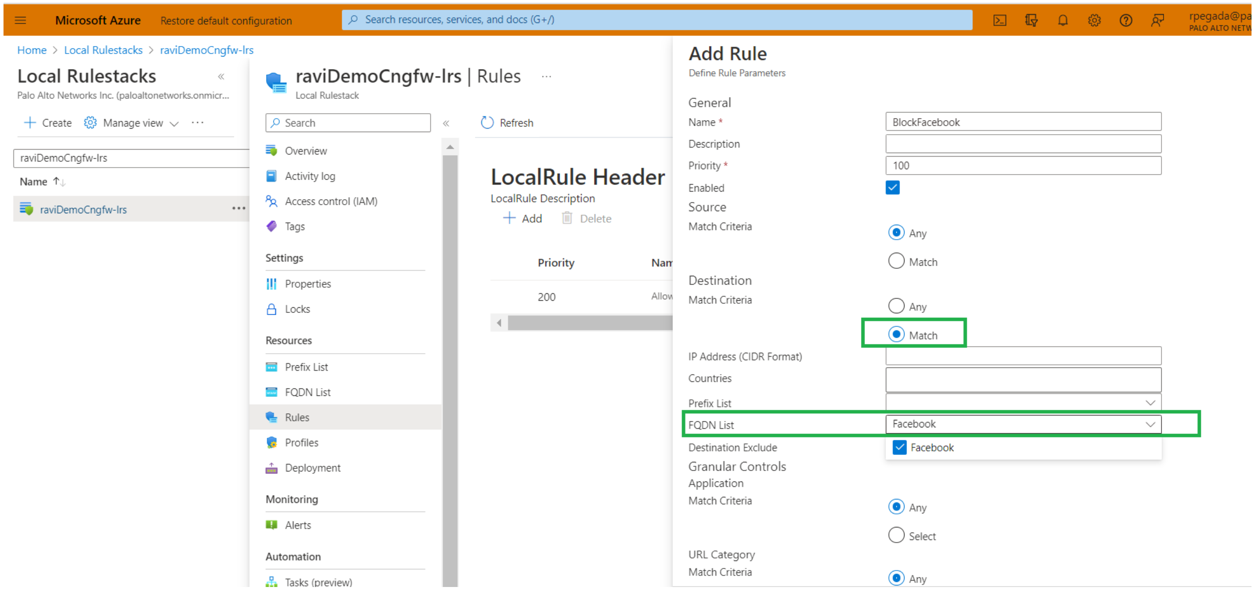
Local Rulestack

FQDN List

An Fully-Qualified Domain Names (FQDN) List is security policy object that allows you to group specific source or destination FQDN that require the same policy enforcement. Because FQDNs can be translated to many different IP address, using an FQDN object is more efficient than specifying IP addresses because FQDNs can be translated to different IP addresses.

Name	FQDN	Description
Facebook	www.facebook.com	

Return to the Rules setting page and add a rule that matches the FQDN list created. Set the action to **Drop** to block Facebook traffic:



Add Rule

Define Rule Parameters

General

Name * BlockFacebook

Description

Priority * 100

Enabled ☑

Source

Match Criteria ● Any ○ Match

Destination

Match Criteria ○ Any ● Match

IP Address (CIDR Format)

Countries

Prefix List

FQDN List: Facebook

Destination Exclude ☑ Facebook

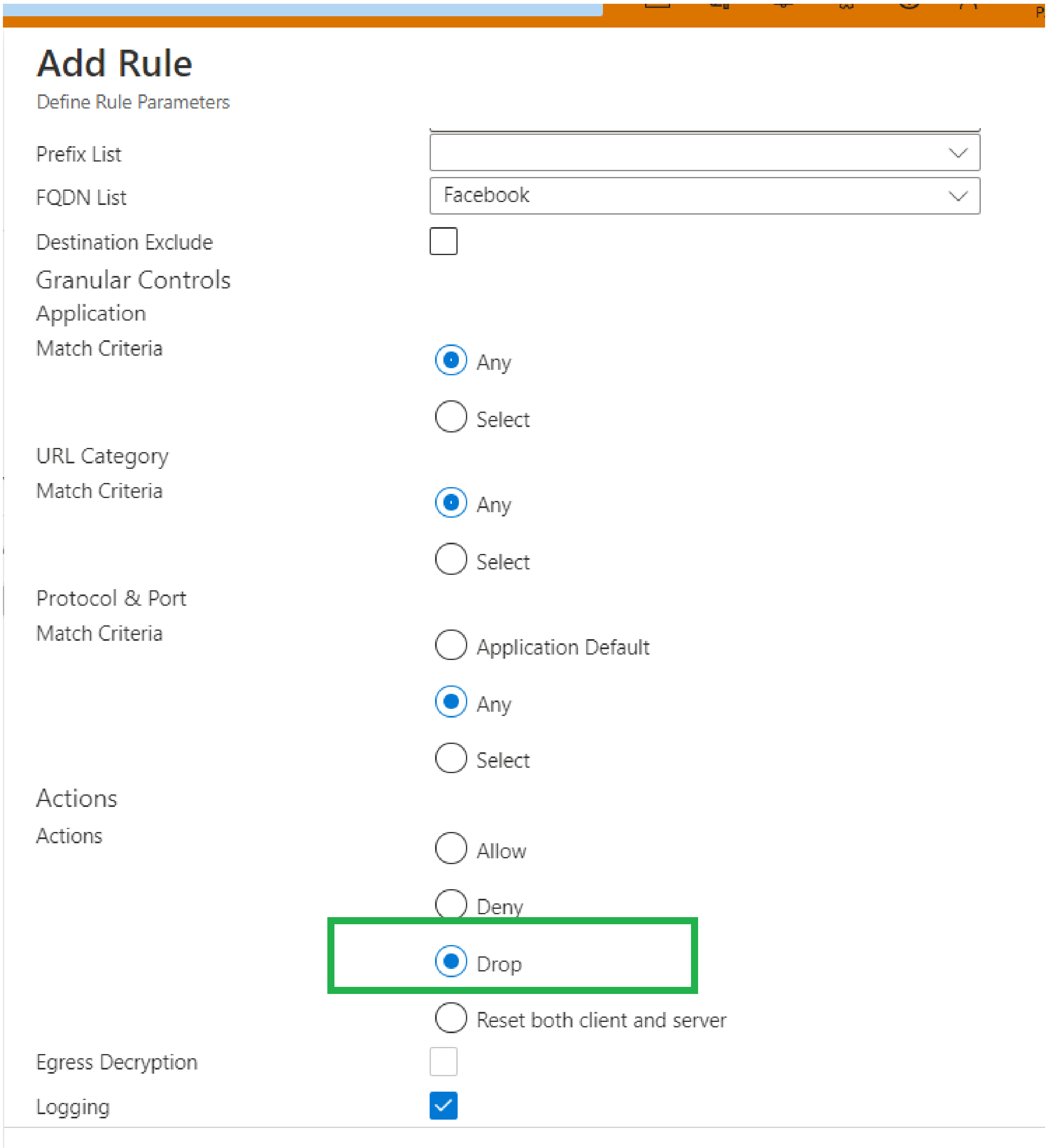
Granular Controls

Application

Match Criteria ● Any ○ Select

URL Category

Match Criteria ● Any



Add Rule

Define Rule Parameters

Prefix List

FQDN List: Facebook

Destination Exclude ☐

Granular Controls

Application

Match Criteria: ● Any ○ Select

URL Category

Match Criteria: ● Any ○ Select

Protocol & Port

Match Criteria: ○ Application Default ● Any ○ Select

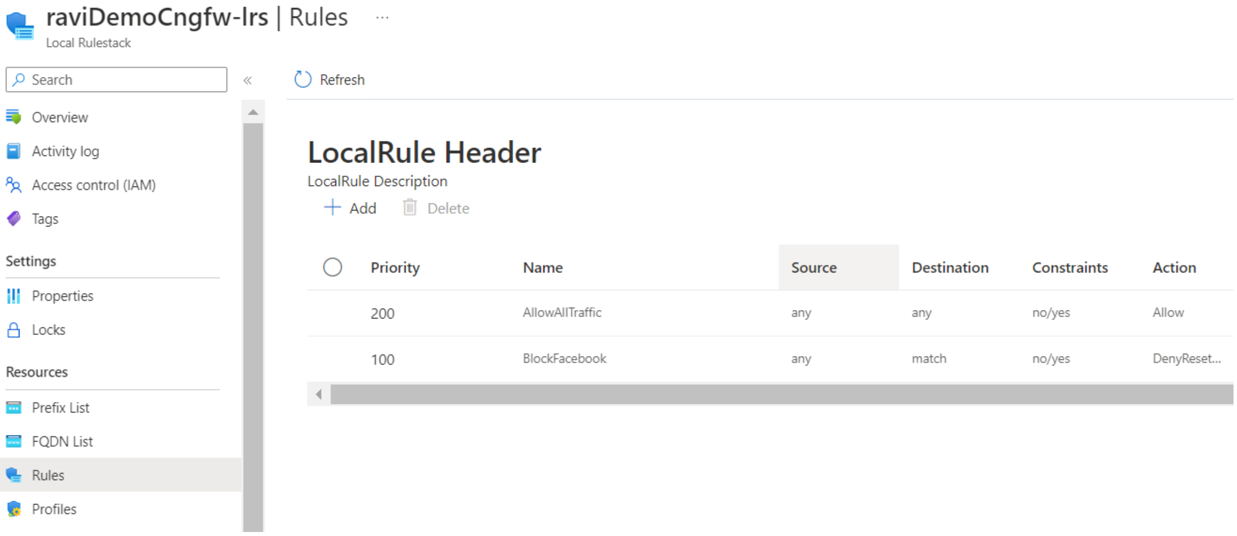
Actions

Actions: ○ Allow ○ Deny ● Drop ○ Reset both client and server

Egress Decryption ☐

Logging ☑

5. Both the rules appear as shown below:



raviDemoCngfw-lrs | Rules

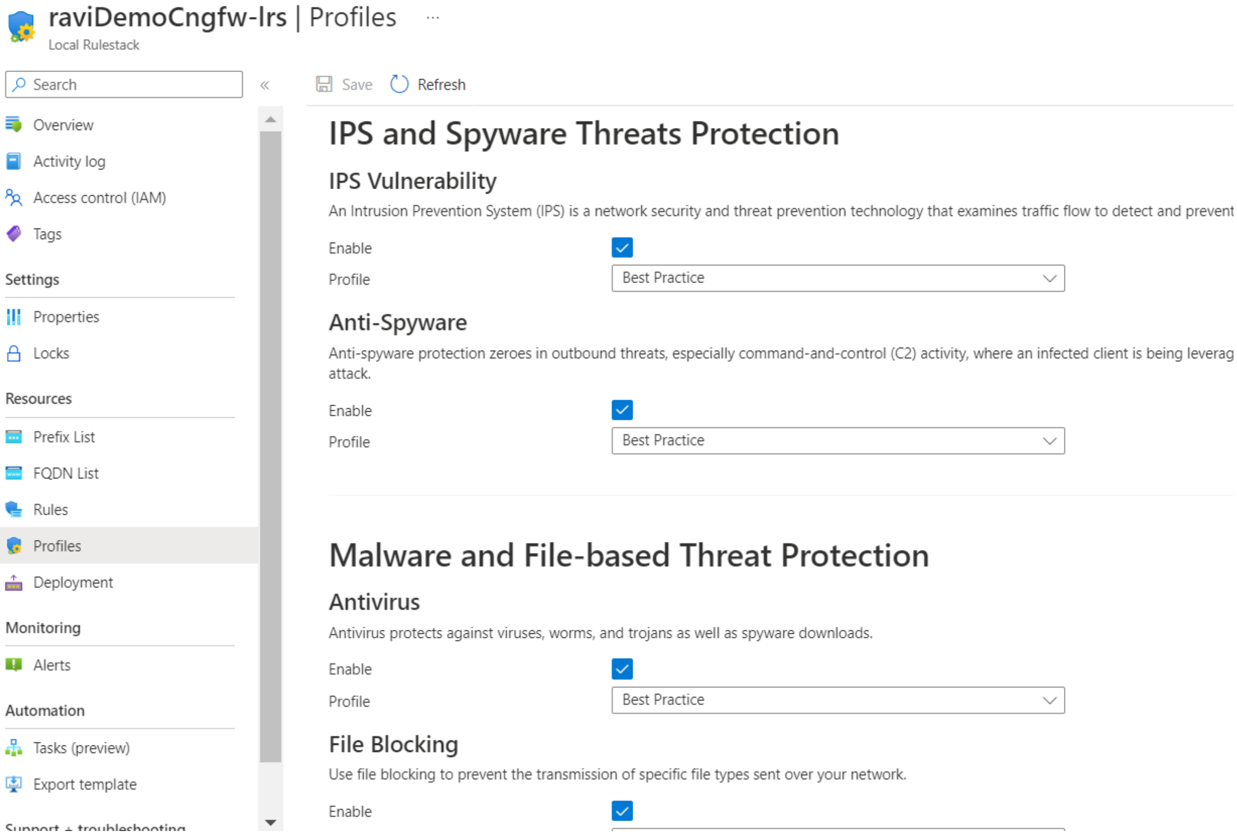
Local Rulestack

LocalRule Header

LocalRule Description

○	Priority	Name	Source	Destination	Constraints	Action
	200	AllowAllTraffic	any	any	no/yes	Allow
	100	BlockFacebook	any	match	no/yes	DenyReset...

6. As part of this Cloud NGFW service, the security profiles are enabled with best practice configurations by default. This means that the traffic is secured with the best security profiles from day one, once the Cloud NGFW is deployed in the network:



raviDemoCngfw-lrs | Profiles

Local Rulestack

IPS and Spyware Threats Protection

IPS Vulnerability

An Intrusion Prevention System (IPS) is a network security and threat prevention technology that examines traffic flow to detect and prevent

Enable ☑

Profile: Best Practice

Anti-Spyware

Anti-spyware protection zeroes in outbound threats, especially command-and-control (C2) activity, where an infected client is being leverag attack.

Enable ☑

Profile: Best Practice

Malware and File-based Threat Protection

Antivirus

Antivirus protects against viruses, worms, and trojans as well as spyware downloads.

Enable ☑

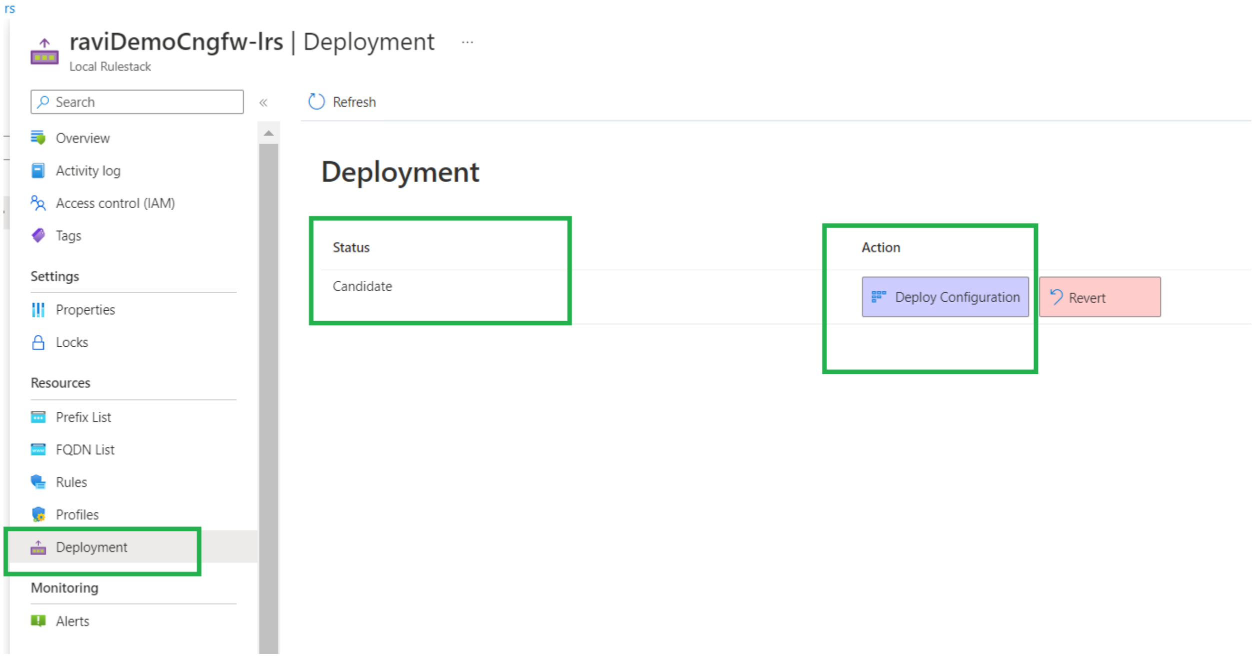
Profile: Best Practice

File Blocking

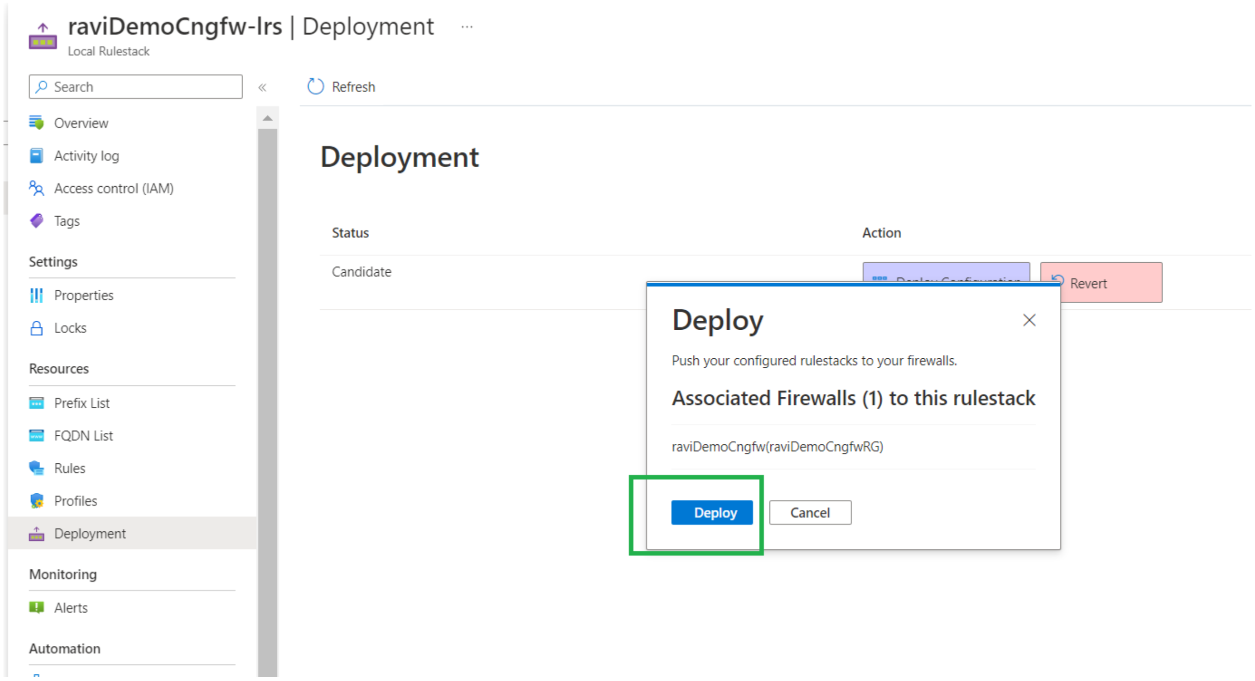
Use file blocking to prevent the transmission of specific file types sent over your network.

Enable ☑

7. Now that the rules have been modified, they should be deployed onto the Local rulestack associated with the Cloud NGFW service. Click the **Deployment** tab to see the page below. The deployment status displays as **Candidate**, which means the configuration was built but not deployed. Next, click **Deploy Configuration** to deploy the configuration onto the Cloud NGFW service. *It is mandatory to do this step as without this the configuration will not be deployed onto the rulestack.*



After clicking **Deploy Configuration,** a pop-up displays the firewalls associated with this rulestack. Click **Deploy** to configure this rulestack on all the associated firewalls:

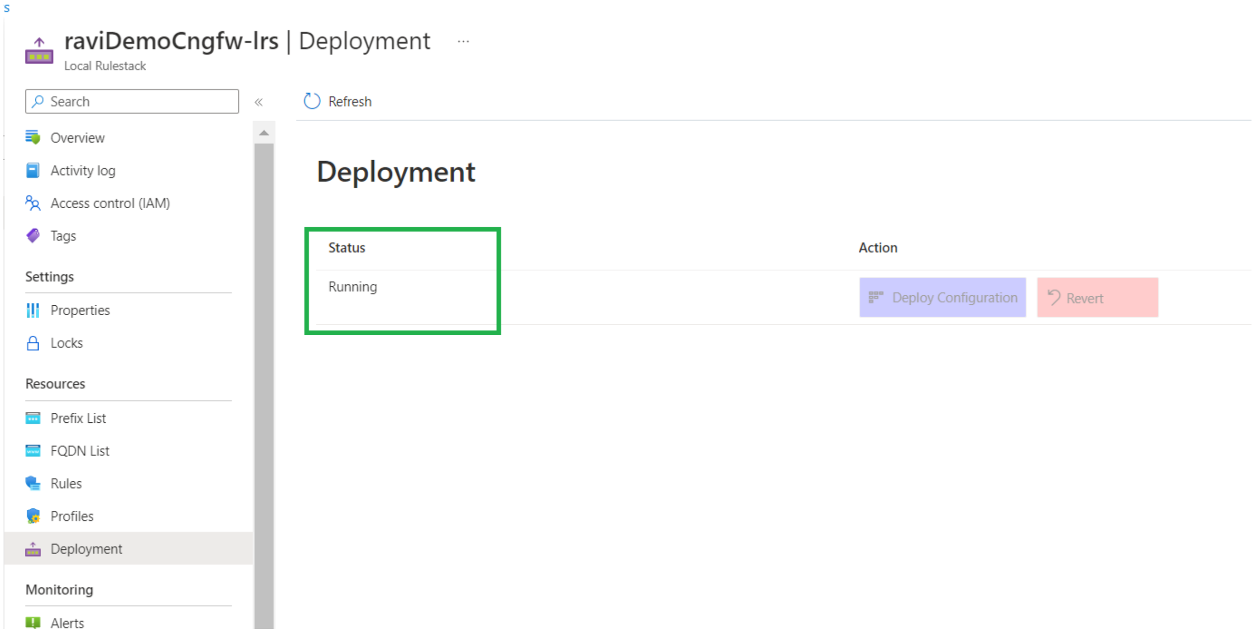


Push your configured rulestacks to your firewalls.

Associated Firewalls (1) to this rulestack

raviDemoCngfw(raviDemoCngfwRG)

After successfully deploying the configuration, the screen displays the deployment status as **Running**



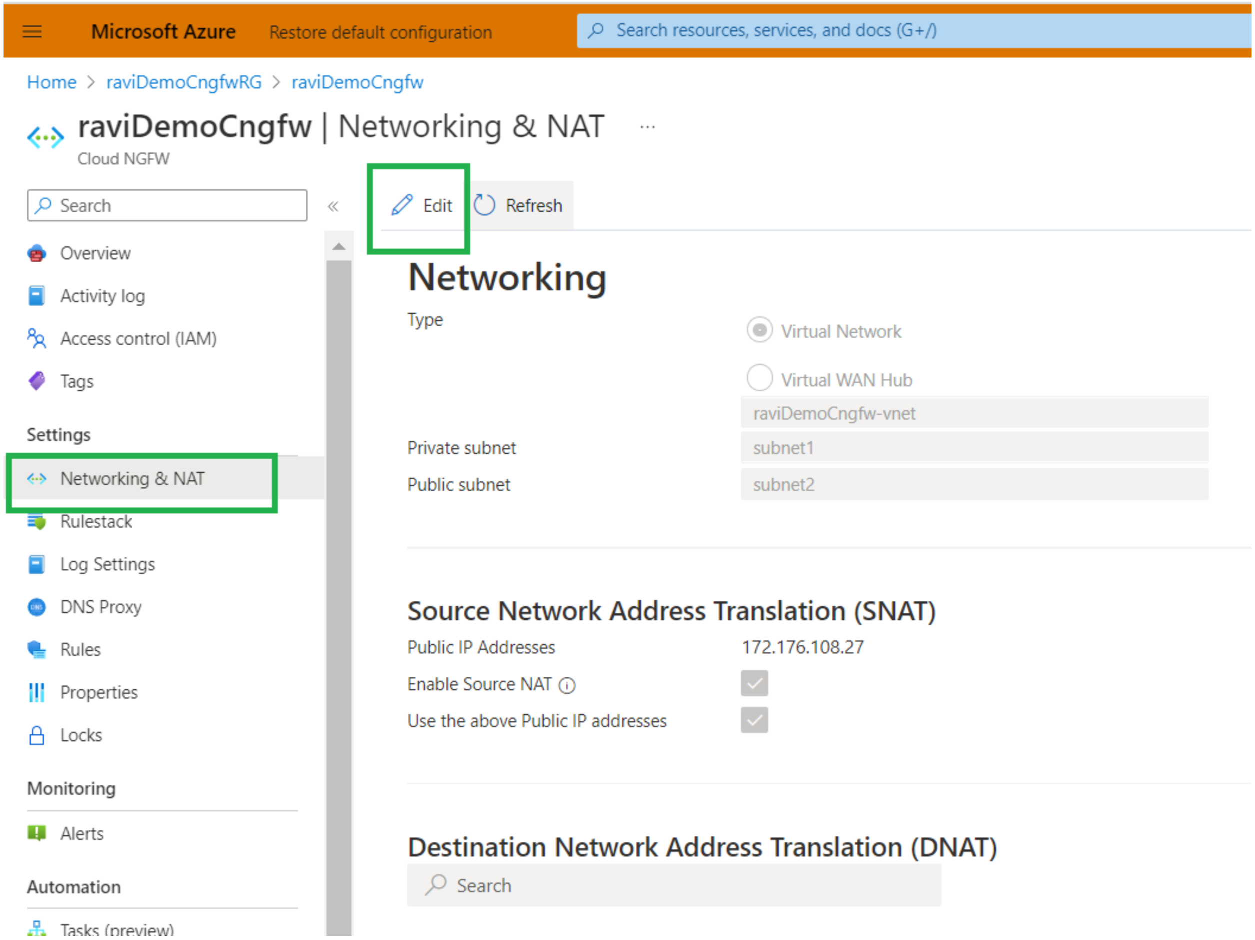
Status	Action
Running	

With this, the Cloud NGFW and Local rulestack are successfully deployed.

Source/Destination NAT rule on Cloud NGFW

Configure a destination NAT rule with frontend configuration on Cloud NGFW to take care of Inbound traffic towards App1 or App2 present on spoke VNet1 or spoke VNet2.

1. Access the **Networking & NAT** settings screen for the loud NGFW resource. The first thing to observe is whether the **Source NAT** setting has been enabled or not. During the creation of the Cloud NGFW resource,(step 6) if Source NAT was enabled, that's how it will show up here.
2. Click **Edit** to add the Destination NAT rule.



Networking

Type: ● Virtual Network ○ Virtual WAN Hub

raviDemoCngfw-vnet

Private subnet: subnet1

Public subnet: subnet2

Source Network Address Translation (SNAT)

Public IP Addresses: 172.176.108.27

Enable Source NAT ☑

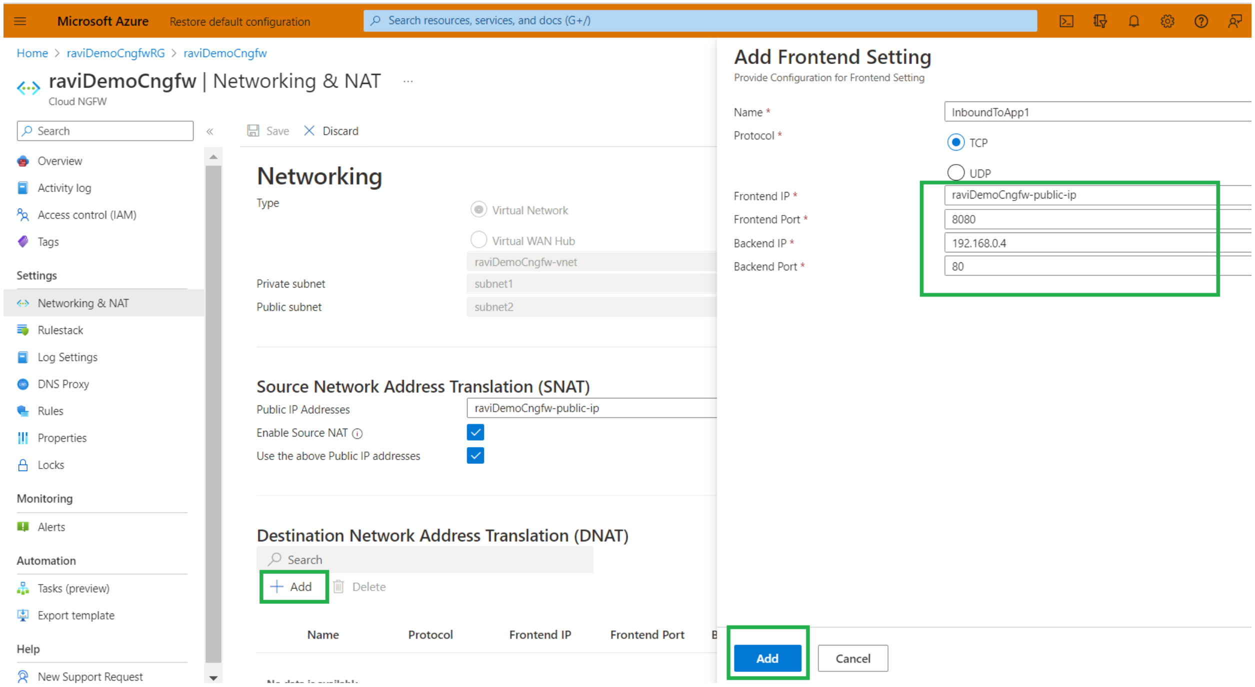
Use the above Public IP addresses ☑

Destination Network Address Translation (DNAT)

Search

3. Add a **Destination NAT** rule with frontend configuration as shown below. Frontend IP is the Public IP address associated with Cloud NGFW (choose this from the drop-down menu). To access App1 (192.168.0.4), deployed on spoke VNet1, on port 80(HTTP), we are going to use Cloud NGFW frontend IP address

and port 8080. After adding the Destination NAT rule, save the configuration by clicking **Add** .



raviDemoCngfw | Networking & NAT

Cloud NGFW

Networking

Type: ● Virtual Network ○ Virtual WAN Hub

raviDemoCngfw-vnet

Private subnet: subnet1

Public subnet: subnet2

Source Network Address Translation (SNAT)

Public IP Addresses: raviDemoCngfw-public-ip

Enable Source NAT ☑

Use the above Public IP addresses ☑

Destination Network Address Translation (DNAT)

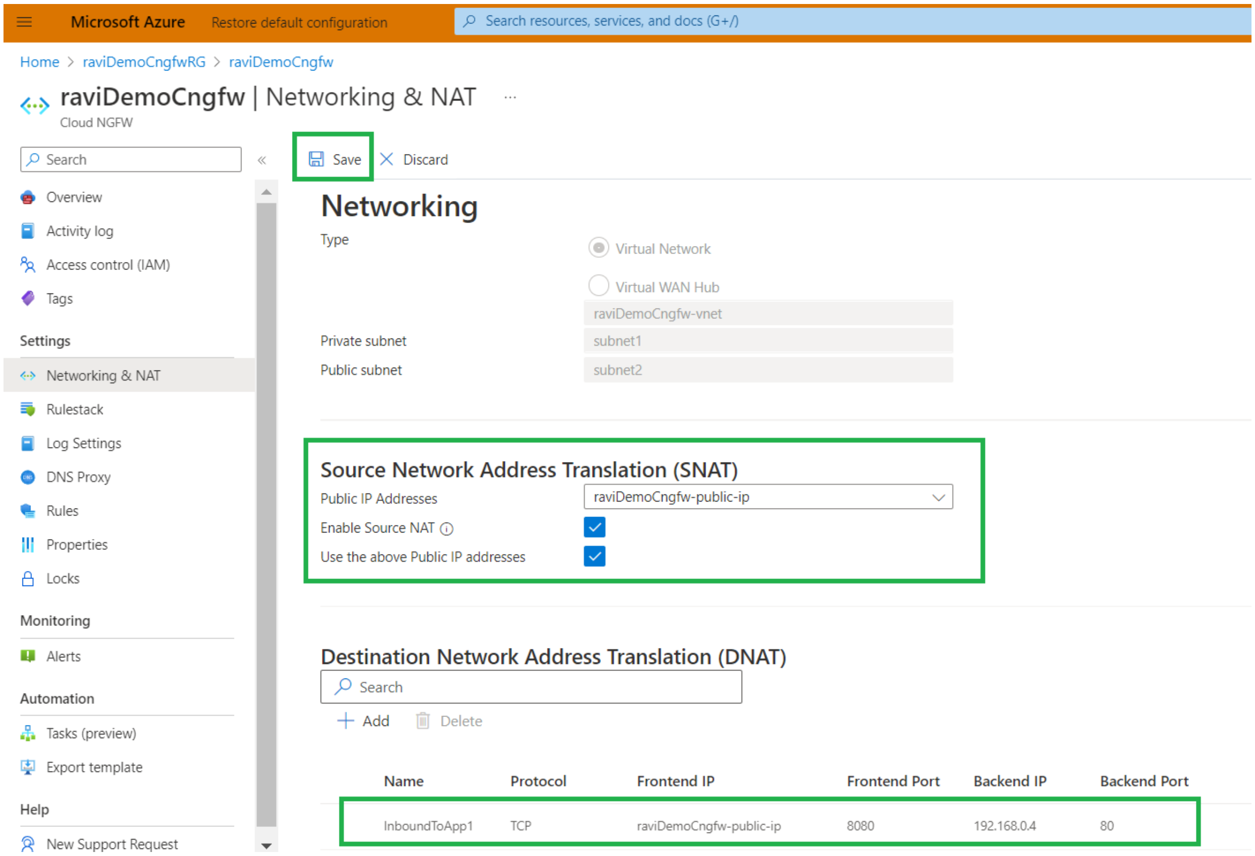
No data is available

Add Frontend Setting

Provide Configuration for Frontend Setting

Name *	InboundToApp1
Protocol *	● TCP ○ UDP
Frontend IP *	raviDemoCngfw-public-ip
Frontend Port *	8080
Backend IP *	192.168.0.4
Backend Port *	80

Once the destination NAT rule has been added, click **Save** to deploy this configuration on to the Cloud NGFW resource:



Networking

Type: ● Virtual Network ○ Virtual WAN Hub

raviDemoCngfw-vnet

Private subnet: subnet1

Public subnet: subnet2

Source Network Address Translation (SNAT)

Public IP Addresses: raviDemoCngfw-public-ip

Enable Source NAT ☑

Use the above Public IP addresses ☑

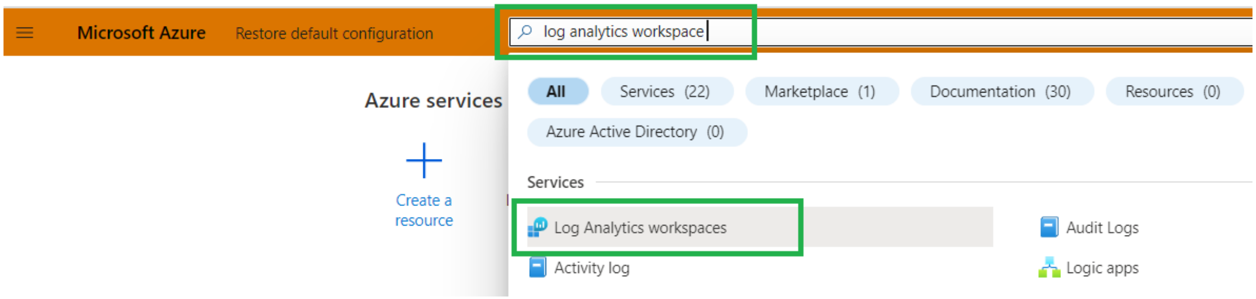
Destination Network Address Translation (DNAT)

Name	Protocol	Frontend IP	Frontend Port	Backend IP	Backend Port
InboundToApp1	TCP	raviDemoCngfw-public-ip	8080	192.168.0.4	80

With this configuration in place, the address http://frondendIP:8080 is redirected to App1 on port 80 through Cloud NGFW. This means that inbound traffic is now flowing through the Cloud NGFW.

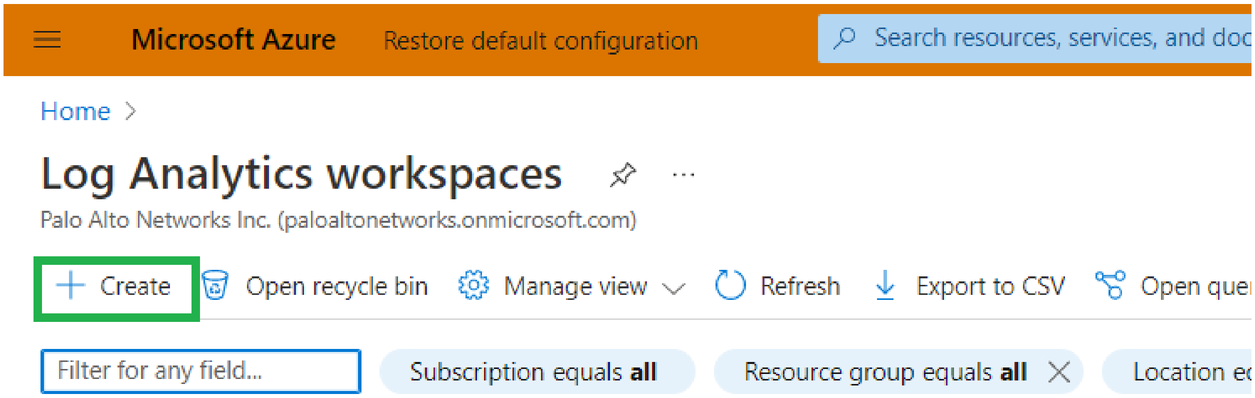
Configure Logging

1. Before configuring Log settings on Cloud NGFW, create the Log Analytics workspace on Azure. Search for **Azure Log Analytics** workspace as shown below and click **Log Analytics Workspaces** service to add it to the workspace:

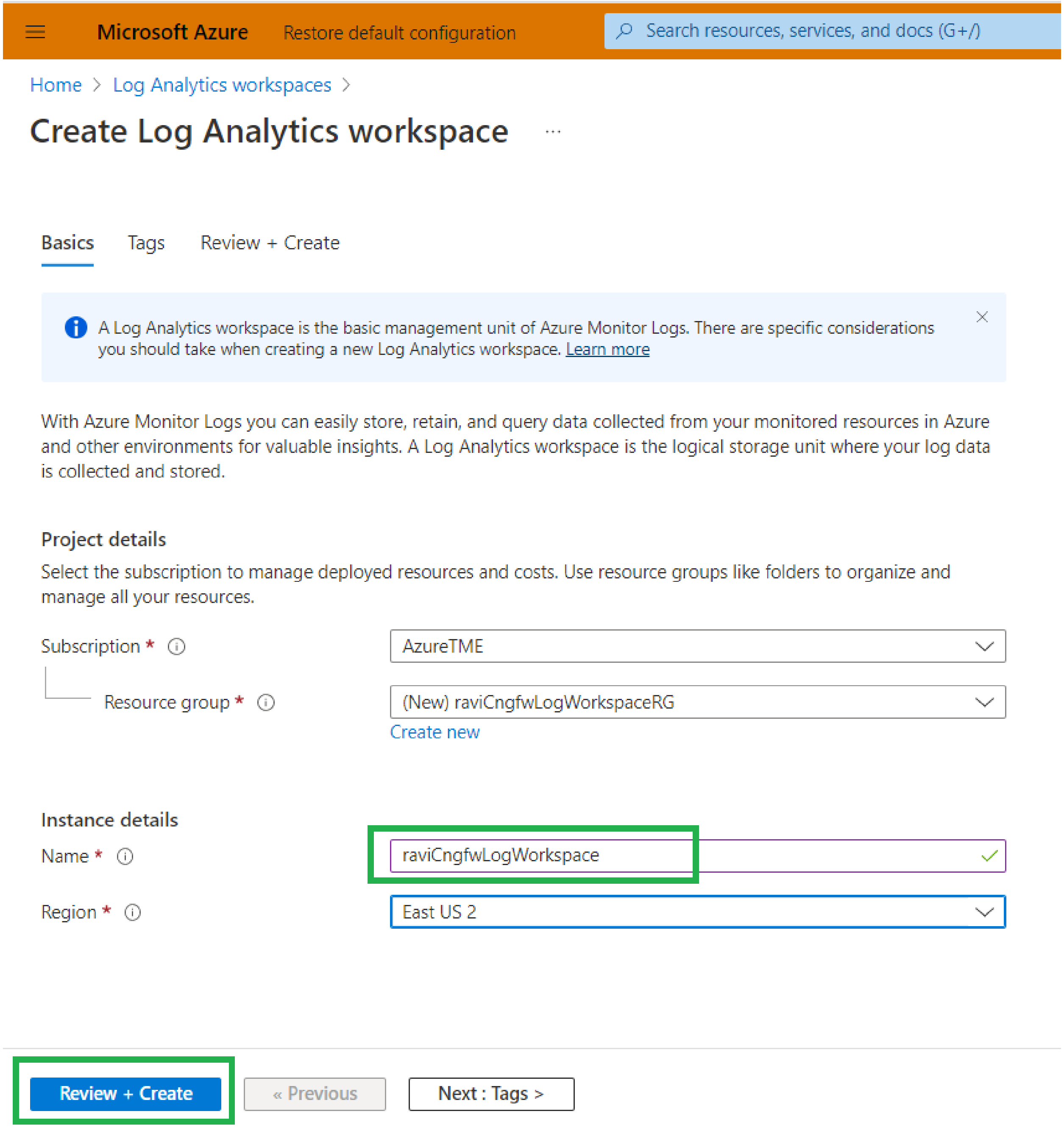


log analytics workspace

2. Click on Create option to create a new Log Analytics Workspace.



3. Create the **Log analytics workspace** as shown below. Make sure that the region is either US-East-2 or US-central:


Microsoft Azure
Restore default configuration
Search resources, services, and docs (G+/)
Home > Log Analytics workspaces >
Basics
Tags
Review + Create
Review + Create
« Previous
Next : Tags >

Create Log Analytics workspace ...

A Log Analytics workspace is the basic management unit of Azure Monitor Logs. There are specific considerations you should take when creating a new Log Analytics workspace. Learn more

With Azure Monitor Logs you can easily store, retain, and query data collected from your monitored resources in Azure and other environments for valuable insights. A Log Analytics workspace is the logical storage unit where your log data is collected and stored.

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * : AzureTME

Resource group * : (New) raviCngfwLogWorkspaceRG

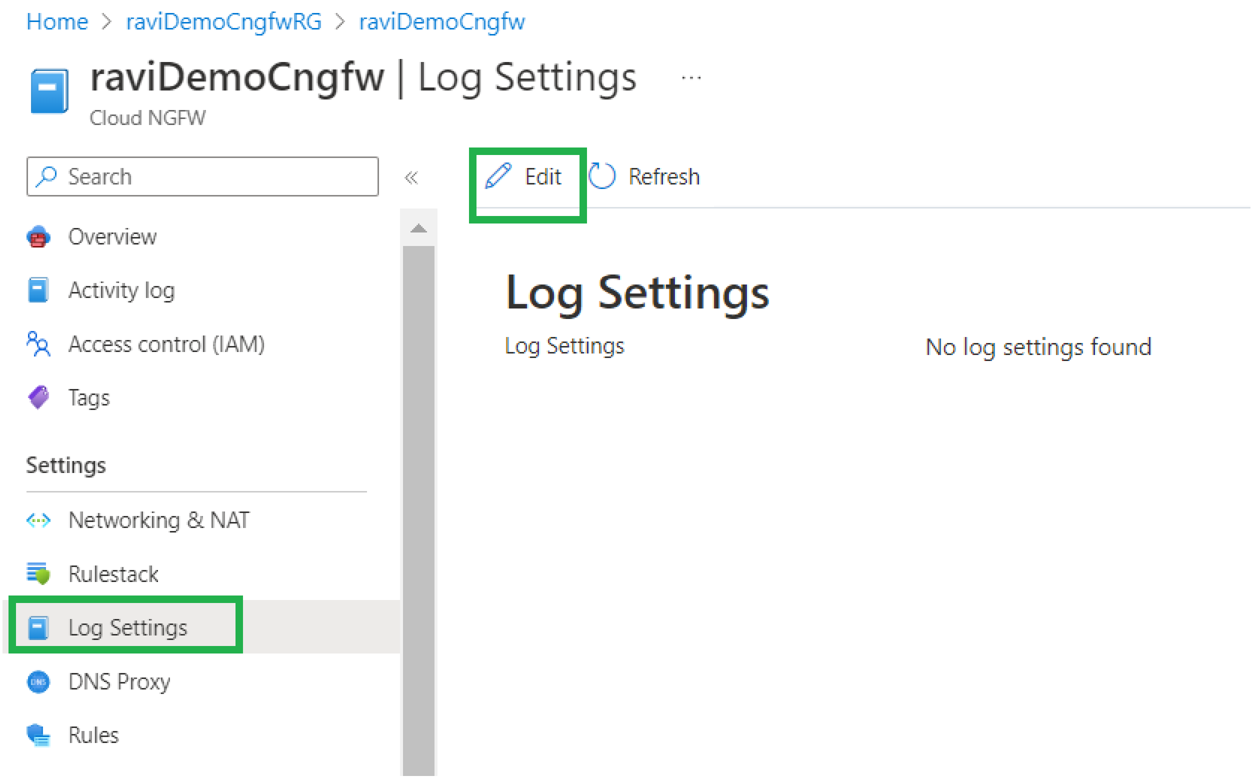
Create new

Instance details

Name * : raviCngfwLogWorkspace

Region * : East US 2

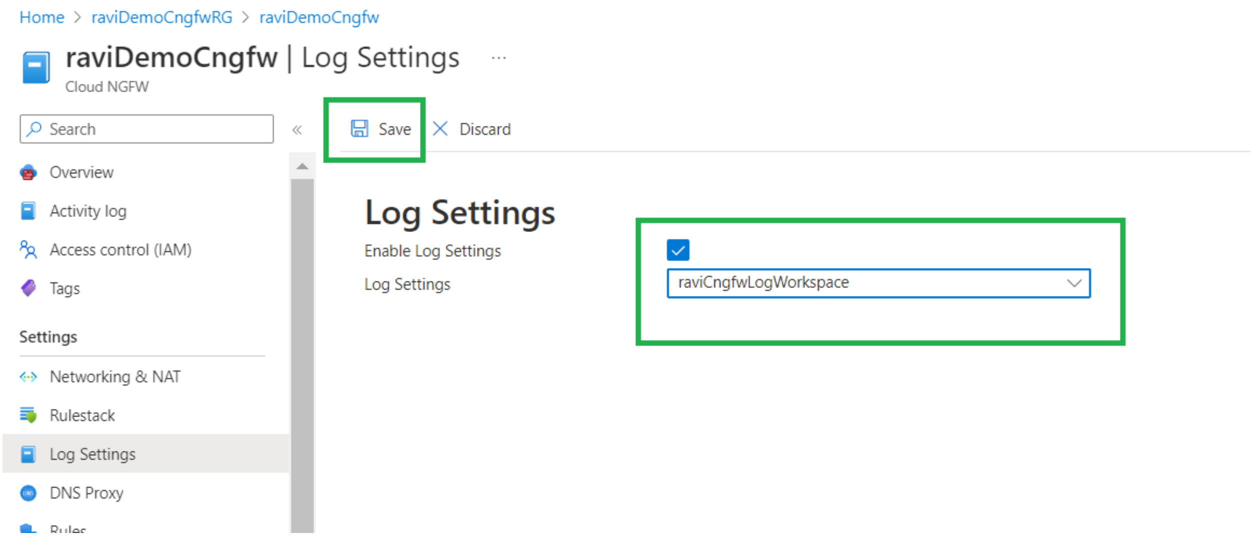
4. Now configure Cloud NGFW Log settings using the Log Analytics workspace created above. Go to the Cloud NGFW resource, select the **Log Settings** section, and click the **Edit** option to choose the Log analytics workspace that has just been created:



Log Settings

Log Settings: No log settings found

5. Enable **Log Settings** and choose the log analytics workspace created in the previous step from the drop-down, and save the configuration:



Log Settings

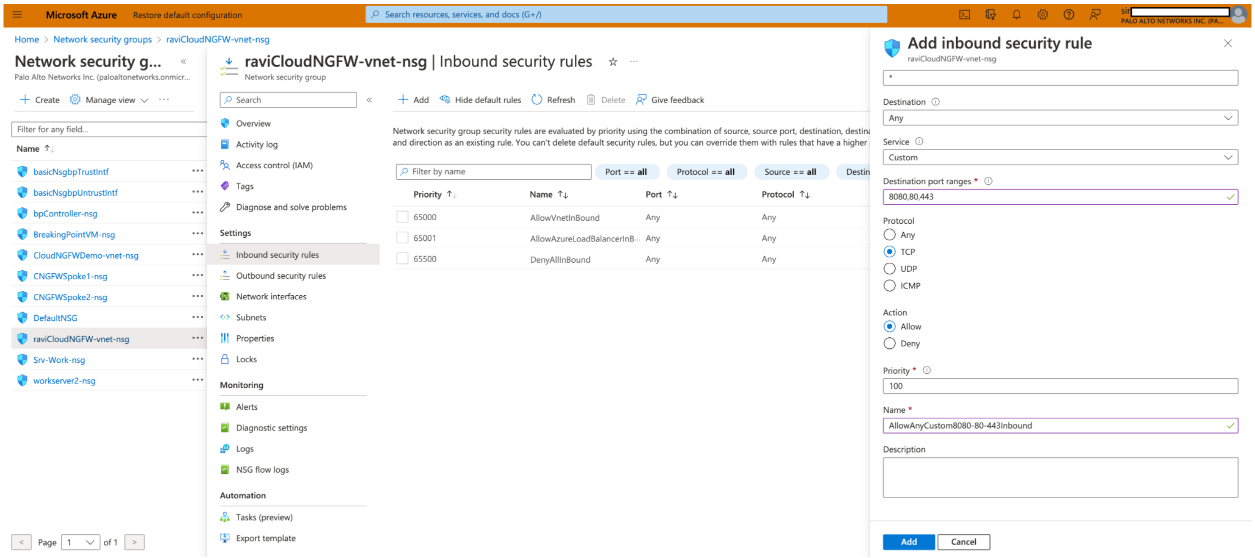
Enable Log Settings ☑

Log Settings: raviCngfwLogWorkspace

Update Network Security Group

Next, update the network security group that was created as part of the Cloud NGFW deployment. This security group is associated with both the Private and Public subnet as part of Hub VNetin the Customer Subscription (refer to the topology).

1. Allow traffic as per Frontend/Destination NAT rule configuration. Also, allow HTTP and HTTPS traffic so that the Internet can be accessed from Application VNets through Cloud NGFW:



Network security group security rules are evaluated by priority using the combination of source, source port, destination, destina and direction as an existing rule. You can't delete default security rules, but you can override them with rules that have a higher

Priority	Name	Port	Protocol
65000	AllowVnetInBound	Any	Any
65001	AllowAzureLoadBalancerInB...	Any	Any
65500	DenyAllInBound	Any	Any

Add inbound security rule
raviCloudNGFW-vnet-nsg

*

Destination: Any

Service: Custom

Destination port ranges *: 8080,80,443

Protocol
○ Any
● TCP
○ UDP
○ ICMP

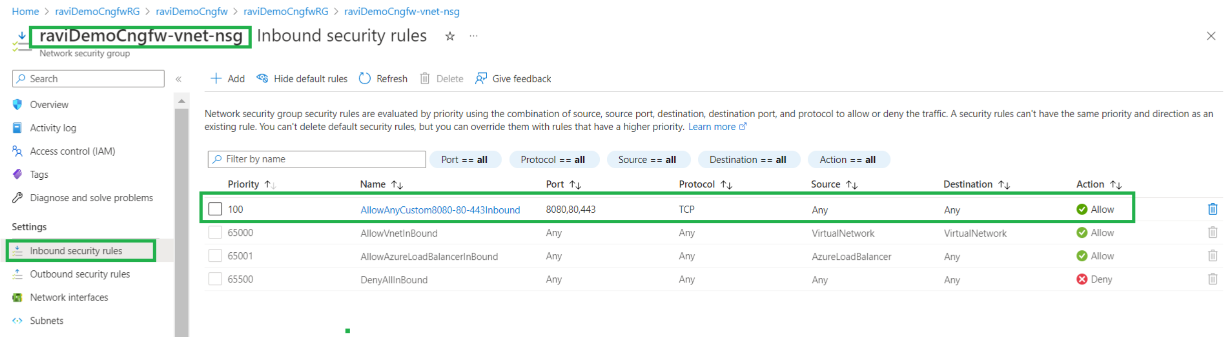
Action
● Allow
○ Deny

Priority *: 100

Name *: AllowAnyCustom8080-80-443Inbound

Description:

2. Click Add to incorporate this inbound security rule:



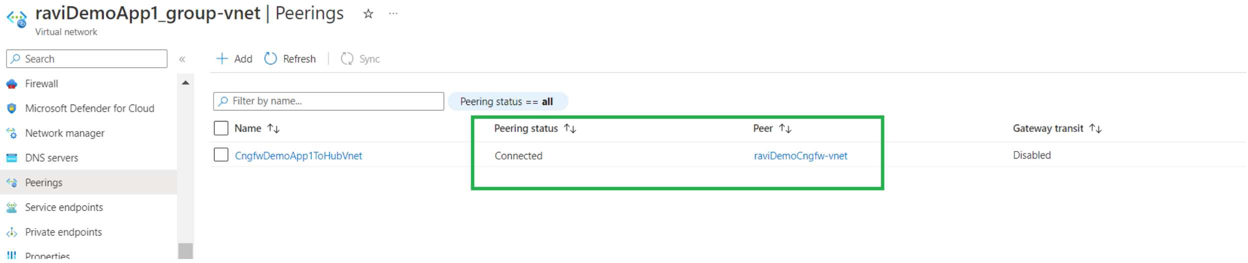
Network security group security rules are evaluated by priority using the combination of source, source port, destination, destination port, and protocol to allow or deny the traffic. A security rules can't have the same priority and direction as an existing rule. You can't delete default security rules, but you can override them with rules that have a higher priority. Learn more

Priority	Name	Port	Protocol	Source	Destination	Action
100	AllowAnyCustom8080-80-443Inbound	8080,80,443	TCP	Any	Any	✓ Allow
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	✓ Allow
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	✓ Allow
65500	DenyAllInBound	Any	Any	Any	Any	✗ Deny

These steps ensure that app1 on spoke VNet1 can access the Internet.

Configure VNetpeering between Hub Vnet(that got created during Cloud NGFW creation) and Spoke Vnets

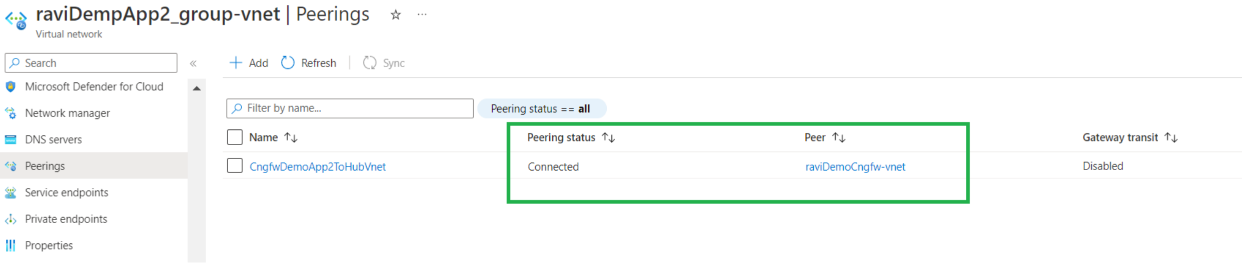
1. Configure VNet peering between spoke VNet1 and Hub VNet. Search for the spoke-VNet1 and select the **Peerings** section. Click **Add** to create a new peering.
2. While adding a peer, give it a name and leave the rest to default settings. Choose the hub virtual network (as peering is from spoke-vnet1) that has to be peered.



raviDemoApp1_group-vnet | Peerings

Name	Peering status	Peer	Gateway transit
CngfwDemoApp1ToHubVnet	Connected	raviDemoCngfw-vnet	Disabled

3. Configure VNetpeering between spoke-VNet2 and Hub VNet by repeating the steps above.

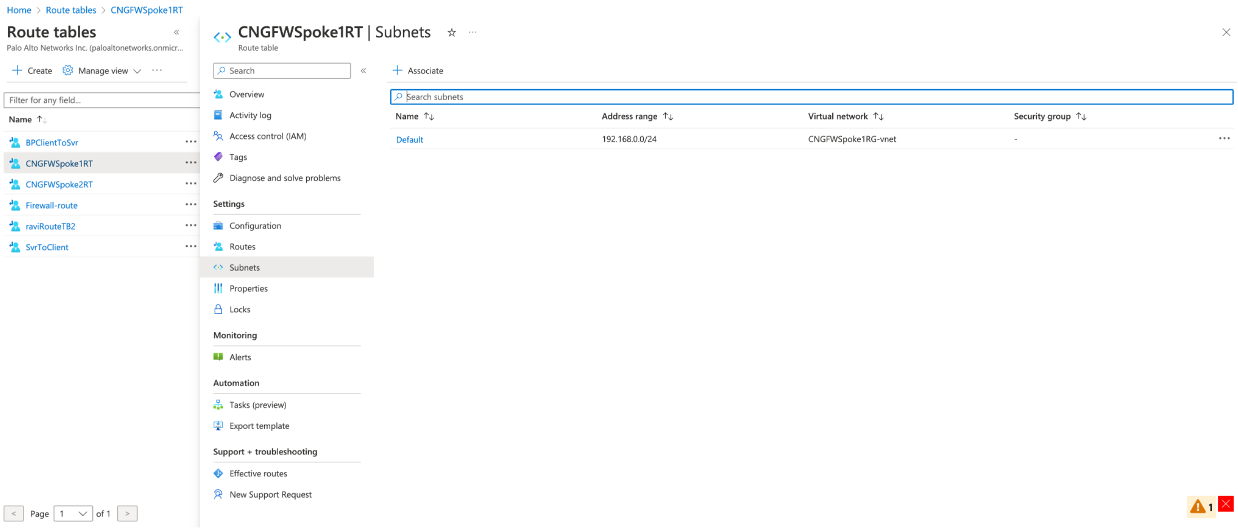


raviDempApp2_group-vnet | Peerings

Name	Peering status	Peer	Gateway transit
CngfwDemoApp2ToHubVnet	Connected	raviDemoCngfw-vnet	Disabled

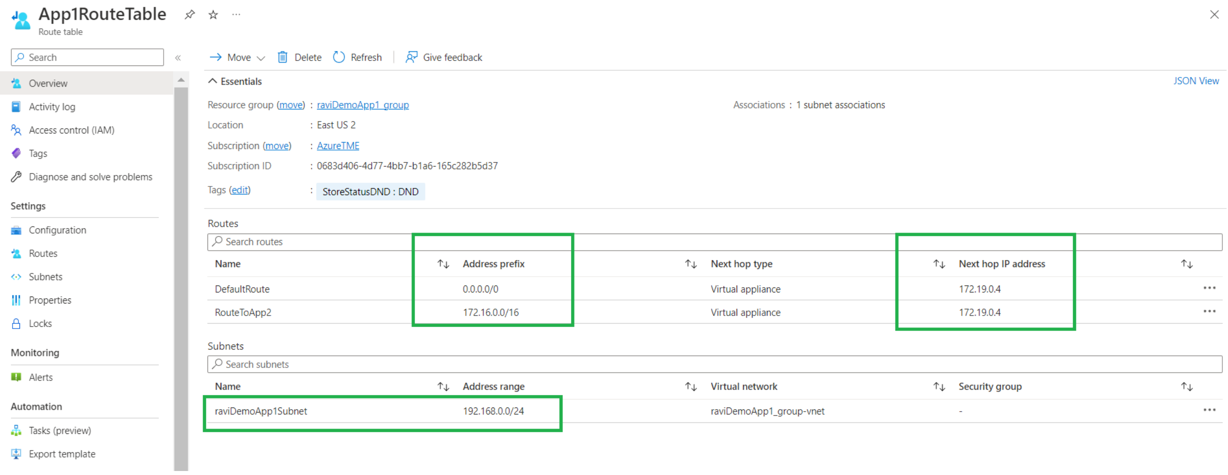
Add a Route Table to route traffic through Cloud NGFW

1. Create a route table by searching for the **Route table** in the Azure search bar. Then, click **Create** to establish a new route table. Fill in the necessary fields and click **Review+create** to create a route table.
2. After creating the route table, select the **Subnets** section and associate it with the App1 subnet from spoke-vnet1:



Name	Address range	Virtual network	Security group
Default	192.168.0.0/24	CNGFWSpoke1RG-vnet	-

3. Configure the default route (for outbound traffic) and route towards the App2 subnet (for east-west traffic) with the next hop as the Cloud NGFW private IP address:



Resource group (move) : raviDemoApp1_group
Location : East US 2
Subscription (move) : AzureTME
Subscription ID : 0683d406-4d77-4bb7-b1a6-165c282b5d37
Tags (edit) : StoreStatusDND : DND
Associations : 1 subnet associations

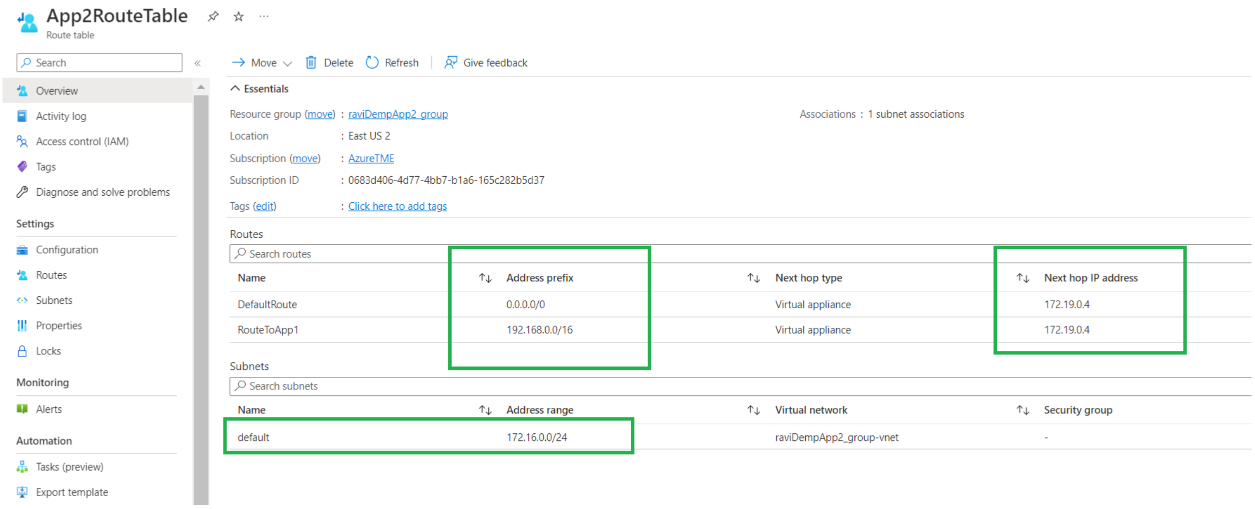
Routes

Name	Address prefix	Next hop type	Next hop IP address
DefaultRoute	0.0.0.0/0	Virtual appliance	172.19.0.4
RouteToApp2	172.16.0.0/16	Virtual appliance	172.19.0.4

Subnets

Name	Address range	Virtual network	Security group
raviDemoApp1Subnet	192.168.0.0/24	raviDemoApp1_group-vnet	-

4. Similarly, associate one more route table with the App2 subnet from spoke-VNet2. Configure a default route (for outbound traffic) and route towards the App1 subnet (for east-west traffic) with the next hop as the Cloud NGFW private IP address:



App2RouteTable

Route table

Resource group (move) : raviDempApp2_group

Location : East US 2

Subscription (move) : AzureTME

Subscription ID : 0683d406-4d77-4bb7-b1a6-165c282b5d37

Tags (edit) : Click here to add tags

Associations : 1 subnet associations

Routes

Name	Address prefix	Next hop type	Next hop IP address
DefaultRoute	0.0.0.0/0	Virtual appliance	172.19.0.4
RouteToApp1	192.168.0.0/16	Virtual appliance	172.19.0.4

Subnets

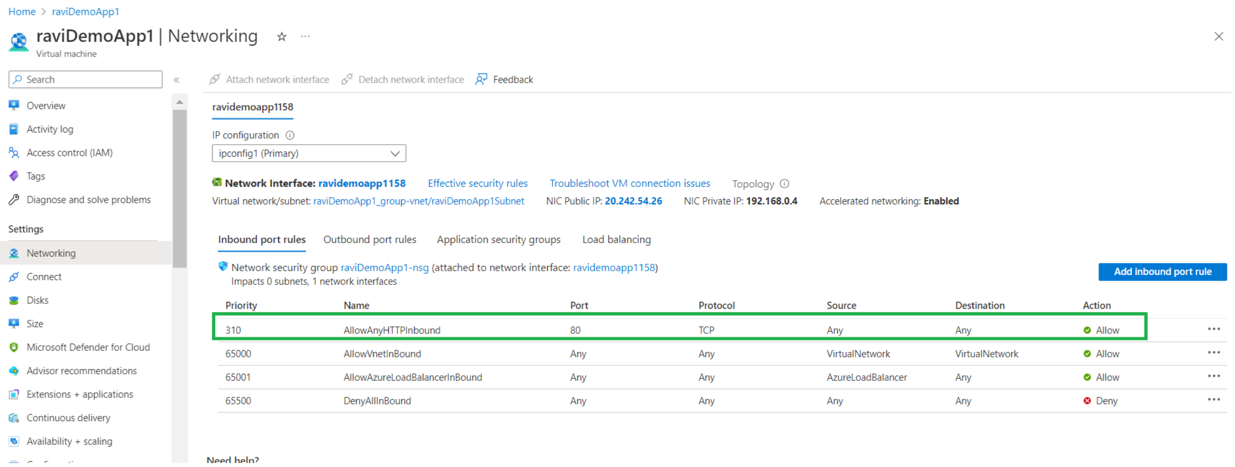
Name	Address range	Virtual network	Security group
default	172.16.0.0/24	raviDempApp2_group-vnet	-

Testing traffic

Test Inbound Traffic

1. To validate the inbound connection towards App1, try to access http://<Cloud NGFW Public IP>:8080.
2. As per the Destination NAT configuration on Cloud NGFW, if http://<Cloud NGFW Public IP>:8080 is accessed, the connection will be redirected to App1 after inspection by Cloud NGFW.

Make sure to allow HTTP traffic on the application server network interface. For this, go to App1, select **Networking**, and add an inbound port rule that allows any HTTP inbound traffic. Configure the source as **IP Addresses**, port as 80, protocol as TCP, and set the **Action** to**Allow:**



Home > raviDemoApp1

raviDemoApp1 | Networking ☆ ···

Virtual machine

ravidemoapp1158

IP configuration ⓘ

ipconfig1 (Primary)

Network Interface: ravidemoapp1158 Effective security rules Troubleshoot VM connection issues Topology ⓘ

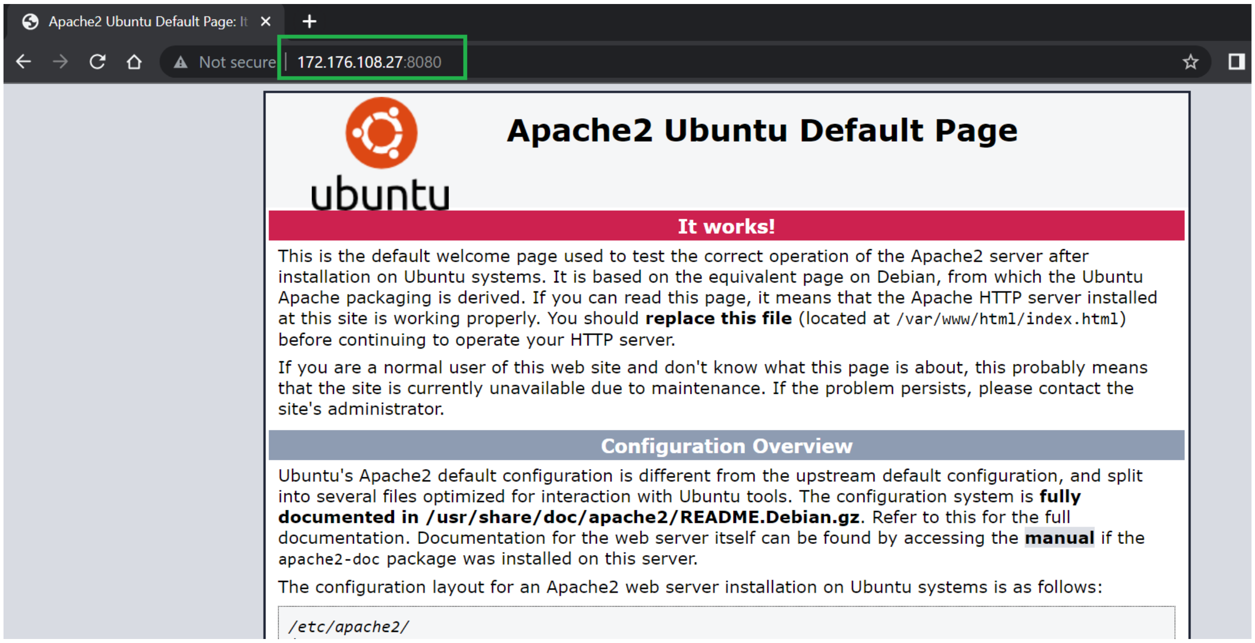
Virtual network/subnet: raviDemoApp1_group-vnet/raviDemoApp1Subnet NIC Public IP: **20.242.54.26** NIC Private IP: **192.168.0.4** Accelerated networking: **Enabled**

Network security group raviDemoApp1-nsg (attached to network interface: ravidemoapp1158)
Impacts 0 subnets, 1 network interfaces

Priority	Name	Port	Protocol	Source	Destination	Action
310	AllowAnyHTTPInbound	80	TCP	Any	Any	Allow
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	Allow
65500	DenyAllInBound	Any	Any	Any	Any	Deny

Need help?

If http:/<Cloud NGFW Public IP>:8080/ is accessed, the following screen appears if the apache server was running as the default web server. Here, once the public IP of the Cloud NGFW was accessed, it will redirect the traffic to App1 on spoke-vnet1 where apache server was running. Since Inbound HTTP is enabled on App1, it will run the apache server that was deployed on App1.



Apache2 Ubuntu Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

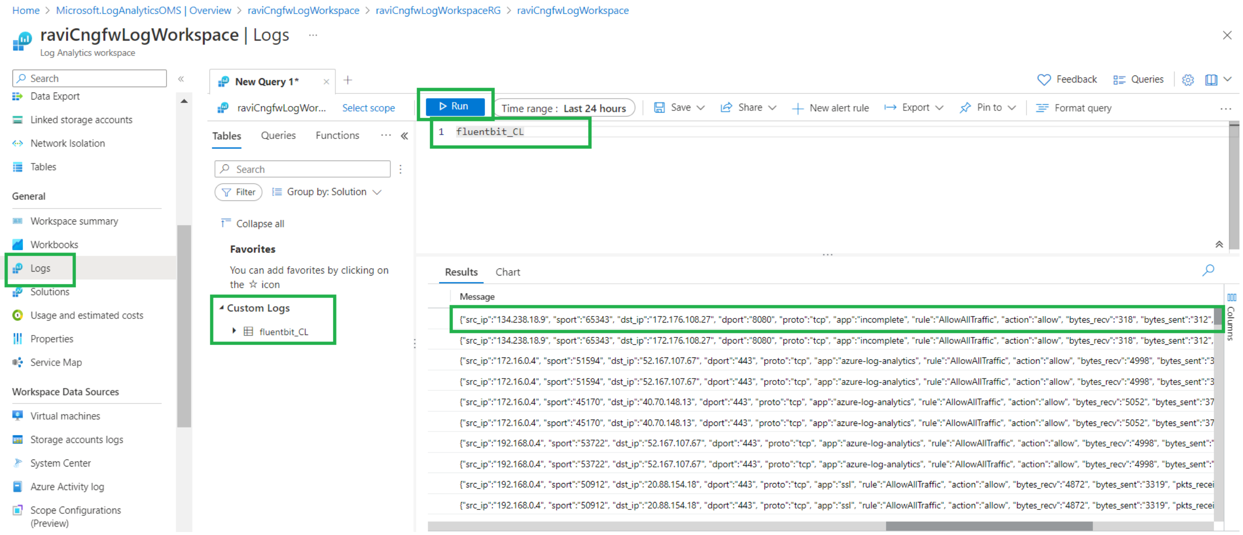
The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
```

Accessing logs

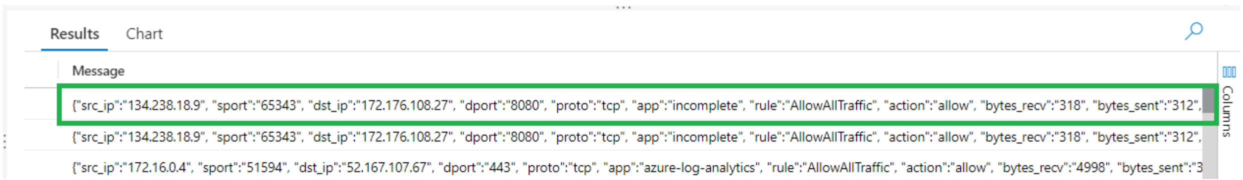
To verify that this particular inbound traffic was processed correctly by Cloud NGFW, go to the **Log Analytics workspace** and verify the logs as shown below.

Within Log analytics workspace **raviCngfwLogWorkspace**, navigate to the **Logs** section, select **Custom Logs** and select **fluentbit_CL** and run the query to get the latest logs:



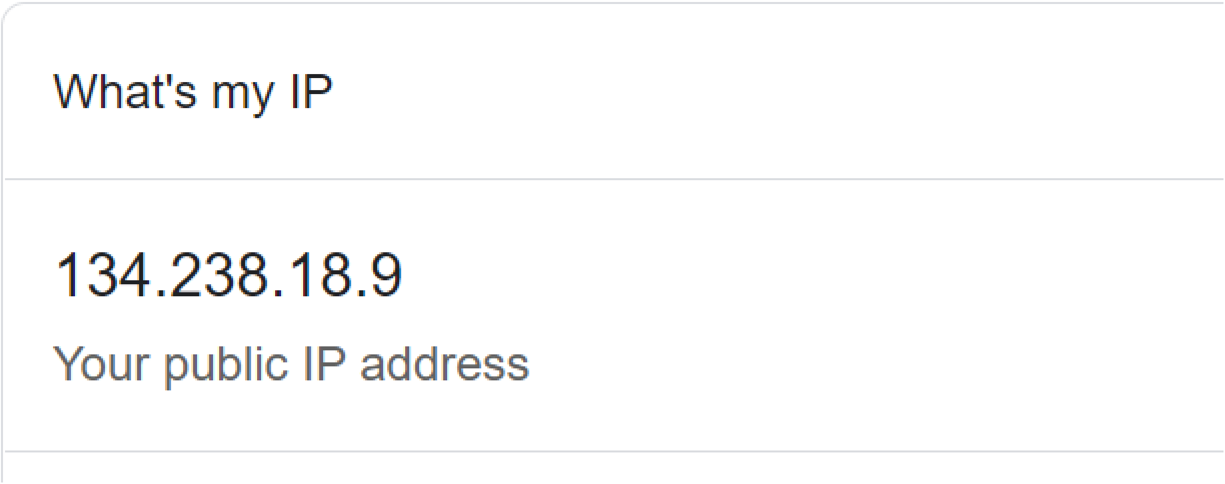
```
fluentbit_CL
```

```
Message
{"src_ip":"134.238.18.9", "sport":"65343", "dst_ip":"172.176.108.27", "dport":"8080", "proto":"tcp", "app":"incomplete", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"318", "bytes_sent":"312",
{"src_ip":"134.238.18.9", "sport":"65343", "dst_ip":"172.176.108.27", "dport":"8080", "proto":"tcp", "app":"incomplete", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"318", "bytes_sent":"312",
{"src_ip":"172.16.0.4", "sport":"51594", "dst_ip":"52.167.107.67", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4998", "bytes_sent":"3
{"src_ip":"172.16.0.4", "sport":"51594", "dst_ip":"52.167.107.67", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4998", "bytes_sent":"3
{"src_ip":"172.16.0.4", "sport":"45170", "dst_ip":"40.70.148.13", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"5052", "bytes_sent":"37
{"src_ip":"172.16.0.4", "sport":"45170", "dst_ip":"40.70.148.13", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"5052", "bytes_sent":"37
{"src_ip":"192.168.0.4", "sport":"53722", "dst_ip":"52.167.107.67", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4998", "bytes_sent":"
{"src_ip":"192.168.0.4", "sport":"53722", "dst_ip":"52.167.107.67", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4998", "bytes_sent":"
{"src_ip":"192.168.0.4", "sport":"50912", "dst_ip":"20.88.154.18", "dport":"443", "proto":"tcp", "app":"ssl", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4872", "bytes_sent":"3319", "pkts_recei
{"src_ip":"192.168.0.4", "sport":"50912", "dst_ip":"20.88.154.18", "dport":"443", "proto":"tcp", "app":"ssl", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4872", "bytes_sent":"3319", "pkts_recei
```



```
Message
{"src_ip":"134.238.18.9", "sport":"65343", "dst_ip":"172.176.108.27", "dport":"8080", "proto":"tcp", "app":"incomplete", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"318", "bytes_sent":"312",
{"src_ip":"134.238.18.9", "sport":"65343", "dst_ip":"172.176.108.27", "dport":"8080", "proto":"tcp", "app":"incomplete", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"318", "bytes_sent":"312",
{"src_ip":"172.16.0.4", "sport":"51594", "dst_ip":"52.167.107.67", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4998", "bytes_sent":"3
```

From the log, it can be seen that the source IP address is the IP address of the machine from which the request originated, and the destination IP address is Cloud NGFW public IP address, and it's hitting the **AllowAllTraffic** rule that has been created in the rulestack. The screenshot below shows the IP address of the machine from which the request originated:



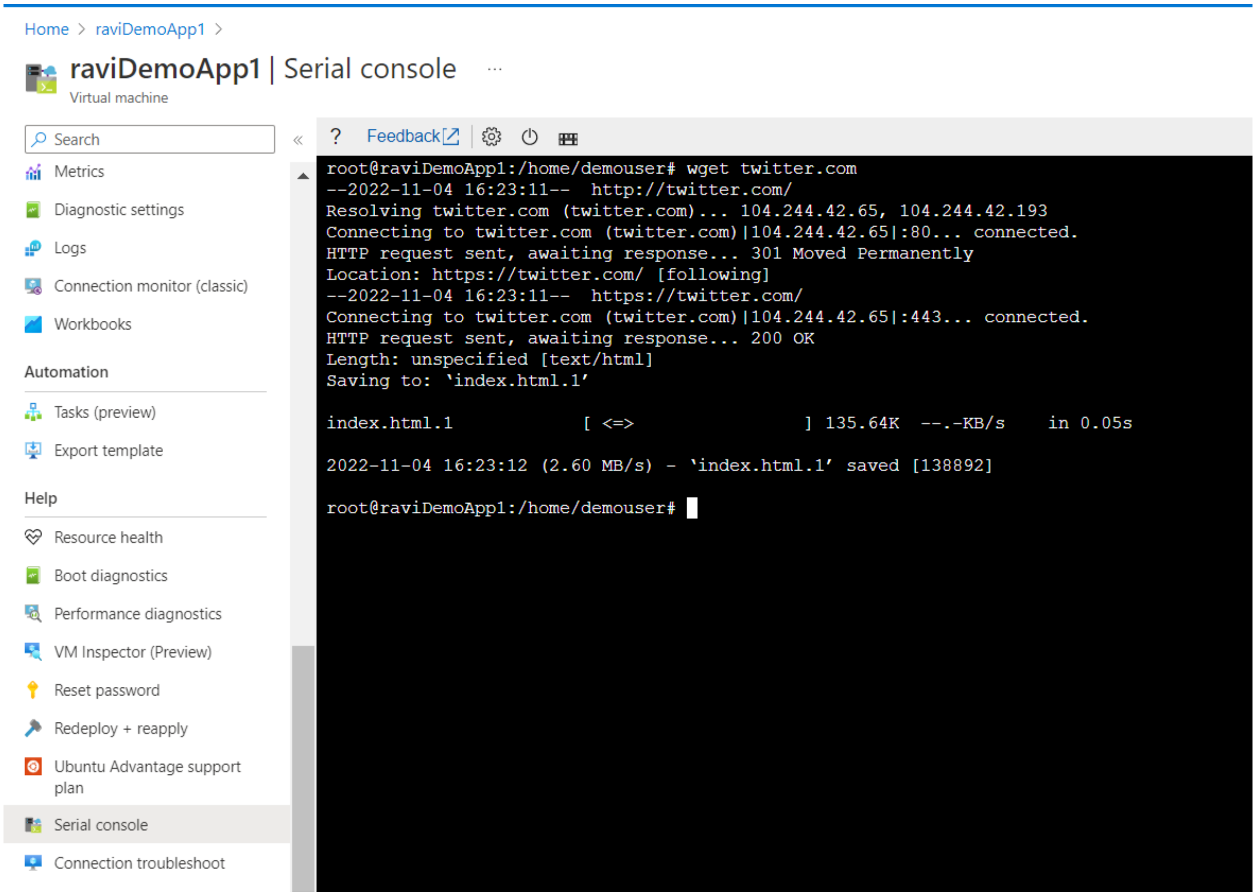
What's my IP

134.238.18.9

Your public IP address

Test Outbound Traffic

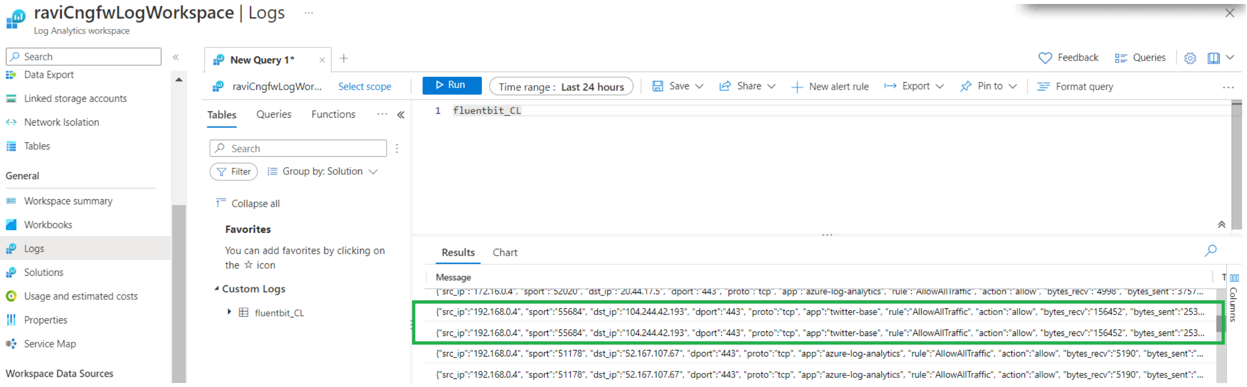
To validate the outbound connection, try to access twitter.com from App1 as shown below. Go to App1, select the **Serial console** section and type the following command:
Wget twitter.com



```
root@raviDemoApp1:/home/demouser# wget twitter.com
--2022-11-04 16:23:11--  http://twitter.com/
Resolving twitter.com (twitter.com)... 104.244.42.65, 104.244.42.193
Connecting to twitter.com (twitter.com)|104.244.42.65|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://twitter.com/ [following]
--2022-11-04 16:23:11--  https://twitter.com/
Connecting to twitter.com (twitter.com)|104.244.42.65|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: ‘index.html.1’

index.html.1            [ <=>                ] 135.64K  --.-KB/s    in 0.05s

2022-11-04 16:23:12 (2.60 MB/s) - ‘index.html.1’ saved [138892]

root@raviDemoApp1:/home/demouser#
```

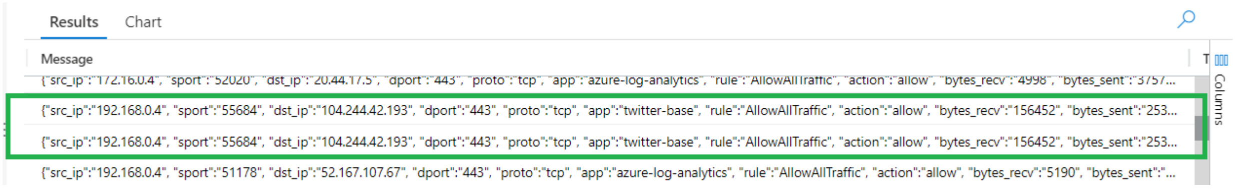
The connection has been established. Verify that this traffic is being processed by Cloud NGFW by going to the **Log Analytics workspace**. Repeat the steps to access logs.

Run the query again to get the latest logs.



```
1  fluentbit_CL
```

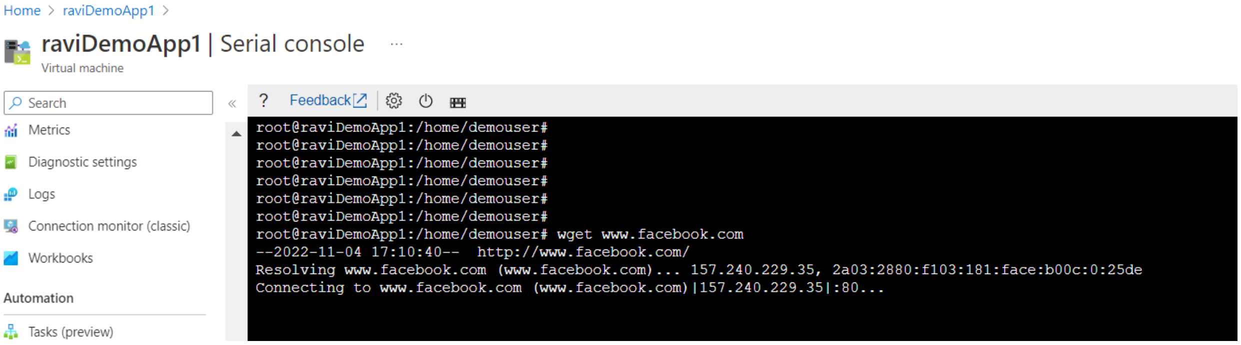
```
{"src_ip":"172.16.0.4", "sport":"52020", "dst_ip":"20.44.17.5", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4998", "bytes_sent":"3757...
{"src_ip":"192.168.0.4", "sport":"55684", "dst_ip":"104.244.42.193", "dport":"443", "proto":"tcp", "app":"twitter-base", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"156452", "bytes_sent":"253...
{"src_ip":"192.168.0.4", "sport":"55684", "dst_ip":"104.244.42.193", "dport":"443", "proto":"tcp", "app":"twitter-base", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"156452", "bytes_sent":"253...
{"src_ip":"192.168.0.4", "sport":"51178", "dst_ip":"52.167.107.67", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"5190", "bytes_sent":"...
{"src_ip":"192.168.0.4", "sport":"51178", "dst_ip":"52.167.107.67", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"5190", "bytes_sent":"...
```



```
{"src_ip":"172.16.0.4", "sport":"52020", "dst_ip":"20.44.17.5", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4998", "bytes_sent":"3757...
{"src_ip":"192.168.0.4", "sport":"55684", "dst_ip":"104.244.42.193", "dport":"443", "proto":"tcp", "app":"twitter-base", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"156452", "bytes_sent":"253...
{"src_ip":"192.168.0.4", "sport":"55684", "dst_ip":"104.244.42.193", "dport":"443", "proto":"tcp", "app":"twitter-base", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"156452", "bytes_sent":"253...
{"src_ip":"192.168.0.4", "sport":"51178", "dst_ip":"52.167.107.67", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"5190", "bytes_sent":"...
```

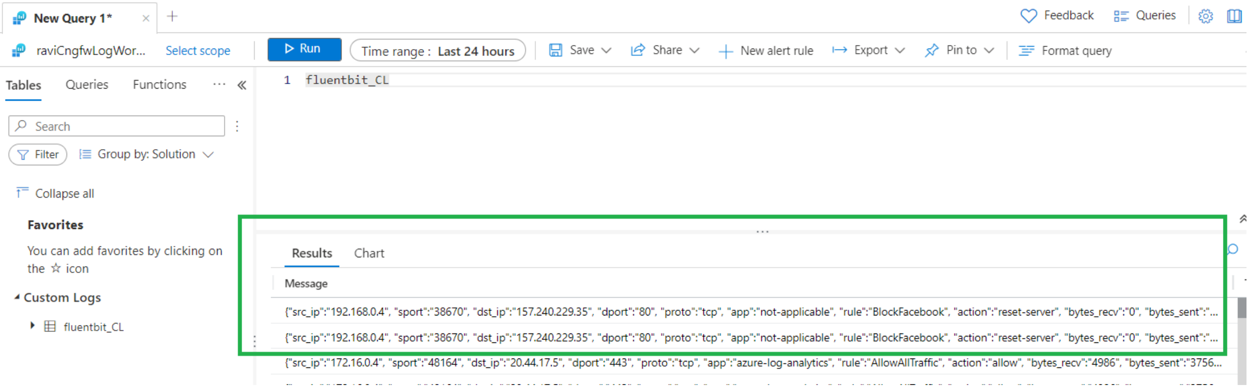
Test Outbound Block Rule

Now try to access Facebook. The traffic to Facebook should get blocked as per the rule configured. Go to App1, select**Serial console** and type the following command: *wget www.facebook.com*



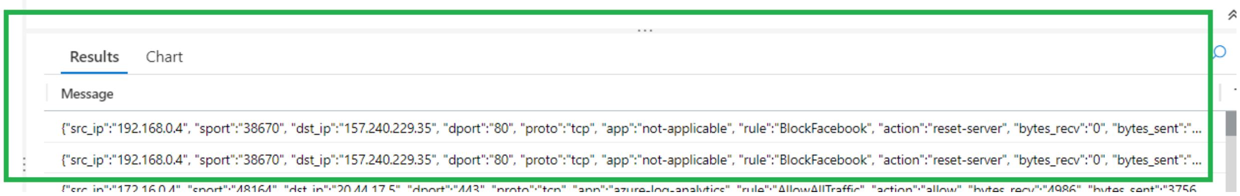
```
root@raviDemoApp1:/home/demouser#
root@raviDemoApp1:/home/demouser#
root@raviDemoApp1:/home/demouser#
root@raviDemoApp1:/home/demouser#
root@raviDemoApp1:/home/demouser#
root@raviDemoApp1:/home/demouser#
root@raviDemoApp1:/home/demouser# wget www.facebook.com
--2022-11-04 17:10:40--  http://www.facebook.com/
Resolving www.facebook.com (www.facebook.com)... 157.240.229.35, 2a03:2880:f103:181:face:b00c:0:25de
Connecting to www.facebook.com (www.facebook.com)|157.240.229.35|:80...
```

Connection won't be established. Go to **Azure Log Analytics** to validate that Cloud NGFW has blocked this connection as per the rRulestack configuration.



```
fluentbit_CL
```

```
{"src_ip":"192.168.0.4", "sport":"38670", "dst_ip":"157.240.229.35", "dport":"80", "proto":"tcp", "app":"not-applicable", "rule":"BlockFacebook", "action":"reset-server", "bytes_recv":"0", "bytes_sent":"...
{"src_ip":"192.168.0.4", "sport":"38670", "dst_ip":"157.240.229.35", "dport":"80", "proto":"tcp", "app":"not-applicable", "rule":"BlockFacebook", "action":"reset-server", "bytes_recv":"0", "bytes_sent":"...
{"src_ip":"172.16.0.4", "sport":"48164", "dst_ip":"20.44.17.5", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4986", "bytes_sent":"3756...
```



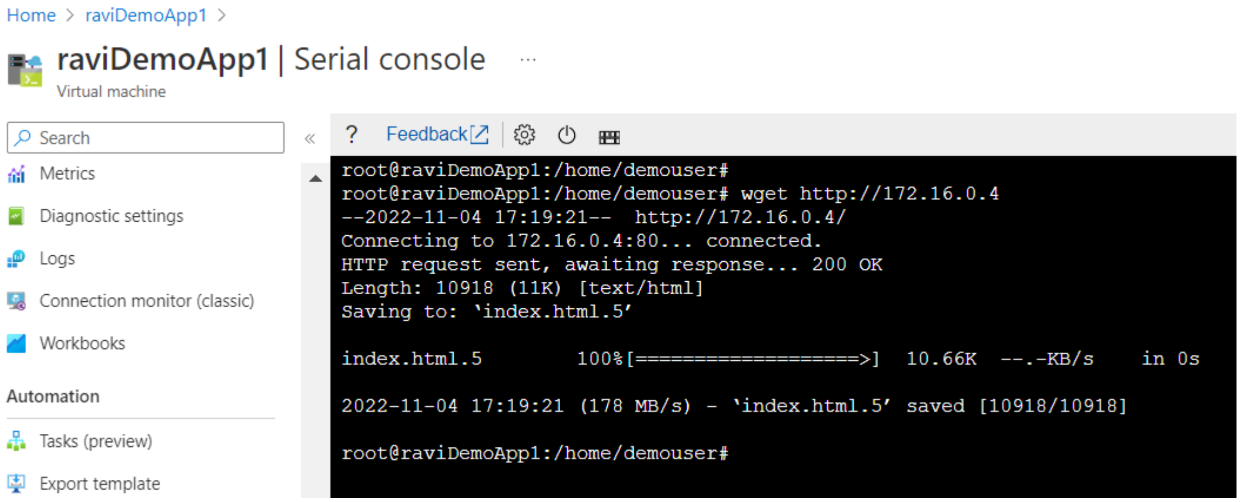
```
{"src_ip":"192.168.0.4", "sport":"38670", "dst_ip":"157.240.229.35", "dport":"80", "proto":"tcp", "app":"not-applicable", "rule":"BlockFacebook", "action":"reset-server", "bytes_recv":"0", "bytes_sent":"...
{"src_ip":"192.168.0.4", "sport":"38670", "dst_ip":"157.240.229.35", "dport":"80", "proto":"tcp", "app":"not-applicable", "rule":"BlockFacebook", "action":"reset-server", "bytes_recv":"0", "bytes_sent":"...
{"src_ip":"172.16.0.4", "sport":"48164", "dst_ip":"20.44.17.5", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4986", "bytes_sent":"3756
```

From these logs, it is evident that the traffic to Facebook was blocked after hitting the **BlockFacebook** rule. This confirms that Cloud NGFW is able to block traffic as per configured rulestack.

Test East-West Traffic flow

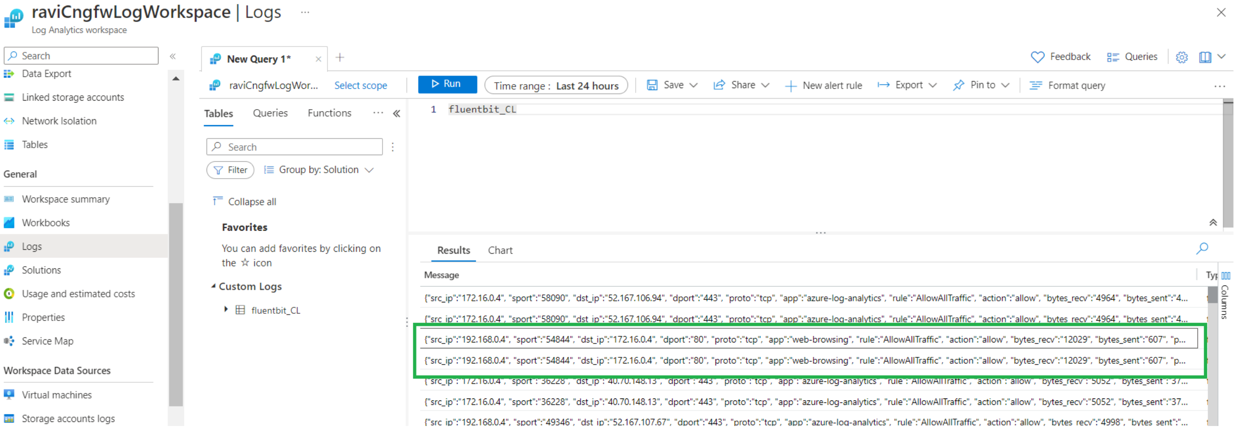
Validate east-west traffic flow by trying to send traffic from App1 to App2.

On App1, execute the following command:
wget http://<App2 IP address>



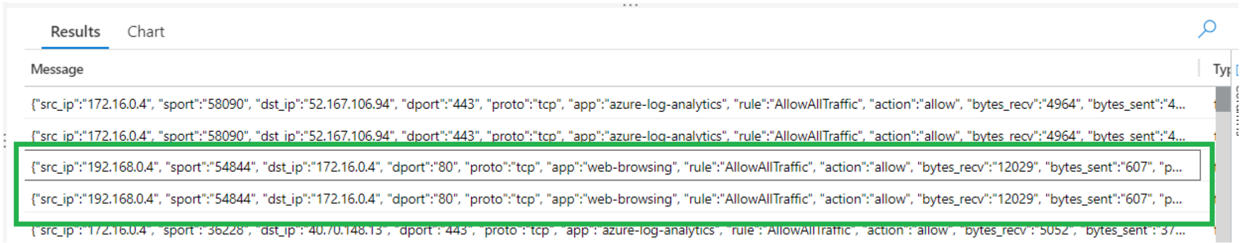
```
root@raviDemoApp1:/home/demouser#
root@raviDemoApp1:/home/demouser# wget http://172.16.0.4
--2022-11-04 17:19:21--  http://172.16.0.4/
Connecting to 172.16.0.4:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10918 (11K) [text/html]
Saving to: 'index.html.5'

index.html.5        100%[===================>]  10.66K  --.-KB/s    in 0s

2022-11-04 17:19:21 (178 MB/s) - 'index.html.5' saved [10918/10918]

root@raviDemoApp1:/home/demouser#
```

The connection has been established. Validate by going to the to **Azure Log Analytics** workspace:



```
1  fluentbit_CL
```

```
Message
{"src_ip":"172.16.0.4", "sport":"58090", "dst_ip":"52.167.106.94", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4964", "bytes_sent":"4...
{"src_ip":"172.16.0.4", "sport":"58090", "dst_ip":"52.167.106.94", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4964", "bytes_sent":"4...
{"src_ip":"192.168.0.4", "sport":"54844", "dst_ip":"172.16.0.4", "dport":"80", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"12029", "bytes_sent":"607", "p...
{"src_ip":"192.168.0.4", "sport":"54844", "dst_ip":"172.16.0.4", "dport":"80", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"12029", "bytes_sent":"607", "p...
{"src_ip":"172.16.0.4", "sport":"36228", "dst_ip":"40.70.148.13", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"5052", "bytes_sent":"37...
{"src_ip":"172.16.0.4", "sport":"36228", "dst_ip":"40.70.148.13", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"5052", "bytes_sent":"37...
{"src_ip":"192.168.0.4", "sport":"49346", "dst_ip":"52.167.107.67", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4998", "bytes_sent":"...
```



```
Message
{"src_ip":"172.16.0.4", "sport":"58090", "dst_ip":"52.167.106.94", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4964", "bytes_sent":"4...
{"src_ip":"172.16.0.4", "sport":"58090", "dst_ip":"52.167.106.94", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"4964", "bytes_sent":"4...
{"src_ip":"192.168.0.4", "sport":"54844", "dst_ip":"172.16.0.4", "dport":"80", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"12029", "bytes_sent":"607", "p...
{"src_ip":"192.168.0.4", "sport":"54844", "dst_ip":"172.16.0.4", "dport":"80", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"12029", "bytes_sent":"607", "p...
{"src_ip":"172.16.0.4", "sport":"36228", "dst_ip":"40.70.148.13", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow", "bytes_recv":"5052", "bytes_sent":"37...
```

From these logs, it is visible that the traffic sent between App1 (192.168.0.4) and App2 (172.16.0.4) is going through the Cloud NGFW service and hitting the **AllowAllTraffic** rule which is part of the local rulestack.

Thus the inbound, outbound, and east-west traffic has been tested and is flowing through the Cloud NGFW service.

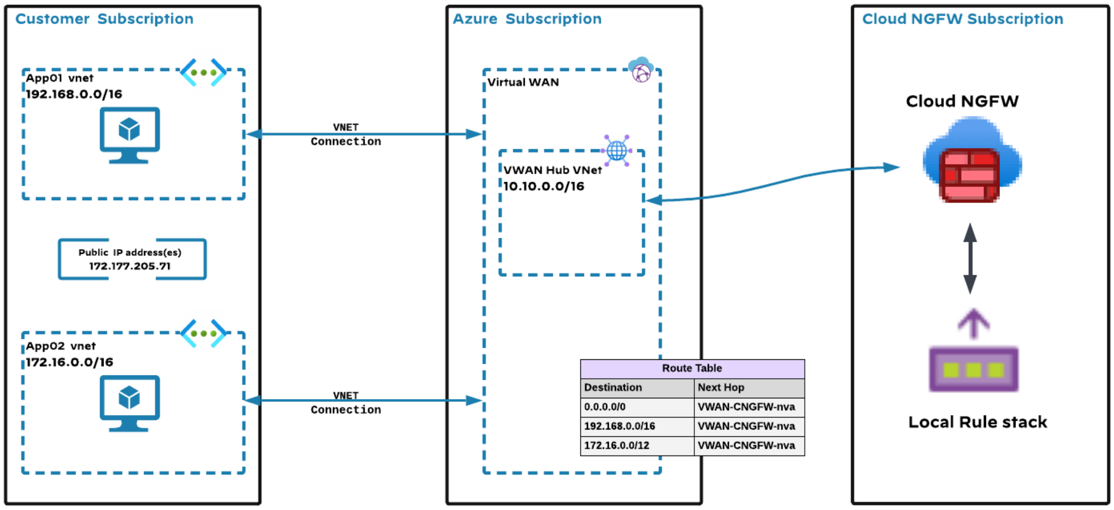
Integrate Cloud NGFW into Azure Virtual WAN(VWAN) Infrastructure

Pre-deployment of Cloud NGFW - setting up the environment

Topology

A hub-spoke topology is used as an example to route traffic through Cloud NGFW. Cloud NGFW supports all topologies.

Integrate Cloud NGFW into Azure Virtual WAN(VWAN)



Route Table

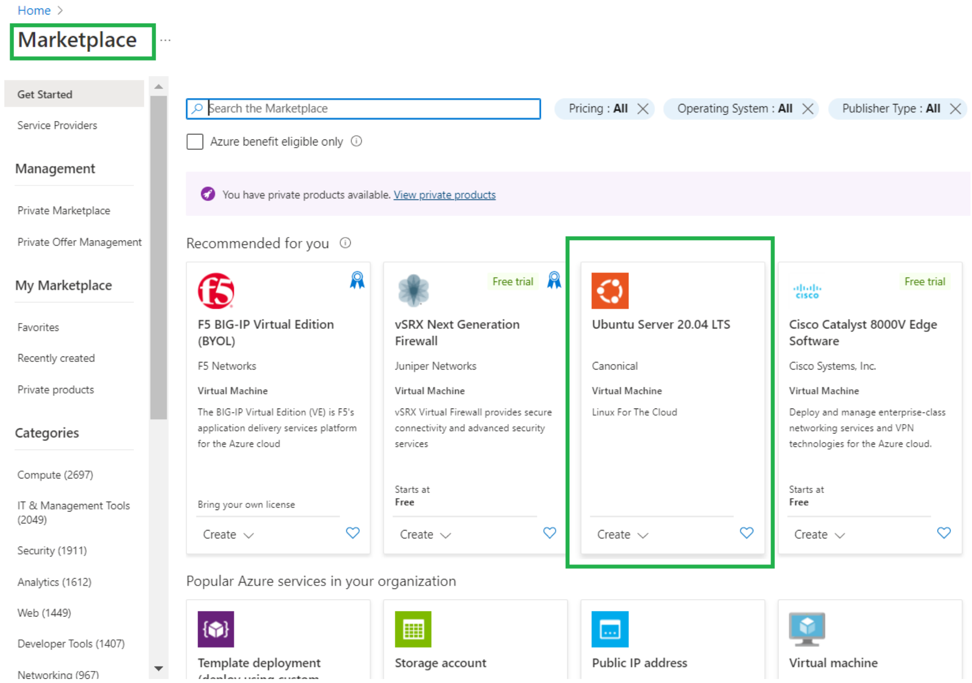
Destination	Next Hop
0.0.0.0/0	VWAN-CNGFW-nva
192.168.0.0/16	VWAN-CNGFW-nva
172.16.0.0/12	VWAN-CNGFW-nva

As per the above topology, in order to set up a lab environment, there should be a Azure Virtual WAN with VWAN Hub, 2 spoke VNets and a virtual machine on each of those spoke VNets that's running a web server (apache2). Create this environment before creating and deploying the Cloud NGFW resources.

Create Spoke VNets with a virtual machine on each of them

Create Spoke App01 VNet with Ubuntu Server

1. Go to Azure Marketplace and search for "*Ubuntu*" Server as shown below:



You have private products available. View private products

Recommended for you

F5 BIG-IP Virtual Edition (BYOL)
F5 Networks
Virtual Machine
The BIG-IP Virtual Edition (VE) is F5's application delivery services platform for the Azure cloud
Bring your own license

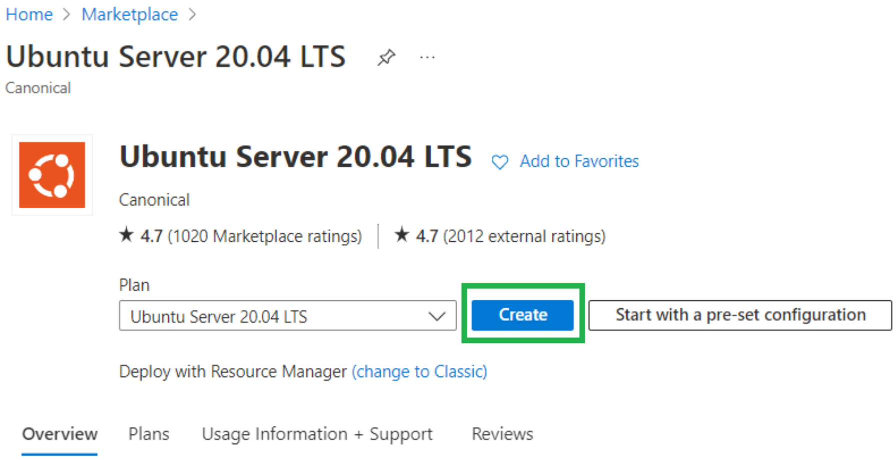
vSRX Next Generation Firewall
Juniper Networks
Virtual Machine
vSRX Virtual Firewall provides secure connectivity and advanced security services
Starts at Free

Ubuntu Server 20.04 LTS
Canonical
Virtual Machine
Linux For The Cloud

Cisco Catalyst 8000V Edge Software
Cisco Systems, Inc.
Virtual Machine
Deploy and manage enterprise-class networking services and VPN technologies for the Azure cloud.
Starts at Free

Popular Azure services in your organization

2. Choose this Ubuntu server and click 'Create' to start the creation of the Ubuntu server:



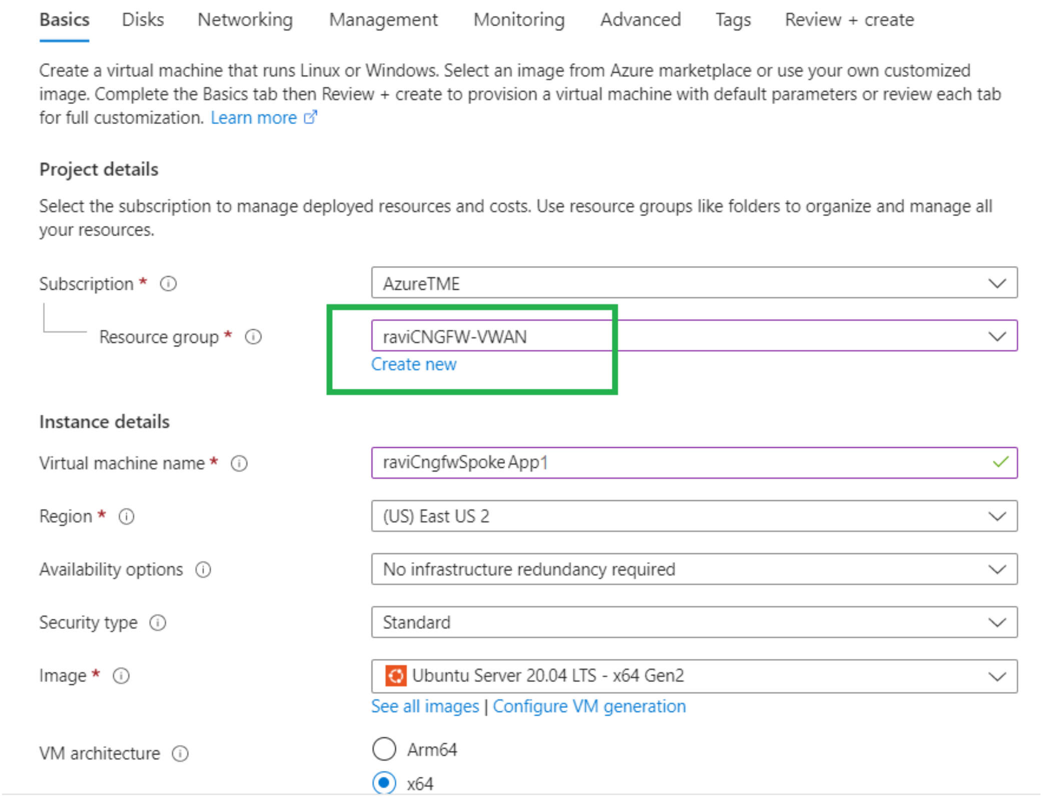
Ubuntu Server 20.04 LTS
Canonical

Ubuntu Server 20.04 LTS
Canonical
★ 4.7 (1020 Marketplace ratings) | ★ 4.7 (2012 external ratings)

Plan: Ubuntu Server 20.04 LTS

Deploy with Resource Manager (change to Classic)

3. Fill in the details (Resource Group, VM Name, Region and the type of image while leaving other fields to default.) to complete the creation of the Ubuntu server.



Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. Learn more

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * : AzureTME

Resource group * : raviCNGFW-VWAN

Create new

Instance details

Virtual machine name * : raviCngfwSpoke App1 ✓

Region * : (US) East US 2

Availability options : No infrastructure redundancy required

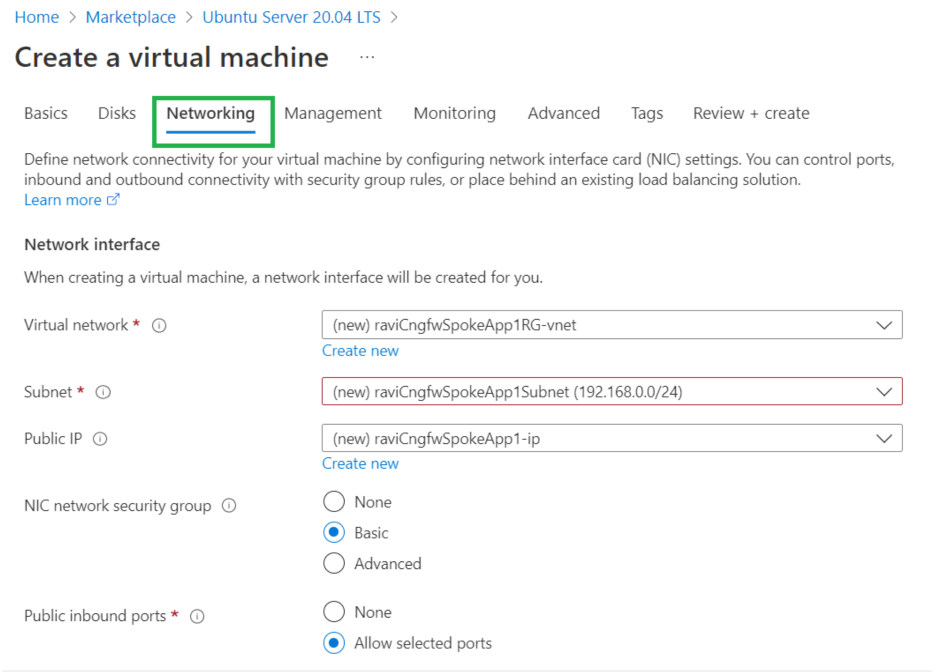
Security type : Standard

Image * : Ubuntu Server 20.04 LTS - x64 Gen2

See all images | Configure VM generation

VM architecture : ○ Arm64 ● x64

4. In the networking section, select an existing VNet or create a new one in which this Ubuntu server will be installed:


Home > Marketplace > Ubuntu Server 20.04 LTS >

Create a virtual machine ...

Define network connectivity for your virtual machine by configuring network interface card (NIC) settings. You can control ports, inbound and outbound connectivity with security group rules, or place behind an existing load balancing solution.
Learn more

Network interface

When creating a virtual machine, a network interface will be created for you.

Virtual network * : (new) raviCngfwSpokeApp1RG-vnet
Create new

Subnet * : (new) raviCngfwSpokeApp1Subnet (192.168.0.0/24)

Public IP : (new) raviCngfwSpokeApp1-ip
Create new

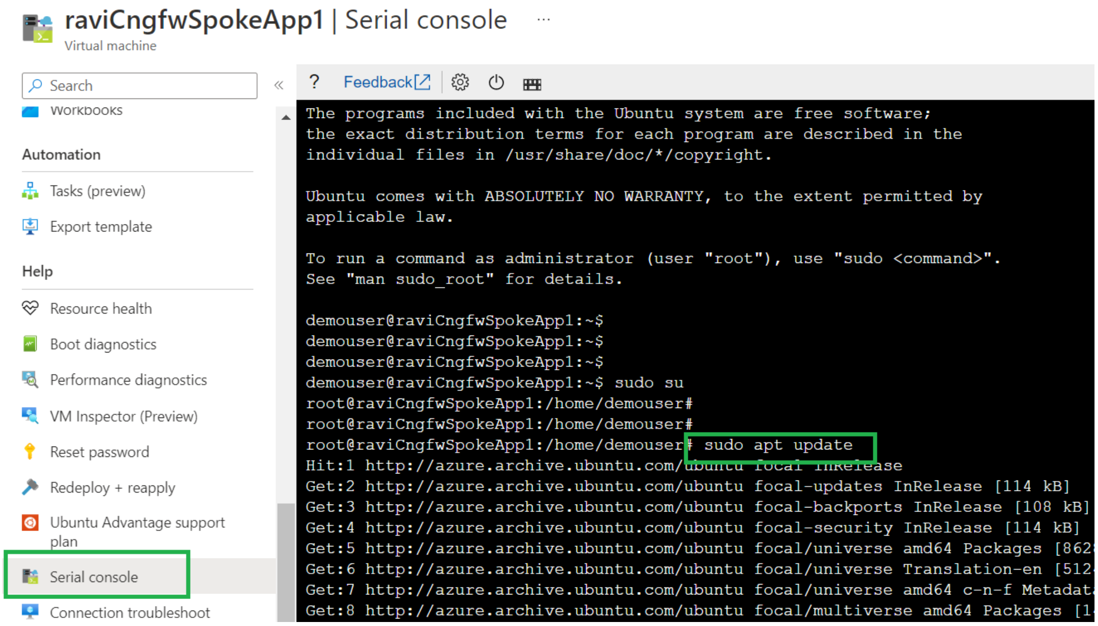
NIC network security group : ○ None ● Basic ○ Advanced

Public inbound ports * : ○ None ● Allow selected ports

5. Review the configuration and create the server.

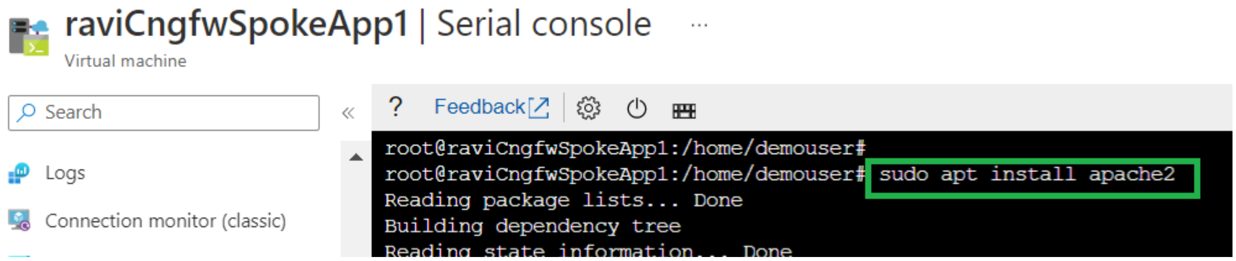
6. Once the Ubuntu server deployment is complete, install an apache server on it. To do this, go to the serial console of the created Ubuntu server and execute the commands below to install an apache server:

   sudo apt update



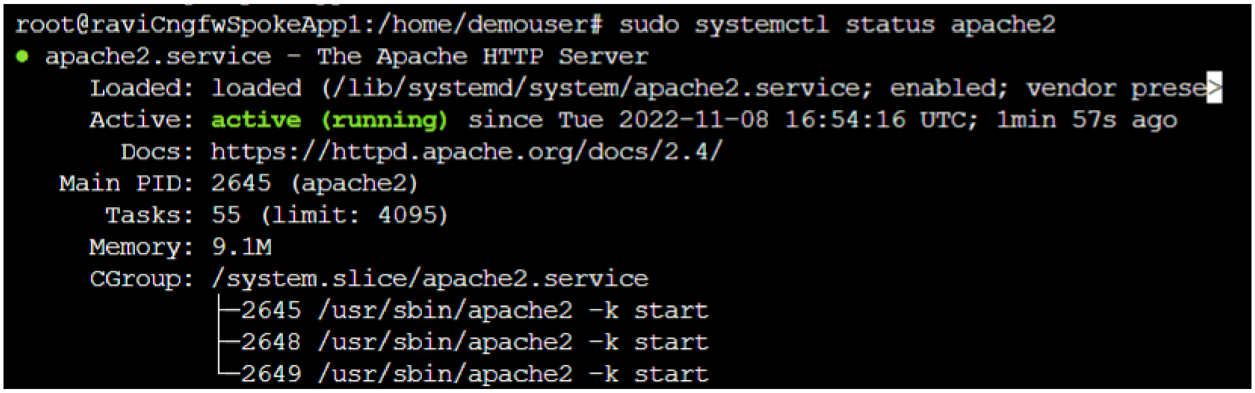
```
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

demouser@raviCngfwSpokeApp1:~$
demouser@raviCngfwSpokeApp1:~$
demouser@raviCngfwSpokeApp1:~$
demouser@raviCngfwSpokeApp1:~$ sudo su
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser# sudo apt update
Hit:1 http://azure.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://azure.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:3 http://azure.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:4 http://azure.archive.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:5 http://azure.archive.ubuntu.com/ubuntu focal/universe amd64 Packages [862
Get:6 http://azure.archive.ubuntu.com/ubuntu focal/universe Translation-en [512
Get:7 http://azure.archive.ubuntu.com/ubuntu focal/universe amd64 c-n-f Metadat
Get:8 http://azure.archive.ubuntu.com/ubuntu focal/multiverse amd64 Packages [1
```

sudo apt install apache2


raviCngfwSpokeApp1 | Serial console
Virtual machine
Search
Logs
Connection monitor (classic)
?
Feedback

```
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser# sudo apt install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

Confirm that the apache server installed successfully using the following command:
sudo systemctl status apache2



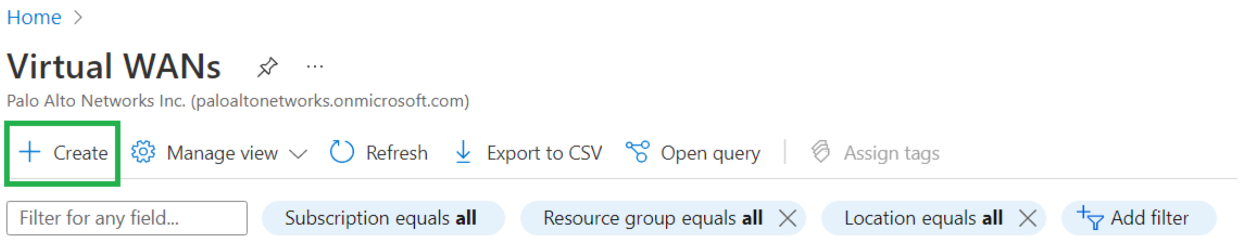
```
root@raviCngfwSpokeApp1:/home/demouser# sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor prese>
     Active: active (running) since Tue 2022-11-08 16:54:16 UTC; 1min 57s ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2645 (apache2)
      Tasks: 55 (limit: 4095)
     Memory: 9.1M
     CGroup: /system.slice/apache2.service
             ├─2645 /usr/sbin/apache2 -k start
             ├─2648 /usr/sbin/apache2 -k start
             └─2649 /usr/sbin/apache2 -k start
```

Create Spoke App02 VNet with Ubuntu Server (For EW-traffic)

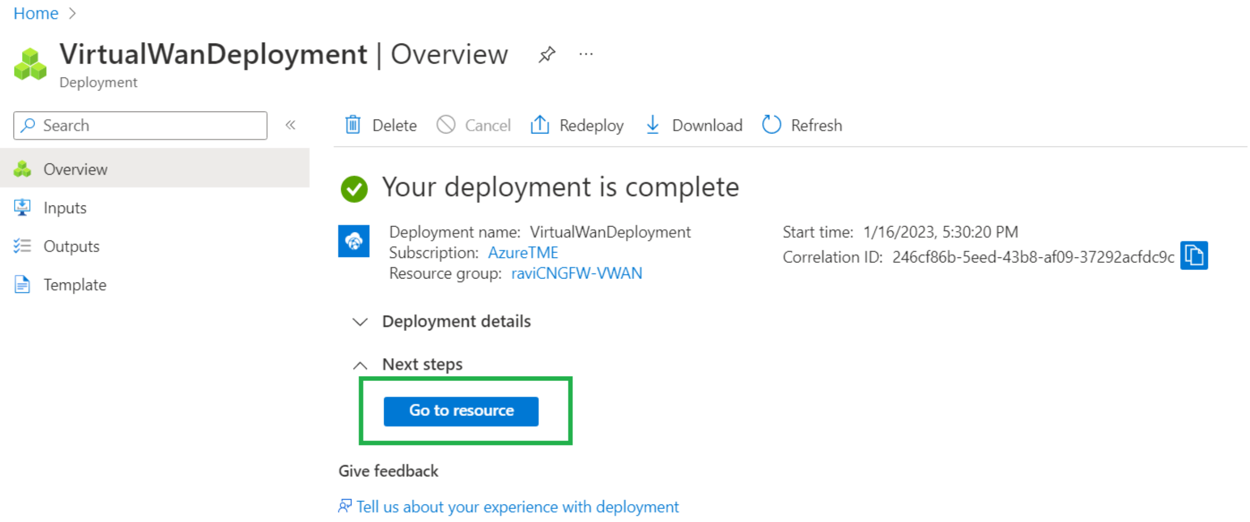
Repeat the above steps to create Spoke VNet2 with Ubuntu server.

Create Azure Virtual WAN with Hub

1. Login to Azure portal and search for Virtual WAN and click on "Create" option to create Virtual WAN Service



On successful creation of Virtual WAN service, you can go to the resource by clicking on "Go to resource" as per the screenshot below.

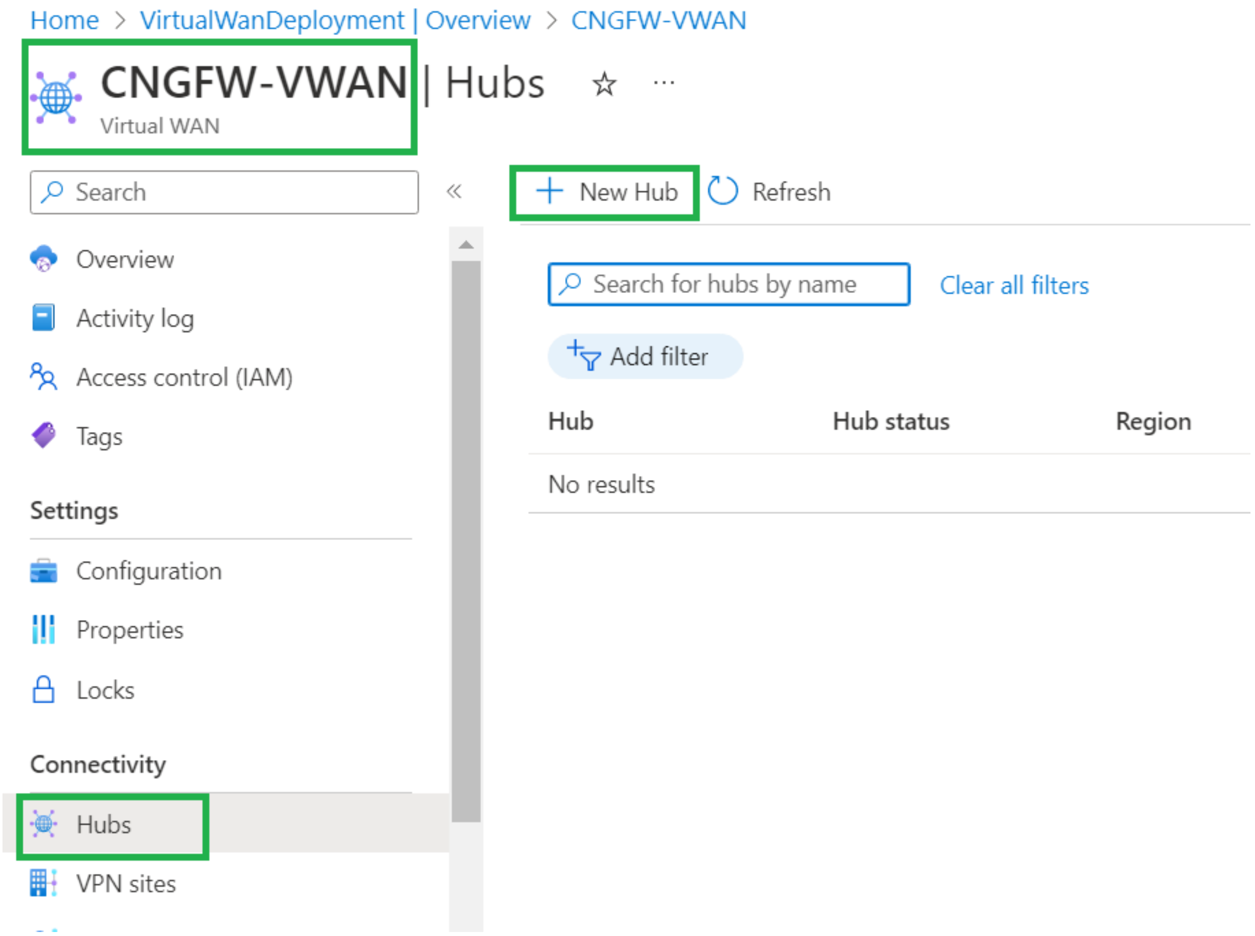


Your deployment is complete

Deployment name: VirtualWanDeployment
Subscription: AzureTME
Resource group: raviCNGFW-VWAN

Start time: 1/16/2023, 5:30:20 PM
Correlation ID: 246cf86b-5eed-43b8-af09-37292acfdc9c

2. Add Hub to the Virtual WAN created.
   After going to the Virtual WAN created in the above step, add a new hub by going to "Connectivity > Hubs" as shown below.



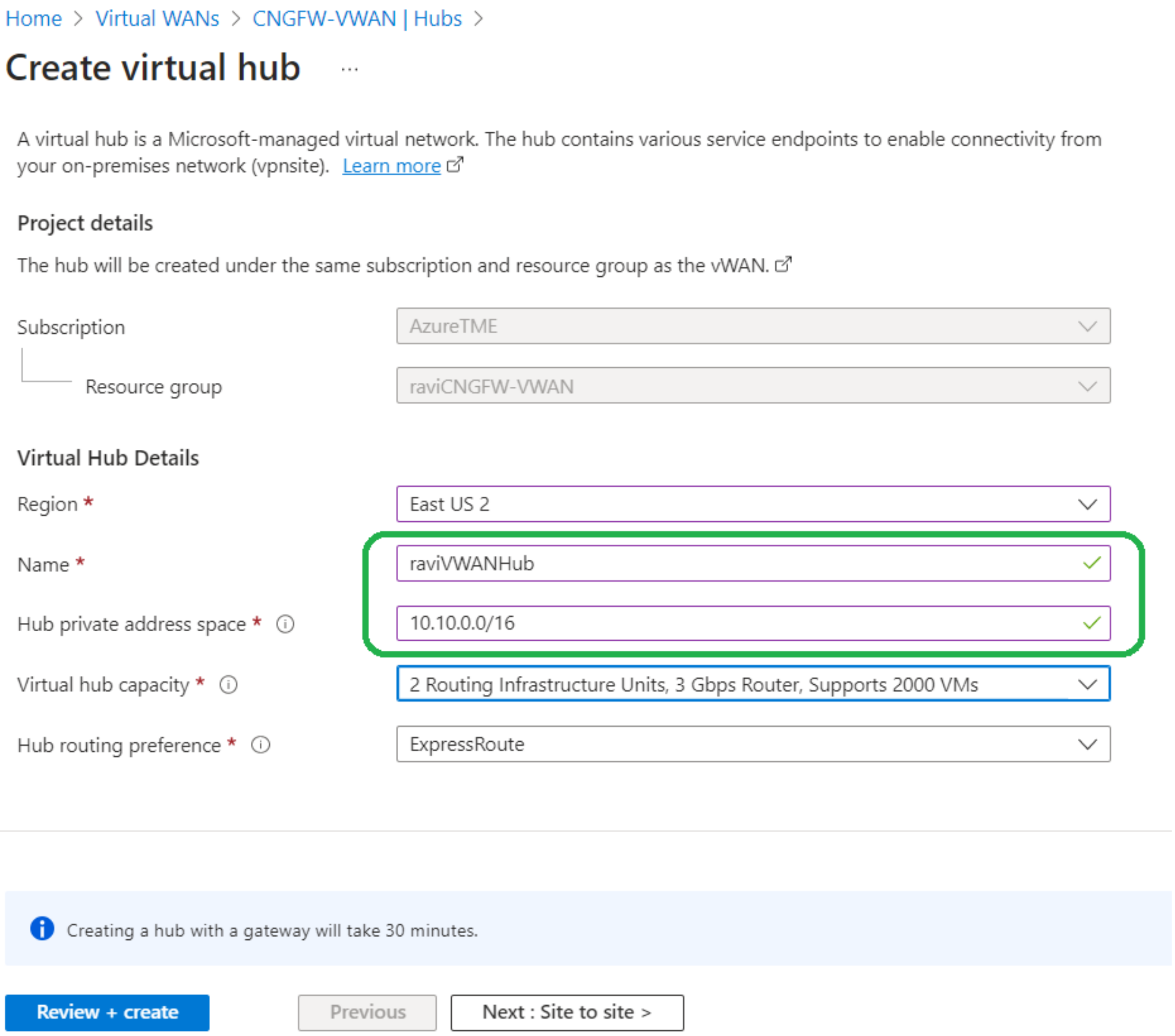
Home > VirtualWanDeployment | Overview > CNGFW-VWAN

CNGFW-VWAN | Hubs

Virtual WAN

Hub	Hub status	Region
No results		

Configure Hub private address space as shown below and click on “**Next : Site to Site >**”



Home > Virtual WANs > CNGFW-VWAN | Hubs >

Create virtual hub ...

A virtual hub is a Microsoft-managed virtual network. The hub contains various service endpoints to enable connectivity from your on-premises network (vpnsite). Learn more

Project details

The hub will be created under the same subscription and resource group as the vWAN.

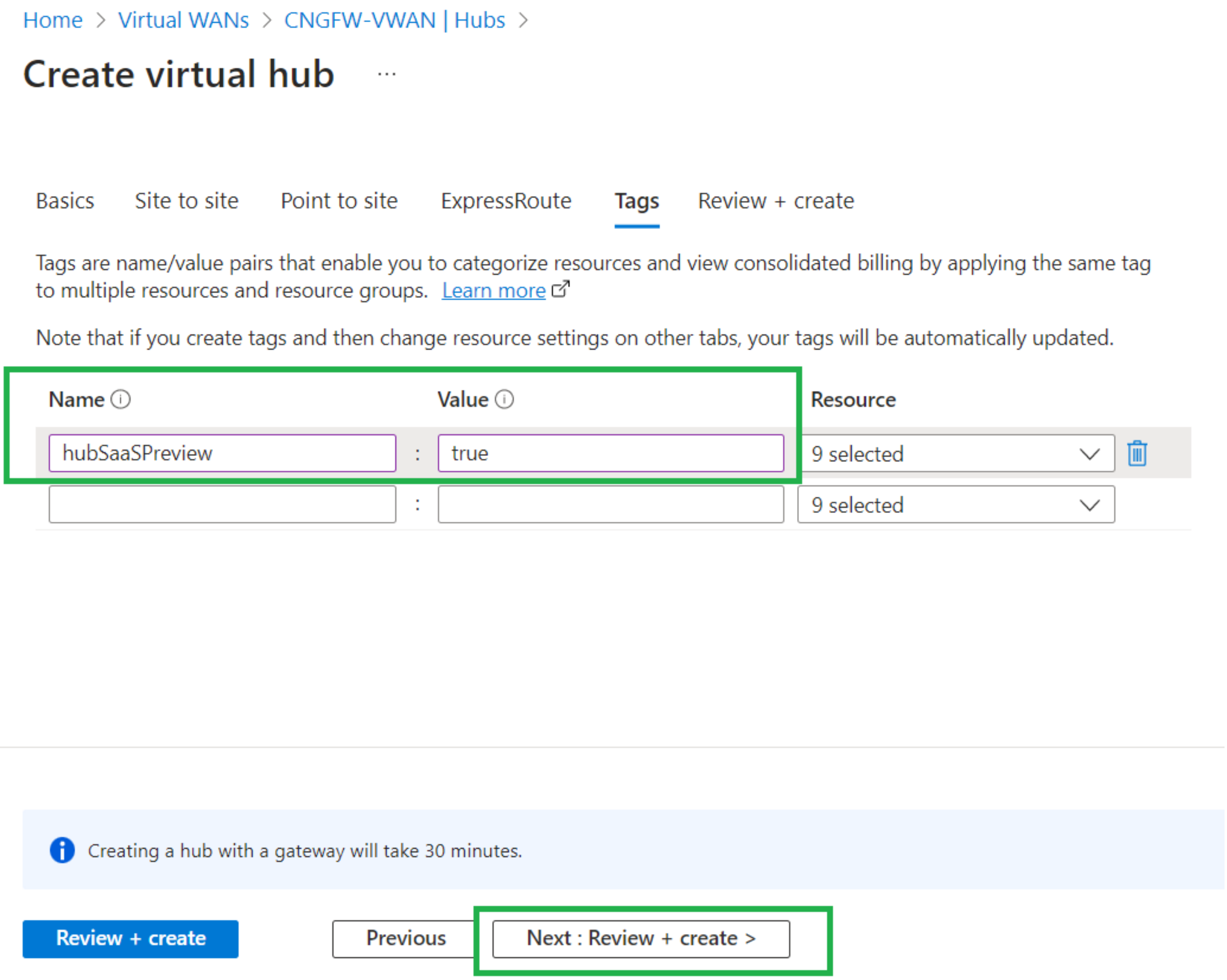
Subscription	AzureTME
Resource group	raviCNGFW-VWAN

Virtual Hub Details

Region *	East US 2
Name *	raviVWANHub ✓
Hub private address space *	10.10.0.0/16 ✓
Virtual hub capacity *	2 Routing Infrastructure Units, 3 Gbps Router, Supports 2000 VMs
Hub routing preference *	ExpressRoute

Creating a hub with a gateway will take 30 minutes.

After this you can directly go to the “Tags” section and configure Tag as shown below.



Create virtual hub ...

Tags are name/value pairs that enable you to categorize resources and view consolidated billing by applying the same tag to multiple resources and resource groups. Learn more

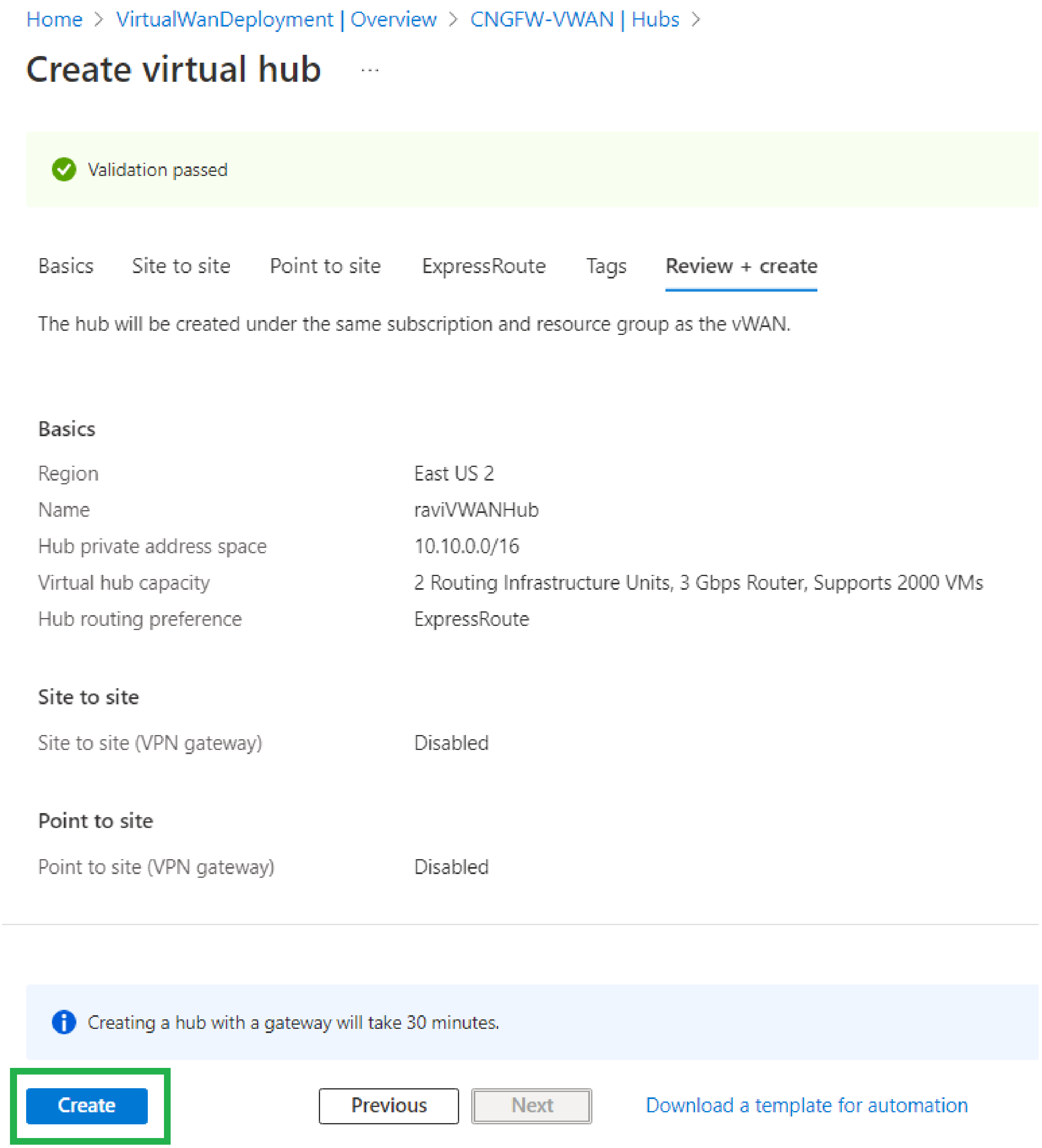
Note that if you create tags and then change resource settings on other tabs, your tags will be automatically updated.

Name		Value	Resource
hubSaaSPreview	:	true	9 selected
	:		9 selected

Creating a hub with a gateway will take 30 minutes.

NOTE: Tag name “hubSaaSPreview” and Value “true” should be provided while creation of the hub and should not be provided after creation of hub.

On successful validation of the configuration, click on “**Create**” to create Virtual WAN Hub



Home > VirtualWanDeployment | Overview > CNGFW-VWAN | Hubs >

Create virtual hub

Validation passed

The hub will be created under the same subscription and resource group as the vWAN.

Basics

Region	East US 2
Name	raviVWANHub
Hub private address space	10.10.0.0/16
Virtual hub capacity	2 Routing Infrastructure Units, 3 Gbps Router, Supports 2000 VMs
Hub routing preference	ExpressRoute

Site to site

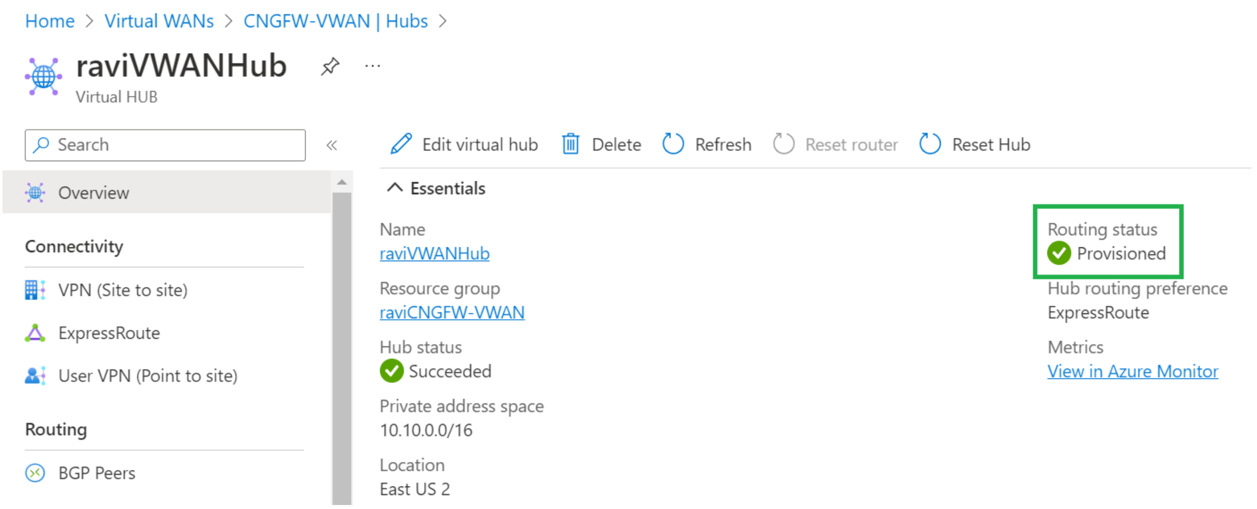
Site to site (VPN gateway)	Disabled

Point to site

Point to site (VPN gateway)	Disabled

Creating a hub with a gateway will take 30 minutes.

After creation of Virtual WAN Hub, make sure that the Routing status is in "**Provisioned**" state


Home > Virtual WANs > CNGFW-VWAN | Hubs >

raviVWANHub

Virtual HUB

Name: raviVWANHub

Resource group: raviCNGFW-VWAN

Hub status: Succeeded

Private address space: 10.10.0.0/16

Location: East US 2

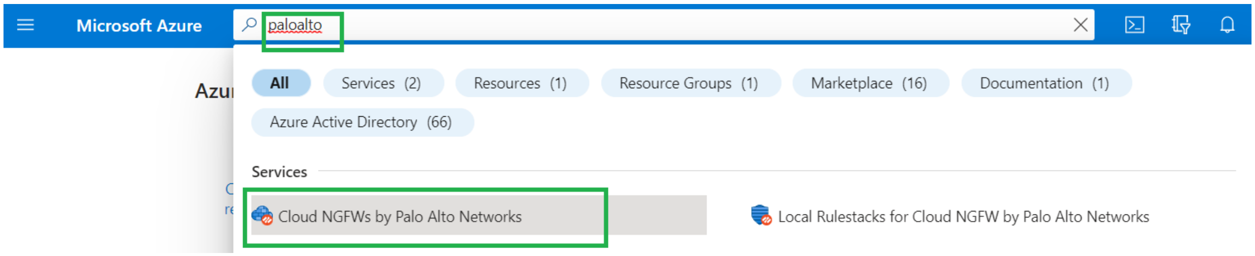
Routing status: Provisioned

Hub routing preference: ExpressRoute

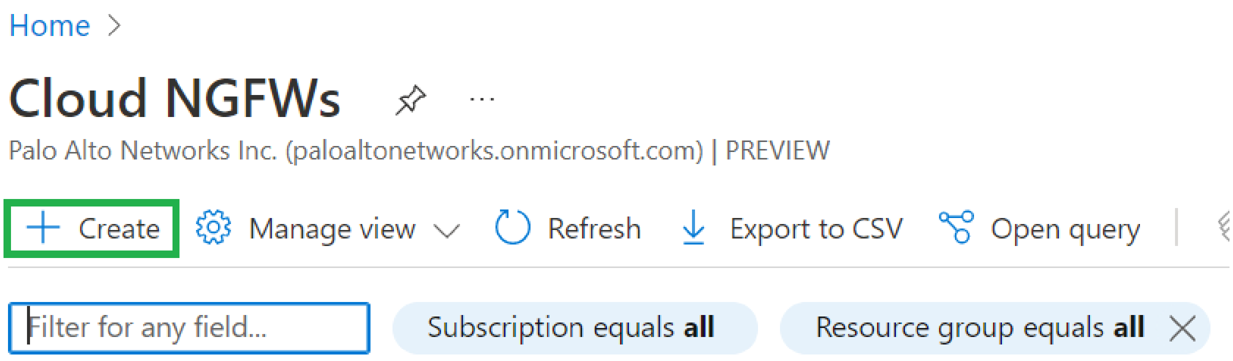
Metrics: View in Azure Monitor

Deployment of Cloud NGFW

1. Now that the Virtual WAN environment is set up, deploy the Cloud NGFW to protect the traffic going through Virtual WAN Hub.
2. Login to Azure portal and search for “paloalto”. This search displays the Cloud NGFW service by Palo Alto Networks:


Microsoft Azure
paloalto
All
Services (2)
Resources (1)
Resource Groups (1)
Marketplace (16)
Documentation (1)
Azure Active Directory (66)
Services
Cloud NGFWs by Palo Alto Networks
Local Rulestacks for Cloud NGFW by Palo Alto Networks

3. Click “Cloud NGFWs” to start the creation of the Palo Alto Networks Cloud NGFW service for Azure.
4. The following screen is the landing page for the Cloud NGFW resource. This screen populates all the available, pre-created Cloud NGFW instances (if not a first-time user). Click **Create** to start the creation of a Cloud NGFW resource:



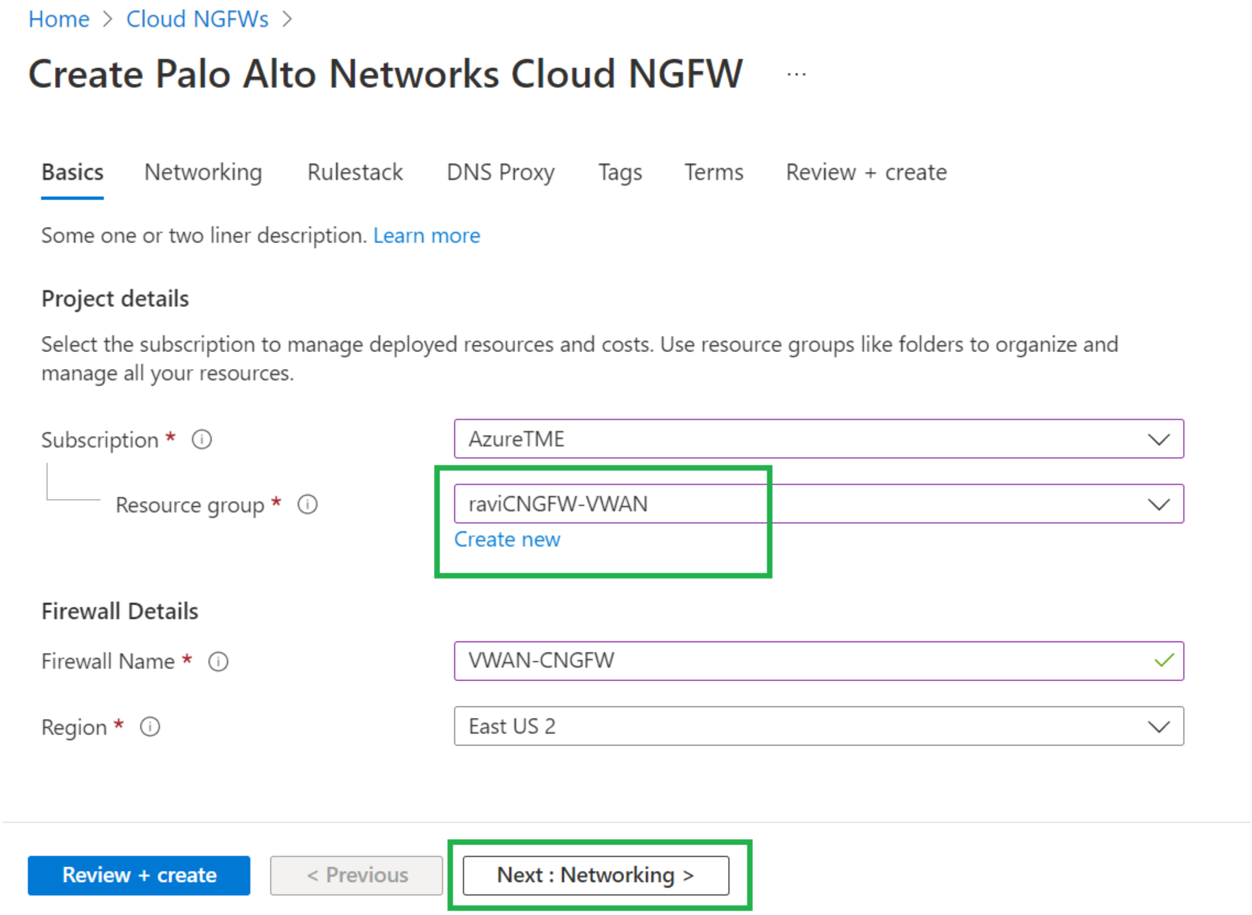
Palo Alto Networks Inc. (paloaltonetworks.onmicrosoft.com) | PREVIEW

Subscription equals **all**

Resource group equals **all**

5. After clicking **Create**, the Create Palo Alto Networks Cloud NGFW screen appears. Use the information in the table below to populate basic information for your Cloud NGFW resource:

Subscription	Automatically selected based on the subscription used while logged in.
Resource Group	Use one of the existing resource groups or create a new resource group (by clicking the **Create New** option) in which the Cloud NGFW resource is created.
Firewall Name	Name of the Cloud NGFW Firewall resource.
Region	Region in which Cloud NGFW is provisioned. For this Private Preview, only US East-2 and US Central regions are supported.



Home > Cloud NGFWs >

Create Palo Alto Networks Cloud NGFW ...

Some one or two liner description. Learn more

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * : AzureTME

Resource group * : raviCNGFW-VWAN

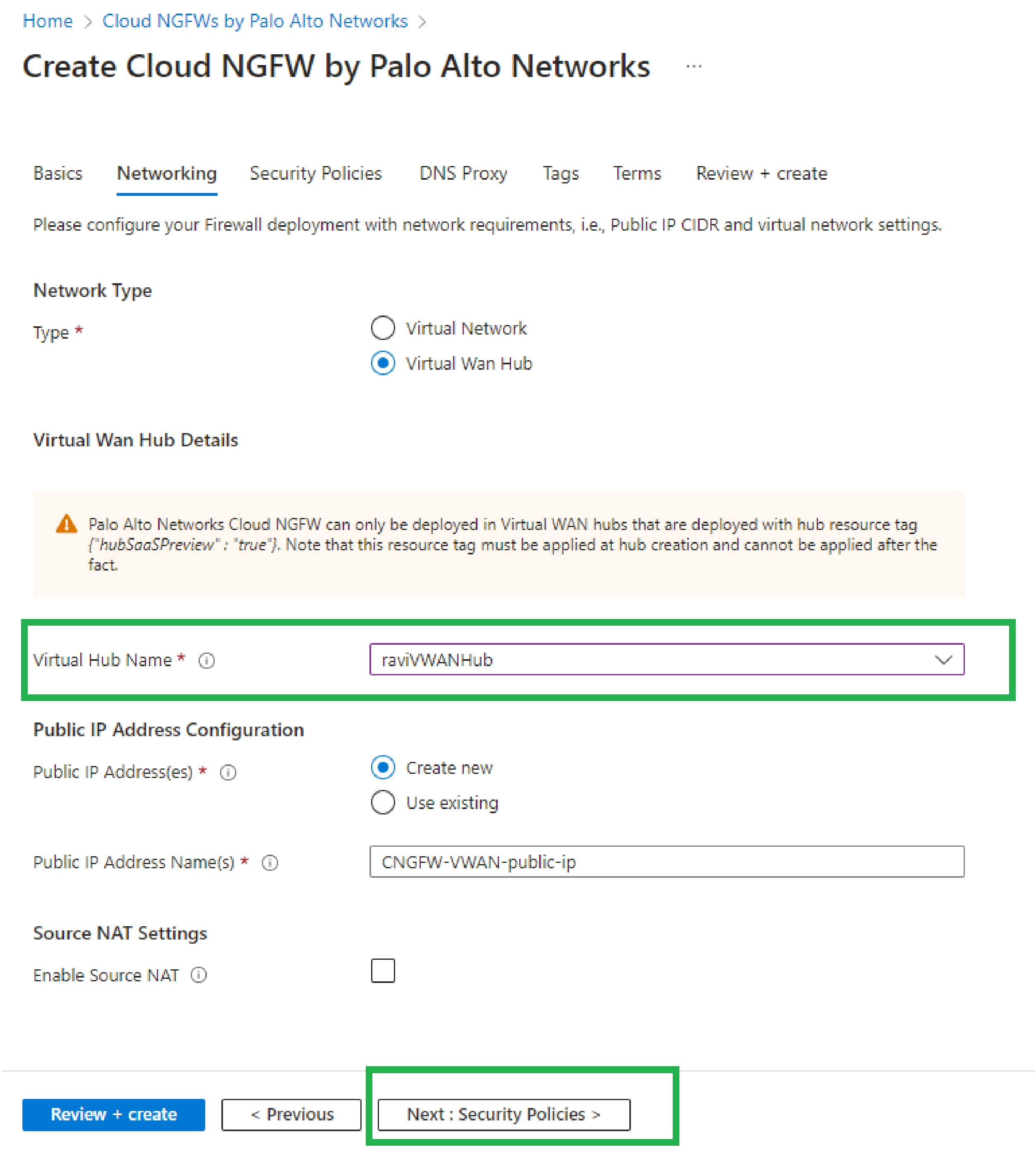
Create new

Firewall Details

Firewall Name * : VWAN-CNGFW ✓

Region * : East US 2

6. Once the details are filled in, click **Next: Networking >** and provide information for your networking environment. Choose the Network Injection type as **Virtual Wan Hub**. And select Virtual Hub Name from the dropdown(this is the Hub which got created in above step.). You can also specify IP addresses. Specify the **Source NAT** option if Network Address Translation (NAT) is used on the traffic going out to the Internet:



Home > Cloud NGFWs by Palo Alto Networks >

Create Cloud NGFW by Palo Alto Networks ...

Please configure your Firewall deployment with network requirements, i.e., Public IP CIDR and virtual network settings.

Network Type

Type * ○ Virtual Network ● Virtual Wan Hub

Virtual Wan Hub Details

Palo Alto Networks Cloud NGFW can only be deployed in Virtual WAN hubs that are deployed with hub resource tag {"*hubSaaSPreview*" : "*true*"}. Note that this resource tag must be applied at hub creation and cannot be applied after the fact.

Virtual Hub Name * ⓘ raviVWANHub

Public IP Address Configuration

Public IP Address(es) * ⓘ ● Create new ○ Use existing

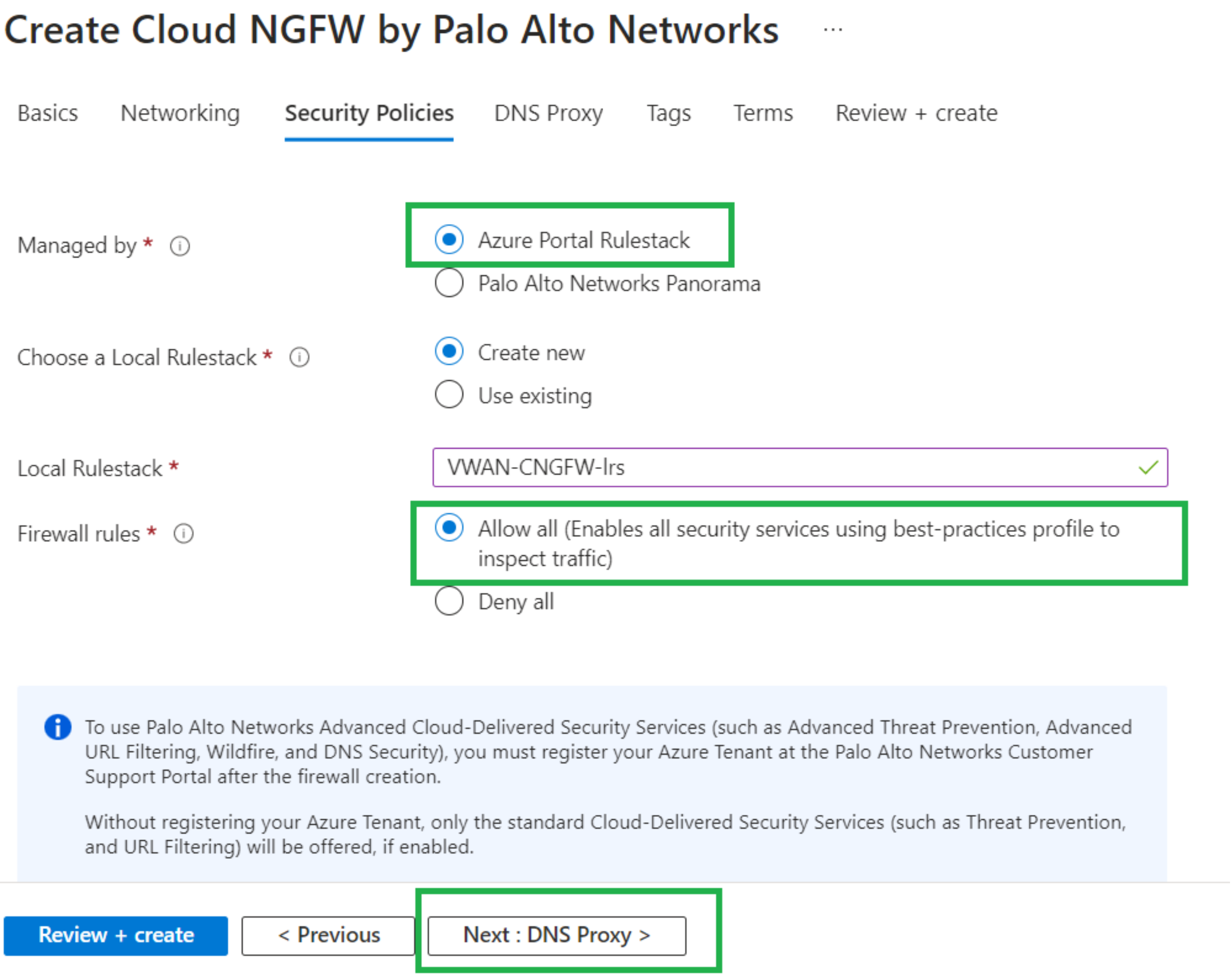
Public IP Address Name(s) * ⓘ CNGFW-VWAN-public-ip

Source NAT Settings

Enable Source NAT ⓘ □

7. Click **Next: Security Policies >** to create a Local Rule stack where rules can be defined. This is a placeholder for the local rule stack. After the creation of Cloud NGFW resource, this rulestack can be modified to add/edit rules, FQDN,

and prefix list. If there is a Local Rule Stack that's already created, select it from the drop-down menu after selecting Use existing option:



Create Cloud NGFW by Palo Alto Networks ...

Managed by * ● Azure Portal Rulestack ○ Palo Alto Networks Panorama

Choose a Local Rulestack * ● Create new ○ Use existing

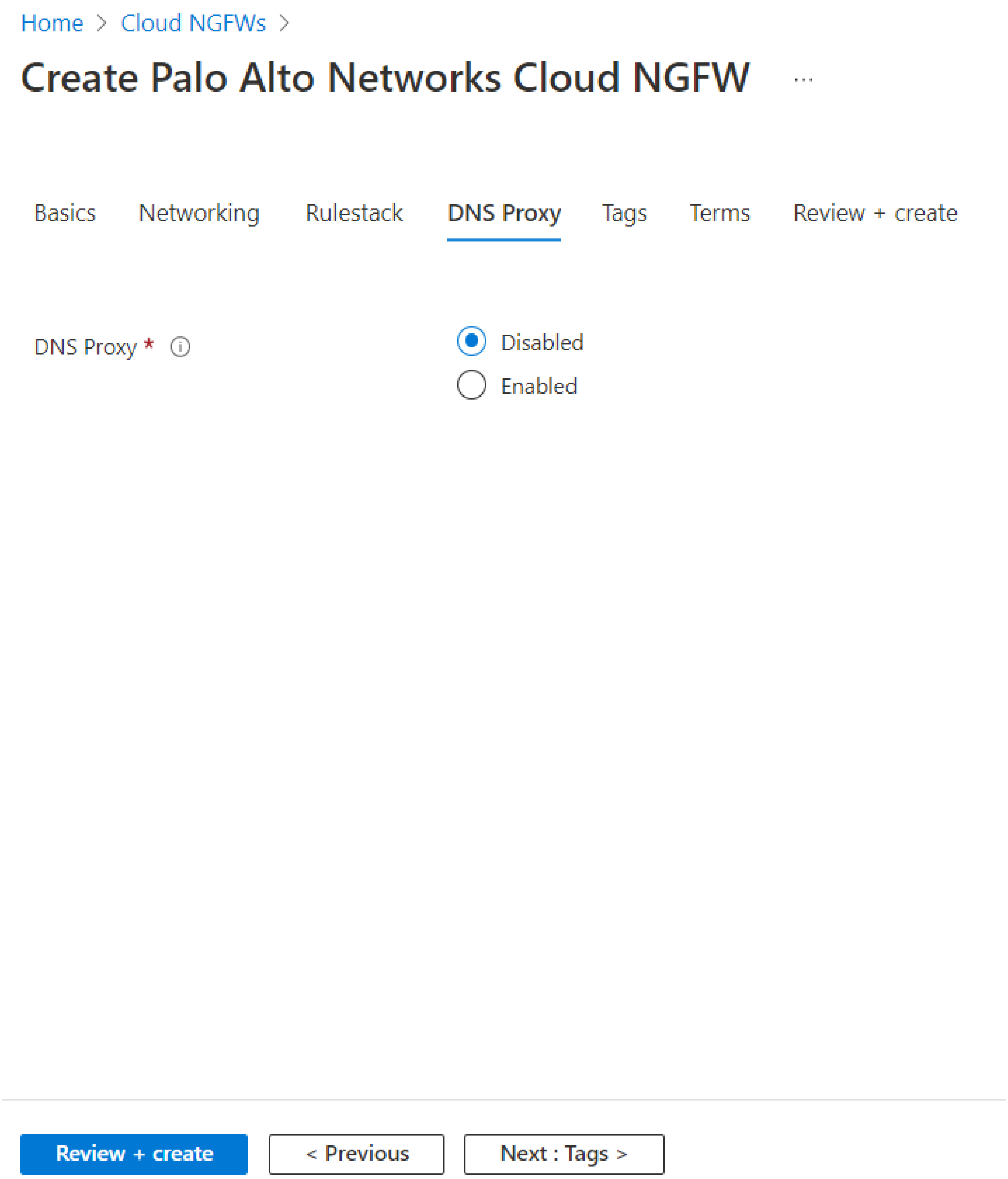
Local Rulestack * VWAN-CNGFW-lrs ✓

Firewall rules * ● Allow all (Enables all security services using best-practices profile to inspect traffic) ○ Deny all

To use Palo Alto Networks Advanced Cloud-Delivered Security Services (such as Advanced Threat Prevention, Advanced URL Filtering, Wildfire, and DNS Security), you must register your Azure Tenant at the Palo Alto Networks Customer Support Portal after the firewall creation.

Without registering your Azure Tenant, only the standard Cloud-Delivered Security Services (such as Threat Prevention, and URL Filtering) will be offered, if enabled.

8. Click **Next: DNS Proxy >** to configure Cloud NGFW as a DNS Proxy. It is disabled by default:



Home > Cloud NGFWs >

Create Palo Alto Networks Cloud NGFW ...

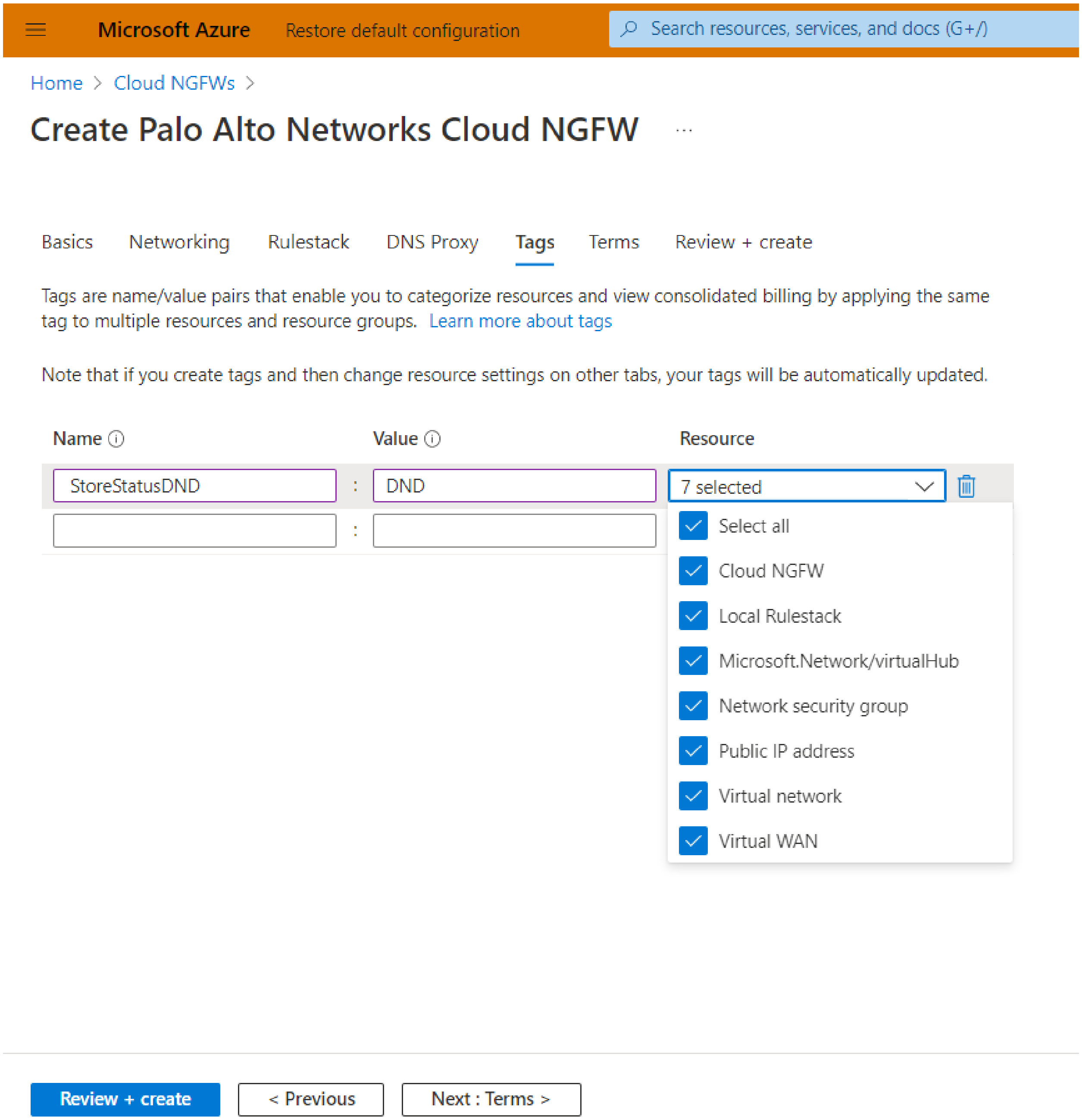
Basics Networking Rulestack **DNS Proxy** Tags Terms Review + create

DNS Proxy * ⓘ

● Disabled

○ Enabled

9. Click **Next: Tags >** to specify tags as per your Azure requirements:



Microsoft Azure
Restore default configuration
Search resources, services, and docs (G+/)
Home > Cloud NGFWs >

Create Palo Alto Networks Cloud NGFW

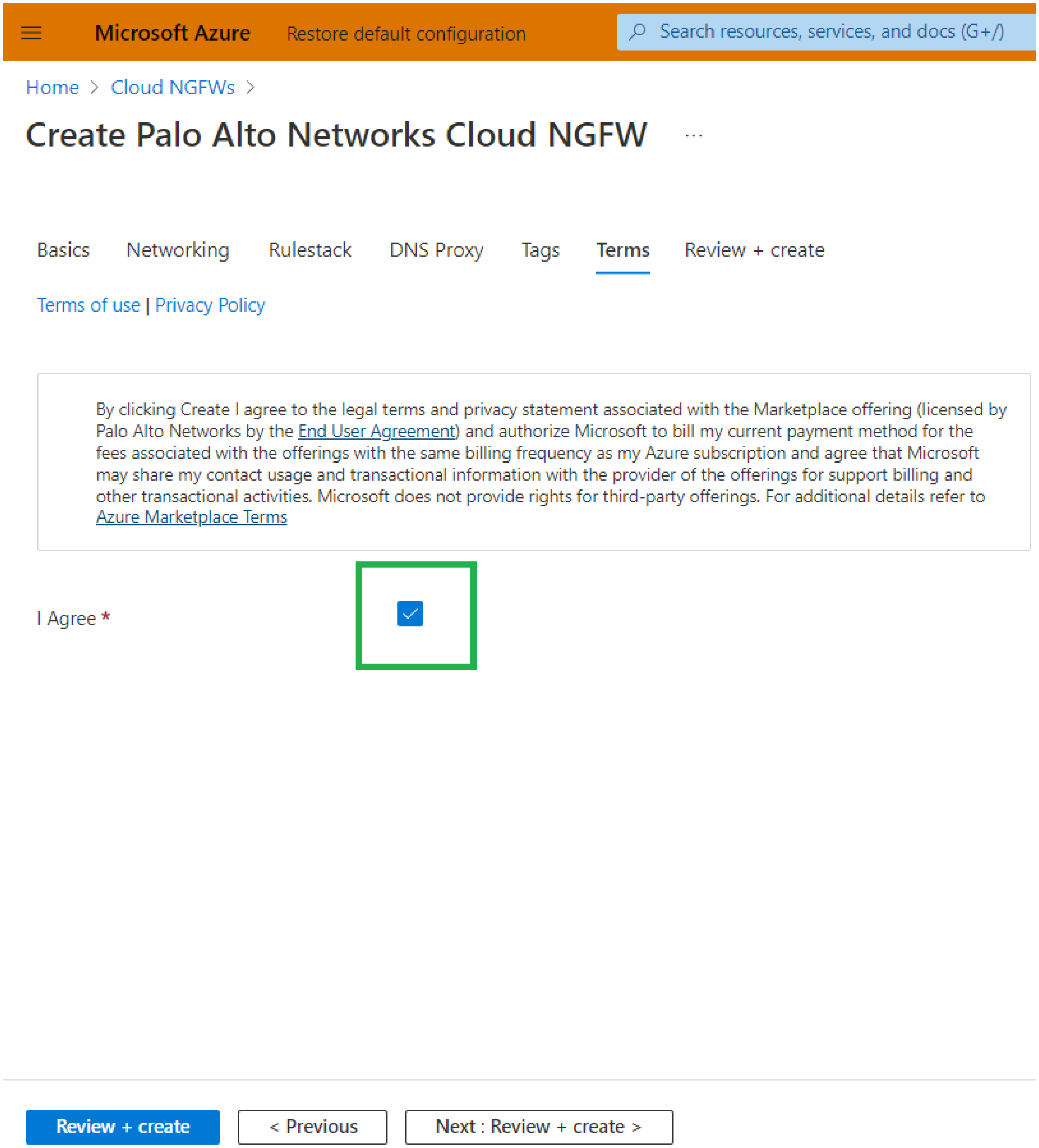
Tags are name/value pairs that enable you to categorize resources and view consolidated billing by applying the same tag to multiple resources and resource groups. Learn more about tags

Note that if you create tags and then change resource settings on other tabs, your tags will be automatically updated.

Name	Value	Resource
StoreStatusDND	DND	7 selected

☑ Select all
☑ Cloud NGFW
☑ Local Rulestack
☑ Microsoft.Network/virtualHub
☑ Network security group
☑ Public IP address
☑ Virtual network
☑ Virtual WAN

10. Click **Next: Terms >** and accept the terms as shown below:



Microsoft Azure
Restore default configuration
Search resources, services, and docs (G+/)
Home > Cloud NGFWs >

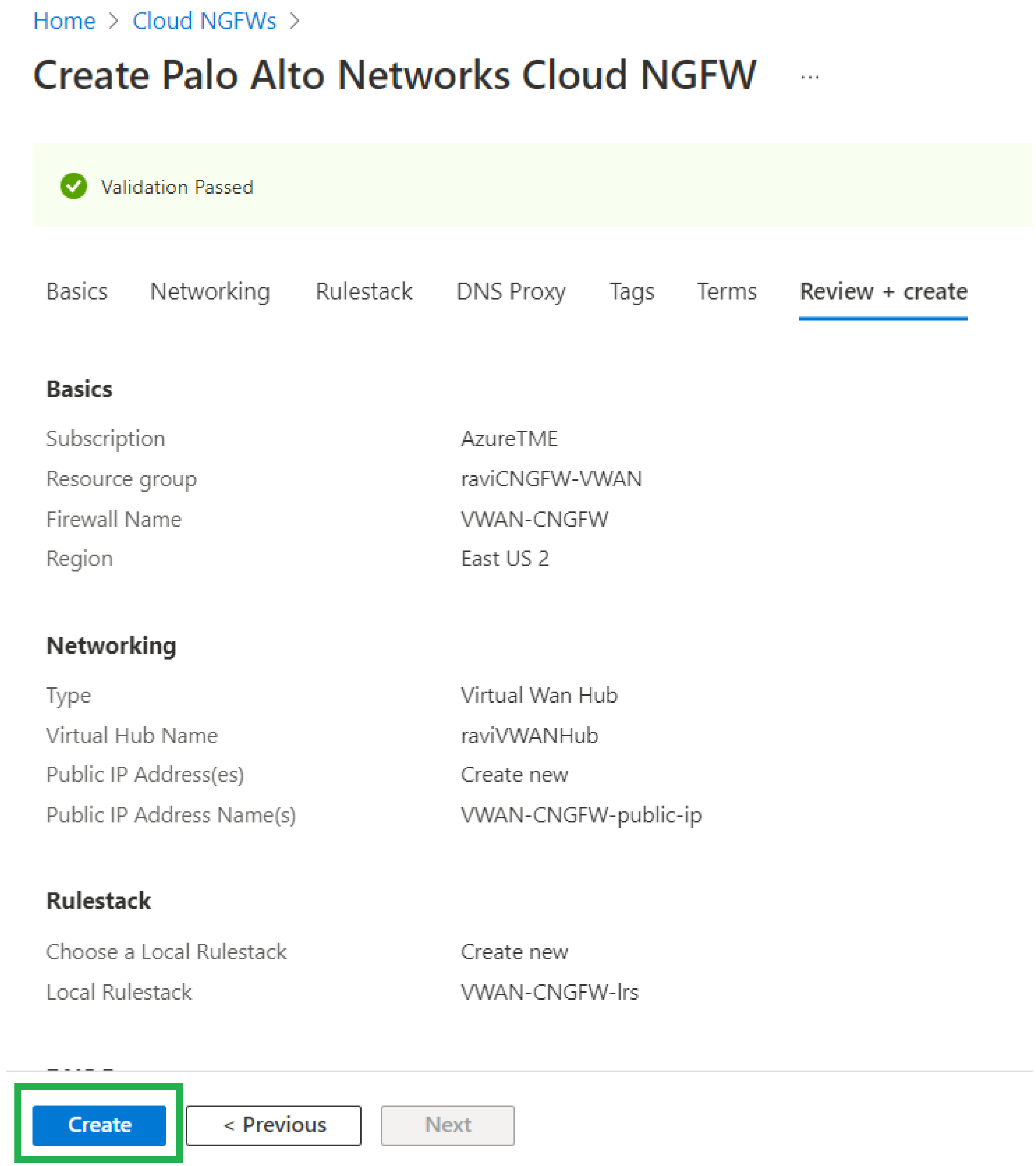
Create Palo Alto Networks Cloud NGFW

Basics
Networking
Rulestack
DNS Proxy
Tags
Terms
Review + create
Terms of use | Privacy Policy

By clicking Create I agree to the legal terms and privacy statement associated with the Marketplace offering (licensed by Palo Alto Networks by the End User Agreement) and authorize Microsoft to bill my current payment method for the fees associated with the offerings with the same billing frequency as my Azure subscription and agree that Microsoft may share my contact usage and transactional information with the provider of the offerings for support billing and other transactional activities. Microsoft does not provide rights for third-party offerings. For additional details refer to Azure Marketplace Terms

I Agree * ☑

11. Click **Next: Review + Create >** and create a Cloud NGFW service. Like any other Azure native service, the resource is validated first and then created. Once the screen shows V**alidation Passed**, click **Create** to deploy the Cloud NGFW service.



Create Palo Alto Networks Cloud NGFW

✓ Validation Passed

Basics

Subscription	AzureTME
Resource group	raviCNGFW-VWAN
Firewall Name	VWAN-CNGFW
Region	East US 2

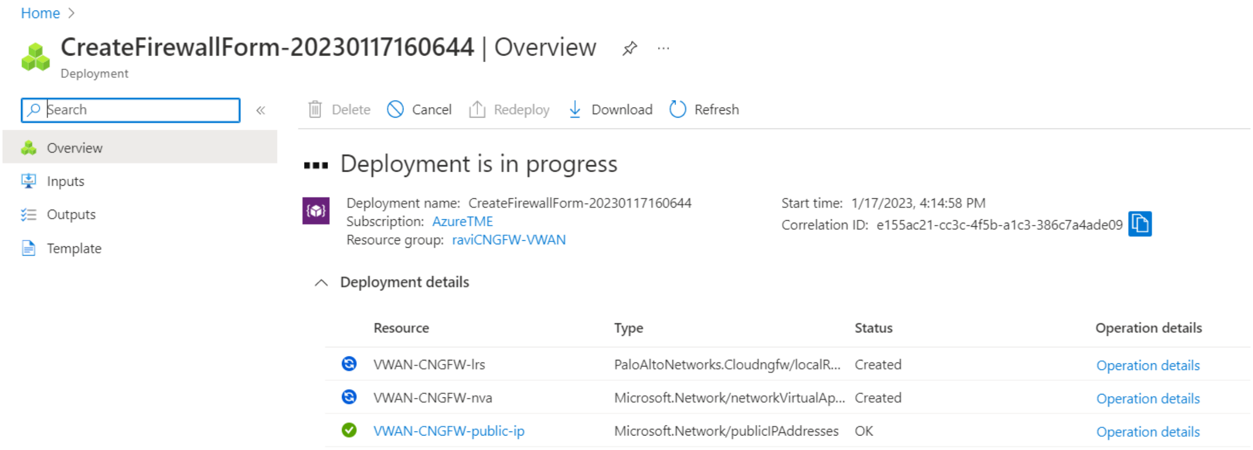
Networking

Type	Virtual Wan Hub
Virtual Hub Name	raviVWANHub
Public IP Address(es)	Create new
Public IP Address Name(s)	VWAN-CNGFW-public-ip

Rulestack

Choose a Local Rulestack	Create new
Local Rulestack	VWAN-CNGFW-lrs

After creating the Cloud NGFW service the deployment progress is displayed:



CreateFirewallForm-20230117160644 | Overview

Deployment

Deployment is in progress

Deployment name: CreateFirewallForm-20230117160644
Subscription: AzureTME
Resource group: raviCNGFW-VWAN

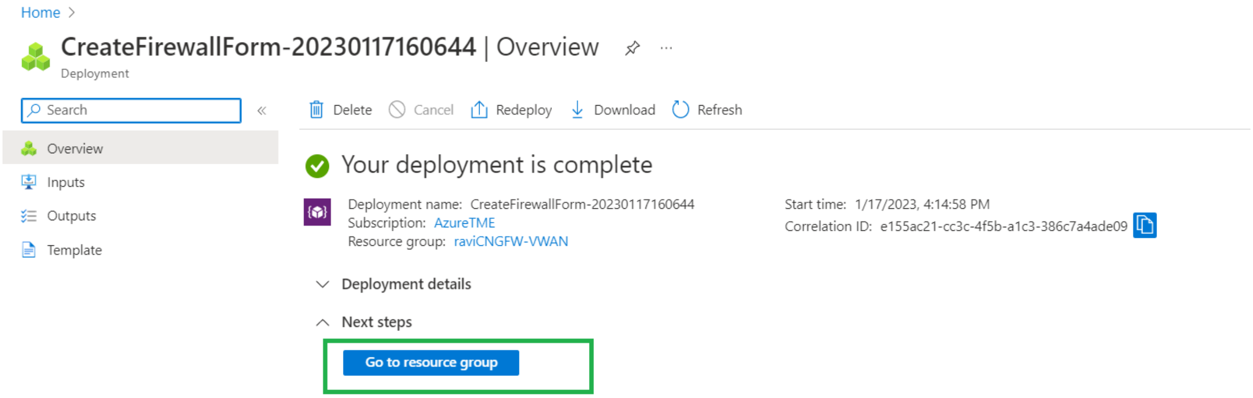
Start time: 1/17/2023, 4:14:58 PM
Correlation ID: e155ac21-cc3c-4f5b-a1c3-386c7a4ade09

Deployment details

Resource	Type	Status	Operation details
VWAN-CNGFW-lrs	PaloAltoNetworks.Cloudngfw/localR...	Created	Operation details
VWAN-CNGFW-nva	Microsoft.Network/networkVirtualAp...	Created	Operation details
VWAN-CNGFW-public-ip	Microsoft.Network/publicIPAddresses	OK	Operation details

The deployment of a Cloud NGFW takes approximately 30 minutes.

On a successful deployment, the screen below appears. Click Go to resource group to verify the resources created for this deployment:



CreateFirewallForm-20230117160644 | Overview

Deployment

Your deployment is complete

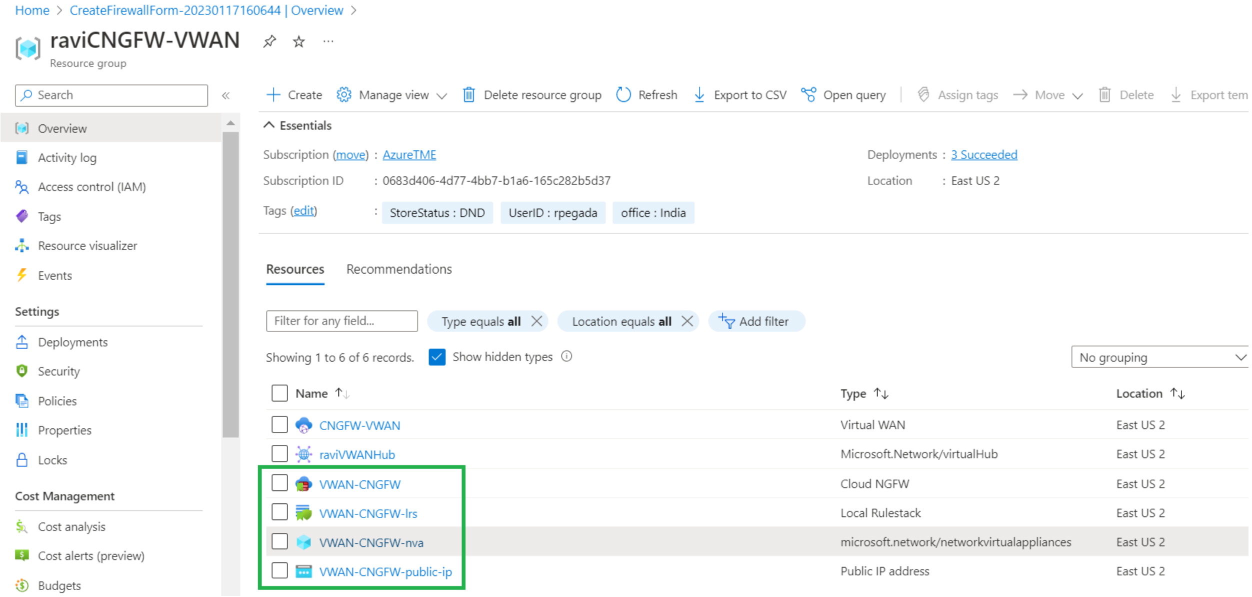
Deployment name: CreateFirewallForm-20230117160644
Subscription: AzureTME
Resource group: raviCNGFW-VWAN

Start time: 1/17/2023, 4:14:58 PM
Correlation ID: e155ac21-cc3c-4f5b-a1c3-386c7a4ade09

Deployment details

Next steps

12. There are Four resources created, which include Cloud NGFW, Local Rule stack, Public IP address and Cloud NGFW-nva:



Subscription (move) : AzureTME

Subscription ID : 0683d406-4d77-4bb7-b1a6-165c282b5d37

Tags (edit) : StoreStatus : DND UserID : rpegada office : India

Deployments : 3 Succeeded

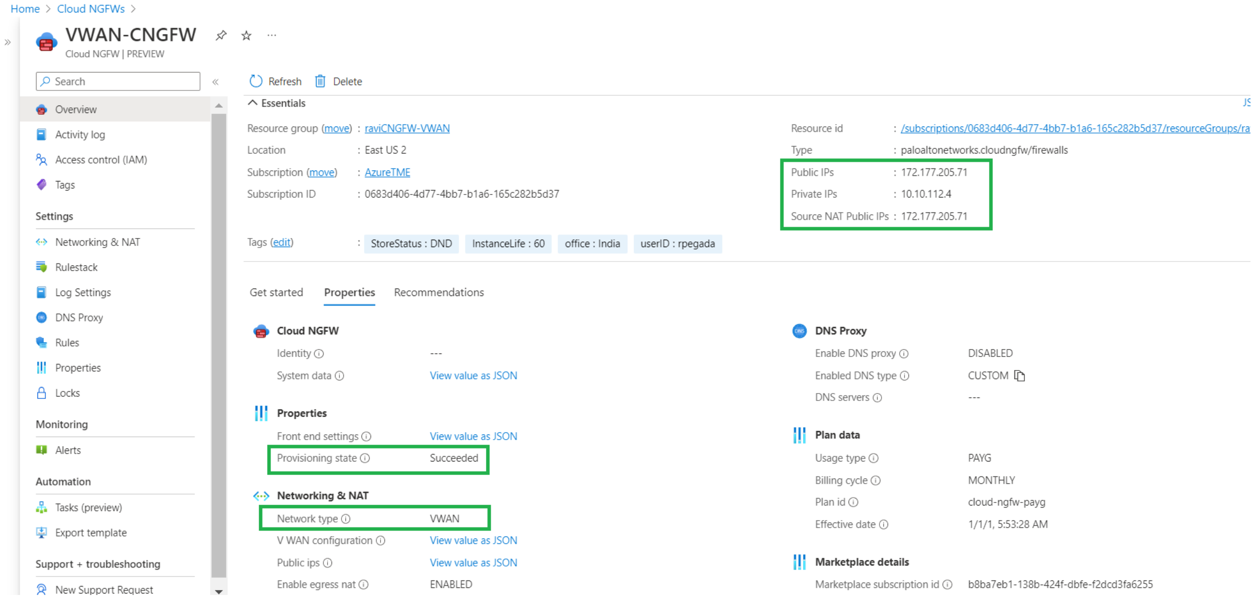
Location : East US 2

Showing 1 to 6 of 6 records. ☑ Show hidden types

□ Name	Type	Location
□ CNGFW-VWAN	Virtual WAN	East US 2
□ raviVWANHub	Microsoft.Network/virtualHub	East US 2
□ VWAN-CNGFW	Cloud NGFW	East US 2
□ VWAN-CNGFW-lrs	Local Rulestack	East US 2
□ VWAN-CNGFW-nva	microsoft.network/networkvirtualappliances	East US 2
□ VWAN-CNGFW-public-ip	Public IP address	East US 2

13. Once the Cloud NGFW resource is created, click on it to verify that the Provisioning state shows **Succeeded**. This screen also displays Public and Private IP addresses that are associated with the Cloud NGFW service. Use this information in further steps of this document to route traffic through the Cloud NGFW service

Also make sure that the Network type is **VWAN**:



Resource group (move) : raviCNGFW-VWAN

Location : East US 2

Subscription (move) : AzureTME

Subscription ID : 0683d406-4d77-4bb7-b1a6-165c282b5d37

Tags (edit) : StoreStatus : DND InstanceLife : 60 office : India userID : rpegada

Resource id : /subscriptions/0683d406-4d77-4bb7-b1a6-165c282b5d37/resourceGroups/ra

Type : paloaltonetworks.cloudngfw/firewalls

Public IPs : 172.177.205.71

Private IPs : 10.10.112.4

Source NAT Public IPs : 172.177.205.71

Cloud NGFW

Identity ---

System data View value as JSON

Properties

Front end settings View value as JSON

Provisioning state Succeeded

Networking & NAT

Network type VWAN

V WAN configuration View value as JSON

Public ips View value as JSON

Enable egress nat ENABLED

DNS Proxy

Enable DNS proxy DISABLED

Enabled DNS type CUSTOM

DNS servers ---

Plan data

Usage type PAYG

Billing cycle MONTHLY

Plan id cloud-ngfw-payg

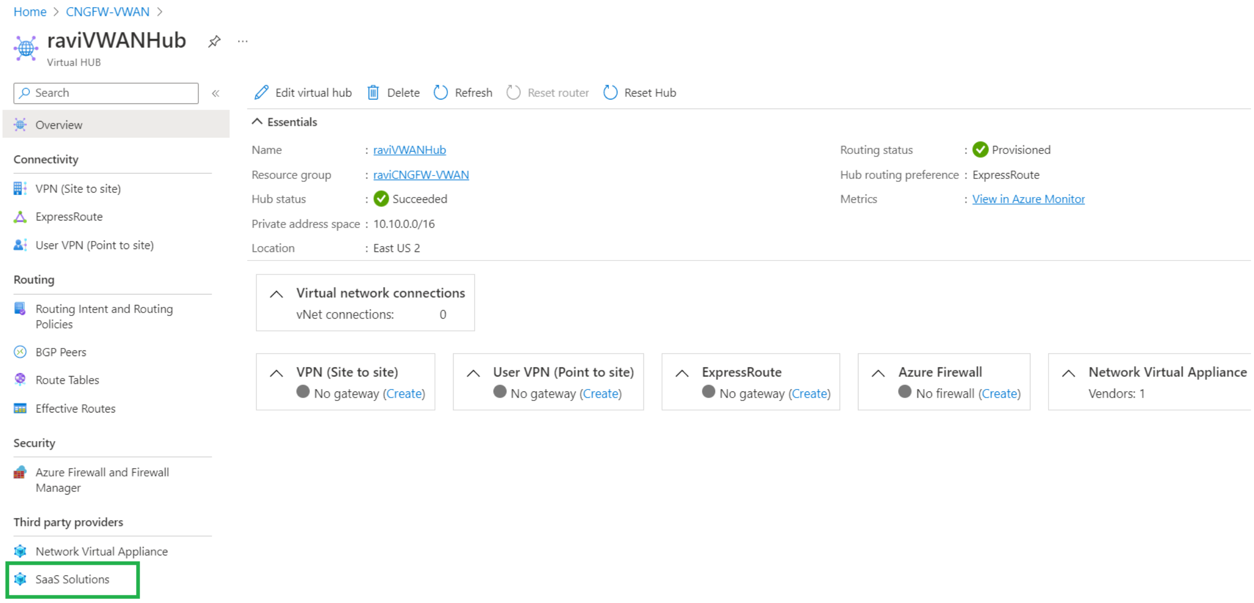
Effective date 1/1/1, 5:53:28 AM

Marketplace details

Marketplace subscription id b8ba7eb1-138b-424f-dbfe-f2dcd3fa6255

Verify for SaaS Solution within Virtual WAN Hub

After successful creation of the Cloud NGFW service with network type as Virtual WAN Hub. Verity for cloud NGFW to be added as a **SaaS Solution** for Virtual WAN Hub used.
Go to the Virtual Hub used while creating the cloud NGFW and click on “SaaS Solutions”



Name	: raviVWANHub	Routing status	: Provisioned
Resource group	: raviCNGFW-VWAN	Hub routing preference	: ExpressRoute
Hub status	: Succeeded	Metrics	: View in Azure Monitor
Private address space	: 10.10.0.0/16		
Location	: East US 2		

Virtual network connections
vNet connections: 0

VPN (Site to site)
No gateway (Create)

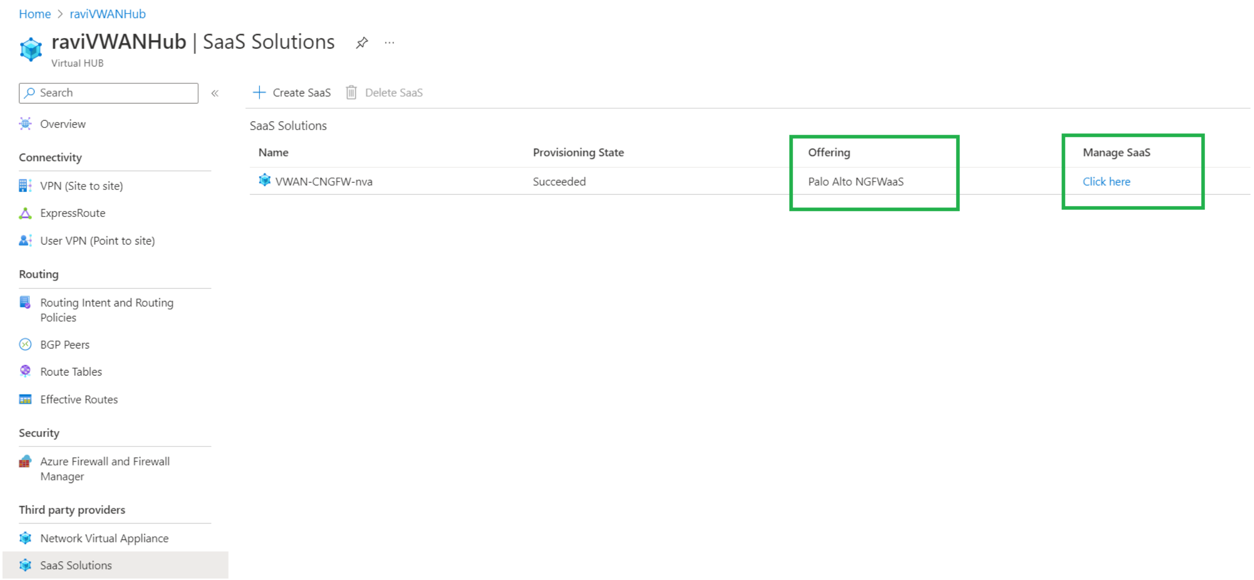
User VPN (Point to site)
No gateway (Create)

ExpressRoute
No gateway (Create)

Azure Firewall
No firewall (Create)

Network Virtual Appliance
Vendors: 1

Cloud NGFW created will be added as a SaaS solution to this Hub as shown below.



Home > raviVWANHub

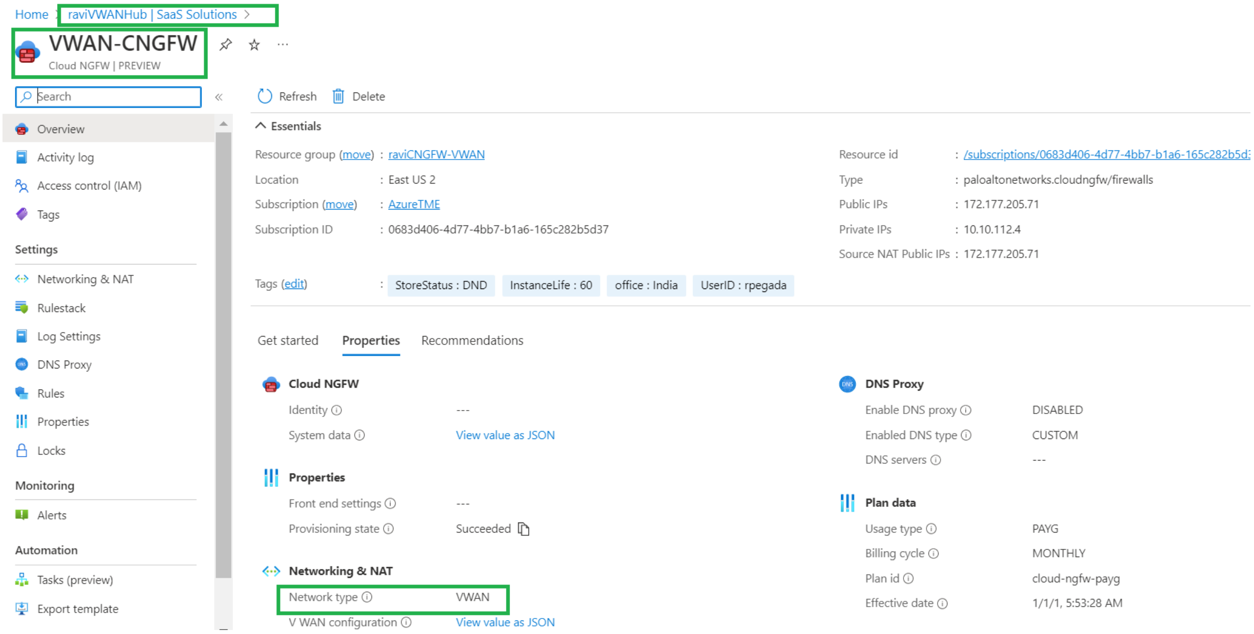
raviVWANHub | SaaS Solutions

Virtual HUB

SaaS Solutions

Name	Provisioning State	Offering	Manage SaaS
VWAN-CNGFW-nva	Succeeded	Palo Alto NGFWaaS	Click here

Within this screen, you can go to Cloud NGFW service created by clicking on “Click here” hyperlink, which is part of the Manage SaaS column.



Home > raviVWANHub | SaaS Solutions >

VWAN-CNGFW

Cloud NGFW | PREVIEW

Essentials

Resource group (move) : raviCNGFW-VWAN

Location : East US 2

Subscription (move) : AzureTME

Subscription ID : 0683d406-4d77-4bb7-b1a6-165c282b5d37

Resource id : /subscriptions/0683d406-4d77-4bb7-b1a6-165c282b5d3

Type : paloaltonetworks.cloudngfw/firewalls

Public IPs : 172.177.205.71

Private IPs : 10.10.112.4

Source NAT Public IPs : 172.177.205.71

Tags (edit) : StoreStatus : DND | InstanceLife : 60 | office : India | UserID : rpegada

Cloud NGFW

Identity ⓘ ---

System data ⓘ View value as JSON

Properties

Front end settings ⓘ ---

Provisioning state ⓘ Succeeded

Networking & NAT

Network type ⓘ VWAN

V WAN configuration ⓘ View value as JSON

DNS Proxy

Enable DNS proxy ⓘ DISABLED

Enabled DNS type ⓘ CUSTOM

DNS servers ⓘ ---

Plan data

Usage type ⓘ PAYG

Billing cycle ⓘ MONTHLY

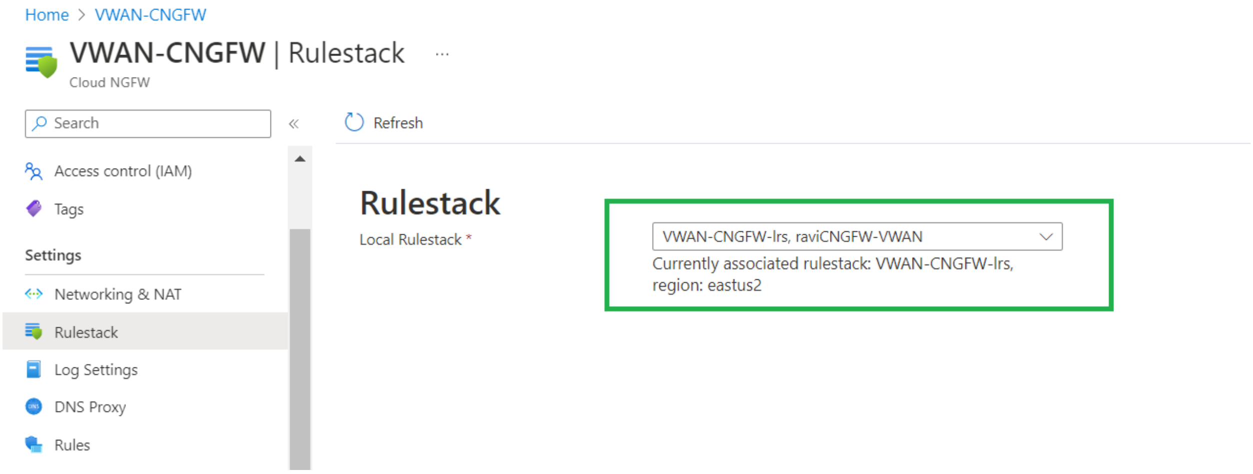
Plan id ⓘ cloud-ngfw-payg

Effective date ⓘ 1/1/1, 5:53:28 AM

Post Deployment of Cloud NGFW

Create/Update Rule stack

1. To update/edit the rulestack, click the **Rulestack** option available in the Cloud NGFW resource. As shown below, this displays the rulestack associated with the cloud NGFW service along with the resource group:



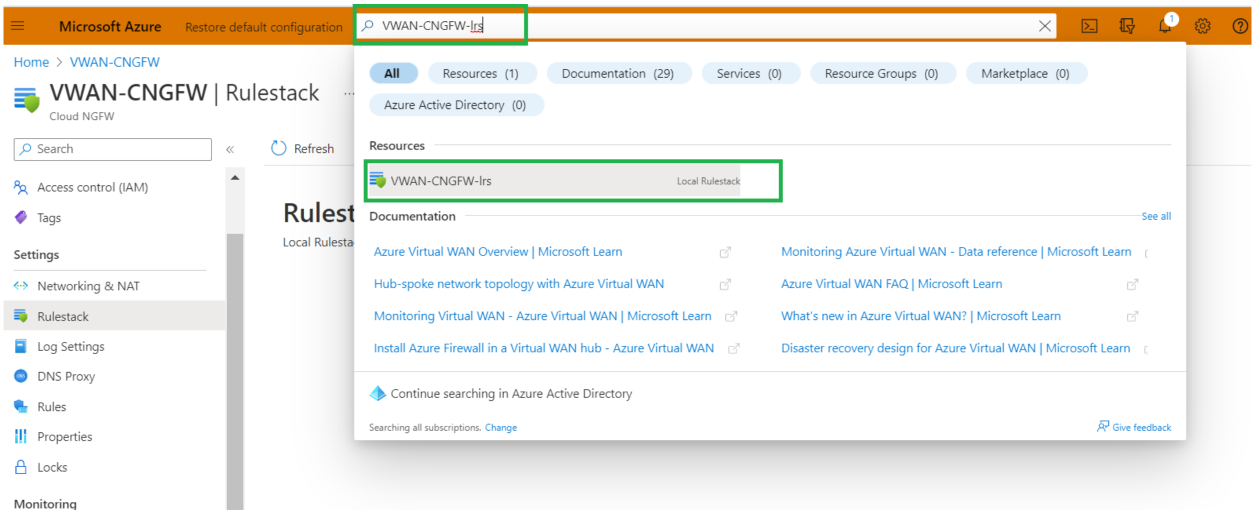
Rulestack

Local Rulestack *: VWAN-CNGFW-lrs, raviCNGFW-VWAN

Currently associated rulestack: VWAN-CNGFW-lrs, region: eastus2

The Cloud NGFW is associated with **VWAN-CNGFW-lrs**.
Next, modify this rulestack to add firewall rules to allow some traffic and block specific traffic.
By default Cloud NGFW blocks all traffic.

2. Search for Local Rulestack in the global search of the Azure portal:



Microsoft Azure   Restore default configuration   VWAN-CNGFW-lrs

Home > VWAN-CNGFW

VWAN-CNGFW | Rulestack

Cloud NGFW

All   Resources (1)   Documentation (29)   Services (0)   Resource Groups (0)   Marketplace (0)   Azure Active Directory (0)

Resources

VWAN-CNGFW-lrs   Local Rulestack

Documentation   See all

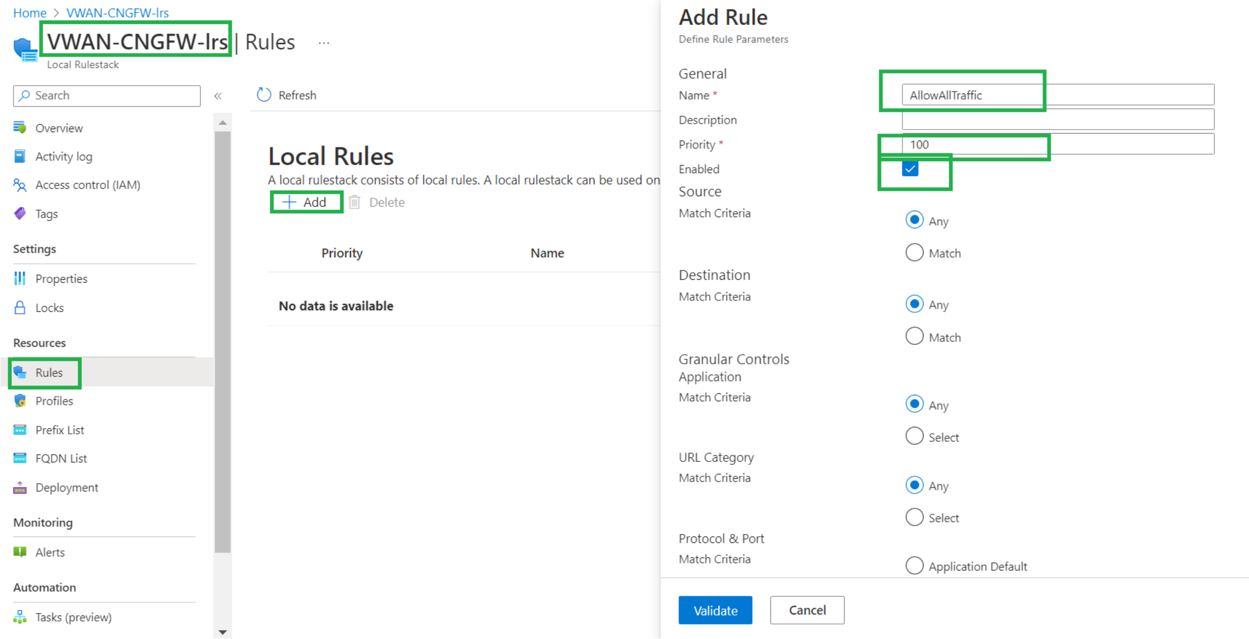
- Azure Virtual WAN Overview | Microsoft Learn
- Hub-spoke network topology with Azure Virtual WAN
- Monitoring Virtual WAN - Azure Virtual WAN | Microsoft Learn
- Install Azure Firewall in a Virtual WAN hub - Azure Virtual WAN
- Monitoring Azure Virtual WAN - Data reference | Microsoft Learn
- Azure Virtual WAN FAQ | Microsoft Learn
- What's new in Azure Virtual WAN? | Microsoft Learn
- Disaster recovery design for Azure Virtual WAN | Microsoft Learn

Continue searching in Azure Active Directory

Searching all subscriptions. Change   Give feedback

Click the Local Rulestacks to navigate to the rulestack associated with your Cloud NGFW service.

3. Click your rulestack(VWAN-CNGFW-lrs) to add rules as shown below. Modify the rules as per your use cases and functionality. Add a rule to allow traffic. Fill in the mandatory fields and use the default settings for the remaining fields:



Home > VWAN-CNGFW-lrs

VWAN-CNGFW-lrs | Rules

Local Rulestack

Local Rules

A local rulestack consists of local rules. A local rulestack can be used on

Priority	Name
No data is available	

Add Rule

Define Rule Parameters

General

Name *: AllowAllTraffic

Description:

Priority *: 100

Enabled: ☑

Source

Match Criteria: ● Any ○ Match

Destination

Match Criteria: ● Any ○ Match

Granular Controls

Application

Match Criteria: ● Any ○ Select

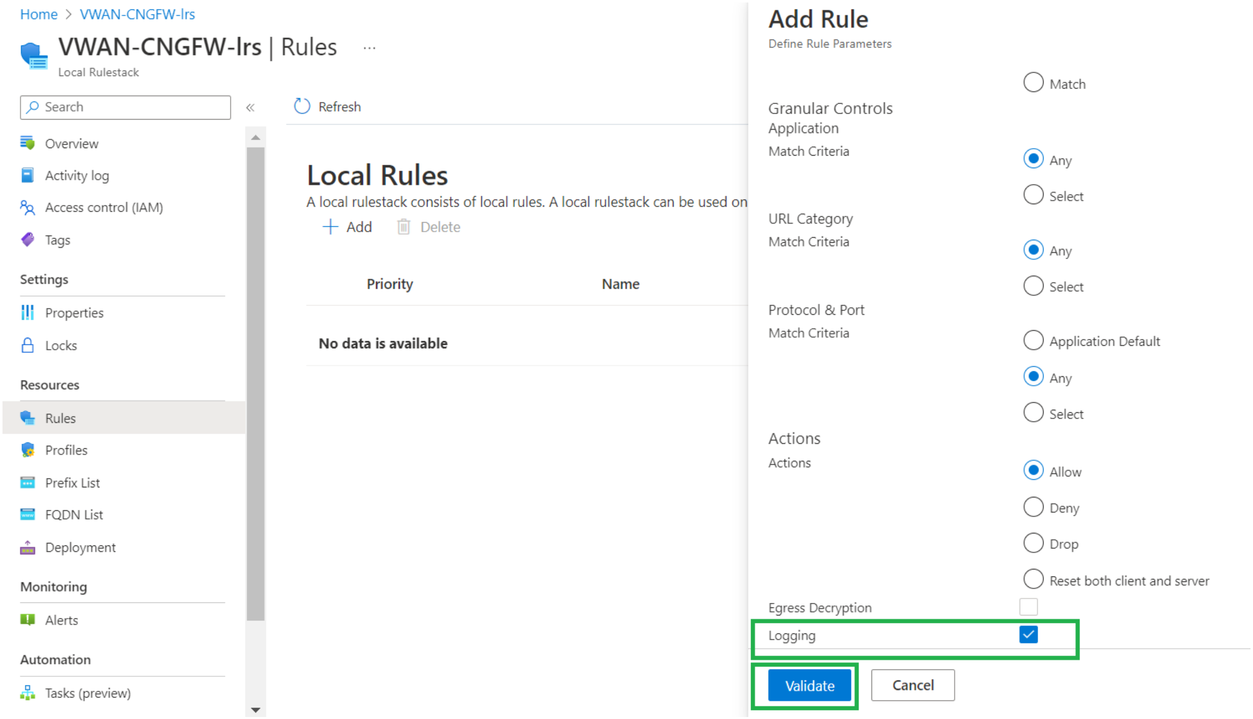
URL Category

Match Criteria: ● Any ○ Select

Protocol & Port

Match Criteria: ○ Application Default

Enable logging as part of the rule configuration, as shown below:



Add Rule

Define Rule Parameters

○ Match

Granular Controls Application Match Criteria: ● Any ○ Select

URL Category Match Criteria: ● Any ○ Select

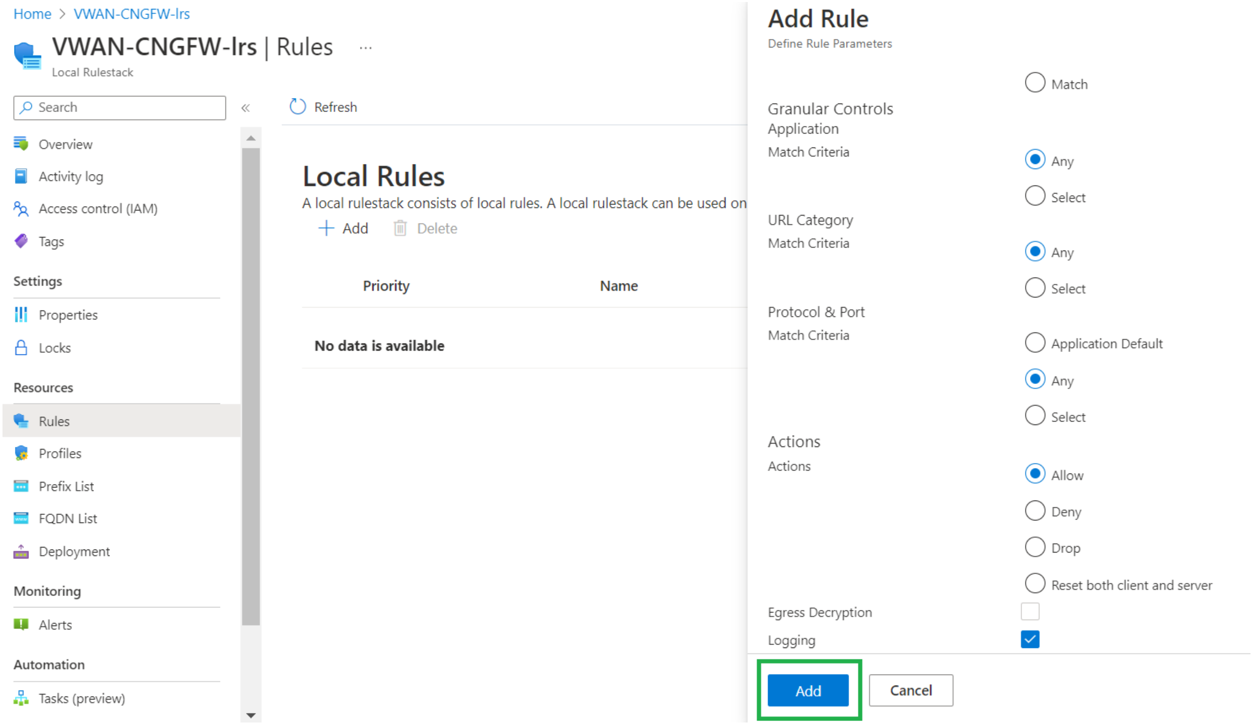
Protocol & Port Match Criteria: ○ Application Default ● Any ○ Select

Actions

Actions: ● Allow ○ Deny ○ Drop ○ Reset both client and server

Egress Decryption ☐

Logging ☑



Add Rule

Define Rule Parameters

○ Match

Granular Controls Application Match Criteria: ● Any ○ Select

URL Category Match Criteria: ● Any ○ Select

Protocol & Port Match Criteria: ○ Application Default ● Any ○ Select

Actions

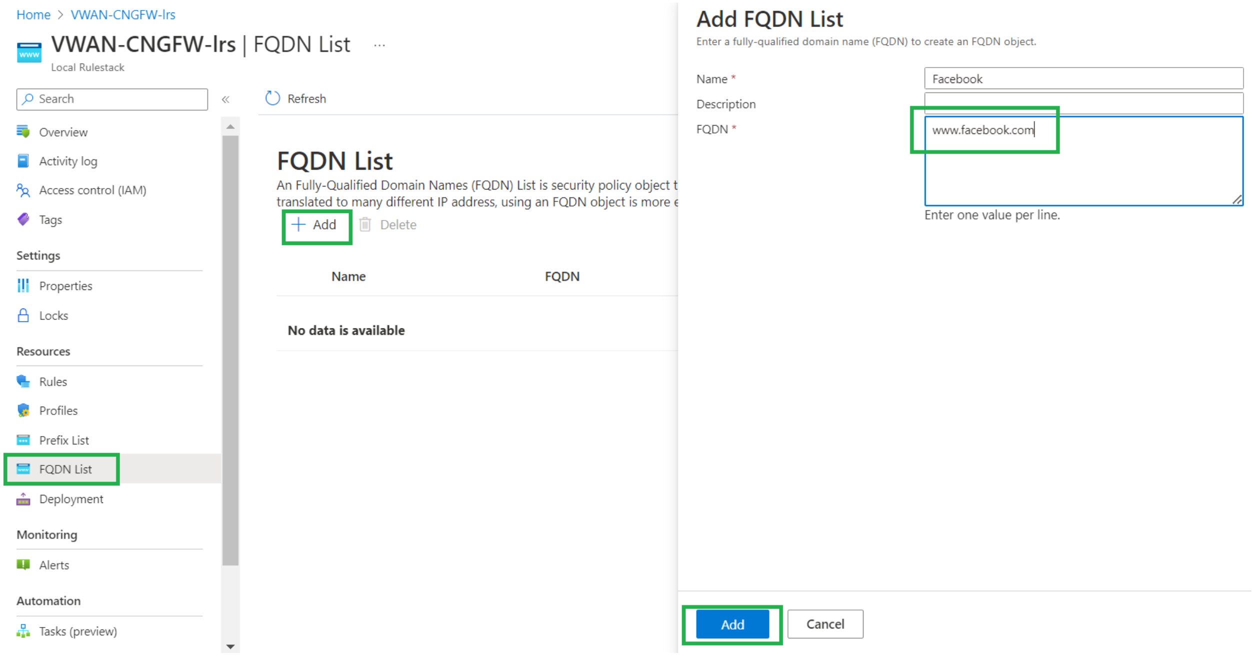
Actions: ● Allow ○ Deny ○ Drop ○ Reset both client and server

Egress Decryption ☐

Logging ☑

Click **Validate** and then **Add** to incorporate the rule.

4. Add an FQDN list that includes Facebook, and use this list to add a rule to block facebook.com:



FQDN List

An Fully-Qualified Domain Names (FQDN) List is security policy object t
translated to many different IP address, using an FQDN object is more e

Add FQDN List

Enter a fully-qualified domain name (FQDN) to create an FQDN object.

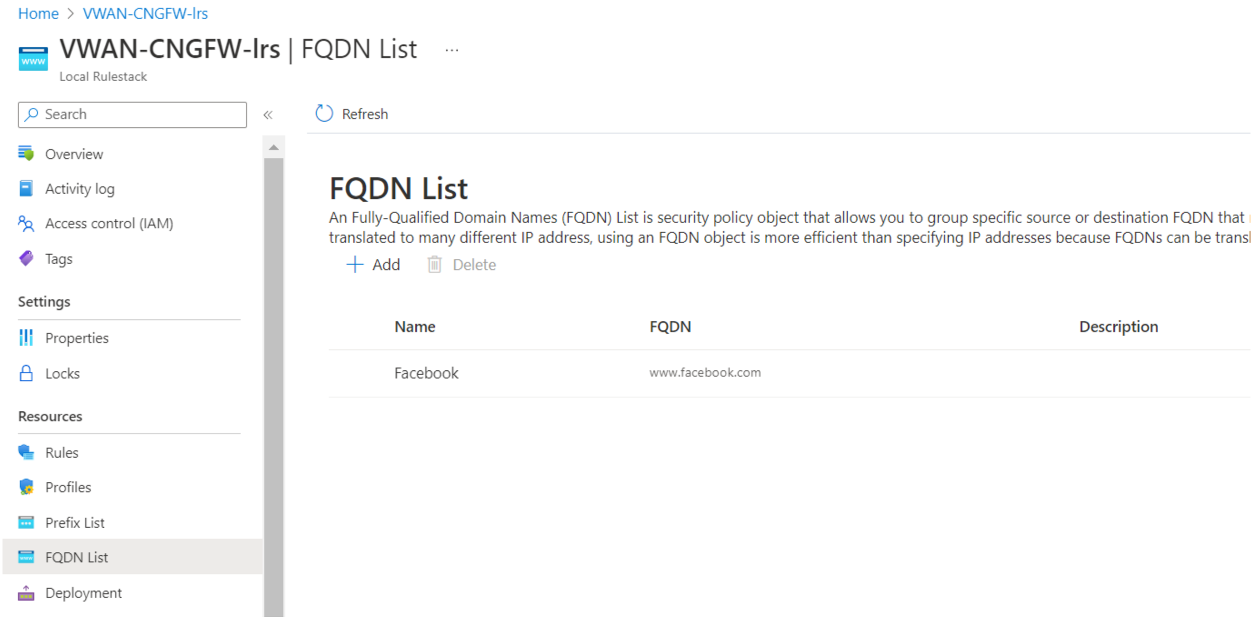
Name * Facebook

Description

FQDN * www.facebook.com

Enter one value per line.

Facebook now appears in the **FQDN List**:

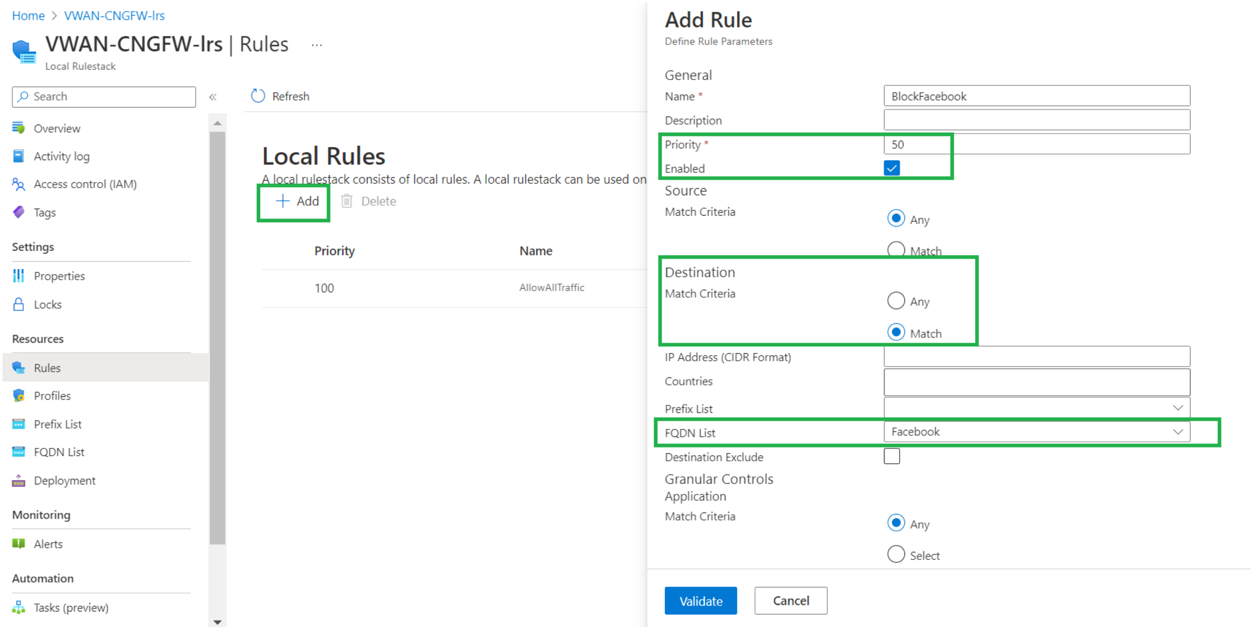


FQDN List

An Fully-Qualified Domain Names (FQDN) List is security policy object that allows you to group specific source or destination FQDN that
translated to many different IP address, using an FQDN object is more efficient than specifying IP addresses because FQDNs can be transl

Name	FQDN	Description
Facebook	www.facebook.com	

Return to the Rules setting page and add a rule that matches the FQDN list created. Set the action to **Drop** to block Facebook traffic:



Local Rules

A local rulestack consists of local rules. A local rulestack can be used on

Priority	Name
100	AllowAllTraffic

Add Rule

Define Rule Parameters

General

Name *	BlockFacebook
Description	
Priority *	50
Enabled	☑

Source

Match Criteria: ● Any ○ Match

Destination

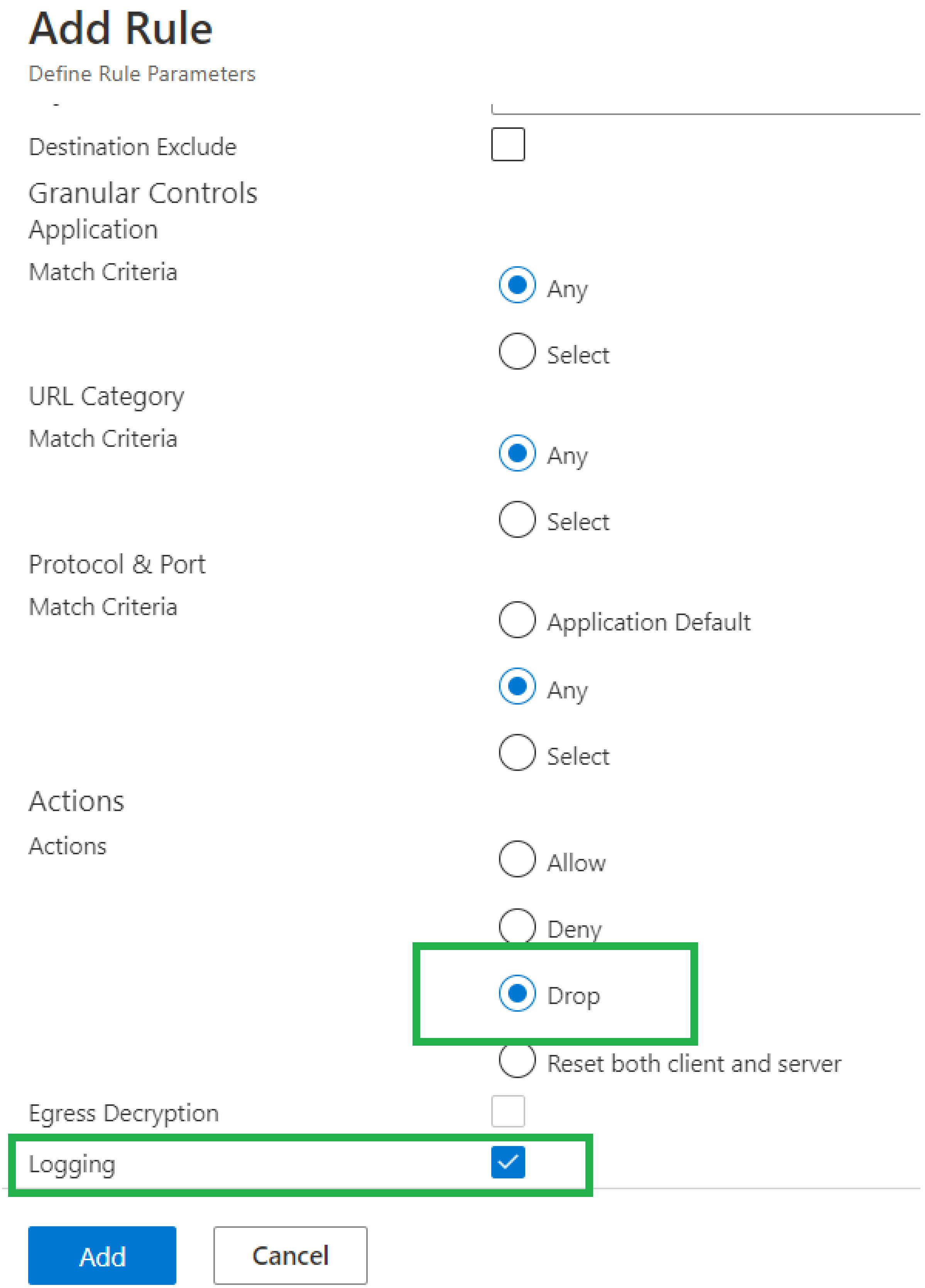
Match Criteria: ○ Any ● Match

IP Address (CIDR Format)	
Countries	
Prefix List	
FQDN List	Facebook
Destination Exclude	□

Granular Controls

Application

Match Criteria: ● Any ○ Select



Add Rule

Define Rule Parameters

Destination Exclude □

Granular Controls

Application

Match Criteria ● Any ○ Select

URL Category

Match Criteria ● Any ○ Select

Protocol & Port

Match Criteria ○ Application Default ● Any ○ Select

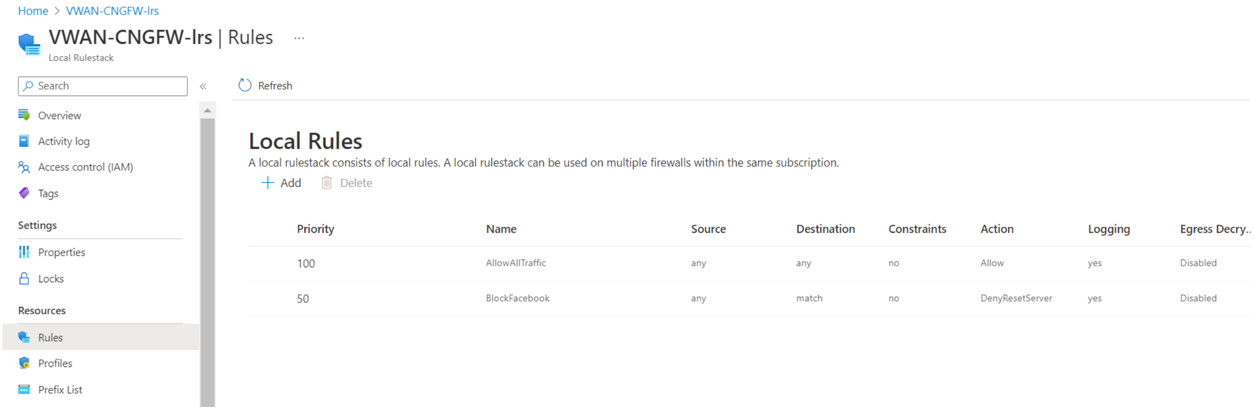
Actions

Actions ○ Allow ○ Deny ● Drop ○ Reset both client and server

Egress Decryption □

Logging ☑

5. Both the rules appear as shown below:

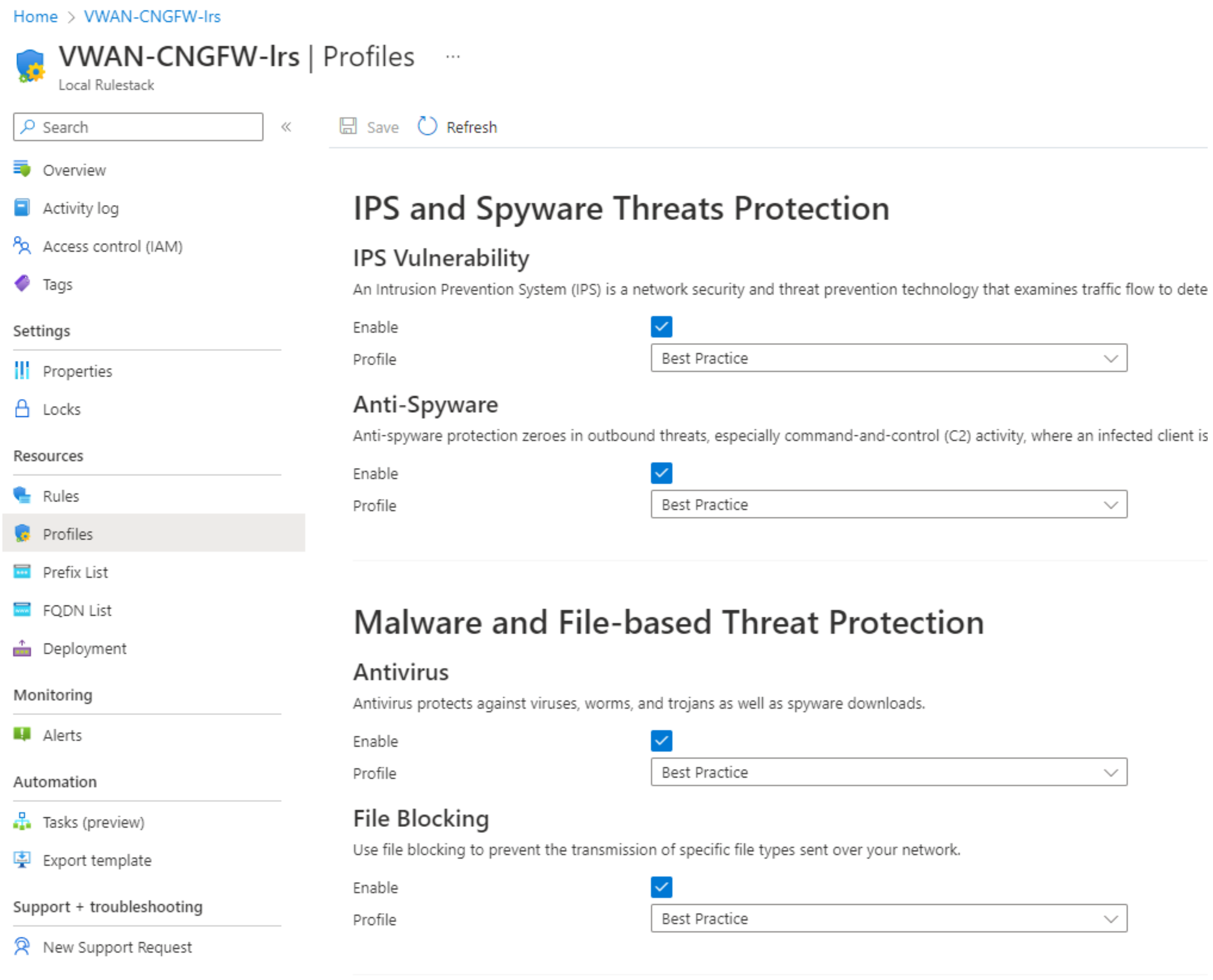


Local Rules

A local rulestack consists of local rules. A local rulestack can be used on multiple firewalls within the same subscription.

Priority	Name	Source	Destination	Constraints	Action	Logging	Egress Decry..
100	AllowAllTraffic	any	any	no	Allow	yes	Disabled
50	BlockFacebook	any	match	no	DenyResetServer	yes	Disabled

6. As part of this Cloud NGFW service, the security profiles are enabled with best practice configurations by default. This means that the traffic is secured with the best security profiles from day one, once the Cloud NGFW is deployed in the network:



IPS and Spyware Threats Protection

IPS Vulnerability

An Intrusion Prevention System (IPS) is a network security and threat prevention technology that examines traffic flow to dete

Enable ☑

Profile: Best Practice

Anti-Spyware

Anti-spyware protection zeroes in outbound threats, especially command-and-control (C2) activity, where an infected client is

Enable ☑

Profile: Best Practice

Malware and File-based Threat Protection

Antivirus

Antivirus protects against viruses, worms, and trojans as well as spyware downloads.

Enable ☑

Profile: Best Practice

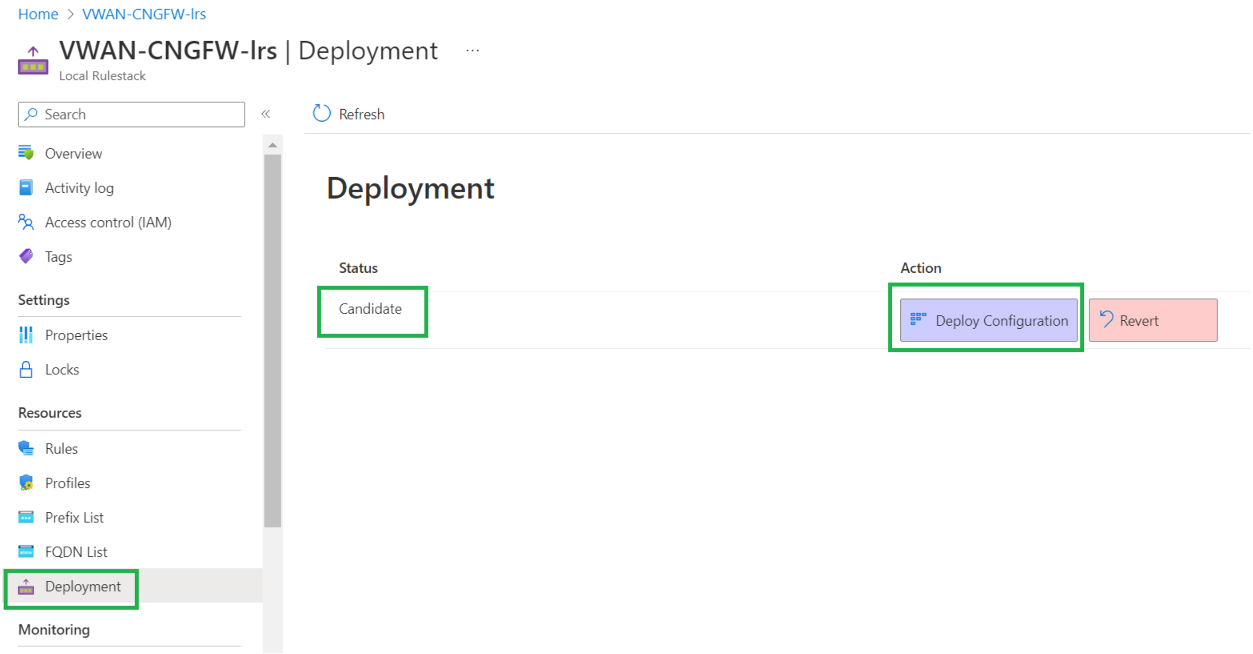
File Blocking

Use file blocking to prevent the transmission of specific file types sent over your network.

Enable ☑

Profile: Best Practice

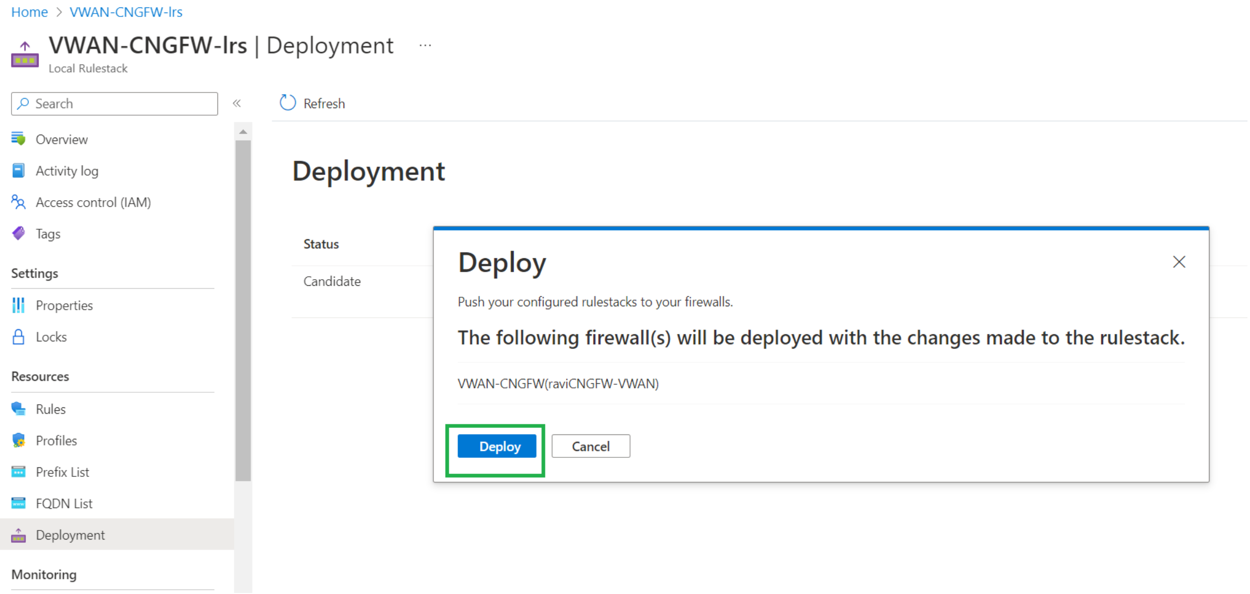
7. Now that the rules have been modified, they should be deployed onto the Local rulestack associated with the Cloud NGFW service. Click the **Deployment** tab to see the page below. The deployment status displays as **Candidate**, which means the configuration was built but not deployed. Next, click **Deploy Configuration** to deploy the configuration onto the Cloud NGFW service. *It is mandatory to do this step as without this the configuration will not be deployed onto the rulestack.*



Deployment

Status	Action
Candidate	Deploy Configuration Revert

After clicking **Deploy Configuration,** a pop-up displays the firewalls associated with this rulestack. Click **Deploy** to configure this rulestack on all the associated firewalls:



Deploy

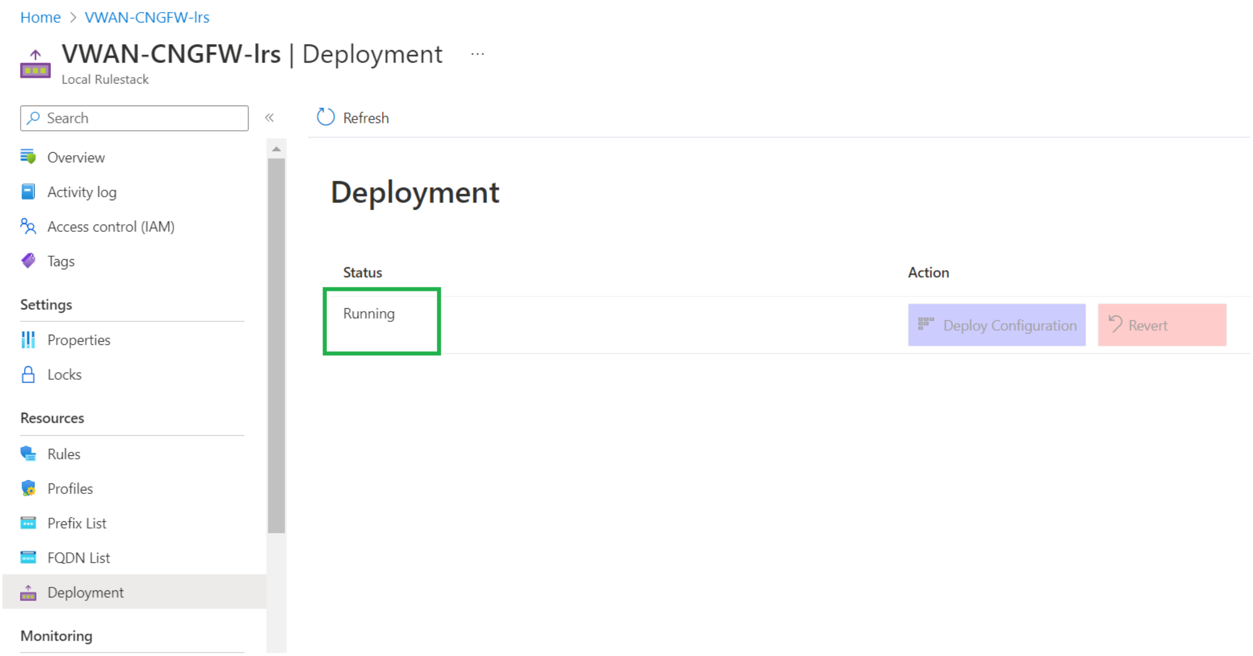
Push your configured rulestacks to your firewalls.

The following firewall(s) will be deployed with the changes made to the rulestack.

VWAN-CNGFW(raviCNGFW-VWAN)

Deploy Cancel

After successfully deploying the configuration, the screen displays the deployment status as **Running**



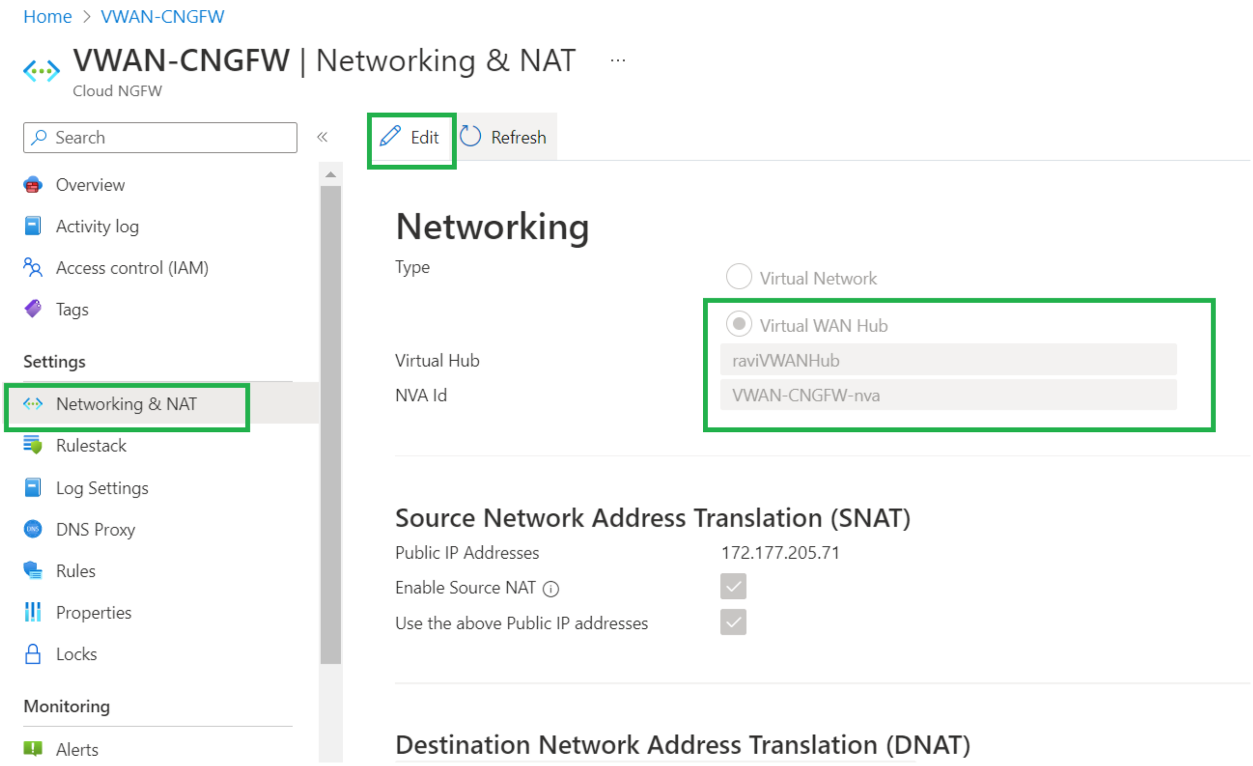
Status	Action
Running	

With this, the Cloud NGFW and Local rulestack are successfully deployed.

Source/Destination NAT rule on Cloud NGFW

Configure a destination NAT rule with frontend configuration on Cloud NGFW to take care of Inbound traffic towards App1 or App2 present on spoke VNet1 or spoke VNet2.

1. Access the **Networking & NAT** settings screen for the cloud NGFW resource. The first thing to observe is whether the network type is selected as “**Virtual WAN Hub**” and **Source NAT** setting has been enabled or not. During the creation of the Cloud NGFW resource,(step 6) if Source NAT was enabled, that’s how it will show up here.
2. Click **Edit** to add the Destination NAT rule.



Networking

Type: ○ Virtual Network ● Virtual WAN Hub

Virtual Hub: raviVWANHub

NVA Id: VWAN-CNGFW-nva

Source Network Address Translation (SNAT)

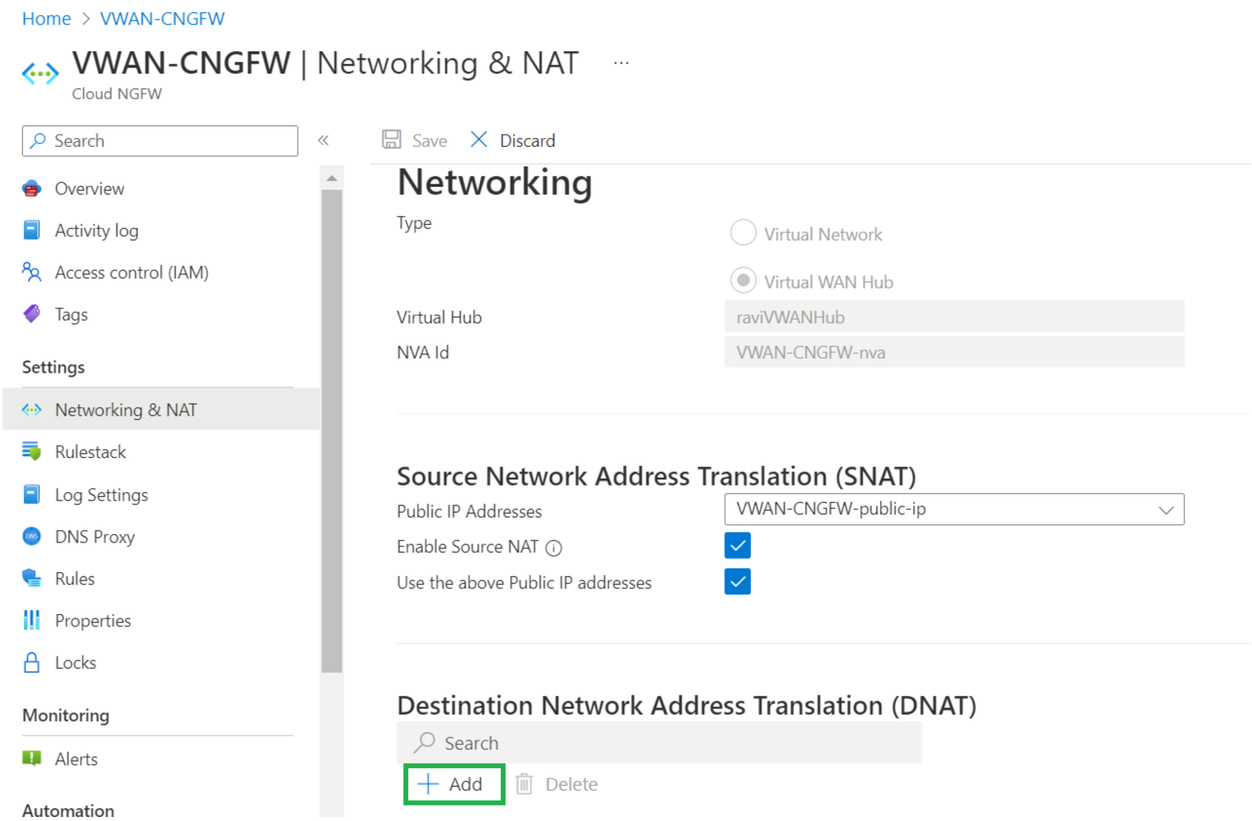
Public IP Addresses: 172.177.205.71

Enable Source NAT ⓘ: ☑

Use the above Public IP addresses: ☑

Destination Network Address Translation (DNAT)

3. Add a **Destination NAT** rule with frontend configuration as shown below. Frontend IP is the Public IP address associated with Cloud NGFW (choose this from the drop-down menu). To access App1 (192.168.0.4), deployed on spoke VNet1, on port 80(HTTP), we are going to use Cloud NGFW frontend IP address and port 8080. After adding the Destination NAT rule, save the configuration by clicking **Add** .



Networking

Type: ○ Virtual Network ● Virtual WAN Hub

Virtual Hub: raviVWANHub

NVA Id: VWAN-CNGFW-nva

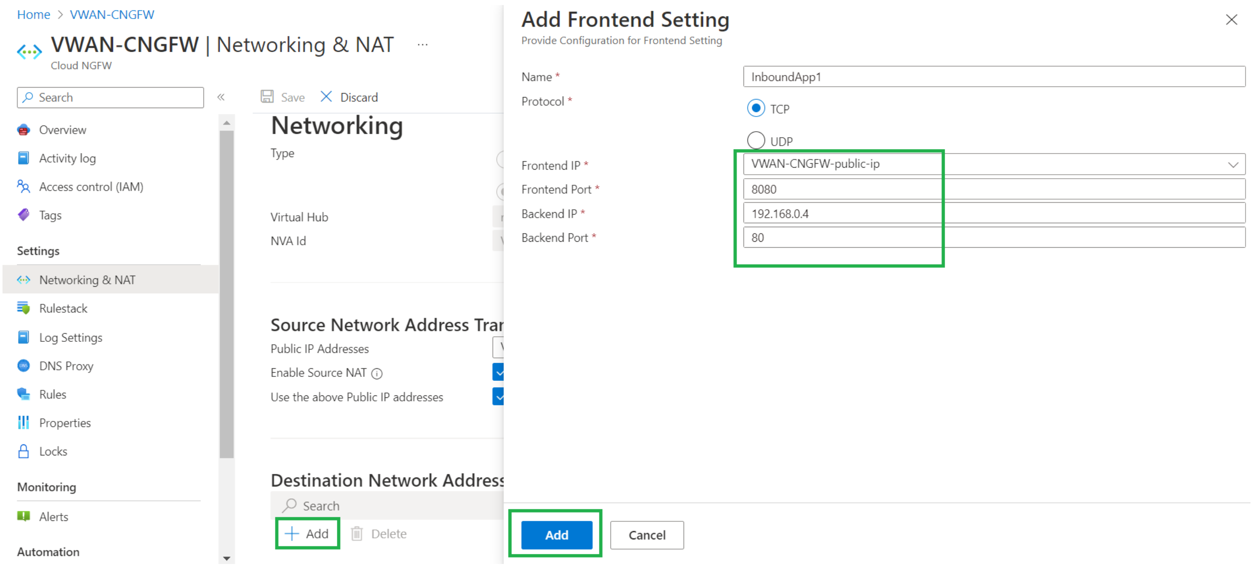
Source Network Address Translation (SNAT)

Public IP Addresses: VWAN-CNGFW-public-ip

Enable Source NAT ⓘ ☑

Use the above Public IP addresses ☑

Destination Network Address Translation (DNAT)



Add Frontend Setting

Provide Configuration for Frontend Setting

Name *: InboundApp1

Protocol *: ● TCP ○ UDP

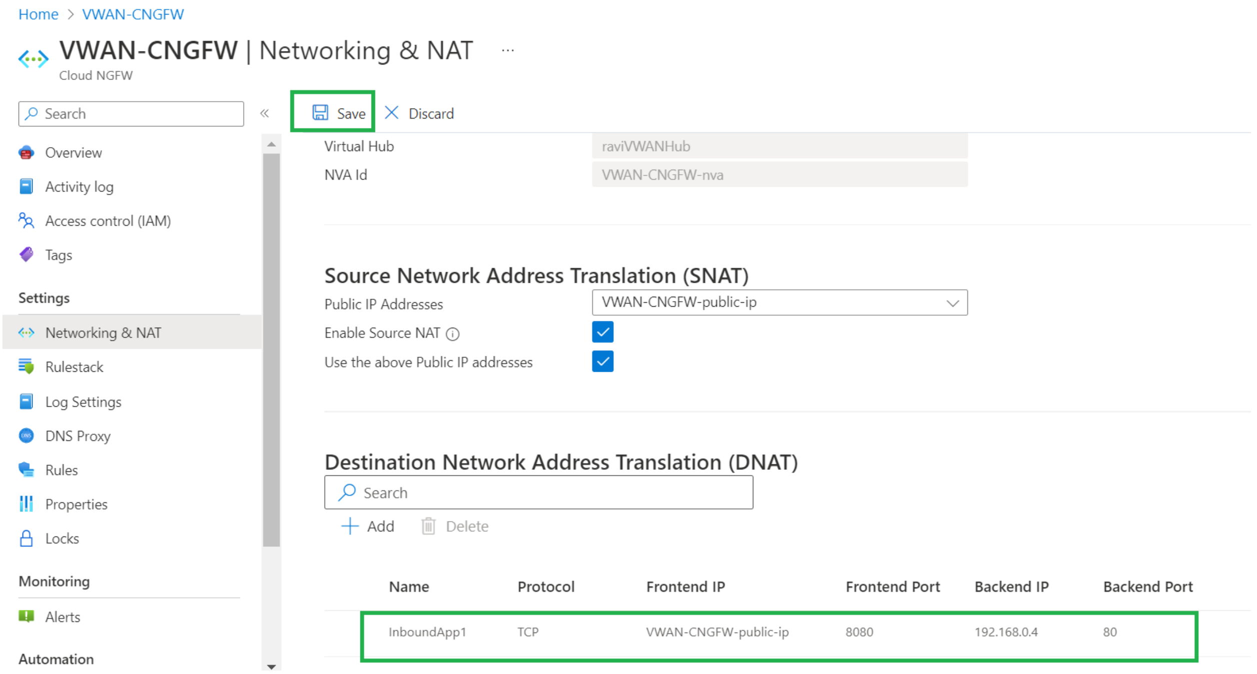
Frontend IP *: VWAN-CNGFW-public-ip

Frontend Port *: 8080

Backend IP *: 192.168.0.4

Backend Port *: 80

Once the destination NAT rule has been added, click **Save** to deploy this configuration on to the Cloud NGFW resource:



VWAN-CNGFW | Networking & NAT

Cloud NGFW

Virtual Hub: raviVWANHub

NVA Id: VWAN-CNGFW-nva

Source Network Address Translation (SNAT)

Public IP Addresses: VWAN-CNGFW-public-ip

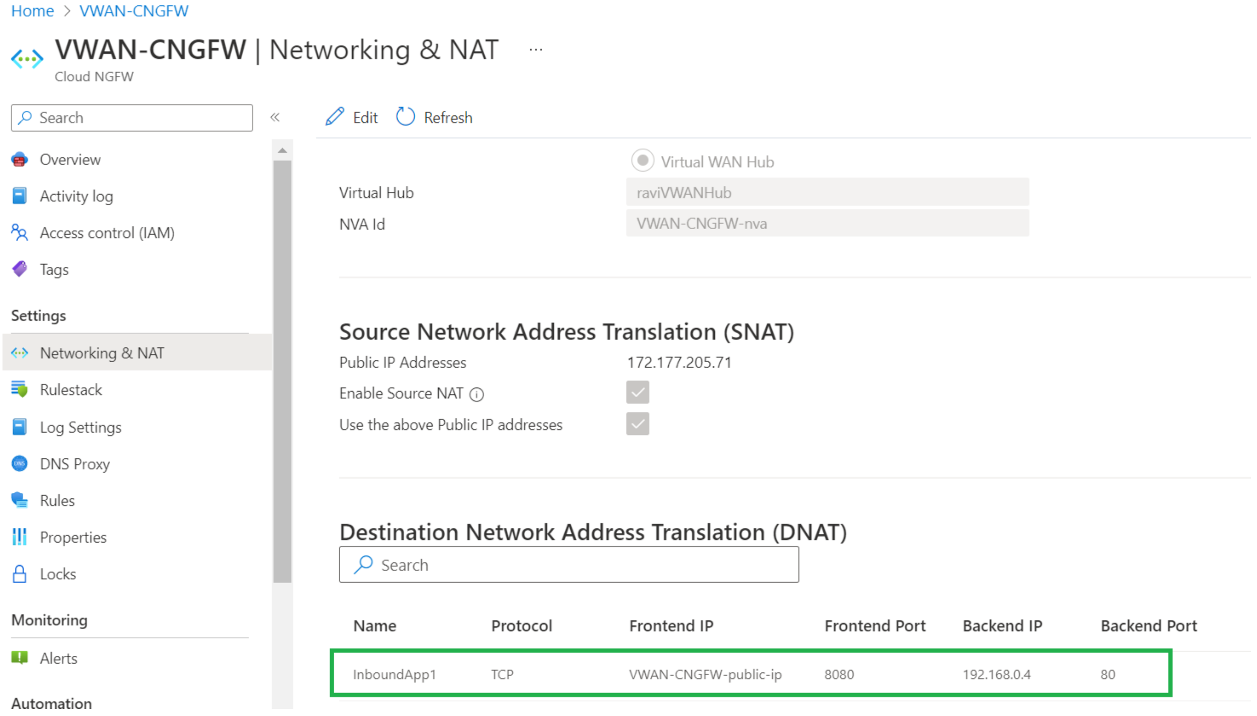
Enable Source NAT ⓘ ☑

Use the above Public IP addresses ☑

Destination Network Address Translation (DNAT)

Name	Protocol	Frontend IP	Frontend Port	Backend IP	Backend Port
InboundApp1	TCP	VWAN-CNGFW-public-ip	8080	192.168.0.4	80

After saving the configuration, the screen would look as shown below



VWAN-CNGFW | Networking & NAT

Cloud NGFW

● Virtual WAN Hub

Virtual Hub: raviVWANHub

NVA Id: VWAN-CNGFW-nva

Source Network Address Translation (SNAT)

Public IP Addresses: 172.177.205.71

Enable Source NAT ⓘ ☑

Use the above Public IP addresses ☑

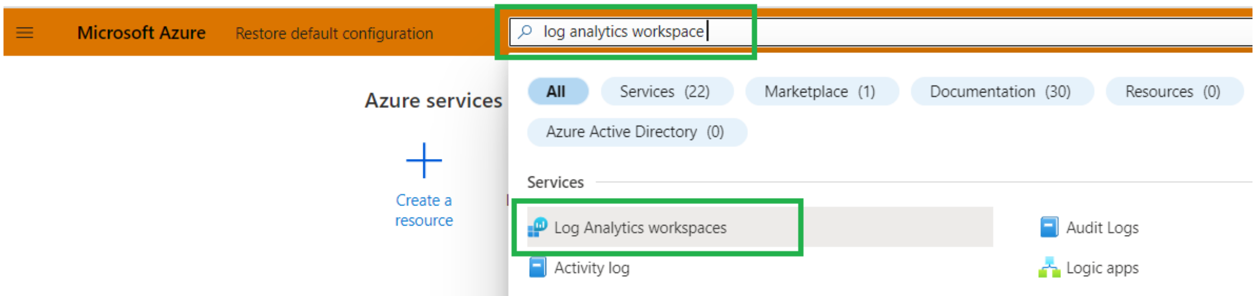
Destination Network Address Translation (DNAT)

Name	Protocol	Frontend IP	Frontend Port	Backend IP	Backend Port
InboundApp1	TCP	VWAN-CNGFW-public-ip	8080	192.168.0.4	80

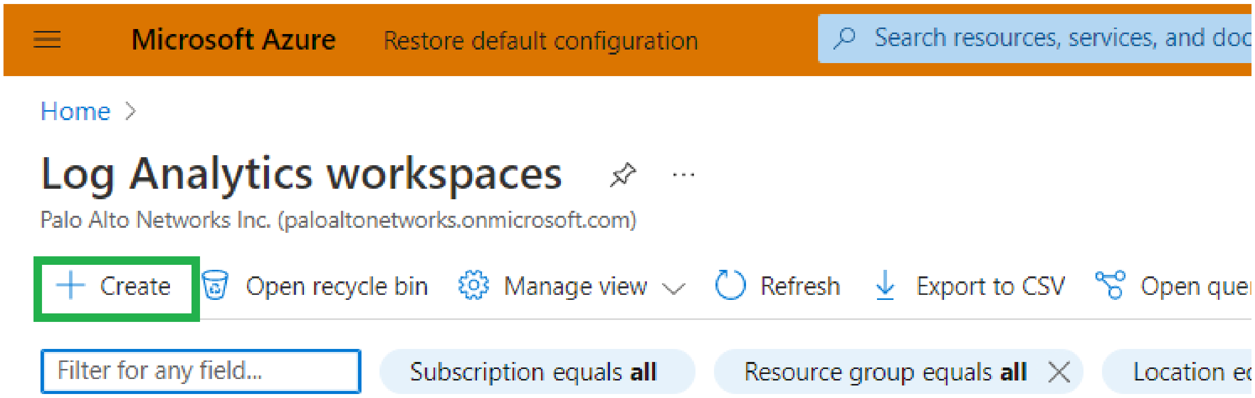
With this configuration in place, the address http://frondendIP:8080 is redirected to App1 on port 80 through Cloud NGFW. This means that inbound traffic is now flowing through the Cloud NGFW.

Configure Logging

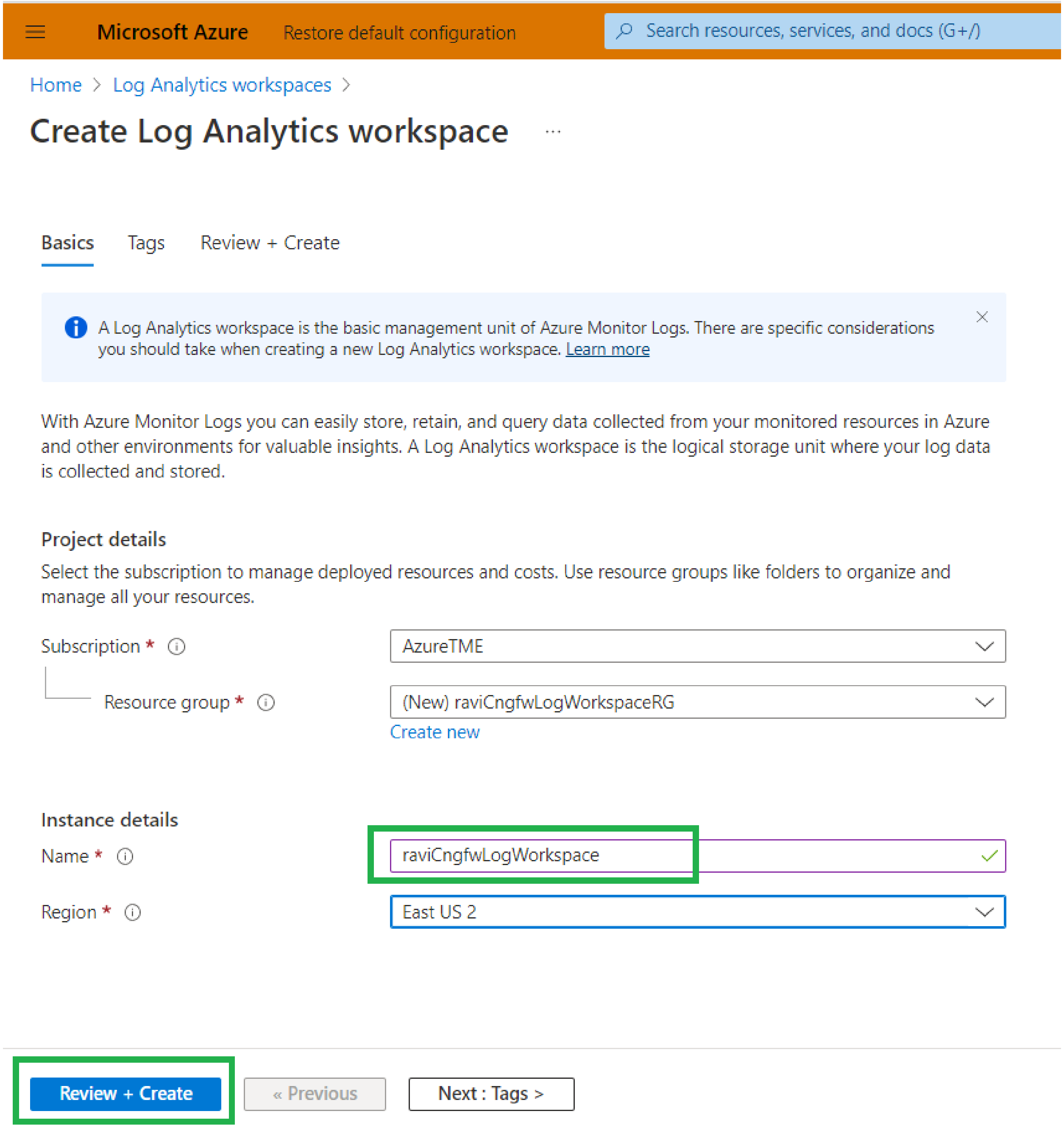
1. Before configuring Log settings on Cloud NGFW, create the Log Analytics workspace on Azure. Search for **Azure Log Analytics** workspace as shown below and click **Log Analytics Workspaces** service to add it to the workspace:



2. Click on Create option to create a new Log Analytics Workspace.



3. Create the **Log analytics workspace** as shown below. Make sure that the region is either US-East-2 or US-central:


Microsoft Azure
Restore default configuration
Search resources, services, and docs (G+/)
Home > Log Analytics workspaces >

Create Log Analytics workspace

Basics
Tags
Review + Create

A Log Analytics workspace is the basic management unit of Azure Monitor Logs. There are specific considerations you should take when creating a new Log Analytics workspace. Learn more

With Azure Monitor Logs you can easily store, retain, and query data collected from your monitored resources in Azure and other environments for valuable insights. A Log Analytics workspace is the logical storage unit where your log data is collected and stored.

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * AzureTME

Resource group * (New) raviCngfwLogWorkspaceRG

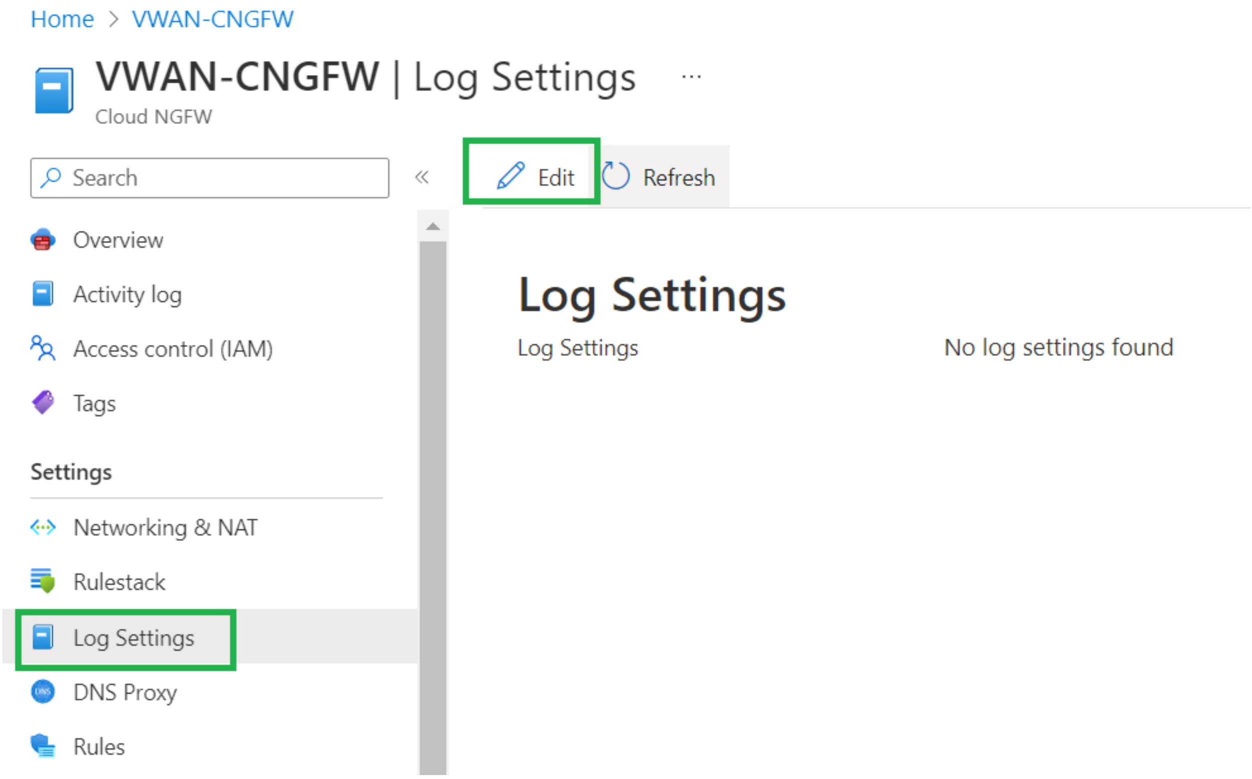
Create new

Instance details

Name * raviCngfwLogWorkspace

Region * East US 2

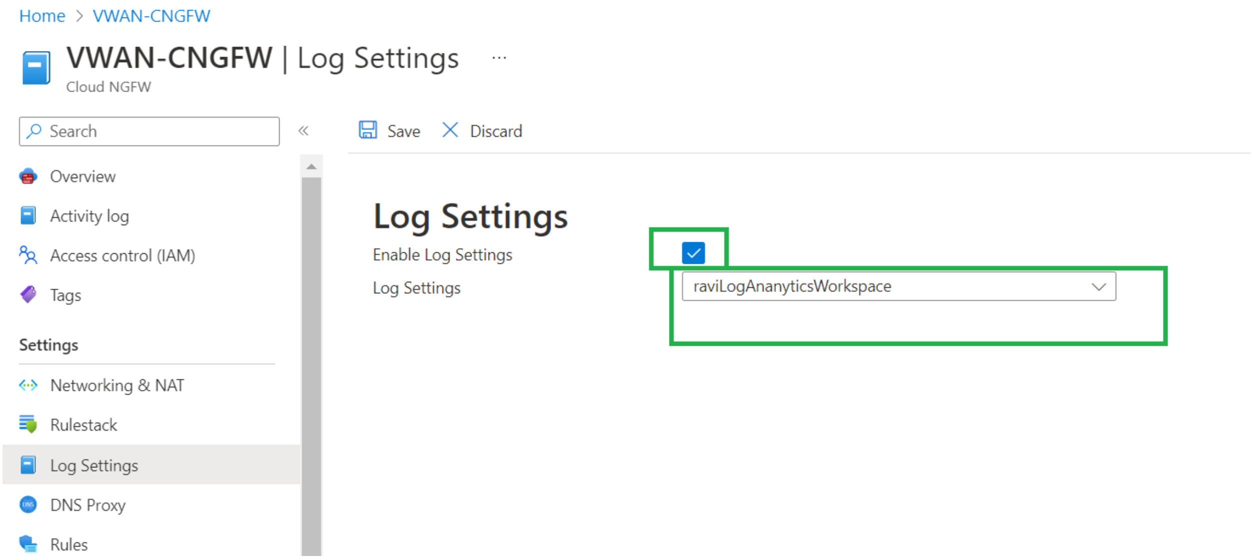
4. Now configure Cloud NGFW Log settings using the Log Analytics workspace created above. Go to the Cloud NGFW resource, select the **Log Settings** section, and click the **Edit** option to choose the Log analytics workspace that has just been created:



Log Settings

Log Settings No log settings found

5. Enable **Log Settings** and choose the log analytics workspace created in the previous step from the drop-down, and save the configuration:



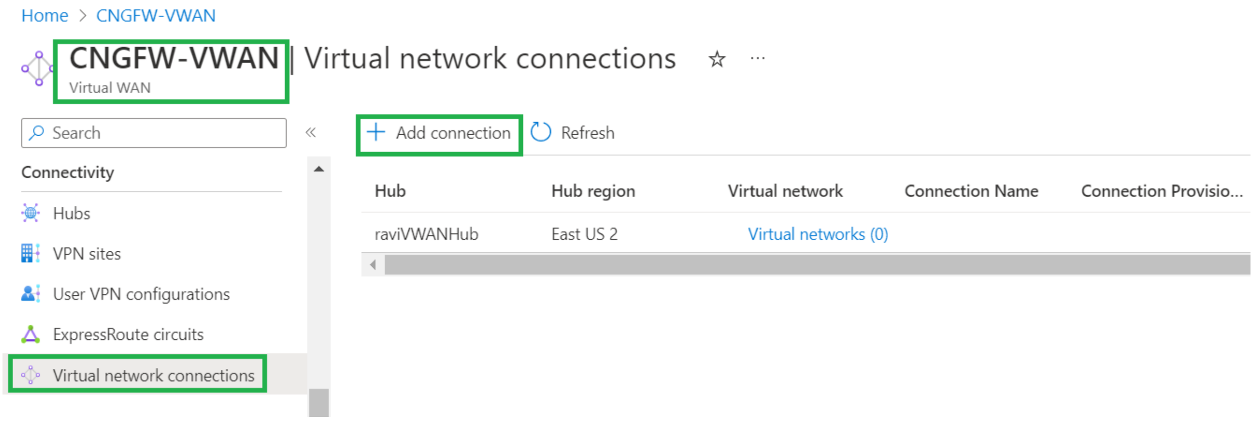
Log Settings

Enable Log Settings ☑

Log Settings raviLogAnanyticsWorkspace

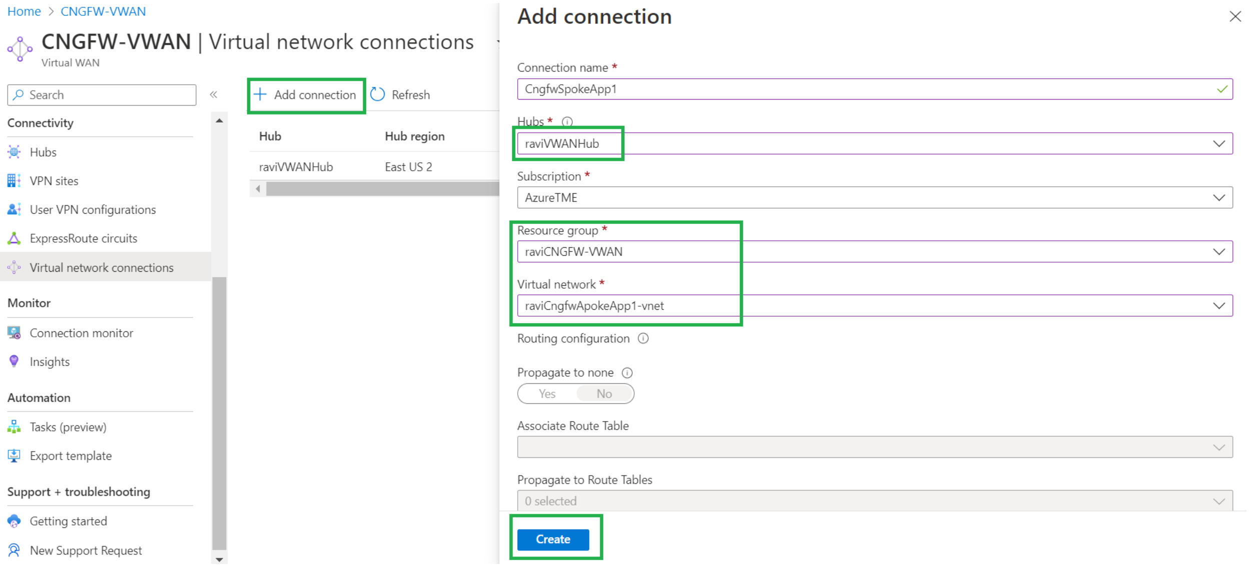
Add Spoke(Application) VNets as Virtual Network Connections to Virtual WAN

1. Add spoke vnets as **Virtual Network Connections** to Virtual WAN Hub by clicking on “**+ Add Connection**” as shown below.



Hub	Hub region	Virtual network	Connection Name	Connection Provisio...
raviVWANHub	East US 2	Virtual networks (0)		

2. Select Spoke1 VNet as Virtual Network while adding the connection as shown below.



Connection name * CngfwSpokeApp1

Hubs * raviVWANHub

Subscription * AzureTME

Resource group * raviCNGFW-VWAN

Virtual network * raviCngfwApokeApp1-vnet

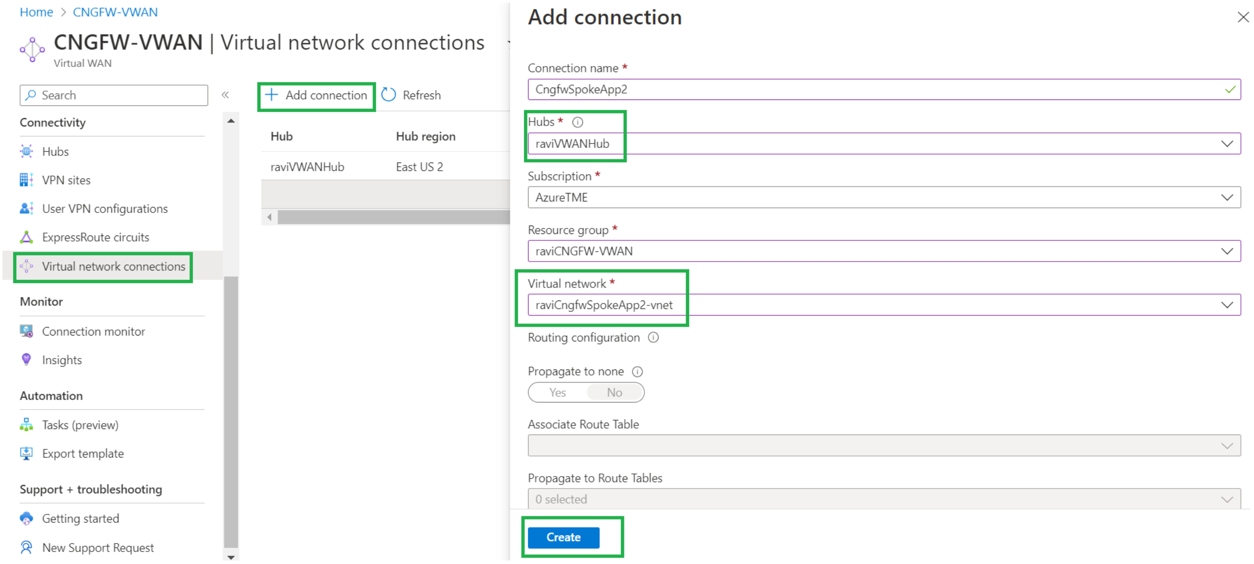
Routing configuration

Propagate to none: Yes No

Associate Route Table

Propagate to Route Tables: 0 selected

3. Similarly select Spoke2 VNet as Virtual Network while adding the connection as shown below.



Hub	Hub region
raviVWANHub	East US 2

Add connection

Connection name * CngfwSpokeApp2

Hubs * raviVWANHub

Subscription * AzureTME

Resource group * raviCNGFW-VWAN

Virtual network * raviCngfwSpokeApp2-vnet

Routing configuration

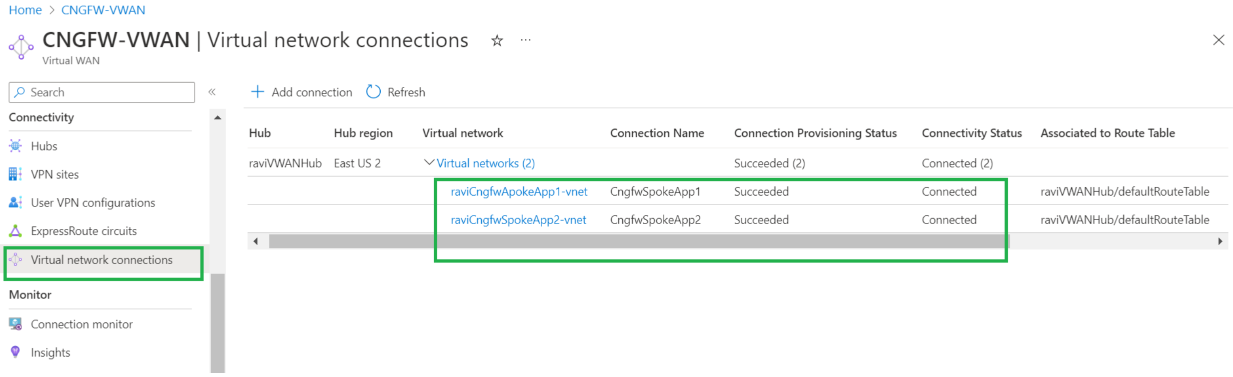
Propagate to none Yes No

Associate Route Table

Propagate to Route Tables 0 selected

Create

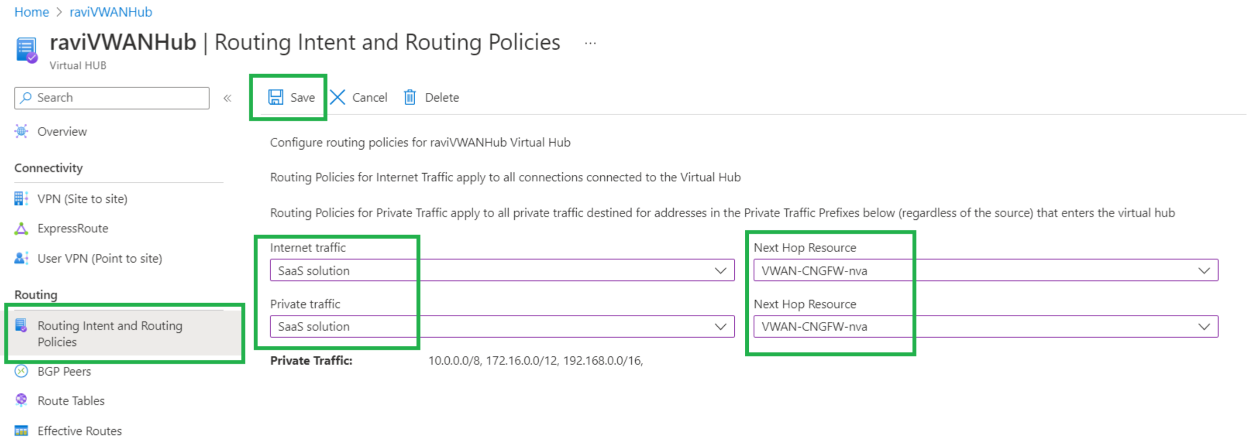
4. After successful addition of the connections, it would look something as shown below. Make sure that the status is in **Connected** state.



Hub	Hub region	Virtual network	Connection Name	Connection Provisioning Status	Connectivity Status	Associated to Route Table
raviVWANHub	East US 2	Virtual networks (2)		Succeeded (2)	Connected (2)	
		raviCngfwApokeApp1-vnet	CngfwSpokeApp1	Succeeded	Connected	raviVWANHub/defaultRouteTable
		raviCngfwSpokeApp2-vnet	CngfwSpokeApp2	Succeeded	Connected	raviVWANHub/defaultRouteTable

Configure VWAN Hub Routing Intent and Routing Policies

1. Routing Policies within Virtual WAN Hub will be used to route traffic through Cloud NGFW service.
2. To route Internet bound traffic and Private Traffic(Spoke to Spoke), configure the next hop as VWAN Cloud NGFW as shown below



Home > raviVWANHub

raviVWANHub | Routing Intent and Routing Policies

Virtual HUB

Save Cancel Delete

Configure routing policies for raviVWANHub Virtual Hub

Routing Policies for Internet Traffic apply to all connections connected to the Virtual Hub

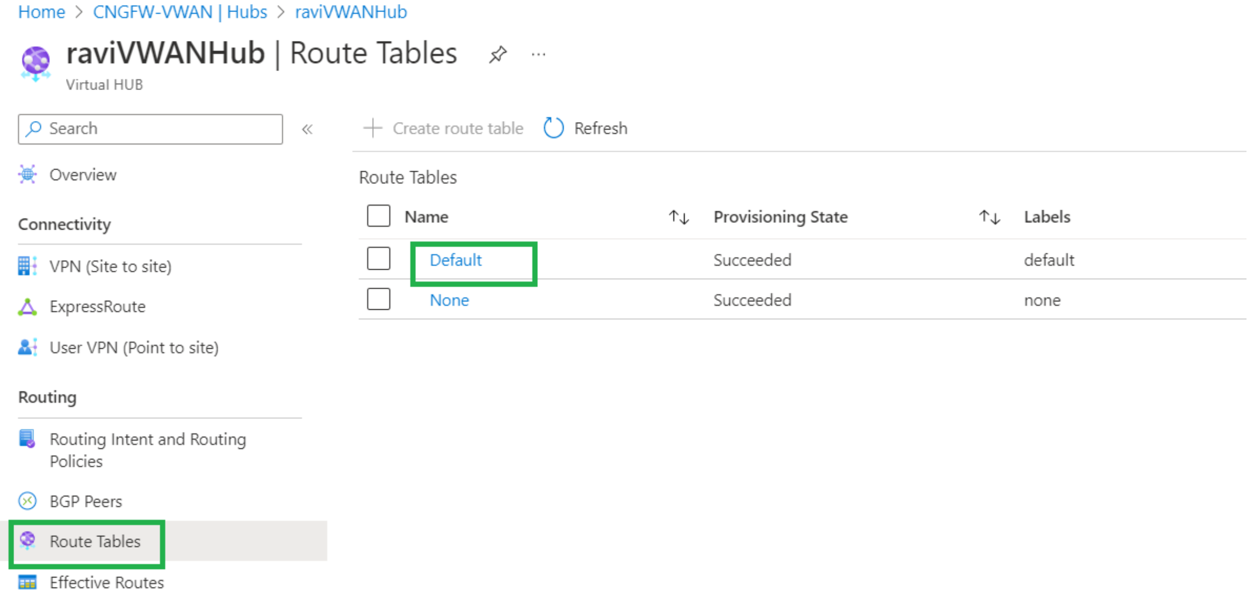
Routing Policies for Private Traffic apply to all private traffic destined for addresses in the Private Traffic Prefixes below (regardless of the source) that enters the virtual hub

Internet traffic: SaaS solution — Next Hop Resource: VWAN-CNGFW-nva

Private traffic: SaaS solution — Next Hop Resource: VWAN-CNGFW-nva

Private Traffic: 10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16,

3. After configuring Routing Policies, check for the routing table to be updated to route traffic through Cloud NGFW
   Click on **Route Tables** and select **Default** routing table



Home > CNGFW-VWAN | Hubs > raviVWANHub

raviVWANHub | Route Tables

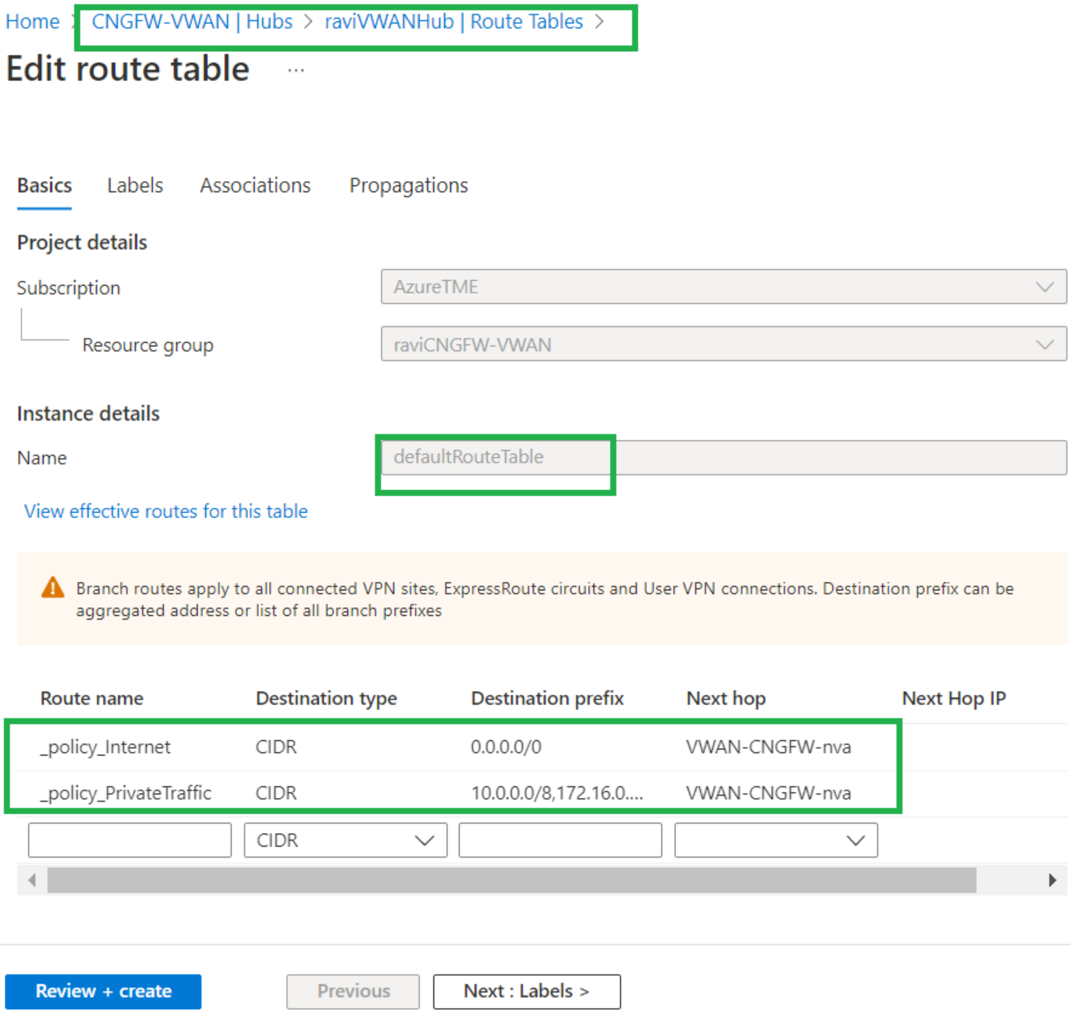
Virtual HUB

Create route table Refresh

Route Tables

☐	Name	Provisioning State	Labels
☐	Default	Succeeded	default
☐	None	Succeeded	none

This will provide the details related to the routes associated with the Default Routing table. Over here we can see that any traffic going out to internet or to other spoke VNets will be routed through Cloud NGFW



Home > CNGFW-VWAN | Hubs > raviVWANHub | Route Tables >

Edit route table ...

Project details

Subscription: AzureTME

Resource group: raviCNGFW-VWAN

Instance details

Name: defaultRouteTable

View effective routes for this table

Branch routes apply to all connected VPN sites, ExpressRoute circuits and User VPN connections. Destination prefix can be aggregated address or list of all branch prefixes

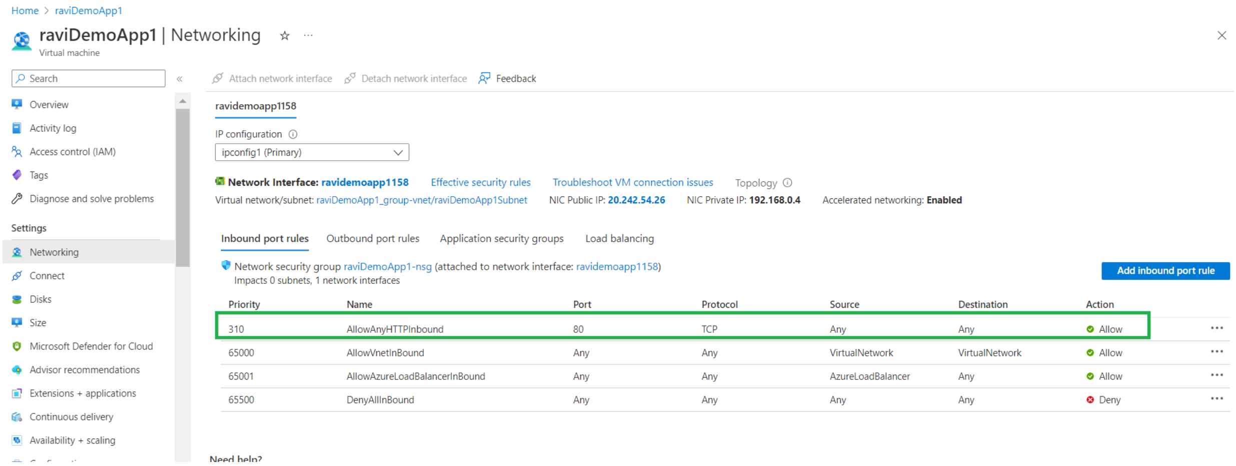
Route name	Destination type	Destination prefix	Next hop	Next Hop IP
_policy_Internet	CIDR	0.0.0.0/0	VWAN-CNGFW-nva	
_policy_PrivateTraffic	CIDR	10.0.0.0/8,172.16.0....	VWAN-CNGFW-nva	
	CIDR			

Testing traffic

Test Inbound Traffic

1. To validate the inbound connection towards App1, try to access http://<Cloud NGFW Public IP>:8080.
2. As per the Destination NAT configuration on Cloud NGFW, if http://<Cloud NGFW Public IP>:8080 is accessed, the connection will be redirected to App1 after inspection by Cloud NGFW.

Make sure to allow HTTP traffic on the application server network interface. For this, go to App1, select **Networking**, and add an inbound port rule that allows any HTTP inbound traffic. Configure the source as **IP Addresses**, port as 80, protocol as TCP, and set the **Action** to**Allow:**



IP configuration: ipconfig1 (Primary)

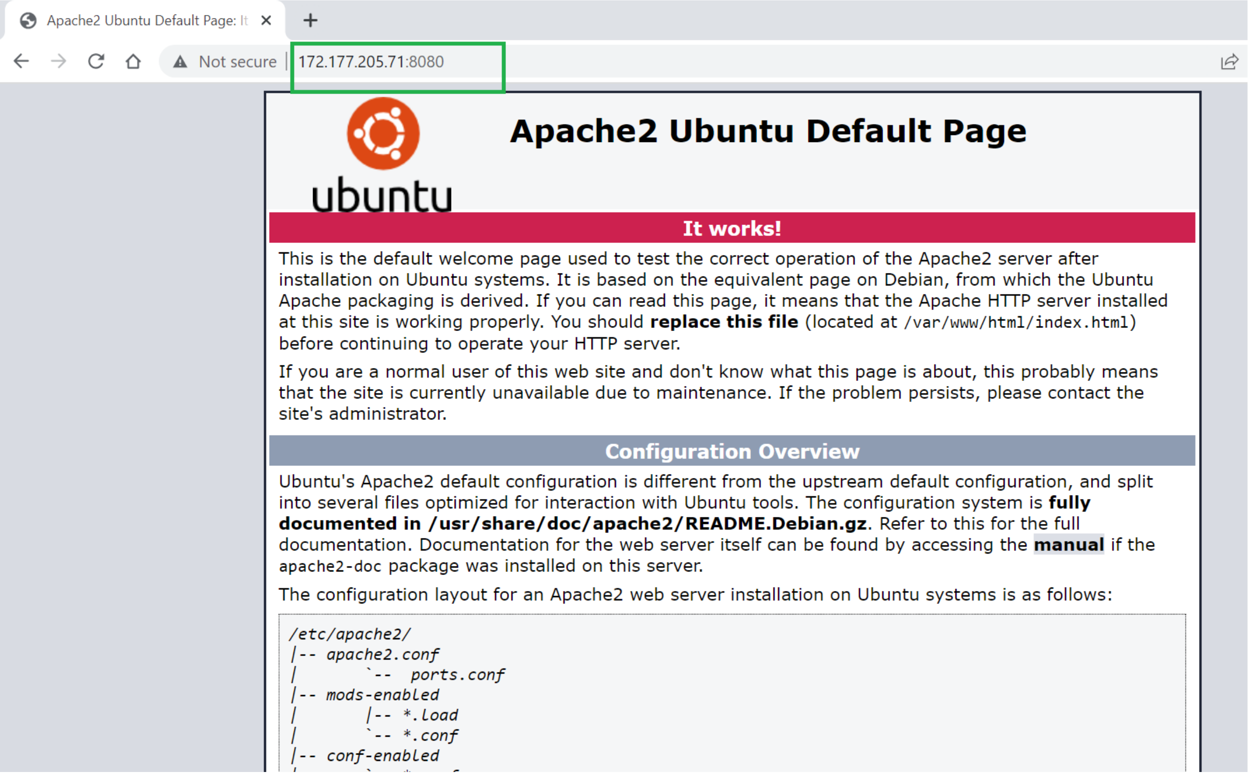
Network Interface: ravidemoapp1158

Virtual network/subnet: raviDemoApp1_group-vnet/raviDemoApp1Subnet   NIC Public IP: **20.242.54.26**   NIC Private IP: **192.168.0.4**   Accelerated networking: **Enabled**

Network security group raviDemoApp1-nsg (attached to network interface: ravidemoapp1158)
Impacts 0 subnets, 1 network interfaces

Priority	Name	Port	Protocol	Source	Destination	Action
310	AllowAnyHTTPInbound	80	TCP	Any	Any	✓ Allow
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	✓ Allow
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	✓ Allow
65500	DenyAllInBound	Any	Any	Any	Any	✗ Deny

If *http:/<Cloud NGFW Public IP>:8080/* is accessed, the following screen appears if the apache server was running as the default web server. Here, once the public IP of the Cloud NGFW was accessed, it will redirect the traffic to App1 on spoke-vnet1 where apache server was running. Since Inbound HTTP is enabled on App1, it will run the apache server that was deployed on App1.



Apache2 Ubuntu Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

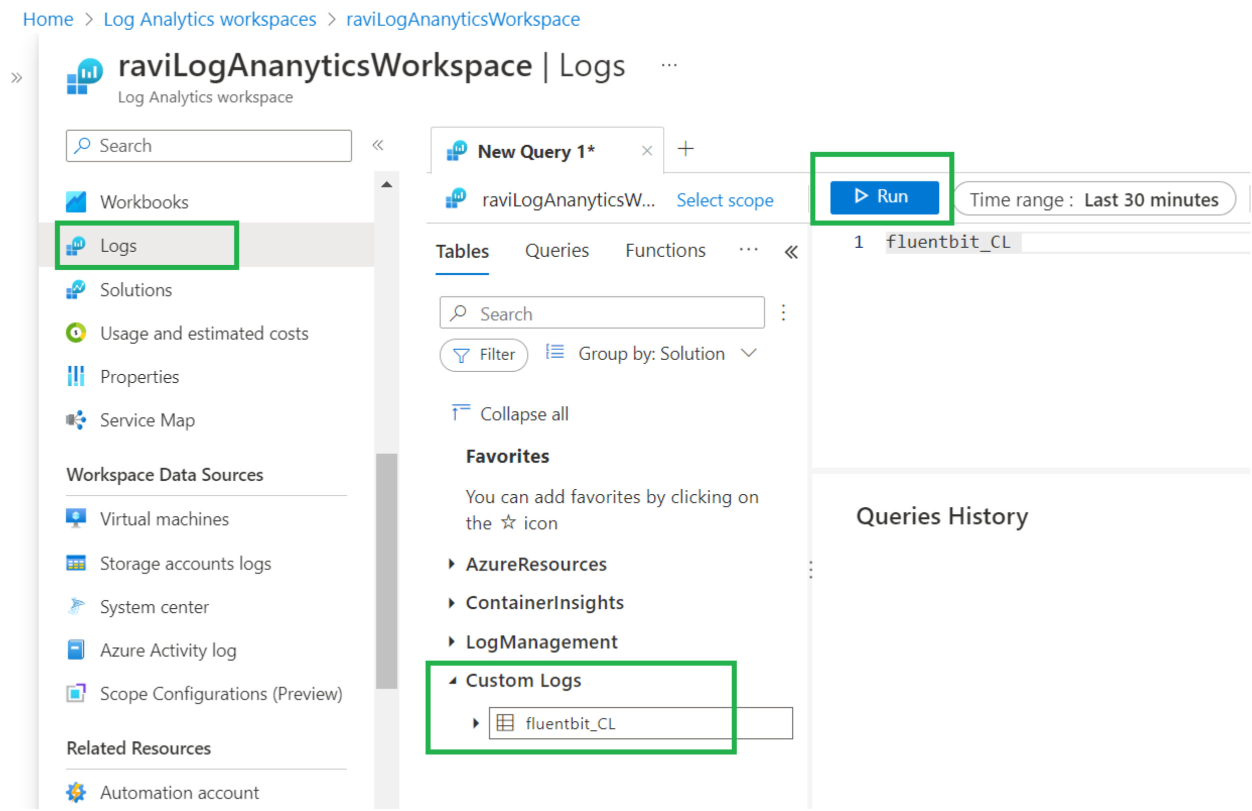
The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|       `--  ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
```

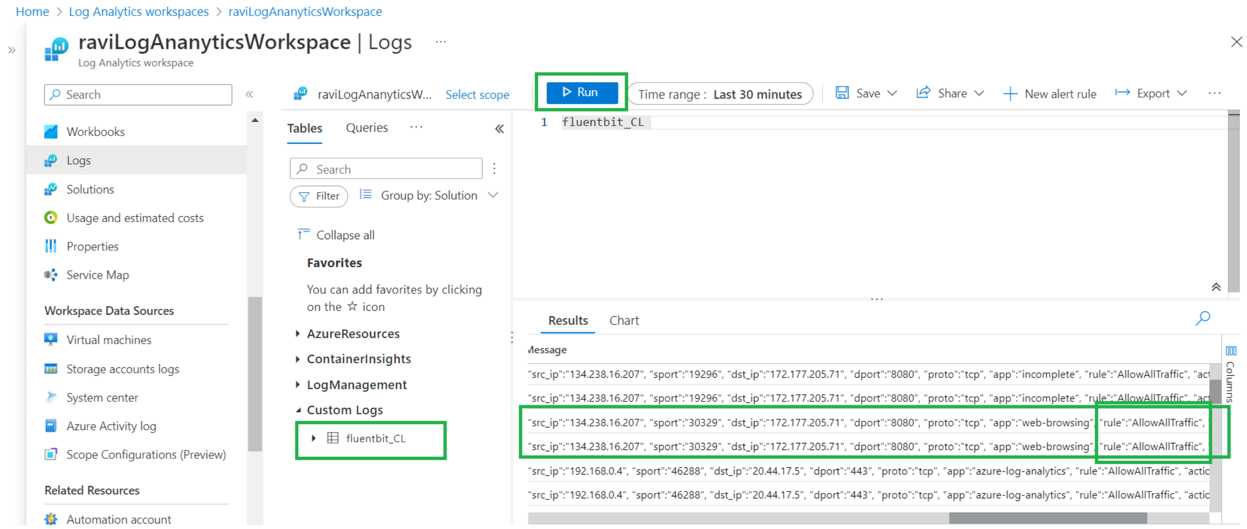
Accessing logs

To verify that this particular inbound traffic was processed correctly by Cloud NGFW, go to the **Log Analytics workspace** and verify the logs as shown below.

Within Log analytics workspace **raviCngfwLogWorkspace**, navigate to the **Logs** section, select **Custom Logs** and select **fluentbit_CL** and **run** the query to get the latest logs:

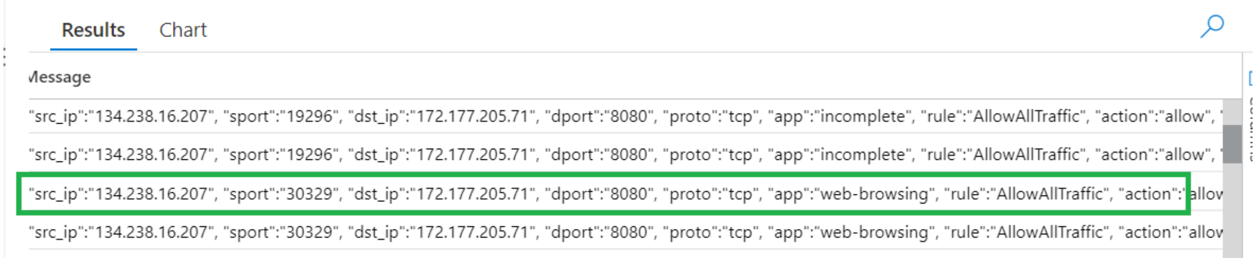


```
1  fluentbit_CL
```



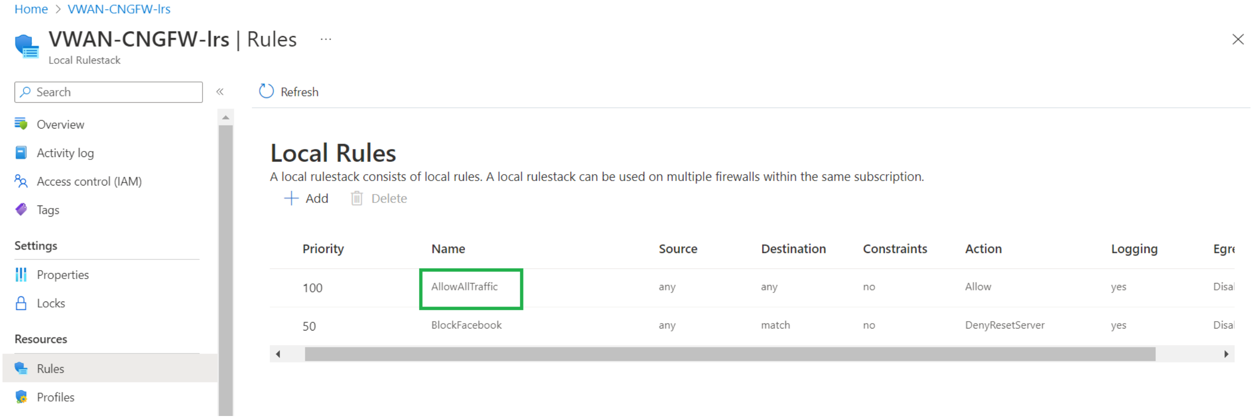
```
1  fluentbit_CL
```

```
Message
"src_ip":"134.238.16.207", "sport":"19296", "dst_ip":"172.177.205.71", "dport":"8080", "proto":"tcp", "app":"incomplete", "rule":"AllowAllTraffic", "act
"src_ip":"134.238.16.207", "sport":"19296", "dst_ip":"172.177.205.71", "dport":"8080", "proto":"tcp", "app":"incomplete", "rule":"AllowAllTraffic", "act
"src_ip":"134.238.16.207", "sport":"30329", "dst_ip":"172.177.205.71", "dport":"8080", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic",
"src_ip":"134.238.16.207", "sport":"30329", "dst_ip":"172.177.205.71", "dport":"8080", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic",
"src_ip":"192.168.0.4", "sport":"46288", "dst_ip":"20.44.17.5", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "actic
"src_ip":"192.168.0.4", "sport":"46288", "dst_ip":"20.44.17.5", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "actic
```



```
Message
"src_ip":"134.238.16.207", "sport":"19296", "dst_ip":"172.177.205.71", "dport":"8080", "proto":"tcp", "app":"incomplete", "rule":"AllowAllTraffic", "action":"allow",
"src_ip":"134.238.16.207", "sport":"19296", "dst_ip":"172.177.205.71", "dport":"8080", "proto":"tcp", "app":"incomplete", "rule":"AllowAllTraffic", "action":"allow",
"src_ip":"134.238.16.207", "sport":"30329", "dst_ip":"172.177.205.71", "dport":"8080", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic", "action":"allov
"src_ip":"134.238.16.207", "sport":"30329", "dst_ip":"172.177.205.71", "dport":"8080", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic", "action":"allov
```

From the log, it can be seen that the source IP address is the IP address of the machine from which the request originated, and the destination IP address is Cloud NGFW public IP address, and it's hitting the **AllowAllTraffic** rule that has been created in the rulestack.

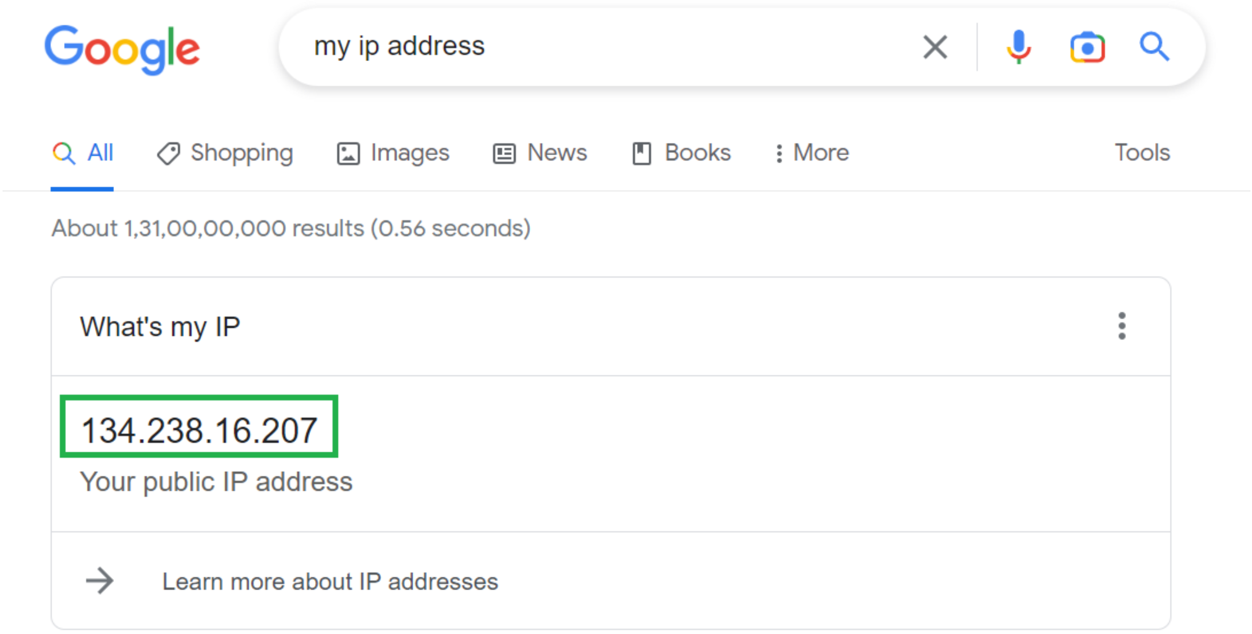


Local Rules

A local rulestack consists of local rules. A local rulestack can be used on multiple firewalls within the same subscription.

Priority	Name	Source	Destination	Constraints	Action	Logging	Egre
100	AllowAllTraffic	any	any	no	Allow	yes	Disa
50	BlockFacebook	any	match	no	DenyResetServer	yes	Disa

The screenshot below shows the IP address of the machine from which the request originated:



About 1,31,00,00,000 results (0.56 seconds)

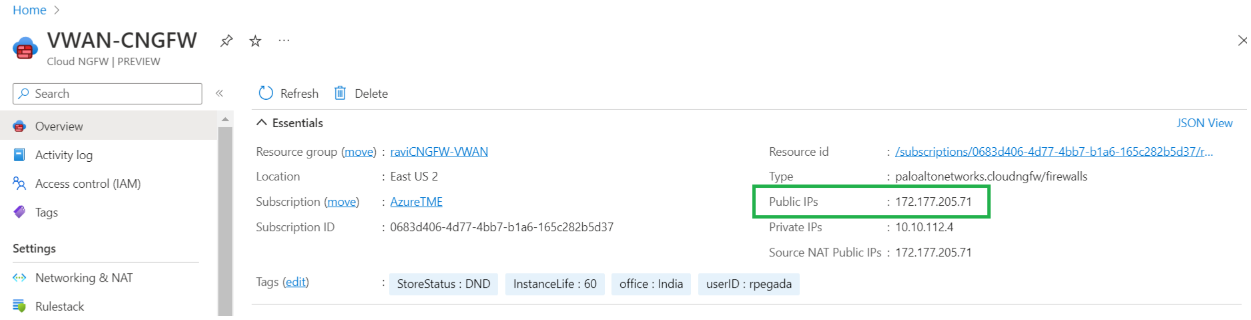
What's my IP

134.238.16.207

Your public IP address

Learn more about IP addresses

And the screenshot below shows the Public IP address of Cloud NGFW

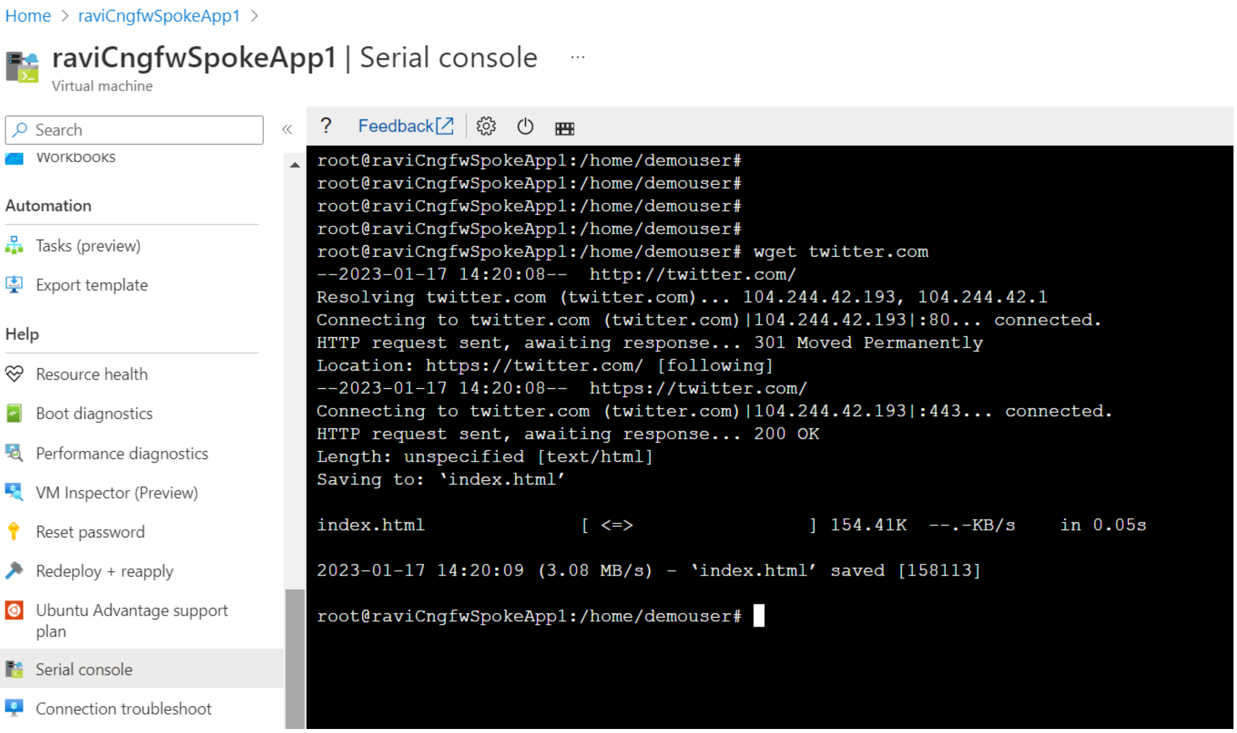


Resource group (move) : raviCNGFW-VWAN
Location : East US 2
Subscription (move) : AzureTME
Subscription ID : 0683d406-4d77-4bb7-b1a6-165c282b5d37
Tags (edit) : StoreStatus : DND InstanceLife : 60 office : India userID : rpegada

Resource id : /subscriptions/0683d406-4d77-4bb7-b1a6-165c282b5d37/r...
Type : paloaltonetworks.cloudngfw/firewalls
Public IPs : 172.177.205.71
Private IPs : 10.10.112.4
Source NAT Public IPs : 172.177.205.71

Test Outbound Traffic

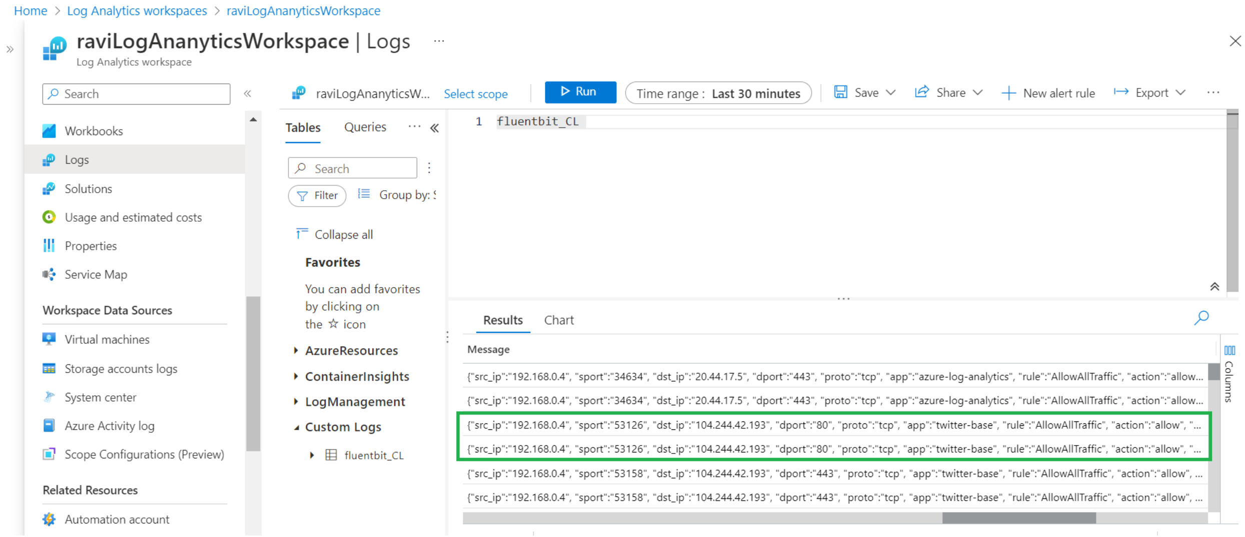
To validate the outbound connection, try to access twitter.com from App1 as shown below. Go to App1, select the **Serial console** section and type the following command:
Wget twitter.com



```
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser# wget twitter.com
--2023-01-17 14:20:08--  http://twitter.com/
Resolving twitter.com (twitter.com)... 104.244.42.193, 104.244.42.1
Connecting to twitter.com (twitter.com)|104.244.42.193|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://twitter.com/ [following]
--2023-01-17 14:20:08--  https://twitter.com/
Connecting to twitter.com (twitter.com)|104.244.42.193|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'index.html'

index.html              [ <=>                ] 154.41K  --.-KB/s    in 0.05s

2023-01-17 14:20:09 (3.08 MB/s) - 'index.html' saved [158113]

root@raviCngfwSpokeApp1:/home/demouser# █
```

The connection has been established. Verify that this traffic is being processed by Cloud NGFW by going to the **Log Analytics workspace**. Repeat the steps to access logs.

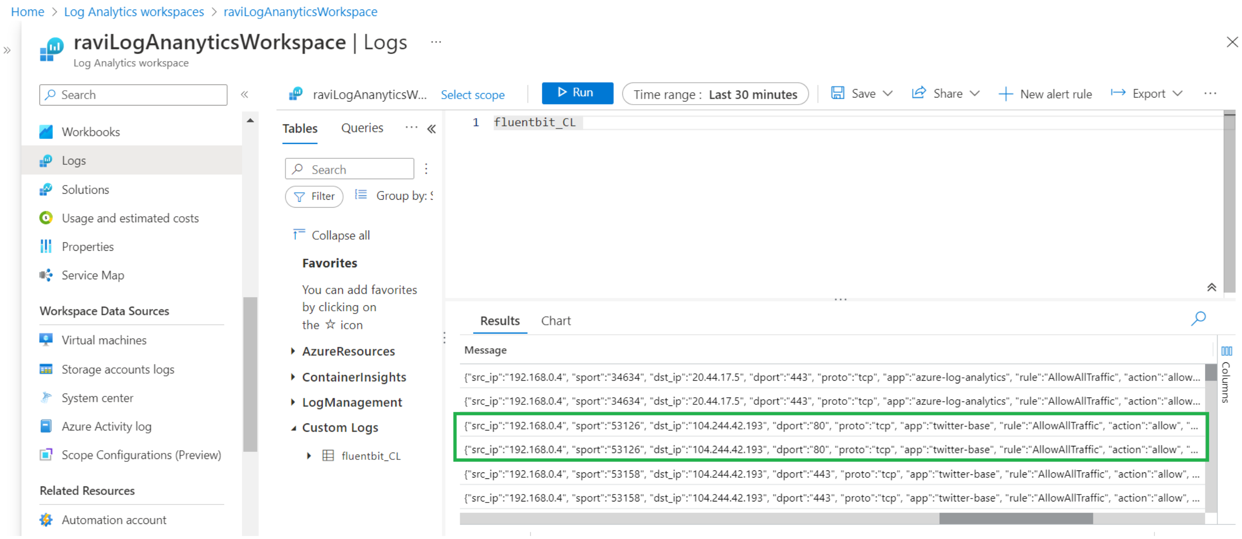
Run the query again to get the latest logs.



```
1  fluentbit_CL
```

Message

```
{"src_ip":"192.168.0.4", "sport":"34634", "dst_ip":"20.44.17.5", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow...
{"src_ip":"192.168.0.4", "sport":"34634", "dst_ip":"20.44.17.5", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow...
{"src_ip":"192.168.0.4", "sport":"53126", "dst_ip":"104.244.42.193", "dport":"80", "proto":"tcp", "app":"twitter-base", "rule":"AllowAllTraffic", "action":"allow", "...
{"src_ip":"192.168.0.4", "sport":"53126", "dst_ip":"104.244.42.193", "dport":"80", "proto":"tcp", "app":"twitter-base", "rule":"AllowAllTraffic", "action":"allow", "...
{"src_ip":"192.168.0.4", "sport":"53158", "dst_ip":"104.244.42.193", "dport":"443", "proto":"tcp", "app":"twitter-base", "rule":"AllowAllTraffic", "action":"allow", ...
{"src_ip":"192.168.0.4", "sport":"53158", "dst_ip":"104.244.42.193", "dport":"443", "proto":"tcp", "app":"twitter-base", "rule":"AllowAllTraffic", "action":"allow", ...
```

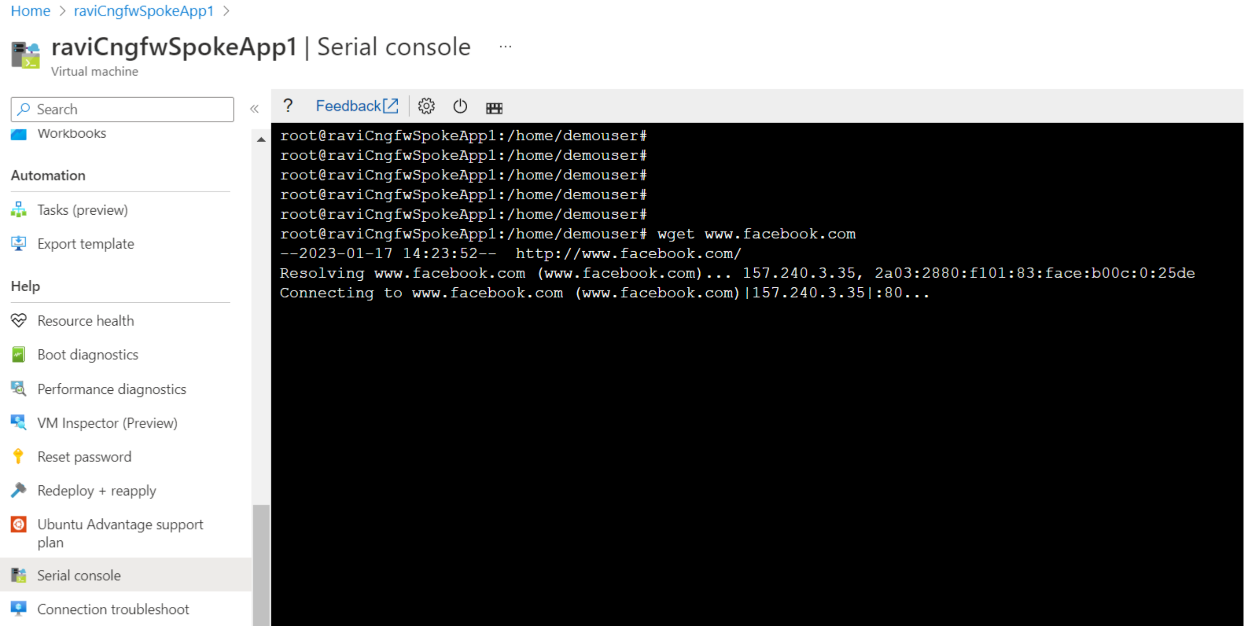


```
fluentbit_CL
```

```
Message
{"src_ip":"192.168.0.4", "sport":"34634", "dst_ip":"20.44.17.5", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow...
{"src_ip":"192.168.0.4", "sport":"34634", "dst_ip":"20.44.17.5", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"allow...
{"src_ip":"192.168.0.4", "sport":"53126", "dst_ip":"104.244.42.193", "dport":"80", "proto":"tcp", "app":"twitter-base", "rule":"AllowAllTraffic", "action":"allow", "...
{"src_ip":"192.168.0.4", "sport":"53126", "dst_ip":"104.244.42.193", "dport":"80", "proto":"tcp", "app":"twitter-base", "rule":"AllowAllTraffic", "action":"allow", "...
{"src_ip":"192.168.0.4", "sport":"53158", "dst_ip":"104.244.42.193", "dport":"443", "proto":"tcp", "app":"twitter-base", "rule":"AllowAllTraffic", "action":"allow", ...
{"src_ip":"192.168.0.4", "sport":"53158", "dst_ip":"104.244.42.193", "dport":"443", "proto":"tcp", "app":"twitter-base", "rule":"AllowAllTraffic", "action":"allow", ...
```

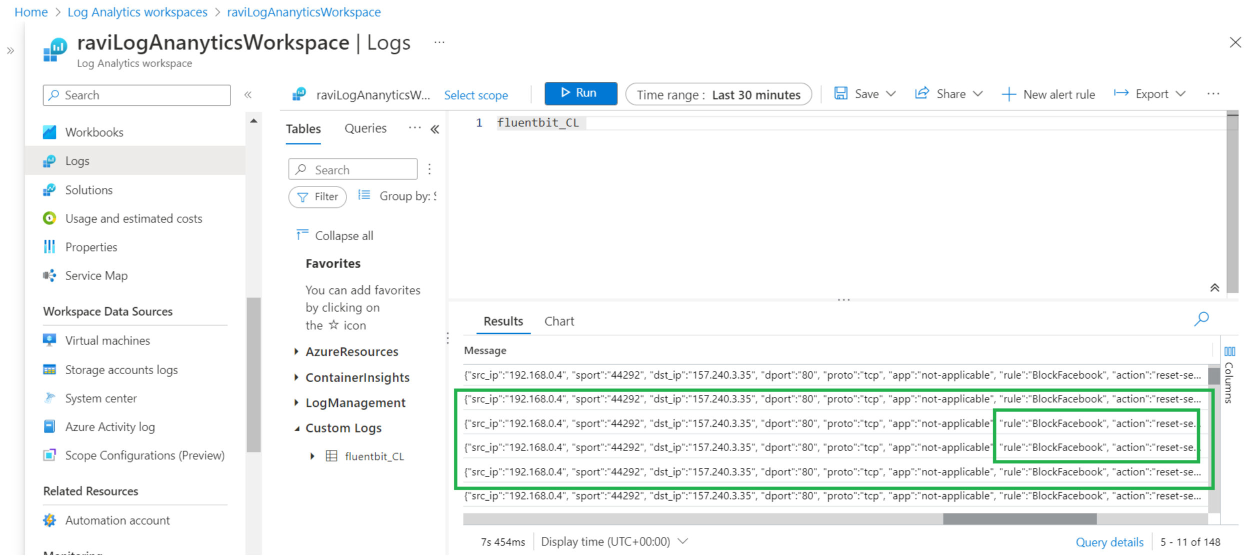
Test Outbound Block Rule

Now try to access Facebook. The traffic to Facebook should get blocked as per the rule configured. Go to App1, select**Serial console** and type the following command: *wget www.facebook.com*



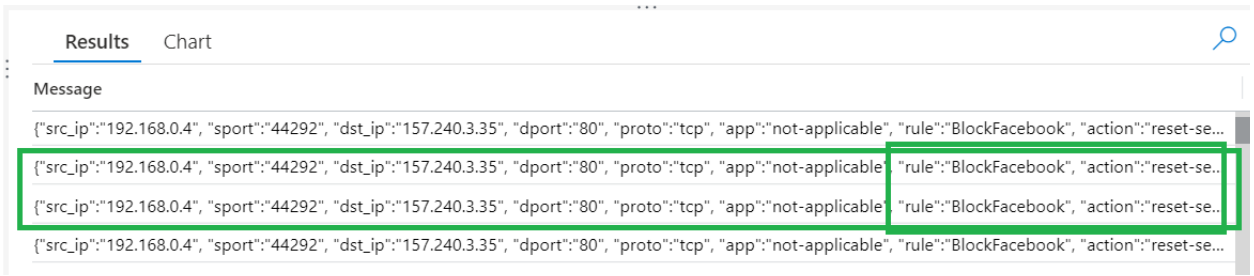
```
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser# wget www.facebook.com
--2023-01-17 14:23:52--  http://www.facebook.com/
Resolving www.facebook.com (www.facebook.com)... 157.240.3.35, 2a03:2880:f101:83:face:b00c:0:25de
Connecting to www.facebook.com (www.facebook.com)|157.240.3.35|:80...
```

Connection won’t be established. Go to **Azure Log Analytics** to validate that Cloud NGFW has blocked this connection as per the rRulestack configuration.



```
fluentbit_CL
```

```
Message
{"src_ip":"192.168.0.4", "sport":"44292", "dst_ip":"157.240.3.35", "dport":"80", "proto":"tcp", "app":"not-applicable", "rule":"BlockFacebook", "action":"reset-se...
{"src_ip":"192.168.0.4", "sport":"44292", "dst_ip":"157.240.3.35", "dport":"80", "proto":"tcp", "app":"not-applicable", "rule":"BlockFacebook", "action":"reset-se...
{"src_ip":"192.168.0.4", "sport":"44292", "dst_ip":"157.240.3.35", "dport":"80", "proto":"tcp", "app":"not-applicable", "rule":"BlockFacebook", "action":"reset-se.
{"src_ip":"192.168.0.4", "sport":"44292", "dst_ip":"157.240.3.35", "dport":"80", "proto":"tcp", "app":"not-applicable", "rule":"BlockFacebook", "action":"reset-se.
{"src_ip":"192.168.0.4", "sport":"44292", "dst_ip":"157.240.3.35", "dport":"80", "proto":"tcp", "app":"not-applicable", "rule":"BlockFacebook", "action":"reset-se...
{"src_ip":"192.168.0.4", "sport":"44292", "dst_ip":"157.240.3.35", "dport":"80", "proto":"tcp", "app":"not-applicable", "rule":"BlockFacebook", "action":"reset-se...
```



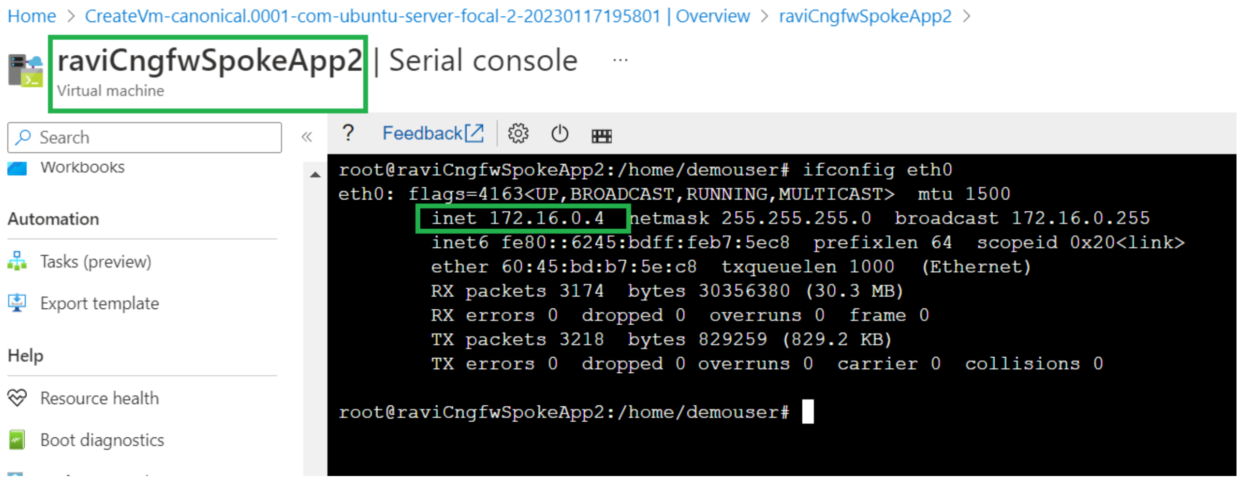
```
Message
{"src_ip":"192.168.0.4", "sport":"44292", "dst_ip":"157.240.3.35", "dport":"80", "proto":"tcp", "app":"not-applicable", "rule":"BlockFacebook", "action":"reset-se...
{"src_ip":"192.168.0.4", "sport":"44292", "dst_ip":"157.240.3.35", "dport":"80", "proto":"tcp", "app":"not-applicable", "rule":"BlockFacebook", "action":"reset-se...
{"src_ip":"192.168.0.4", "sport":"44292", "dst_ip":"157.240.3.35", "dport":"80", "proto":"tcp", "app":"not-applicable", "rule":"BlockFacebook", "action":"reset-se...
{"src_ip":"192.168.0.4", "sport":"44292", "dst_ip":"157.240.3.35", "dport":"80", "proto":"tcp", "app":"not-applicable", "rule":"BlockFacebook", "action":"reset-se...
```

From these logs, it is evident that the traffic to Facebook was blocked after hitting the **BlockFacebook** rule. This confirms that Cloud NGFW is able to block traffic as per configured rulestack.

Test East-West Traffic flow

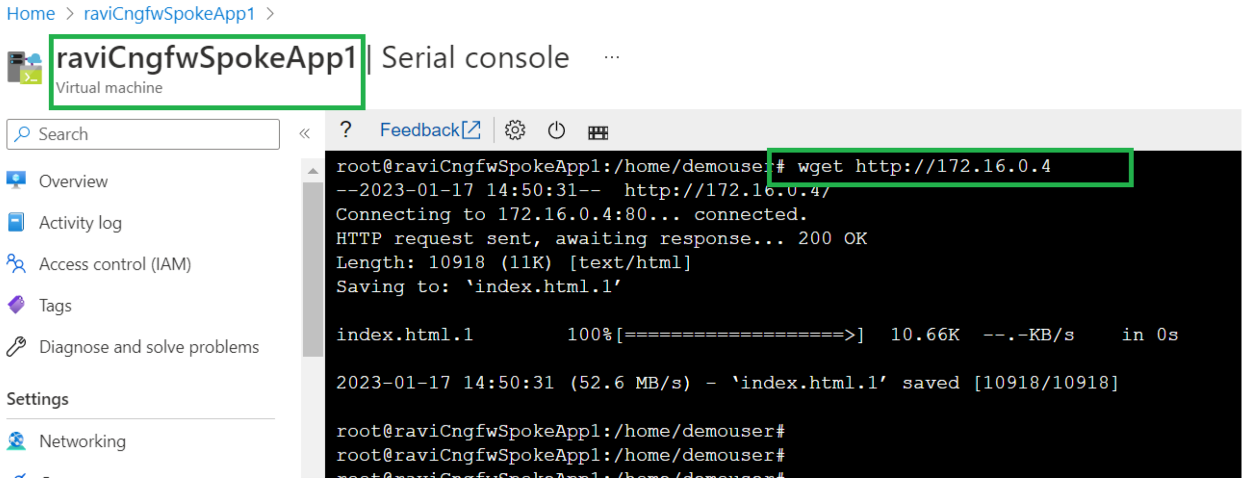
Validate east-west traffic flow by trying to send traffic from App1 to App2.

App2 IP address can be checked as shown below


Home > CreateVm-canonical.0001-com-ubuntu-server-focal-2-20230117195801 | Overview > raviCngfwSpokeApp2 >
raviCngfwSpokeApp2 | Serial console

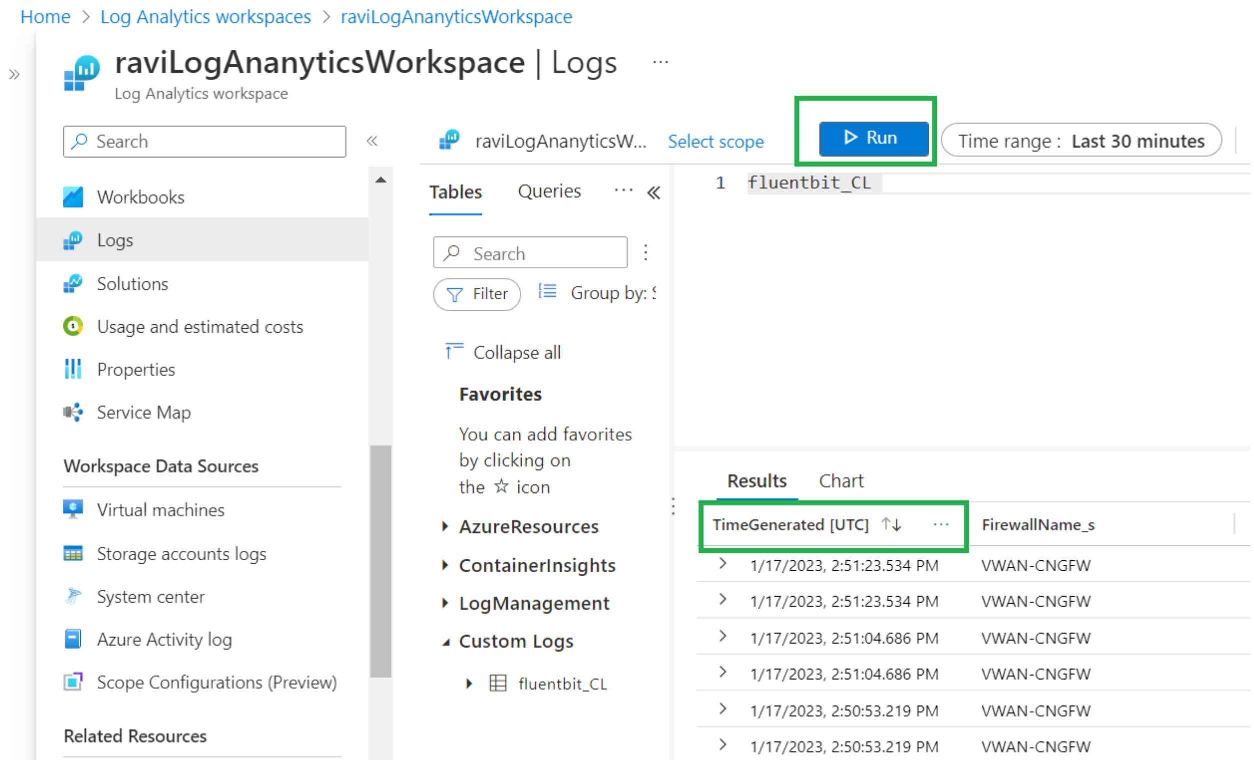
```
root@raviCngfwSpokeApp2:/home/demouser# ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.16.0.4  netmask 255.255.255.0  broadcast 172.16.0.255
        inet6 fe80::6245:bdff:feb7:5ec8  prefixlen 64  scopeid 0x20<link>
        ether 60:45:bd:b7:5e:c8  txqueuelen 1000  (Ethernet)
        RX packets 3174  bytes 30356380 (30.3 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 3218  bytes 829259 (829.2 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@raviCngfwSpokeApp2:/home/demouser#
```

On App1, execute the following command:
wget http://<App2 IP address>



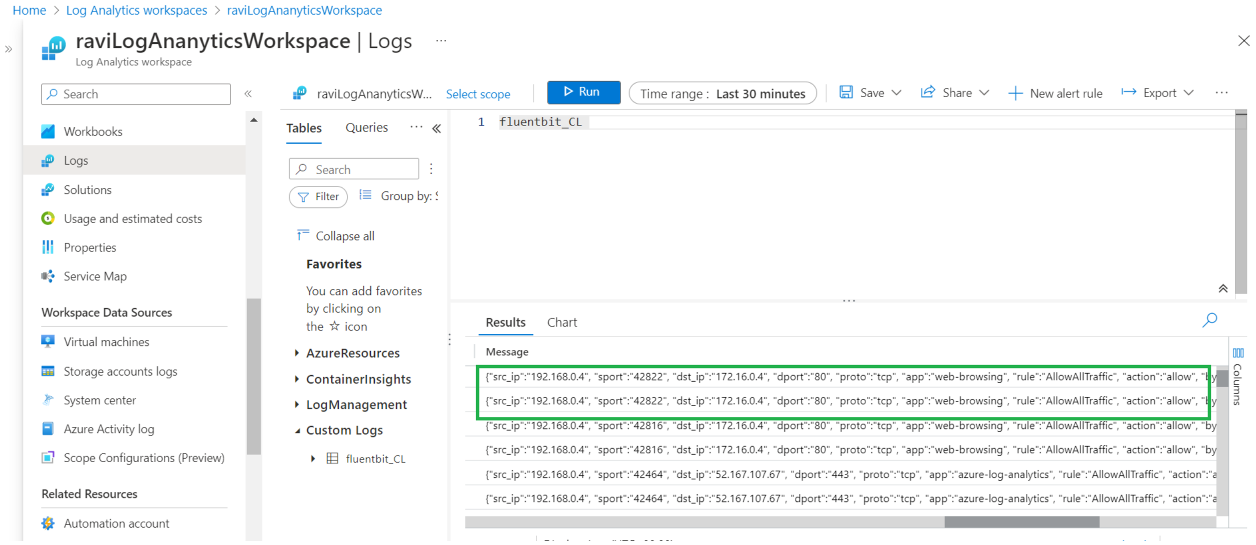
```
root@raviCngfwSpokeApp1:/home/demouser# wget http://172.16.0.4
--2023-01-17 14:50:31--  http://172.16.0.4/
Connecting to 172.16.0.4:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10918 (11K) [text/html]
Saving to: 'index.html.1'

index.html.1        100%[===================>]  10.66K  --.-KB/s    in 0s

2023-01-17 14:50:31 (52.6 MB/s) - 'index.html.1' saved [10918/10918]

root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser#
root@raviCngfwSpokeApp1:/home/demouser#
```

The connection has been established. Validate by going to the to **Azure Log Analytics** workspace:
After running the query, make sure you have sorted the logs based on TimeGenerated to see the latest logs on the top of the list.



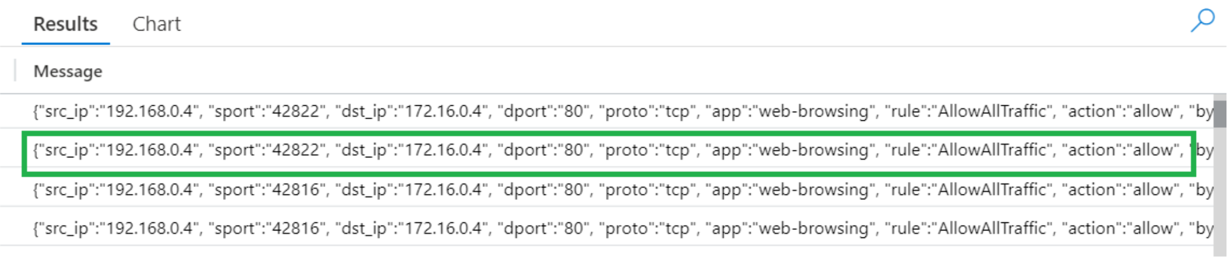
```
fluentbit_CL
```

TimeGenerated [UTC]	FirewallName_s
1/17/2023, 2:51:23.534 PM	VWAN-CNGFW
1/17/2023, 2:51:23.534 PM	VWAN-CNGFW
1/17/2023, 2:51:04.686 PM	VWAN-CNGFW
1/17/2023, 2:51:04.686 PM	VWAN-CNGFW
1/17/2023, 2:50:53.219 PM	VWAN-CNGFW
1/17/2023, 2:50:53.219 PM	VWAN-CNGFW



```
fluentbit_CL
```

```
Message
{"src_ip":"192.168.0.4", "sport":"42822", "dst_ip":"172.16.0.4", "dport":"80", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic", "action":"allow", "by
{"src_ip":"192.168.0.4", "sport":"42822", "dst_ip":"172.16.0.4", "dport":"80", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic", "action":"allow", "by
{"src_ip":"192.168.0.4", "sport":"42816", "dst_ip":"172.16.0.4", "dport":"80", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic", "action":"allow", "by
{"src_ip":"192.168.0.4", "sport":"42816", "dst_ip":"172.16.0.4", "dport":"80", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic", "action":"allow", "by
{"src_ip":"192.168.0.4", "sport":"42464", "dst_ip":"52.167.107.67", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"a
{"src_ip":"192.168.0.4", "sport":"42464", "dst_ip":"52.167.107.67", "dport":"443", "proto":"tcp", "app":"azure-log-analytics", "rule":"AllowAllTraffic", "action":"a
```



```
Message
{"src_ip":"192.168.0.4", "sport":"42822", "dst_ip":"172.16.0.4", "dport":"80", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic", "action":"allow", "by
{"src_ip":"192.168.0.4", "sport":"42822", "dst_ip":"172.16.0.4", "dport":"80", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic", "action":"allow", "by
{"src_ip":"192.168.0.4", "sport":"42816", "dst_ip":"172.16.0.4", "dport":"80", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic", "action":"allow", "by
{"src_ip":"192.168.0.4", "sport":"42816", "dst_ip":"172.16.0.4", "dport":"80", "proto":"tcp", "app":"web-browsing", "rule":"AllowAllTraffic", "action":"allow", "by
```

From these logs, it is visible that the traffic sent between App1 (192.168.0.4) and App2 (172.16.0.4) is going through the Cloud NGFW service and hitting the **AllowAllTraffic** rule which is part of the local rulestack.

Thus the inbound, outbound, and east-west traffic has been tested and is flowing through the Cloud NGFW service.

Resources

Contact

For any support, please email **cloud-ngfw-azure@paloaltonetworks.com** or reach out to your SE/CE.