

Branch Application To Cloud Application Connectivity

Overview

This quickstart guide will provide all the steps to create a secure service between a branch application and/or user and an application hosted in Azure Cloud using NetFoundry Overlay Fabric (NFOF).



Important

Assumption is that the NF Fabric is already up.

Through NF Web Console UI

Create and Deploy NF Gateway in Branch Datacenter

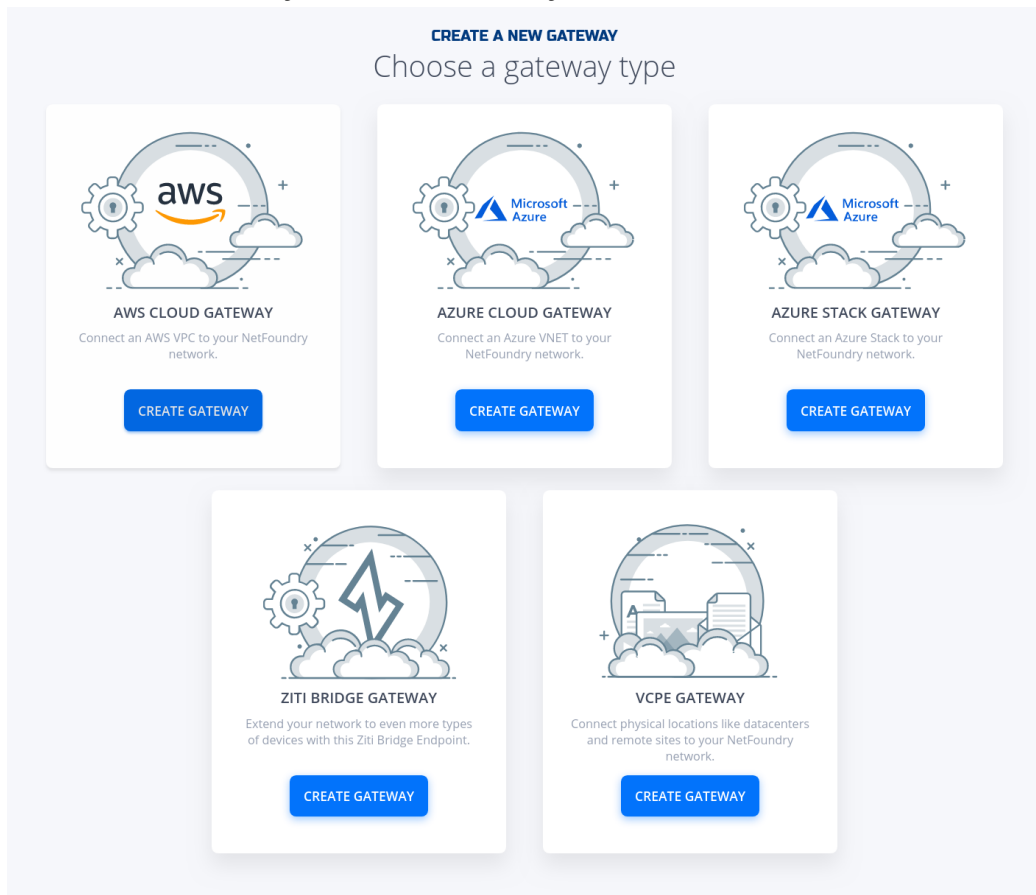
This section will guide a user through the steps on how to create a NF Manage Gateway in the NF Console UI and install it in the Branch Datacenter.



1. Navigate to Manage Gateways Page
2. Click on + sign in the top right corner.



3. Click on "Create Gateway" on the VCPE Gateway Card



4. Fill in the required information and click on "Create"


CREATE A NEW GATEWAY

Enter your Gateway attributes

GATEWAY NAME REQUIRED

CLOUD REGION REQUIRED

ENABLE HIGH AVAILABILITY ENABLE TO EDIT DETAILS


 **ENABLE HIGH AVAILABILITY FOR ULTIMATE REDUNDANCY** OFF
DISABLED

CREATE

5. Copy the Client Registration Key

CONGRATULATIONS

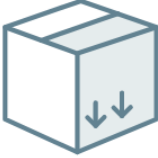
Your Gateway has been created



YOURBRANCHGATEWAYNAME REGISTRATION KEY CLICK TO COPY

`FB18C6EBEAD65B723E9C86B6882A3298ED77DBBF`


Download and install instructions



DOWNLOAD INSTALL PACKAGE

vCPE Gateway to connect to private networks, branches, & data centers using this customer-managed virtual gateway.

DOWNLOAD



VIEW INSTALL INSTRUCTIONS

Connect public or private IP space inside your vCPE using a customer-managed cloud gateway.

VIEW

6. Click on "Download" button on the Installation Package Card

The screenshot shows the NetFoundry website with a navigation bar at the top containing 'SOLUTIONS', 'PARTNERS', 'RESOURCES', 'ABOUT', and a 'START NOW' button. Below the navigation bar are six installation package cards arranged in a 2x3 grid. Each card features a circular icon with a gear and a plus sign, representing a different hypervisor or cloud provider. The cards are:

- AWS Cloud Gateway:** Deploy secure, high performance, global networks as easily and effectively as you spin up your AWS instances. Download it now from the AWS Marketplace. Not sure how to install it? [Click Here](#) to read our simple install instructions. Button: VISIT MARKETPLACE
- Azure Cloud Gateway:** Improve the agility, security and performance of your cloud applications in Azure. Download it now from the Azure Marketplace. Not sure how to install it? [Click Here](#) to read our simple install instructions. Button: VISIT MARKETPLACE
- OVA for VMWare ESXi 5.0 or greater:** This is a V-CPE Gateway to connect to private networks, branches, & datacenters using this customer-managed virtual gateway. Not sure how to install it? [Click Here](#) to read our simple install instructions. Button: DOWNLOAD
- Microsoft Hyper-V VHD Image:** This is a V-CPE Gateway to connect to private networks, branches, & datacenters using this customer-managed virtual gateway. Not sure how to install it? [Click Here](#) to read our simple install instructions.
- KVM QCOW2 Image:** This is a V-CPE Gateway to connect to private networks, branches, & datacenters using this customer-managed virtual gateway. Not sure how to install it? [Click Here](#) to read our simple install instructions.
- OVA for Virtual Box:** This is a V-CPE Gateway to connect to private networks, branches, & datacenters using this customer-managed virtual gateway. Not sure how to install it? [Click Here](#) to read our simple install instructions.

7. Download the correct image for the desired Hypervisor.

8. Follow the installation procedure linked in the description of each image type (i.e. "Click Here").

9. Once installed, login into it locally with ssh and register it using the key copied in the previous step. Run the following command `sudo nfnreg "reg key"`

```
Infadmin@yourBranchGatewayName ~]$ sudo nfnreg 368FD32C7F098838A4C0C3D991D38BD13D89EA87
Created symlink from /etc/systemd/system/multi-user.target.wants/dvn-driver.service to /usr/lib/systemd/system/dvn-driver.service.
Created symlink from /etc/systemd/system/multi-user.target.wants/dvn.service to /usr/lib/systemd/system/dvn.service.
Registration complete
Infadmin@yourBranchGatewayName ~]$
```

10. Once registered, one should see the gateway status turn to green in NF Console UI

MANAGE GATEWAYS

Manage Gateways | Manage Clients | Manage Azure Virtual WAN Sites | Manage Endpoint Groups

Type to Filter

<input type="radio"/>	Gateway Label	Status	Type	Location
<input type="radio"/>	YourBranchGatewayName	Online	V-CPE Gateway	US East

11. Done

Create and Deploy NF Azure Gateway

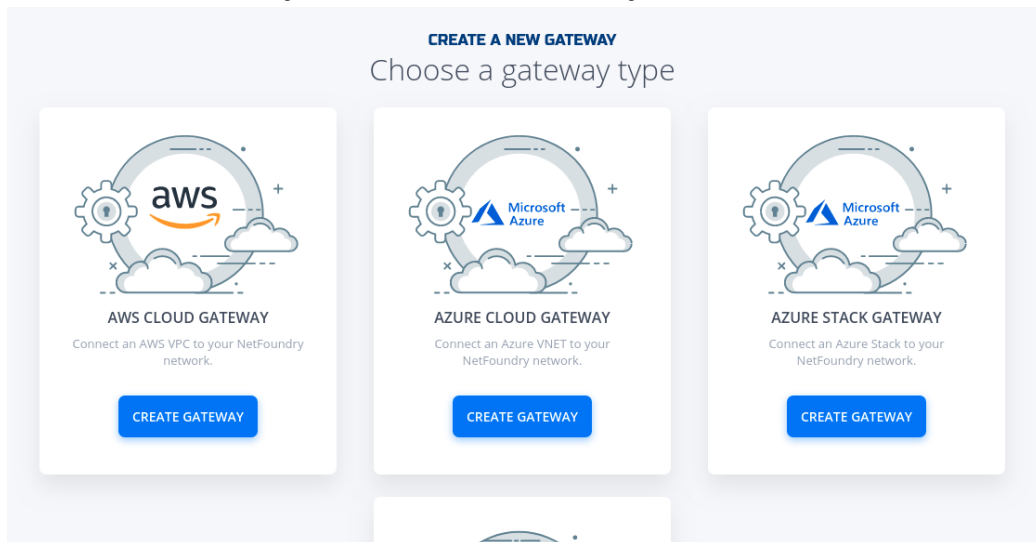
This section will guide a user through the steps on how to create a NF Manage Gateway in the NF Console UI and install it in the Azure vNet.



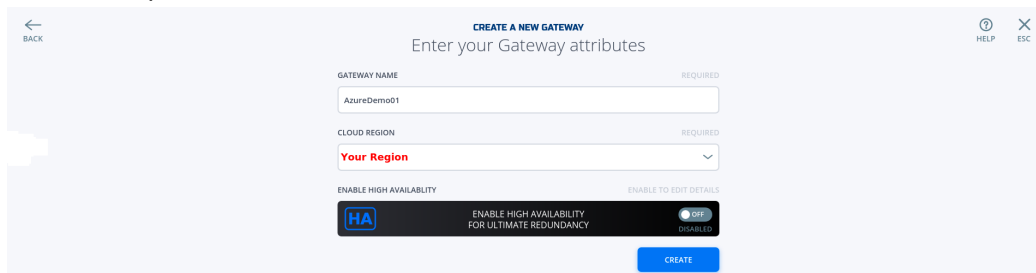
1. Navigate to Manage Gateways Page
2. Click on + sign in the top right corner.



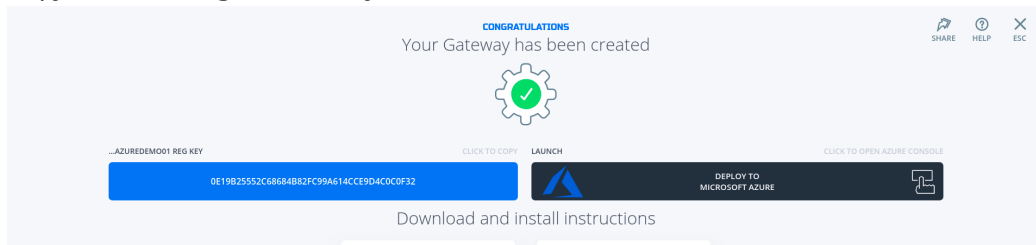
3. Click on "Create Gateway" on the Azure Cloud Gateway Card



4. Fill in the required information and click on "Create"



5. Copy the Client Registration Key



6. Click on "Deploy to Microsoft Azure". It will take you to the Azure Portal and ask you for your login credentials.

7. You will be presented with the template that needs to be filled. The first section is the Basics regarding your Subscription and Resource Group this gateway will be deployed in.

BASICS

Subscription *	<input type="text" value="Your Subscription Name"/>
Resource group *	<input type="text" value="Your Resource Group Name"/> Create new
Location *	<input type="text" value="(US) East US"/>

8. The second section related to resources associated with this gateway. e.g. vm name, ip address space, security groups, etc. you will paste the registration key copied in step 5. You will also need the public ssh key to use for access to this gateway remotely.

SETTINGS

Location	<input type="text" value="Your Region"/>
Network Interface Name	<input type="text" value="azuredemo01-if"/>
Security Group Name	<input type="text" value="azuredemo01-sg"/>
Virtual Network Name	<input type="text" value="azuredemo01-vnet"/>
Address Prefix	<input type="text" value="10.0.8.0/24"/>
Subnet Name	<input type="text" value="default"/>
Subnet Prefix	<input type="text" value="10.0.8.0/24"/>
Public Ip Address Name	<input type="text" value="azuredemo01-ip"/>
Public Ip Address Type	<input type="text" value="Dynamic"/>
Public Ip Address Sku	<input type="text" value="Basic"/>
Virtual Machine Name	<input type="text" value="azuredemo01"/>
Virtual Machine RG	<input type="text" value="nf-sandbox"/>
Os Disk Type	<input type="text" value="Premium_LRS"/>
Virtual Machine Size	<input type="text" value="Standard_B1ms"/>
Nfreg Key * ⓘ	<input type="text" value="....."/>
Admin Username ⓘ	<input type="text" value="nfadmin"/>
Ssh Key Data * ⓘ	<input type="text" value="ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQACjga67wcolSXaD1bswknLrejRYtZ..."/>

9. You will need to agree to Azure Marketplace Terms and Conditions and click to "Purchase" to continue.

TERMS AND CONDITIONS

[Azure Marketplace Terms](#) | [Azure Marketplace](#)

By clicking "Purchase," I (a) agree to the applicable legal terms associated with the offering; (b) authorize Microsoft to charge or bill my current payment method for the fees associated the offering(s), including applicable taxes, with the same billing frequency as my Azure subscription, until I discontinue use of the offering(s); and (c) agree that, if the deployment involves 3rd party offerings, Microsoft may share my contact information and other details of such deployment with the publisher of that offering.

I agree to the terms and conditions stated above

Purchase

10. If the NF Gateway was deployed successfully. Here is the view of the Resource Group and NF Conole UI.

The screenshot displays the Azure portal interface. On the left, the navigation pane shows the 'nf-sandbox' resource group. The main area shows the 'Overview' tab for the resource group, displaying subscription information and a table of resources. A notification banner at the top right indicates a successful deployment of the 'Microsoft.Template' to the resource group. Below the resource group overview, the 'MANAGE GATEWAYS' section is visible, showing a table with one gateway entry: 'AzureDemo01', which is 'Online' and of type 'Azure Private Gateway'.

Gateway Label	Status	Type	Location	Cloud Provider
AzureDemo01	Online	Azure Private Gateway	Your Region	

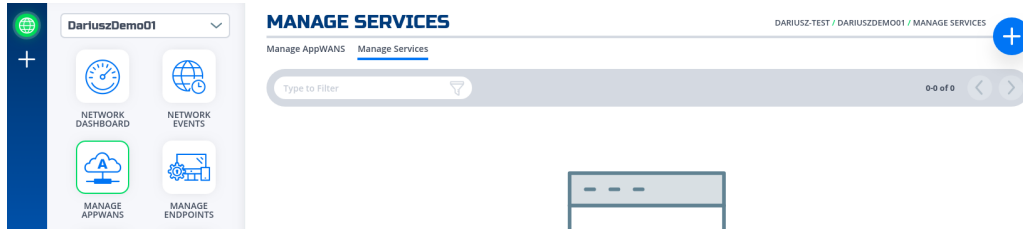
11. Done

Create IP Host Service

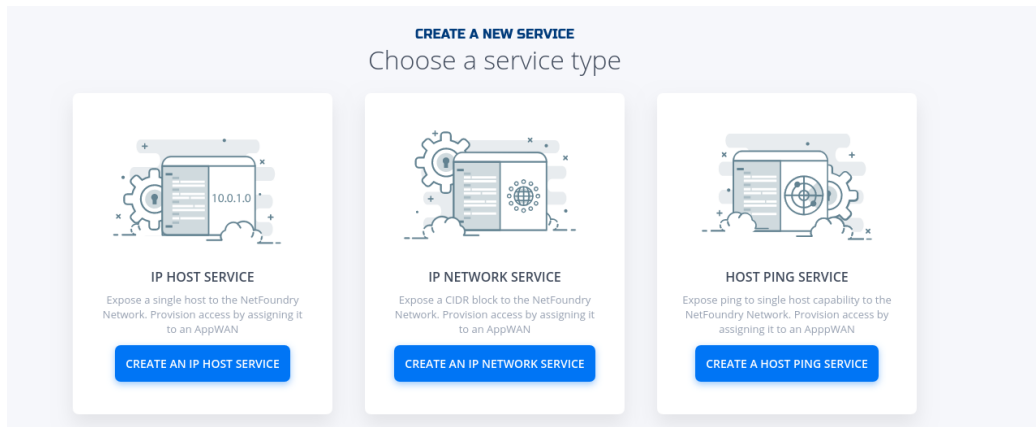
This section will guide a user through the steps on how to create a NF Service.



1. Navigate to Manage Services Page under Manage Appwans
2. Click on + sign in the top right corner.



3. Click on "Create an IP Host Service"



4. Fill in the required information for SSH and click on "Create"

5. If successfully, the service is green.

The screenshot shows a web interface for managing services. On the left is a navigation menu with icons for Network Dashboard, Network Events, Manage AppWans, and Manage Endpoints. The main area is titled 'MANAGE SERVICES' and includes a search bar and a table of services. The table has columns for Service Name, Type, Protocol, IP Address, Intercept IP, and Port Range. One service, 'DemoServiceSsh', is listed with a green dot next to its name, indicating it is active.

Service Name	Type	Protocol	IP Address	Intercept IP	Port Range
<input checked="" type="radio"/> DemoServiceSsh	IP Host	TCP	10.0.8.5	10.0.8.5	22 - 22

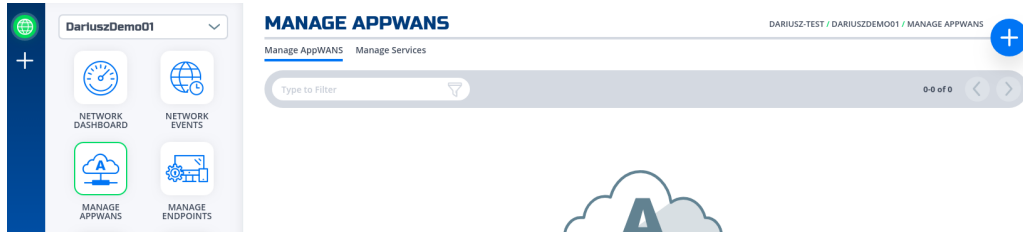
6. Done

Create AppWan

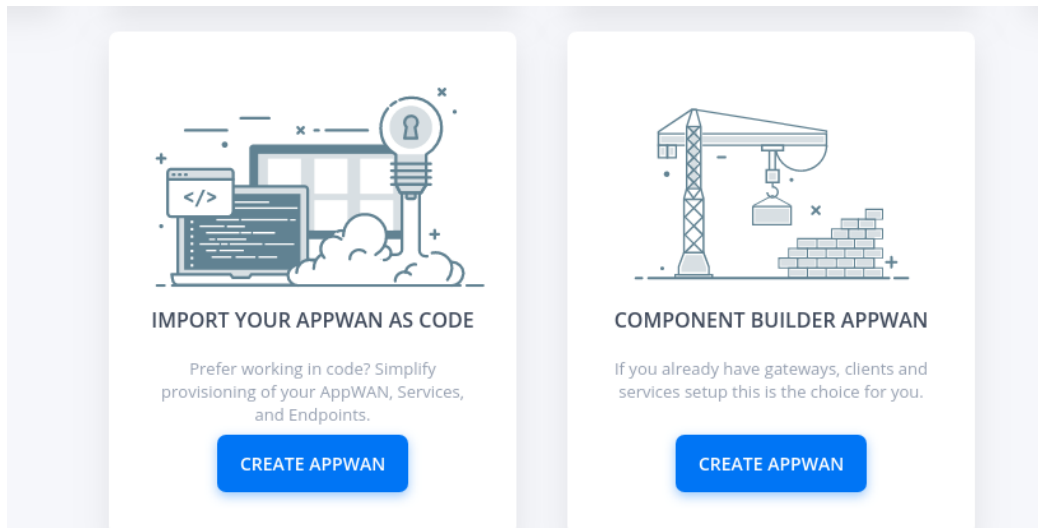
This section will guide a user through the steps on how to enable service connectivity to users by creating an appwan.



1. Navigate to Manage AppWANS Page under Manage Appwans
2. Click on + sign in the top right corner.



3. Click on "Component Builder Appwan"



4. Move the desired gateway (e.g. DemoGateway01) from "Available" Gateways to "Selected" Endpoints. Move the desired service (e.g. DemoServiceSsh) from "Available" to "Selected"

Services.

CREATE A NEW APPWAN
Choose from existing components, or add new ones

1 APPWAN NAME REQUIRED

DemoAppWan

2 ADD CLIENTS, GATEWAYS, OR ENDPOINT GROUPS

Search for Endpoints

AVAILABLE GROUPS ADD NEW +

SELECTED ENDPOINTS

● YourBranchGatewayName⚙️

←→

AVAILABLE CLIENTS ADD NEW +

←→

AVAILABLE GATEWAYS ADD NEW +

● AzureDemo01

3 ADD SERVICES

Search for a Service

AVAILABLE SERVICES ADD NEW +

SELECTED SERVICES

DemoServiceSsh

←→


CREATE


5. Click on "Create".






YOUR APPWAN SUMMARY

Your AppWAN has been created! A network summary is below.

What's next? Finish connecting your network by registering new clients and gateways.

HINT **NEW CLIENTS**
Share Client Registration Info 

HINT **NEW GATEWAYS**
Tap to Launch and Register 

- 1 APPWAN NAME**
DemoAppWan 
- 2 ENDPOINTS**
 - CLIENTS** [SHARE NEW CLIENTS](#)
 - GATEWAYS** [REGISTER NEW GATEWAYS](#)
 -  [YOURBRANCHGATEWAYNAME](#) 
- 3 SERVICES**
 - SERVICE DEFINITIONS**
 -  [DemoServiceSsh](#) 
- 4 ENDPOINT GROUPS**
 - GROUPS**

6. Done

Test Connectivity to Application Server

Route to vNet

The private IP of NF Gateway (e.g. YourBranchGatewayName) needs to be the next hop to reach the vNet in Azure (e.g. 10.0.8.0/24). Thus, a static route will need to be configured in one of the branch routers. NF Gateway can support dynamic routing if needed (e.g.bgp, ospf)



Steps

1. Log in to a Client App Host in Branch DataCenter
2. Run `ssh username@privatelPofServerAppHostInAzure`

```
nfadmin@azuredemoapp:~  
PS C:\>  
PS C:\>  
PS C:\> ssh nfadmin@10.0.8.5  
[nfadmin@azuredemoapp ~]$  
[nfadmin@azuredemoapp ~]$ ls  
[nfadmin@azuredemoapp ~]$ pwd  
/home/nfadmin  
[nfadmin@azuredemoapp ~]$  
[nfadmin@azuredemoapp ~]$
```

Programmatically

via Python and Terraform



Python Modules

For the code clarity, we have broken down the code into multiple Python modules

1. NF REST CRUD (Create,Read, Update and Delete) operations
2. Get MOP Session Token
3. Create NF Network
4. Create NF Gateway(s)
5. Create NF Service(s)
6. Create NF AppWan(s)
7. Wrapper Script to Create NF Resources based on Resource yaml file



Environment Setup Requirements

1. `~/env` to store NF Credentials in (e.g. `clientId`, `clientSecret`) to obtain a session token for NF API
2. Export Azure Credentials (e.g, `export ARM_TENANT_ID, ARM_CLIENT_ID, ARM_CLIENT_SECRET, ARM_SUBSCRIPTION_ID`) to enable resource gateway creation in Azure Resource Group via Terraform.
3. Terraform and Python3 installed in path.

Additional Information:

1. The new Resource Group in Azure is created based on then name provided in `Resource.yml`, if one does not exist already in the same region (e.g. `centralus`). The action `delete gateway` will delete the RG as well even if it was an existing RG. If one does not want to delete the RG, the command `terraform state rm "{tf resource name for RG}"` needs to be run before running the gateway delete step. This will ensure that the RG is not deleted.
2. A new vNet will be created and NF Gateway will be placed in it.
3. Environment means the NF Console Environment used (e.g. `production`), not Azure.



Steps

1. Clone this repo (git clone <https://github.com/netfoundry/mop.git>)
2. Update Resource yaml file with the desired options to feed into the wrapper script as described in the following code snippet. All Resource.yml Options
3. Run this from the root folder to create GW in NF Console UI and Azure.

```
python3 quickstarts/docs/api/python/source/netfoundry/nf_resources.py --file quickstarts/docs/api/python/etc/nf_resources.yml
```

Required Configuration Parameters for Gateway Creation

```
environment: production
network_action: get
network_name: DemoNet01
gateway_list:
- action: create
  cloud: azure
  count: 1
  names: []
  region: westus
  regionalCidr: [10.20.10.0/24]
  regkeys: []
  resourceGroup:
    name: demoPythonTerraform01
    region: centralus
  tag: TerraformDemo
terraform:
  bin: terraform
  output: 'no'
  source: ./quickstarts/docs/terraform
  work_dir: .
```

4. After the script is run successfully, one can see that the gateway name and registration key were saved in Resource.yml file. The name is created automatically based on region and gateway type joined with x and gateway count (AZCPEGW means an azure type gateway in NF console). One can create more than one gateway in the same region by increasing the count to more than 1.

```
environment: production
gateway_list:
- action: create
  cloud: azure
  count: 1
  names:
  - AZCPEGWx0xWESTUS
  region: westus
  regionalCidr:
  - 10.20.10.0/24
  regkeys:
  - 21DB86724EC3F31C11C1C9D68CE5ECD6A06F057E
  resourceGroup:
    name: demoPythonTerraform01
```



```

region: centralus
tag: TerraformDemo
network_action: get
network_name: DemoNet01
terraform:
  bin: terraform
  output: 'no'
  source: ./quickstarts/docs/terraform
  work_dir: .

```

5. Create a test server vm on the same vNet if not already present.

6. Update the Resoure.yaml file to include the Service option to create the NF service on the gateway create in the previous step. Don't forget to change the action on the gateway to "get".

```

environment: production
gateway_list:
- action: get
  cloud: azure
  count: 1
  names:
  - AZCPEGWx0xWESTUS
  region: westus
  regionalCidr:
  - 10.20.10.0/24
  regkeys:
  - 21DB86724EC3F31C11C1C9D68CE5ECD6A06F057E
  resourceGroup:
  name: demoPythonTerraform01

```

```
region: centralus
tag: TerraformDemo
network_action: get
network_name: DemoNet01
terraform:
  bin: terraform
  output: 'no'
  source: ./quickstarts/docs/terraform
  work_dir: .
services:
- action: create
  gateway: AZCPEGWx0xWESTUS
  ip: 10.20.10.5
  port: 22
  name:
  type: host
```

7. After the script run again successfully, the service section should have been populated with the service name as so.

```
services:
- action: create
  gateway: AZCPEGWx0xWESTUS
  ip: 10.20.10.5
  name: AZCPEGWx0xWESTUS--10.20.10.5--22
  port: 22
  type: host
```

MANAGE SERVICES / MANAGE SERVICES +

Manage AppWANs Manage Services

Type to Filter 1-1 of 1 < >

Service Name	Type	Protocol	IP Address	Intercept IP	Port Range
<input checked="" type="radio"/> AZCPEGWx0xWESTUS--10.20.10.5-22	IP Host	TCP	10.20.10.5	10.20.10.5	22 - 22

8. Create a gateway in the branch as the steps in the UI section indicated. We will provide code snippets for private hypervisors deployment through python in later releases (e.g. vSphere)

MANAGE GATEWAYS

Manage Gateways Manage Clients Manage Azure Virtual WAN Sites Manage Endpoint Groups

Type to Filter

Gateway Label	Status	Type	Location
<input type="radio"/> YourBranchGatewayName	● Online	S V-CPE Gateway	US East

9. Update the Resoure.yaml file to include the AppWan option to create the NF AppWan tying the gateway, client and service created in the previous steps. Don't forget to change the action on the service option to "get".

```
environment: production
gateway_list:
- action: get
  cloud: azure
  count: 1
  names:
```

```
- AZCPEGWx0xWESTUS
region: westus
regionalCidr:
- 10.20.10.0/24
regkeys:
- 21DB86724EC3F31C11C1C9D68CE5ECD6A06F057E
resourceGroup:
  name: demoPythonTerraform01
  region: centralus
tag: TerraformDemo
network_action: get
network_name: DemoNet01
services:
- action: get
  gateway: AZCPEGWx0xWESTUS
  ip: 10.20.10.5
  name: AZCPEGWx0xWESTUS--10.20.10.5--22
  port: 22
  type: host
terraform:
  bin: terraform
  output: 'no'
  source: ./quickstarts/docs/terraform
  work_dir: .
appwans:
- action: create
  endpoints:
  - BranchGatewayName
  - ClientName
  name: appwan-ssh-22
  services:
  - AZCPEGWx0xWESTUS--10.20.10.5--22
```

10. After the script ran again successfully, the connectivity should have been up.



YOUR APPWAN SUMMARY

Your AppWAN's details are below. Looking to make some changes?
Follow the hints below to edit your AppWAN, Endpoints, and or Services.

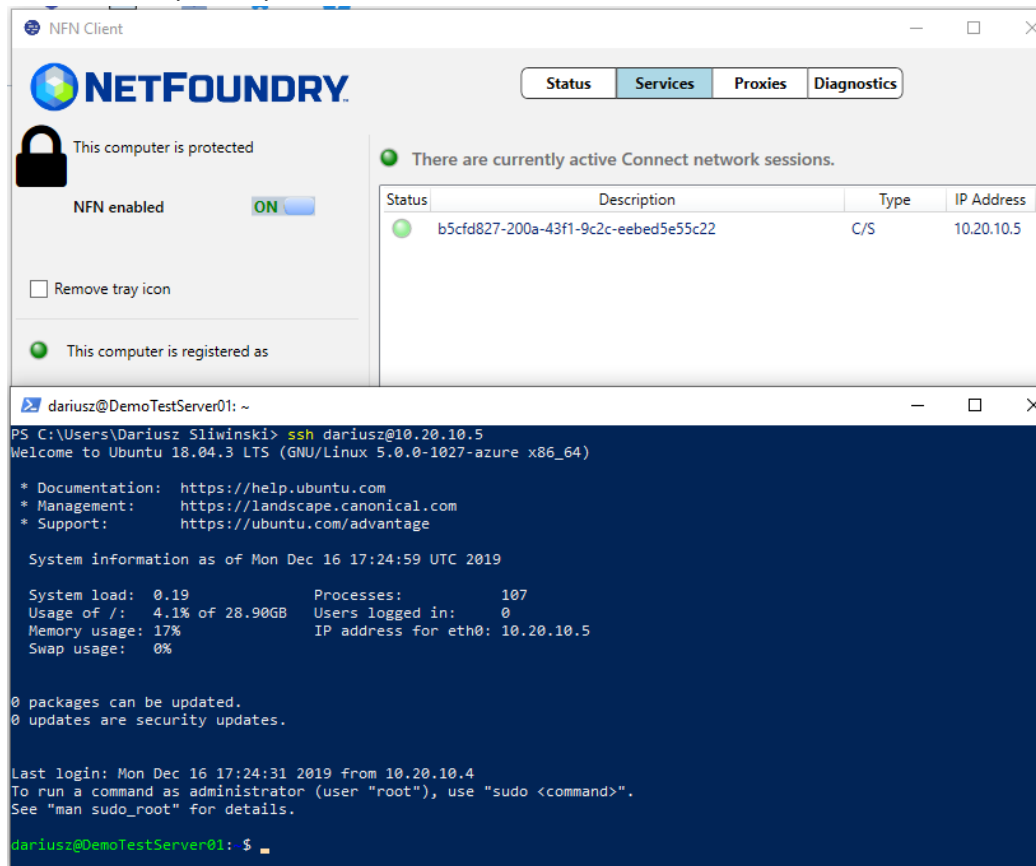
HINT EDIT YOUR APPWAN
Tap the Edit icon to make changes

HINT ENDPOINTS & SERVICES
Tap to Edit Advanced Options

- 1 APPWAN NAME
[appwan-ssh-22](#)
- 2 ENDPOINTS
CLIENTS
 - [ClientName](#)
GATEWAYS
 - [BRANCHGATEWAYNAME](#)
- 3 SERVICES
SERVICE DEFINITIONS
[AZCPEGWx0xWESTUS--10.20.10.5-22](#)
- 4 ENDPOINT GROUPS
GROUPS

 Want to add another environment with the same services or endpoints? **TAP TO CLONE** 

11. To test connectivity, log in to the Remote Client or Branch App Server and run ssh "username"@ "privatelp"



The screenshot shows the NFN Client interface with the NETFOUNDRY logo and tabs for Status, Services, Proxies, and Diagnostics. A notification states "This computer is protected" and "NFN enabled" is turned ON. Below, it says "This computer is registered as". To the right, a table lists active Connect network sessions:

Status	Description	Type	IP Address
●	b5cfd827-200a-43f1-9c2c-eebed5e55c22	C/S	10.20.10.5

Below the interface is a terminal window showing an SSH session from a Windows PowerShell prompt to a Linux server:

```
PS C:\Users\Dariusz Sliwinski> ssh dariusz@10.20.10.5
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 5.0.0-1027-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Mon Dec 16 17:24:59 UTC 2019

System load:  0.19          Processes:    107
Usage of /:   4.1% of 28.90GB Users logged in:  0
Memory usage: 17%         IP address for eth0: 10.20.10.5
Swap usage:   0%

0 packages can be updated.
0 updates are security updates.

Last login: Mon Dec 16 17:24:31 2019 from 10.20.10.4
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

dariusz@DemoTestServer01: $
```

12. To delete resources created, just follow the reverse order. Change the action to delete for AppWans first, then other resources as indicated in the code snippets.

```
appwans:
- action: delete
endpoints:
- BranchGatewayName
- ClientName
name: null
services:
- AZCPEGWx0xWESTUS--10.20.10.5--22
```

13. Services

```
services:
- action: delete
gateway: AZCPEGWx0xWESTUS
ip: 10.20.10.5
name: null
port: 22
type: host
```

14. Endpoints - will delete all resources in Azure as well.

- a. terraform state rm "{tf resource name for RG}" // run this before the python script if Resource Group needs to be preserved 1.

```
gateway_list:
- action: delete
  cloud: azure
  count: 1
  names: []
  region: westus
  regionalCidr:
  - 10.20.10.0/24
  regkeys: []
  resourceGroup:
    name: demoPythonTerraform01
    region: centralus
  tag: TerraformDemo
```

15. Network

```
environment: production
network_action: delete
network_name: DemoNet01
```

16. Done

via Jenkins

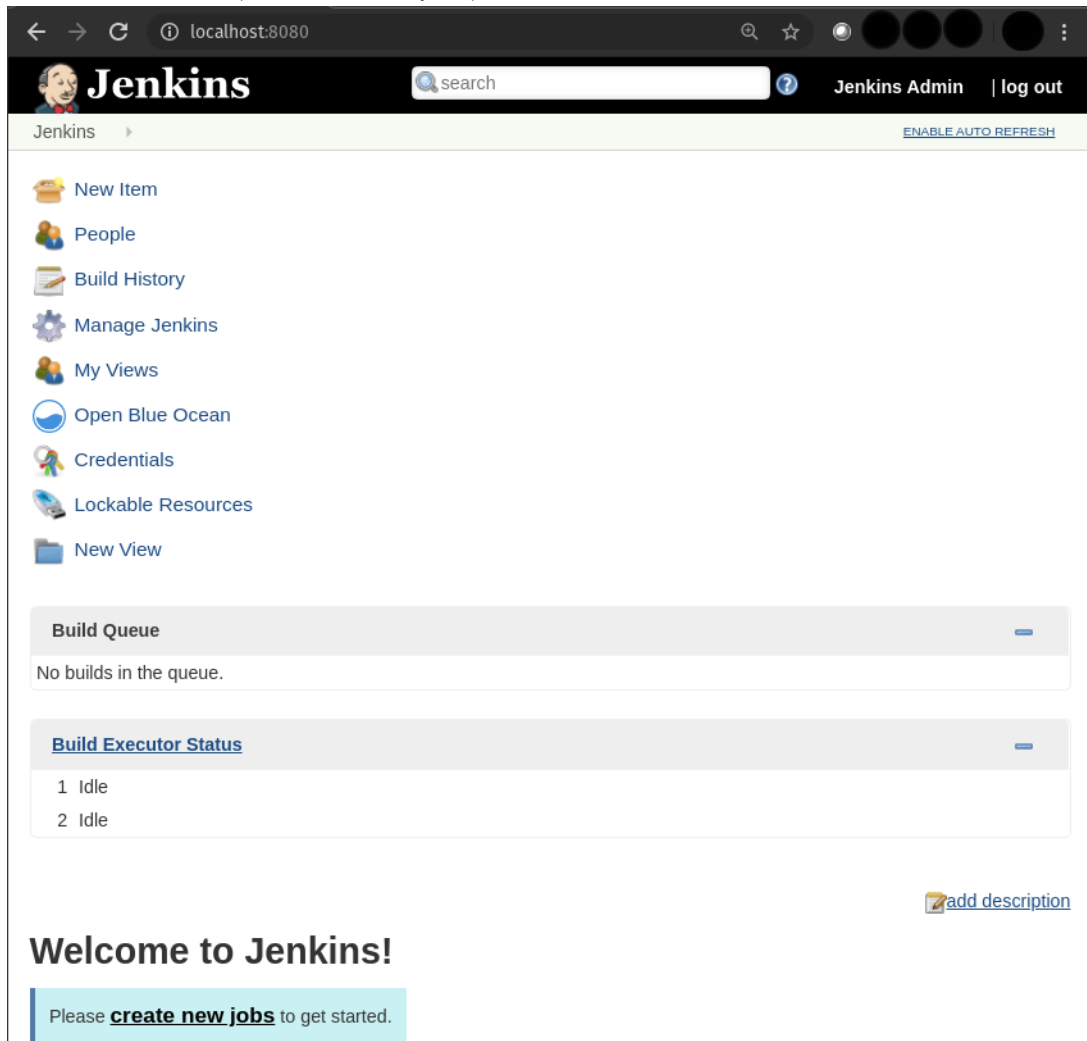
In this section, we will use Resource yaml along with Jenkinsfile to show how to automate the steps further by creating the Jenkins Job



Jenkins Requirements

1. java
2. docker

Then follow jenkins installation using docker to install Jenkins on the localhost and choose "Install suggested plugins". After successful installation, one should be able to reach the Jenkins Dashboard (8080 is default port).



The screenshot shows the Jenkins dashboard in a browser window. The address bar displays 'localhost:8080'. The page header includes the Jenkins logo, a search bar, and the user 'Jenkins Admin' with a 'log out' link. A navigation menu on the left lists various options: New Item, People, Build History, Manage Jenkins, My Views, Open Blue Ocean, Credentials, Lockable Resources, and New View. The main content area features two panels: 'Build Queue' (showing 'No builds in the queue.') and 'Build Executor Status' (showing two 'Idle' executors). At the bottom, there is a 'Welcome to Jenkins!' message and a button to 'add description'. A light blue box at the bottom contains the text: 'Please **create new jobs** to get started.'



Note

If one wants to add the gateway deployed in the Private DataCenter and/or NF Client, it must be created prior to running the next steps. Otherwise the options of APPWAN_PRIVATE_GATEWAY and APPWAN_PRIVATE_CLIENT can be left blank and added after the appwan is created using the steps described in the Console UI section above. GATEWAY_NAME and SERVICE_NAME are automatically generated by the scripts in this version. GATEWAY_NAME = "GW TYPE"+x0x+"LOCATION OF AZURE GW", e.g. AZCPEGWx0xWESTUS; SERVICE_NAME = "GW NAME"--"SERVICE IP"--"SERVICE PORT", e.g. AZCPEGWx0xWESTUS--10.20.10.5--22.



Setting Up Jenkins Pipeline

1. Login to Jenkins

2.



Click on " New Item"

3. Name you Project, select pipeline option and click "Ok"

Enter an item name

NetFoundryDeploy2Cloud

» Required field

Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining something other than software build.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project
Suitable for projects that need a large number of different configurations, such as...

OK

4. In the pipeline details, fill in the scm details as seen in the image below and click "Save".

Everything default apart from:

- Repository Url: <https://github.com/netfoundry/mop.git>

b. Script Path: pipeline/netfoundrydeploy2cloud.jenkinsfile

Pipeline

Definition: Pipeline script from SCM

SCM: Git

Repositories

Repository URL:

Credentials:

Branches to build

Branch Specifier (blank for 'any'):

Repository browser: (Auto)

Additional Behaviours:

Script Path:

Lightweight checkout:

[Pipeline Syntax](#)

5. Set up users for Azure API and NF MOP API access -- More on Credentials setup

Jenkins

Jenkins > Credentials

- New Item
- People
- Build History
- Manage Jenkins
- My Views
- Open Blue Ocean
- Lockable Resources
- Credentials**
- System
- New View

Build Queue

No builds in the queue.

Credentials

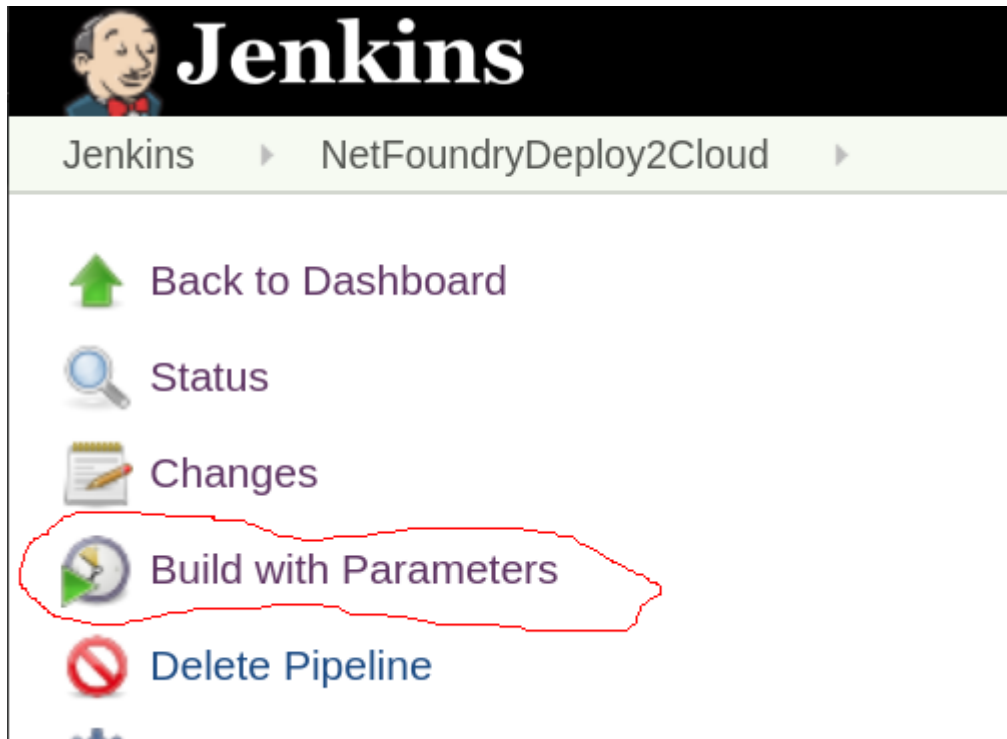
T	P	Store ↓	Domain	ID
		Jenkins	(global)	azure_user_creds 164b
		Jenkins	(global)	sandbox-mop-user Qj9K

Icon: [S](#) [M](#) [L](#)

Stores scoped to Jenkins

P	Store ↓
	Jenkins (global)

6. Run Jenkinsjob by selecting on the pipeline created in the previous step. Click on "Build with Parameters"





To create the resources

1. Fill in the Azure Details (e.g. RG, Tenant Id, etc) and select the following:
 - a. NF Environment, e.g. production
 - b. NETWORK_ACTION - create
 - c. NETWORK_NAME, e.g. DEMONET
 - d. GATEWAY_ACTION - create
 - e. If Azure RG needs to be preserved, then KEEP_RG option must be left checked.
 - f. LOCATION, e.g. westus - location where the Azure GW will be deployed in
 - g. SUBNET_PREFIX, e.g. 10.20.10.0/24 - the subnet used for the vNet in the location of the Azure GW deployment.

The screenshot shows the Jenkins configuration page for the 'NetFoundryDeploy2Cloud' pipeline. The parameters are as follows:

Parameter	Value
AZURE_TENANT_ID	[Redacted]
AZURE_SUBSCRIPTION_ID	[Redacted]
RESOURCE_GROUP_NAME	[Redacted]
RESOURCE_GROUP_LOC	centralus
ENVIRONMENT	sandbox
NETWORK_ACTION	create
NETWORK_NAME	DEMONET
GATEWAY_ACTION	create
GATEWAY_NAME	
SERVICE_ACTION	get
SERVICE_NAME	
SERVICE_IP	
SERVICE_PORT	
APPWAN_ACTION	get
APPWAN_NAME	
APPWAN_PRIVATE_GATEWAY	
APPWAN_PRIVATE_CLIENT	
APPWAN_SERVICE	
LOCATION	westus
SUBNET_PREFIX	10.20.10.0/24

The 'Build' button is located at the bottom center of the configuration area.

2. Run Jenkins job again by selecting on the pipeline created in the previous step. Click on "Build with Parameters"

3. Fill in service and appwan details by selecting the following:

- a. KEEP_RG - not selected
- b. NF Environment, e.g. production
- c. SERVICE_ACTION - create
- d. APPWAN_ACTION - create
- e. GATEWAY_NAME, e.g. AZCPEGWx0xWESTUS (this is created in the previous step automatically)
- f. SERVICE_NAME, e.g. AZCPEGWx0xWESTUS--10.20.10.5--22 (this is created automatically during this step)
- g. SERVICE_IP, e.g. 10.20.10.5
- h. SERVICE_PORT, e.g. 22
- i. APPWAN_NAME, e.g. appwan-ssh-22
- j. APPWAN_PRIVATE_GATEWAY, e.g. private-gateway-name (this is created outside of the jenkins job, prior to running this step)
- k. APPWAN_PRIVATE_CLIENT, e.g. client-name (this is created outside of the jenkins job, prior to running this step)

I. APPWAN_SERVICE, e.g. AZCPEGWx0xWESTUS--10.20.10.5--22

Jenkins x +

localhost:8080/job/NetFoundryDeploy2Cloud/build?delay=0sec

Jenkins > NetFoundryDeploy2Cloud >

- Build with Parameters
- Delete Pipeline
- Configure
- Full Stage View
- Open Blue Ocean
- Rename
- Pipeline Syntax

Build History trend

Atom feed for all Atom feed for failures

AZURE_TENANT_ID	<input type="text"/>	Tenant ID in Azure
AZURE_SUBSCRIPTION_ID	<input type="text"/>	Subscription ID in Azure
RESOURCE_GROUP_NAME	<input type="text"/>	RG Name in Azure
RESOURCE_GROUP_LOC	<input type="text"/>	RG Location in Azure
ENVIRONMENT	<input type="text" value="sandbox"/>	<input type="checkbox"/> KEEP_RG Not to check this if RG can be deleted Select NF Console Environment to spin the network and gateways in
NETWORK_ACTION	<input type="text" value="get"/>	Selection an action to perform on the network in NF
NETWORK_NAME	<input type="text" value="DEMONET"/>	Name to be used to create a network with
GATEWAY_ACTION	<input type="text" value="get"/>	Selection an action to perform on the gateway in NF Network
GATEWAY_NAME	<input type="text" value="AZCPEGWx0xWESTUS"/>	Name of NF Gateway generated in NF Console
SERVICE_ACTION	<input type="text" value="create"/>	Selection an action to perform on the service in NF Network
SERVICE_NAME	<input type="text" value="AZCPEGWx0xWESTUS--10.20.10.5--22"/>	Name of NF Service generated in NF Console
SERVICE_IP	<input type="text" value="10.20.10.5"/>	IP of NF Service App
SERVICE_PORT	<input type="text" value="22"/>	IP of NF Service App
APPWAN_ACTION	<input type="text" value="create"/>	Selection an action to perform on the appwan in NF Network
APPWAN_NAME	<input type="text" value="appwan-ssh-22"/>	Name of NF APPWAN to be used in NF Console
APPWAN_PRIVATE_GATEWAY	<input type="text" value="private-gateway-name"/>	Endpoint Name in Private Datacenter Gateway to be included in AppWan
APPWAN_PRIVATE_CLIENT	<input type="text" value="client-name"/>	Endpoint Name for Client to be included in AppWan
APPWAN_SERVICE	<input type="text" value="AZCPEGWx0xWESTUS--10.20.10.5--22"/>	Service Name to be included in AppWan
LOCATION	<input type="text" value="westus"/>	Azure Cloud DC Location where to deploy GW
SUBNET_PREFIX	<input type="text" value="10.20.10.0/24"/>	Subnet CIDR in Azure Cloud DC Location where to deploy GW

Build



To delete the resources

1. Run Jenkins job again by selecting on the pipeline created in the previous step. Click on "Build with Parameters"
2. Fill in the Azure Details (e.g. RG, Tenant Id, etc) and select the following:
 - a. NF Environment, e.g. production
 - b. NETWORK_ACTION - delete
 - c. NETWORK_NAME, e.g. DEMONET
 - d. GATEWAY_ACTION - delete

Pipeline View

Jenkins

Jenkins > NetFoundryDeploy2Cloud >

- Back to Dashboard
- Status
- Changes
- Build with Parameters
- Delete Pipeline
- Configure
- Full Stage View
- Open Blue Ocean
- Rename
- Pipeline Syntax

Build History [trend](#)

[Atom feed for all](#) [Atom feed for failures](#)

Pipeline NetFoundryDeploy2Cloud

This build requires parameters:

AZURE_TENANT_ID
Tenant ID in Azure

AZURE_SUBSCRIPTION_ID
Subscription ID in Azure

RESOURCE_GROUP_NAME
RG Name in Azure

RESOURCE_GROUP_LOC
RG Location in Azure

KEEP_RG
Not to check this if RG can be deleted

ENVIRONMENT
Select NF Console Environment to spin the network and gateways in

NETWORK_ACTION
Selection an action to perform on the network in NF

NETWORK_NAME
Name to be used to create a network with

GATEWAY_ACTION
Selection an action to perform on the gateway in NF Network

GATEWAY_NAME
Name of NF Gateway generated in NF Console

SERVICE_ACTION
Selection an action to perform on the service in NF Network

SERVICE_NAME
Name of NF Service generated in NF Console

SERVICE_IP
IP of NF Service App

SERVICE_PORT
IP of NF Service App

APPWAN_ACTION
Selection an action to perform on the appwan in NF Network

APPWAN_NAME
Name of NF APPWAN to be used in NF Console

APPWAN_PRIVATE_GATEWAY
Endpoint Name in Private Datacenter Gateway to be included in AppWan

APPWAN_PRIVATE_CLIENT
Endpoint Name for Client to be included in AppWan

APPWAN_SERVICE
Service Name to be included in AppWan

LOCATION
Azure Cloud DC Location where to deploy GW

SUBNET_PREFIX
Subnet CIDR in Azure Cloud DC Location where to deploy GW

Build

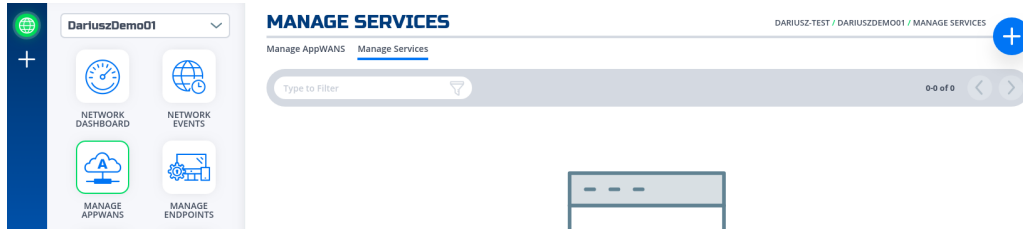
3. Done

Create IP Host Service

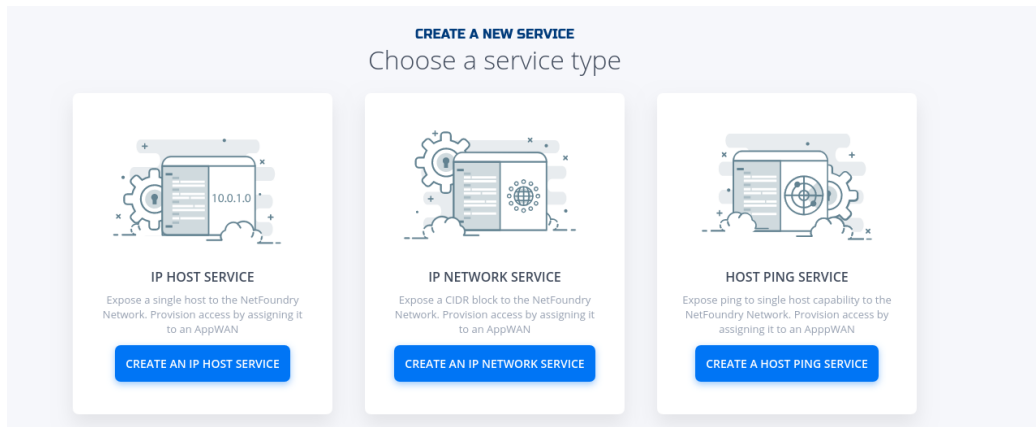
This section will guide a user through the steps on how to create a NF Service.



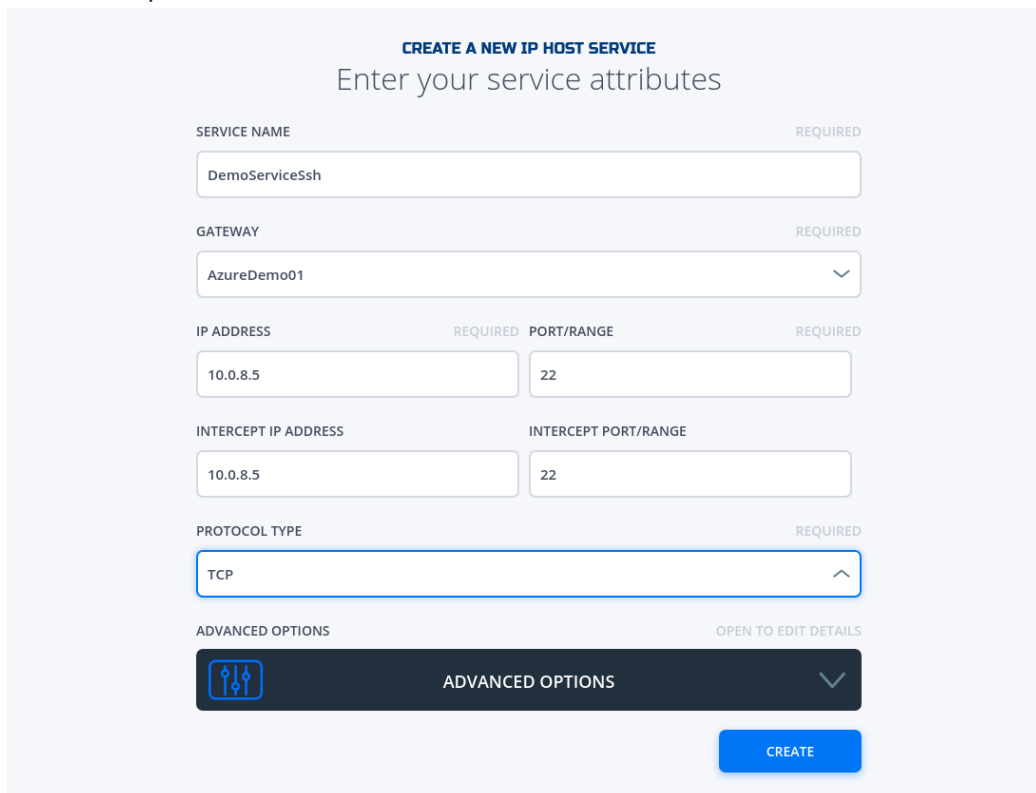
1. Navigate to Manage Services Page under Manage Appwans
2. Click on + sign in the top right corner.



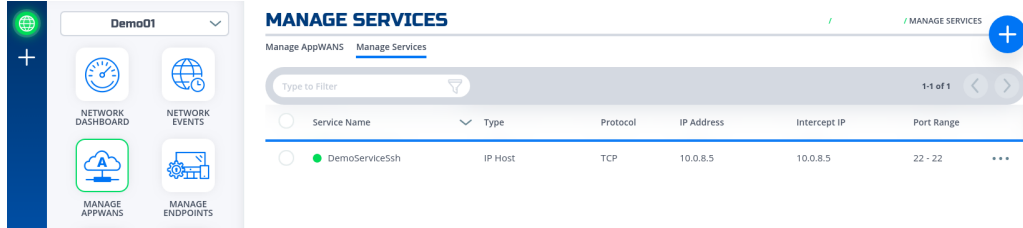
3. Click on "Create an IP Host Service"



4. Fill in the required information for SSH and click on "Create"



5. If successfully, the service is green.



The screenshot shows a web interface for managing services. On the left is a navigation menu with icons for Network Dashboard, Network Events, Manage AppWans, and Manage Endpoints. The main area is titled 'MANAGE SERVICES' and includes a search bar and a table of services. The table has columns for Service Name, Type, Protocol, IP Address, Intercept IP, and Port Range. One service, 'DemoServiceSsh', is listed with a green status indicator.

Service Name	Type	Protocol	IP Address	Intercept IP	Port Range
DemoServiceSsh	IP Host	TCP	10.0.8.5	10.0.8.5	22 - 22

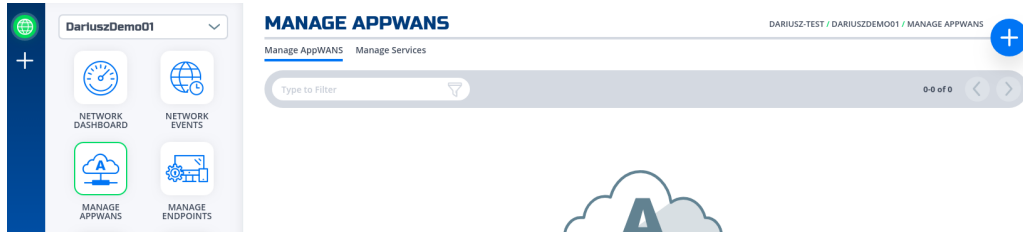
6. Done

Create AppWan

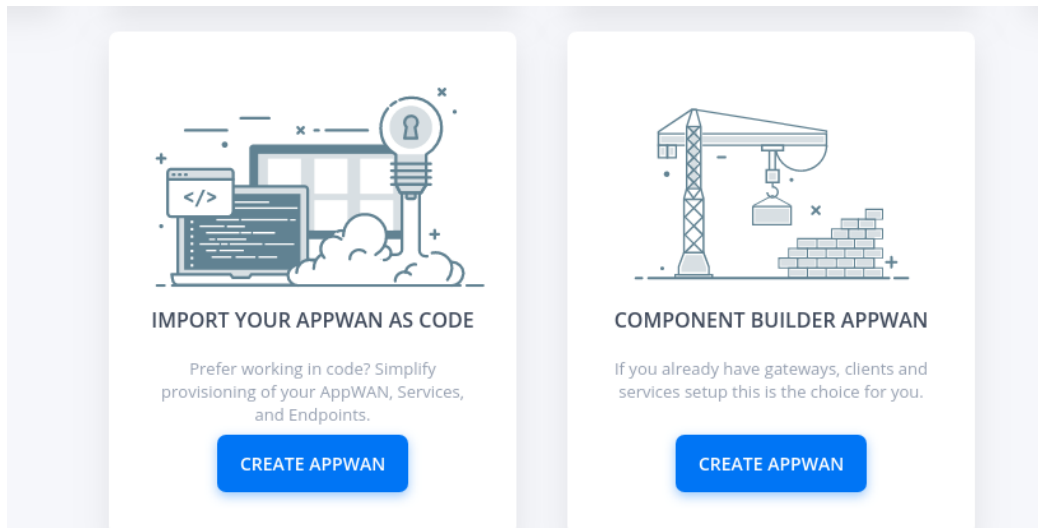
This section will guide a user through the steps on how to enable service connectivity to users by creating an appwan.



1. Navigate to Manage AppWANS Page under Manage Appwans
2. Click on + sign in the top right corner.



3. Click on "Component Builder Appwan"



4. Move the desired gateway (e.g. DemoGateway01) from "Available" Gateways to "Selected" Endpoints. Move the desired service (e.g. DemoServiceSsh) from "Available" to "Selected"

Services.

CREATE A NEW APPWAN

Choose from existing components, or add new ones

1 APPWAN NAME REQUIRED

2 ADD CLIENTS, GATEWAYS, OR ENDPOINT GROUPS

AVAILABLE GROUPS ADD NEW +

AVAILABLE CLIENTS ADD NEW +

AVAILABLE GATEWAYS ADD NEW +

- AzureDemo01

SELECTED ENDPOINTS

- YourBranchGatewayName

3 ADD SERVICES

AVAILABLE SERVICES ADD NEW +

SELECTED SERVICES

- DemoServiceSsh

CREATE


5. Click on "Create".

YOUR APPWAN SUMMARY


Your AppWAN has been created! A network summary is below.


What's next? Finish connecting your network by registering new clients and gateways.

HINT NEW CLIENTS


Share Client Registration Info 

HINT NEW GATEWAYS


Tap to Launch and Register 

- 1 APPWAN NAME**
DemoAppWan 
- 2 ENDPOINTS**

<u>CLIENTS</u>	SHARE NEW CLIENTS
<u>GATEWAYS</u>	REGISTER NEW GATEWAYS

 -  [YOURBRANCHGATEWAYNAME !\[\]\(abba236d415e988a44fd320623ba9ff7_img.jpg\)](#)
- 3 SERVICES**

<u>SERVICE DEFINITIONS</u>

 -  [DemoServiceSsh !\[\]\(a0eb3c6c24908e2ffb06a6e01c517ab5_img.jpg\)](#)
- 4 ENDPOINT GROUPS**

<u>GROUPS</u>

6. Done