Azure Project: Azure Hub-and-Spoke Firewall Lab (Cloud Shell)

Objective

The objective of this lab is to design and deploy a **beginner-friendly Hub-and-Spoke network topology in Microsoft Azure** using **Azure Firewall (Basic)**. The Hub hosts the centralized firewall, while the Spoke contains a private Ubuntu VM running Nginx. A **DNAT rule** is configured to securely publish the web page through the firewall, and a **route table** is applied to ensure that the firewall inspects all outbound traffic from the Spoke.

Tools & Services Used

- Azure Cloud Shell (Bash)

- Azure Resource Group

- Azure Virtual Machine (Ubuntu)

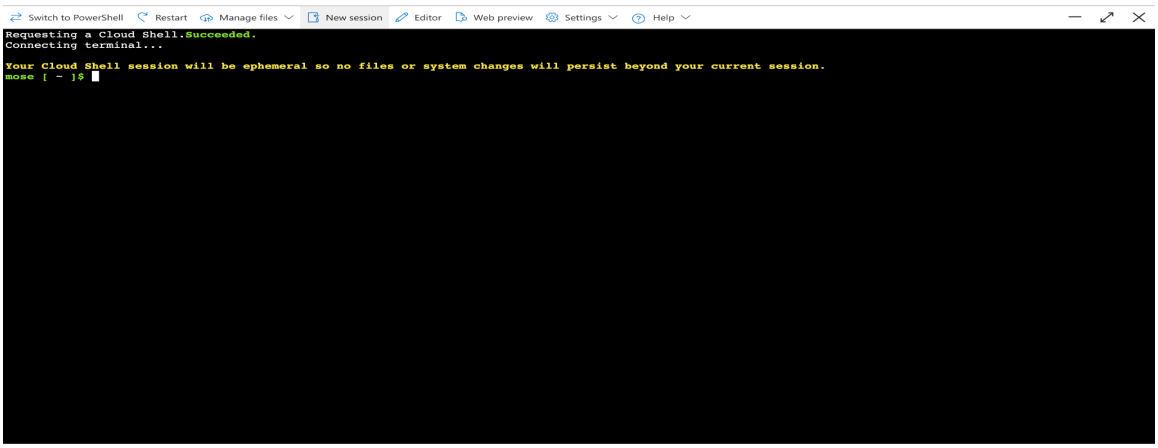
- DNAT

- Nginx Web Server

Step-by-Step Implementation

Step 1: Open Azure Cloud Shell

Go to portal.azure.com → Open Cloud Shell → Select Bash.



Step 2: Variables & Resource Group

Command:

RG=rg-azfw-lab –LOC=eastus –HUB_VNET=hub-vnet –SPOKE_VNET=spoke-web-vnet –AZFW_NAME=azfw –FW_PIP=fw-pip –SPOKE_SUBNET=web-subnet –az group create -n $RG -l $LOC

```
mose [ ~ ]$ RG=rg-azfw-lab
LOC=eastus
HUB_VNET=hub-vnet
SPOKE_VNET=spoke-web-vnet
AZFW_NAME=azfw
FW_PIP=fw-pip
SPOKE_SUBNET=web-subnet

az group create -n $RG -l $LOC
{
  "id": "/subscriptions/a63c3193-6450-4099-95e2-8c41ba85ae57/resourceGroups/rg-azfw-lab",
  "location": "eastus",
  "managedBy": null,
  "name": "rg-azfw-lab",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null,
  "type": "Microsoft.Resources/resourceGroups"
}
mose [ ~ ]$
```

Step 3: Peering (Hub ↔ Spoke)

Command:

az network vnet peering create -g $RG --vnet-name $HUB_VNET -n hub-to-spoke --remote-vnet $SPOKE_VNET --allow-vnet-access --allow-forwarded-traffic

az network vnet peering create -g $RG --vnet-name $SPOKE_VNET -n spoke-to-hub --remote-vnet $HUB_VNET --allow-vnet-access --allow-forwarded-traffic

```
mose [ ~ ]$ # Hub VNet with the required AzureFirewallSubnet
az network vnet create \
  -g $RG -n $HUB_VNET -l $LOC \
  --address-prefixes 10.0.0.0/16 \
  --subnet-name AzureFirewallSubnet --subnet-prefix 10.0.1.0/24

# Spoke VNet with one workload subnet
az network vnet create \
  -g $RG -n $SPOKE_VNET -l $LOC \
  --address-prefixes 10.1.0.0/16 \
  --subnet-name $SPOKE_SUBNET --subnet-prefix 10.1.1.0/24
```

```
mose [ ~ ]$ # Hub -> Spoke
az network vnet peering create -g $RG \
  --vnet-name $HUB_VNET -n hub-to-spoke \
  --remote-vnet $SPOKE_VNET --allow-vnet-access --allow-forwarded-traffic

# Spoke -> Hub
az network vnet peering create -g $RG \
  --vnet-name $SPOKE_VNET -n spoke-to-hub \
  --remote-vnet $HUB_VNET --allow-vnet-access --allow-forwarded-traffic
```

Step 4: Create the Web VM (private only) + Nginx

Command:

az vm create –g $RG -n webl --image Ubuntu2404 –size Standard_Bls --admin-username azureuser --ssh-key-values ~/.ssh/id_rsa.pub --vnet-name $SPOKE_VNET --subnet $SPOKE_SUBNET \ –-public-ip-address "" --nsg "" --custom-data cloud-init-nginx.yaml

```
mose [ ~ ]$ cat > cloud-init-nginx.yaml <<'EOF'
#cloud-config
package_update: true
packages:
  - nginx
runcmd:
  - systemctl enable --now nginx
EOF

az vm create \
  -g $RG -n web1 --image Ubuntu2404 \
  --size Standard_B1s \
  --admin-username azureuser \
  --ssh-key-values ~/.ssh/id_rsa.pub \
  --vnet-name $SPOKE_VNET --subnet $SPOKE_SUBNET \
  --public-ip-address "" \
  --nsg "" \
  --custom-data cloud-init-nginx.yaml
The default value of '--size' will be changed to 'Standard_D2s_v5' from 'Standard_DS1_v2' in a future release.
{
  "fqdns": "",
  "id": "/subscriptions/a63c3193-6450-4099-95e2-8c41ba85ae57/resourceGroups/rg-azfw-lab/providers/Microsoft.Compute/virtualMachines/web1",
  "location": "eastus",
  "macAddress": "7C-1E-52-45-C8-E3",
  "powerState": "VM running",
  "privateIpAddress": "10.1.1.4",
  "publicIpAddress": "",
  "resourceGroup": "rg-azfw-lab"
}
```

Step 5: Deploy Azure Firewall (Basic) in the Hub

Command:
az network public-ip create -g $RG -n $FW_PIP -l $LOC --sku Standard --allocation-method Static az network firewall create -g $RG -n $AZFW_NAME -l $LOC --sku AZFW_VNet --tier Basic az network firewall ip-config create -g $RG -f $AZFW_NAME -n azfw-ipconfig --public-ip-address $FW_PIP --vnet-name $HUB_VNET FW_PRIV_IP=$(az network firewall show -g $RG -n $AZFW_NAME --query "ipConfigurations[0].privateIPAddress" -o tsv FW_PUB_IP=$(az network public-ip show -g $RG -n $FW_PIP --query ipAddress -o tsv) echo "Firewall Private IP: $FW_PRIV_IP" echo "Firewall Public IP: $FW_PUB_IP"

```
AzureFirewallManagementSubnet",
  "name": "AzureFirewallManagementSubnet",
  "privateEndpointNetworkPolicies": "Disabled",
  "privateLinkServiceNetworkPolicies": "Enabled",
  "provisioningState": "Succeeded",
  "resourceGroup": "rg-azfw-lab",
  "type": "Microsoft.Network/virtualNetworks/subnets"
}
Name                          Prefix
----------------------------  -----------
AzureFirewallSubnet           10.0.1.0/24
AzureFirewallManagementSubnet  10.0.2.0/26
mose [ ~ ]$
```

Step 6: DNAT to the Private Web VM

Commands:
WEB_PRIV_IP=$(az vm show -g $RG -n web1 -d --query privateIps -o tsv) echo "Web VM Private IP: $WEB_PRIV_IP" # Create a DNAT rule so http://$FW_PUB_IP:8080 reaches http://$WEB_PRIV_IP:80 az network firewall nat-rule create -g $RG -f $AZFW_NAME --collection-name dnat-web --priority 100 --name web-http --protocols TCP --source-addresses "*" --destination-addresses $FW_PUB_IP --destination-ports 8080 --translated-address $WEB_PRIV_IP --translated-port 80

```
Creating rule collection 'dnat-web'.

{
  "description": null,
  "destinationAddresses": [
    "20.120.96.85"
  ],
  "destinationPorts": [
    "8080"
  ],
  "name": "web-http",
  "protocols": [
    "TCP"
  ],
  "sourceAddresses": [
    "*"
  ],
  "sourceIpGroups": [],
  "translatedAddress": "10.1.1.4",
  "translatedFqdn": null,
  "translatedPort": "80"
}
```

Step 7: Route Spoke Outbound Traffic via the Firewall

Command:

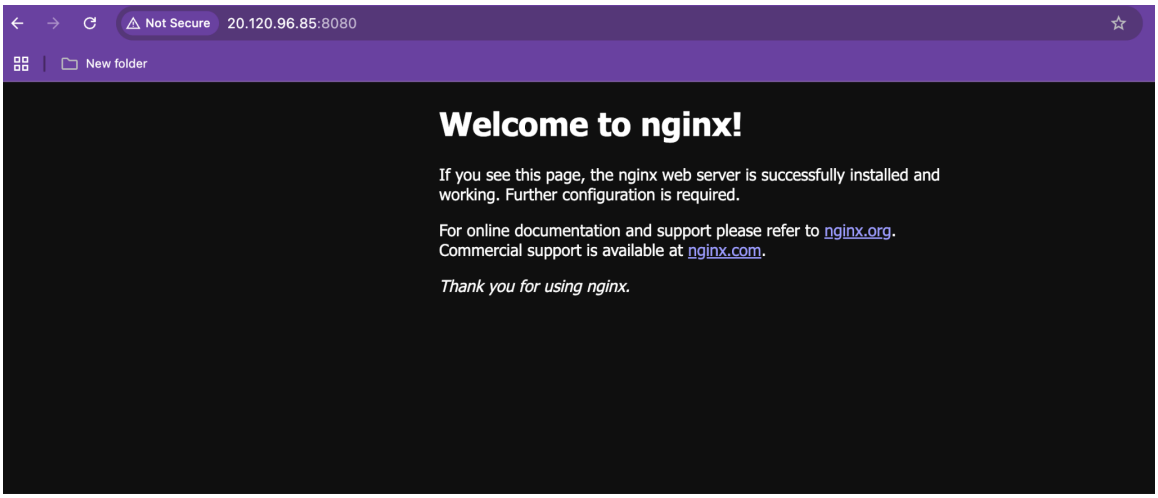
az network route-table create -g $RG -n rt-spoke az network route-table route create -g $RG --route-table-name rt-spoke -n default-to-fw --address-prefix 0.0.0.0/0 --next-hop-type VirtualAppliance --next-hop-ip-address $FW_PRIV_IP # Associate the route table with the spoke subnet az network vnet subnet update -g $RG --vnet-name $SPOKE_VNET --name $SPOKE_SUBNET --route-table rt-spoke

```
/home/mose/.azure/cliextensions/azure-firewall/azext_firewall/vendored_sdks/__init__.py:6: UserWarning: pkg_resources is deprecated as an API. See ht
tps://setuptools.pypa.io/en/latest/pkg_resources.html. The pkg_resources package is slated for removal as early as 2025-11-30. Refrain from using thi
s package or pin to Setuptools<81.
  __import__('pkg_resources').declare_namespace(__name__)
Firewall Private IP  : 10.0.1.4
Firewall Public  IP  : 20.120.96.85
Mgmt Public IP (Basic): 172.191.179.143
```

Step 8: Verify From Browser

Command:

http://20.120.96.85:8080



Step 8A: Verify In Shell

Command:
curl -I http://20.120.96.85:8080

Results

- Built a working **hub-and-spoke** network in Azure.
- Stood up **Azure Firewall (Basic)** as the single ingress/egress control point.
- Deployed a **private** Ubuntu/Nginx VM (no public IP) in the spoke.
- Published the site **safely via DNAT**.
- Forced all spoke **outbound** through the firewall using a **UDR**.