

Use Azure network security groups (NSGs) to filter network traffic between Azure resources in an Azure Virtual Network. An NSG is assigned to an Azure VNet subnet and contains inbound and outbound rules.

## Learning Objectives

In this lab, you will learn how to do the following:

Log into the Azure CLI

Understand Azure network security groups (NSGs)

Create an NSG for a subnet using Azure CLI

Confirm that the NSG is assigned using Azure CLI

Check that the NSG is assigned in the Azure Portal

Remove the NSG using Azure CLI

Login to azure :

Az login -u \$username -p \$password

To create a new Azure Virtual Network and a subnet:

```
az network vnet create --resource-group $resource --name $vnetName --address-prefix 10.0.0.0/24 --subnet-name subnet01 --subnet-prefix 10.0.0.0/28
```

In this lab, we will create a new Azure network security group (NSG) for this subnet using Azure CLI.

```
Terminal
$ echo $username
user-zknoezsbkeou@oreilly-cloudlabs.com
$ echo $password
tJPhAmWCaDXBL0
$ az network vnet create --resource-group $resource --name $vnetName --address-prefix 10.0.0.0/24 --subnet-name subnet01 --subnet-prefix 10.0.0.0/28
{
  "newVNet": {
    "addressSpace": {
      "addressPrefixes": [
        "10.0.0.0/24"
      ]
    },
    "dhcpOptions": {
      "dnsServers": [
      ]
    },
    "enableDdosProtection": false,
    "enableVpnProtection": null,
    "encryption": null,
    "etag": "W/\"e0e06e44-1628-4abe-a6ca-12069212d055\"",
    "extendedLocation": null,
    "flowTimeoutInMinutes": null,
    "id": "/subscriptions/dac10b38-92f3-4492-85f4-4f5006fa0c22/resourceGroups/user-zknoezsbkeou/providers/Microsoft.Network/virtualNetworks/vnet179305410",
    "ipAllocations": null,
    "location": "eastus",
    "name": "vnet179305410",
    "provisioningState": "Succeeded",
    "resourceGroup": "user-zknoezsbkeou",
    "resourceId": "/subscriptions/dac10b38-92f3-4492-85f4-4f5006fa0c22/resourceGroups/user-zknoezsbkeou/providers/Microsoft.Network/virtualNetworks/vnet179305410",
    "subnets": [
      {
        "addressPrefix": "10.0.0.0/28",
        "addressPrefixes": null,
        "applicationGatewayIpConfigurations": null,
        "delegations": [
        ],
        "etag": "W/\"e0e06e44-1628-4abe-a6ca-12069212d055\"",
        "id": "/subscriptions/dac10b38-92f3-4492-85f4-4f5006fa0c22/resourceGroups/user-zknoezsbkeou/providers/Microsoft.Network/virtualNetworks/vnet179305410/subnets/subnet01",
        "ipAllocations": null,
        "ipConfigurationProfiles": null,
        "ipConfigurations": null,
        "name": "subnet01",
        "natGateway": null,
        "networkSecurityGroup": null,

```

## Understand Azure NSGs

Similar to other computer networks, you should secure the traffic coming in (ingress) or going out (egress) of our network. Microsoft Azure offers three main services to secure and monitor traffic at the network level:

Azure Firewall

Azure network security groups (NSG)

Azure application security groups (ASG)

In this lab we are talking about NSGs. Using Azure network security groups (NSGs), you can filter inbound and outbound Azure subnet traffic based on its source IP address, destination IP address, and port number. Similar to route tables, NSGs are assigned to Azure Virtual Network subnets.

In this lab, we will create a new NSG and assign it to an existing Azure VNet subnet using the Azure CLI.

Use the following command to confirm that the Azure VNet is present:

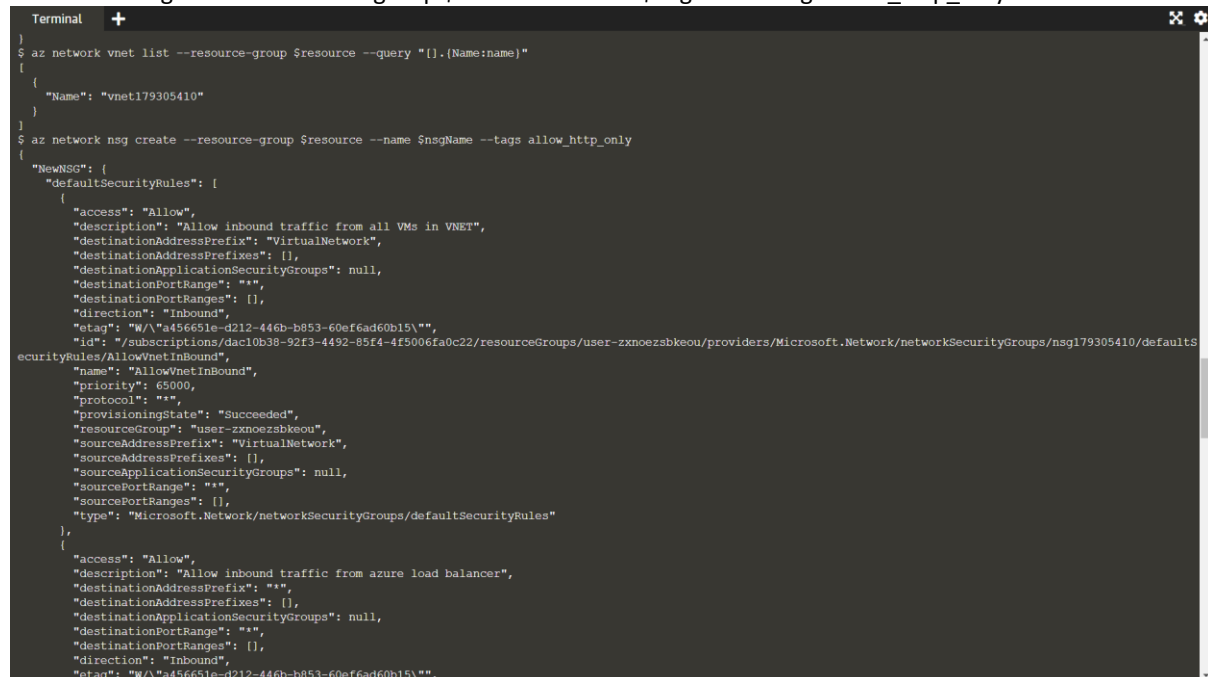
```
$ az network vnet list --resource-group $resource --query "[].{Name:name}"
[
  {
    "Name": "vnet179305410"
  }
]
$
```

In the next step, let's create a new NSG and assign it to our subnet.

Configure an NSGs using Azure CLI

Use the following command to create a new NSG:

```
az network nsg create --resource-group $resource --name $nsgName --tags allow_http_only
```



```
Terminal +
$ az network nsg create --resource-group $resource --name $nsgName --tags allow_http_only
{
  "Name": "vnet179305410"
}
$ az network nsg create --resource-group $resource --name $nsgName --tags allow_http_only
{
  "NewNSG": {
    "defaultSecurityRules": [
      {
        "access": "Allow",
        "description": "Allow inbound traffic from all VMs in VNET",
        "destinationAddressPrefix": "VirtualNetwork",
        "destinationAddressPrefixes": [],
        "destinationApplicationSecurityGroups": null,
        "destinationPortRange": "*",
        "destinationPortRanges": [],
        "direction": "Inbound",
        "etag": "W/\"a456651e-d212-446b-b853-60ef6ad60b15\"",
        "id": "/subscriptions/dac10b38-92f3-4492-85f4-4f5006fa0c22/resourceGroups/user-zxnoezsbkeou/providers/Microsoft.Network/networkSecurityGroups/nsg179305410/defaultSecurityRules/AllowVnetInBound",
        "name": "AllowVnetInBound",
        "priority": 65000,
        "protocol": "*",
        "provisioningState": "Succeeded",
        "resourceGroup": "user-zxnoezsbkeou",
        "sourceAddressPrefix": "VirtualNetwork",
        "sourceAddressPrefixes": [],
        "sourceApplicationSecurityGroups": null,
        "sourcePortRange": "*",
        "sourcePortRanges": [],
        "type": "Microsoft.Network/networkSecurityGroups/defaultSecurityRules"
      },
      {
        "access": "Allow",
        "description": "Allow inbound traffic from azure load balancer",
        "destinationAddressPrefix": "*",
        "destinationAddressPrefixes": [],
        "destinationApplicationSecurityGroups": null,
        "destinationPortRange": "*",
        "destinationPortRanges": [],
        "direction": "Inbound",
        "etag": "W/\"a456651e-d212-446b-b853-60ef6ad60b15\"",

```

The command "az network nsg create" creates a new Network Security Group (NSG) with the specified name and resource group, and applies the specified tags to it. The NSG can be used to filter inbound and outbound traffic to and from Azure resources.

In this particular command, the value "allow\_http\_only" is being used as a tag for the NSG. This tag could be used to indicate that only HTTP traffic should be allowed through the NSG, and that all other traffic should be blocked. However, it's important to note that tags are just labels and do not actually configure any network security rules.

To create the NSG with actual network security rules, additional commands will need to be run. For example, you might use the "az network nsg rule create" command to create a rule that allows HTTP traffic to pass through the NSG.

We already created an Azure Virtual Network and a subnet for you. Imagine that an Azure Virtual Machine (VM) is deployed to this subnet and you need to protect it by securing its network traffic. This imaginary VM will be a web server, so you need to make sure all incoming (inbound/ingress) HTTP(S) traffic is allowed. You need to block other traffic.

Use the following command to create a new rule to allow TCP ports 80 and 443 from any source, including the public internet and add it to your NSG:

```
az network nsg rule create --resource-group $resource --nsg-name $nsgName --name allow_http --priority 200
--source-address-prefixes Internet --source-port-ranges '*' --destination-address-prefixes '*' --destination-port-
ranges 80 443 --access Allow --protocol Tcp --direction Inbound --description "Allow from internet IP addresses
on ports 80 and 443."
```

Here are the command parameters:

--resource-group: Parent resource group for the NSG.

--nsg-name: The NSG name to which to add this rule.

--name: The name for the new rule.

--source-address-prefixes: The incoming resource address; a space-separated list of CIDR prefixes or IP ranges. Alternatively, you can specify one of the values VirtualNetwork, AzureLoadBalancer, Internet, or \* (to match all IPs). It also supports all available service tags, such as ApiManagement, SqlManagement, AzureMonitor, etc.

--source-port-ranges: The incoming source ports.

--destination-address-prefixes: The destination address; a space-separated list of CIDR prefixes or IP ranges. Alternatively, specify you can specify one of the values VirtualNetwork, AzureLoadBalancer, Internet or \* (to match all IPs). It also supports all available service tags such as ApiManagement, SqlManagement, AzureMonitor, etc.

--destination-port-ranges: The destination port range.

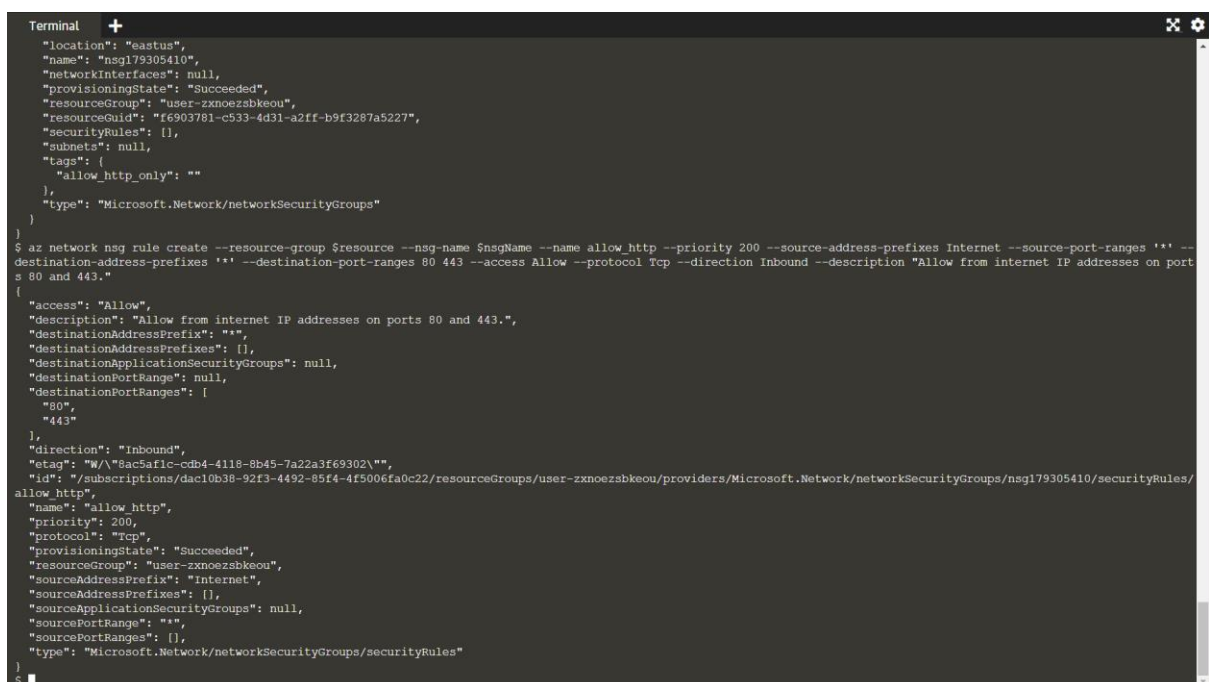
--access: Allow or Deny.

--protocol: \*, Ah, Esp, Icmp, Tcp, or Udp.

--direction: Inbound or Outbound.

--priority: Rule priority, between 100 (highest priority) and 4096 (lowest priority). This must be unique for each rule in the collection. Rules with a lower priority number will be processed first.

--description: Parent resource group for the NSG.



```

Terminal
"location": "eastus",
"name": "nsg179305410",
"networkInterfaces": null,
"provisioningState": "Succeeded",
"resourceGroup": "user-zxnoezsbkeou",
"resourceId": "r6903781-c533-4d31-a2ff-b9f3287a5227",
"securityRules": [],
"subnets": null,
"tags": {
  "allow_http_only": ""
},
"type": "Microsoft.Network/networkSecurityGroups"
}
$ az network nsg rule create --resource-group $resource --nsg-name $nsgName --name allow_http --priority 200 --source-address-prefixes Internet --source-port-ranges '*' --destination-address-prefixes '*' --destination-port-ranges 80 443 --access Allow --protocol Tcp --direction Inbound --description "Allow from internet IP addresses on port 80 and 443."
{
  "access": "Allow",
  "description": "Allow from internet IP addresses on ports 80 and 443.",
  "destinationAddressPrefix": "*",
  "destinationAddressPrefixes": [],
  "destinationApplicationSecurityGroups": null,
  "destinationPortRange": null,
  "destinationPortRanges": [
    "80",
    "443"
  ],
  "direction": "Inbound",
  "etag": "W/\"8ac5af1c-cdb4-4118-8b45-7a22a3f69302\"",
  "id": "/subscriptions/dac10b38-92f3-4492-85f4-4f5006fa0c22/resourceGroups/user-zxnoezsbkeou/providers/Microsoft.Network/networkSecurityGroups/nsg179305410/securityRules/allow_http",
  "name": "allow_http",
  "priority": 200,
  "protocol": "Tcp",
  "provisioningState": "Succeeded",
  "resourceGroup": "user-zxnoezsbkeou",
  "sourceAddressPrefix": "Internet",
  "sourceAddressPrefixes": [],
  "sourceApplicationSecurityGroups": null,
  "sourcePortRange": "*",
  "sourcePortRanges": [],
  "type": "Microsoft.Network/networkSecurityGroups/securityRules"
}

```

Important: A single NSG can be assigned to multiple subnets. However, a subnet can only have one NSG.

Confirm the NSG Creation and Assignment using Azure CLI

Now let's confirm that the NSG exists and is assigned to our subnet. Use the following command to get a list of all NSGs in your resource group:

```
az network nsg list --resource-group $resource --query "[].{Name:name}"
```

Now, you can use the following command to confirm the NSG is assigned to your subnet:

```
az network vnet subnet show --resource-group $resource --name subnet01 --vnet-name $vnetName --query "{Name:name, NSG_ID: networkSecurityGroup.id}"
```

```

$ az network nsg list --resource-group $resource --query "[].{Name:name}"
{
  {
    "Name": "nsg179305410"
  }
}
$ az network vnet subnet show --resource-group $resource --name subnet01 --vnet-name $vnetName --query "{Name:name, NSG_ID: networkSecurityGroup.id}"
{
  "NSG_ID": null,
  "Name": "subnet01"
}
$

```

All resources in your subnet, including our imaginary VM (web server) are now protected by the new NSG. Only HTTP traffic can reach the VM at this point. You can always add or move rules to your NSG to match your project needs.

You can also check your Azure subnet NSGs in the Azure portal. Let's look at that in the next step.

Check the NSG in the Azure Portal

Confirm that you can see subnet01 in the associated subnets list.

Under Settings, click on Inbound security rules.

The screenshot shows the Azure portal interface. The top section displays 'All resources' for the subscription 'O'REILLY MEDIA CLOUD LABS (oreilly-cloudlabs.com)'. It lists two resources: 'nsg179305410' (Network security group) and 'vnet179305410' (Virtual network), both located in 'East US' under the 'cloudlabs50' subscription.

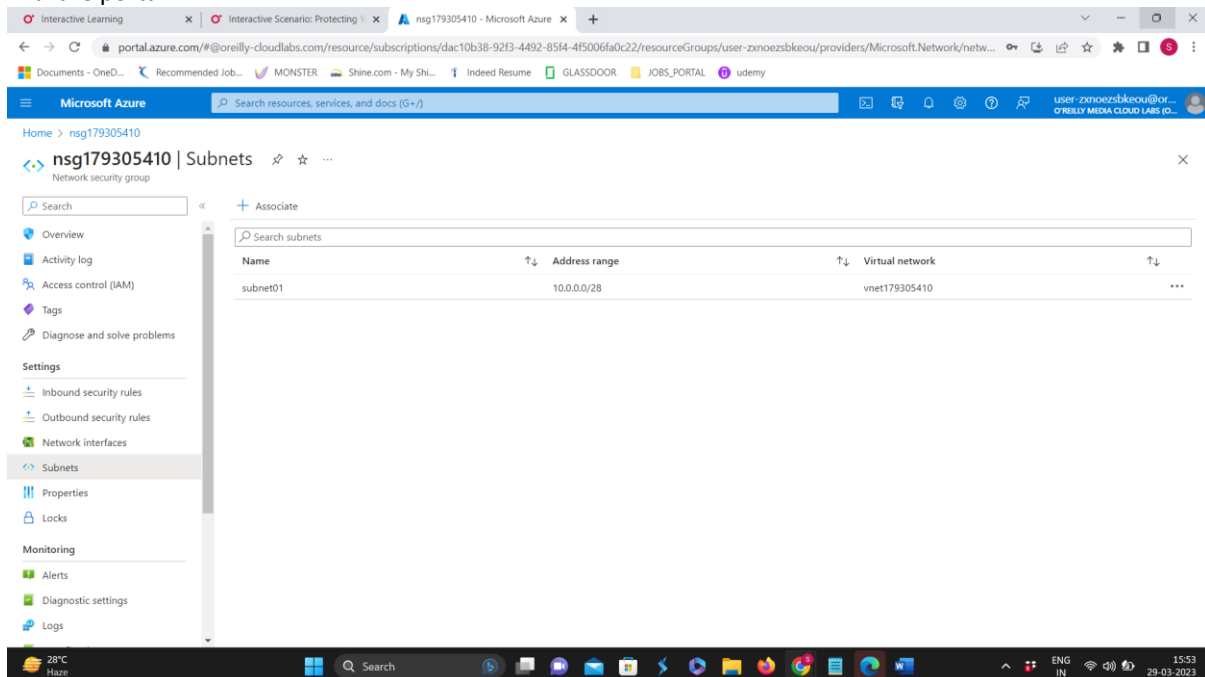
The bottom section shows the 'Inbound security rules' page for the network security group 'nsg179305410'. It displays a table of security rules with the following columns: Priority, Name, Port, Protocol, Source, and Destination.

Priority	Name	Port	Protocol	Source	Destination
200	allow_http	80,443	Tcp	Internet	Any
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork
65001	AllowAzureLoadBalan...	Any	Any	AzureLoadBalancer	Any
65500	DenyAllInBound	Any	Any	Any	Any

Update subnet to add nsg .

```
Terminal +
$ az network vnet subnet update -g $resource --vnet-name $vnetName --name subnet01 --network-security-group nsg179305410
{
  "addressPrefix": "10.0.0.0/28",
  "addressPrefixes": null,
  "applicationGatewayIpConfigurations": null,
  "delegations": [],
  "etag": "W/\"b6b62958-079c-40ac-9061-7dddcbbb2146\"",
  "id": "/subscriptions/dac10b38-92f3-4492-85f4-4f5006fa0c22/resourceGroups/user-zxnoezsbkeou/providers/Microsoft.Network/virtualNetworks/vnet179305410/subnets/subnet01",
  "ipAllocations": null,
  "ipConfigurationProfiles": null,
  "ipConfigurations": null,
  "name": "subnet01",
  "natGateway": null,
  "networkSecurityGroup": {
    "defaultSecurityRules": null,
    "etag": null,
    "flowLogs": null,
    "flushConnection": null,
    "id": "/subscriptions/dac10b38-92f3-4492-85f4-4f5006fa0c22/resourceGroups/user-zxnoezsbkeou/providers/Microsoft.Network/networkSecurityGroups/nsg179305410",
    "location": null,
    "name": null,
    "networkInterfaces": null,
    "provisioningState": null,
    "resourceGroup": "user-zxnoezsbkeou",
    "resourceGuid": null,
    "securityRules": null,
    "subnets": null,
    "tags": null,
    "type": null
  },
  "privateEndpointNetworkPolicies": "Disabled",
  "privateEndpoints": null,
  "privateLinkServiceNetworkPolicies": "Enabled",
  "provisioningState": "Succeeded",
  "purpose": null,
  "resourceGroup": "user-zxnoezsbkeou",
  "resourceNavigationLinks": null,
  "routeTable": null,
  "serviceAssociationLinks": null,
  "serviceEndpointPolicies": null,
  "serviceEndpoints": null,
  "type": "Microsoft.Network/virtualNetworks/subnets"
}
```

In azure portal



Subnet01 is added to nsg .