Network Security Groups

**Network Security Groups (NSGs)** are built-in tools for network control that allow us to control incoming and outgoing traffic on a network interface or at the subnet level. They contain sets of rules that allow or deny specific traffic to specific resources or subnets in Azure. An NSG can be associated with either a subnet (by applying security rules to all resources associated with the subnet) or a **Network Interface Card (NIC)**, which is done by applying security rules to the **Virtual Machine (VM)** associated with the NIC.
We will cover the following recipes in this chapter:

- Creating a new NSG in the Azure portal
- Creating a new NSG with PowerShell
- Creating a new allow rule in an NSG
- Creating a new deny rule in an NSG
- Creating a new NSG rule with PowerShell
- Assigning an NSG to a subnet
- Assigning an NSG to a network interface
- Assigning an NSG to a subnet with PowerShell
- Creating an **Application Security Group** (**ASG**) (Advanced Security Group)
- Associating an ASG with a VM
- Creating rules with an NSG and an ASG

**Technical requirements**

For this chapter, the following is required:

- An Azure subscription
- Azure PowerShell


**Creating a new NSG in the Azure portal**

As a first step to more effectively control network traffic, we are going to create a new NSG.

**Getting ready**

Before you start, open your browser and go to the Azure portal, at [https://portal.azure.com](https://portal.azure.com).
How to do it...

To create a new NSG using the Azure portal, we must follow these steps:

1. In the Azure portal, select **Create a resource** and choose **Network security group** under **Networking** (or search for **network security group** in the search bar).

2. The parameters we need to define for the deployment are **Subscription**, **Resource group**, **Name**, and **Region**. An example of the required parameters is shown in **Figure 3.1**:

![Create network security group](image)

   **Figure 3.1: Creating a new NSG using the Azure portal**

After the deployment has been validated and started (it takes a few moments to complete), the NSG is ready for use.

How it works...

The NSG deployment can be initiated during a VM deployment. This will associate the NSG to the NIC associated with the deployed VM. In this case, the NSG is already associated with the resource, and rules defined in the NSG will apply only to the associated VM.

If the NSG is deployed separately, as seen in this recipe, it is not associated and the rules that are created within it are not applied until an association has been created with the NIC or the subnet. When it is associated with a subnet, the NSG rules will apply to all resources on the subnet.

Let’s move on to the next recipe to understand how to create a new NSG using PowerShell.
Creating a new NSG with PowerShell

Alternatively, we can create an NSG using PowerShell. The advantage of this approach is that we can add NSG rules in a single script, creating custom rules right after the NSG is created. This allows us to automate the deployment process and create our own default rules right after the NSG has been created.

Getting ready

Open the PowerShell console and make sure you are connected to your Azure subscription. Refer to Chapter 1, Azure Virtual Network, for a refresher on how to do this.

How to do it...

To deploy a new NSG, execute the following command:

```powershell
New-AzNetworkSecurityGroup -Name "nsg1" -ResourceGroupName "Packt-Networking-Script" -Location "westeurope"
```

How it works...

The script is using the Resource Group (RG) that was deployed in Chapter 1, Azure Virtual Network (we will use the same RG for all deployments). Otherwise, a new RG needs to be deployed prior to executing the script. The final outcome will be the same as creating a new NSG using the Azure portal: a new NSG will be created with default rules. An advantage of using PowerShell is that we can add additional rules during deployment that will help automate the process. You will see an example of this in the Creating a new NSG rule with PowerShell recipe later in this chapter.

In this recipe, you learned to create a new NSG using PowerShell. Let's move on to the next recipe to learn how to allow rules in NSG using the Azure portal.
Creating a new allow rule in an NSG

When a new NSG is created, only the default rules are present, which allow all outbound traffic and block all inbound traffic. To change these, additional rules need to be created. First, we are going to show you how to create a new rule to allow inbound traffic.

Getting ready

Before you start, open your browser and go to the Azure portal at https://portal.azure.com. Locate the previously created NSG.

How to do it...

To create a new NSG allow rule using the Azure portal, we must follow these steps:

1. In the NSG pane, locate the Inbound security rules option under Settings.
2. Click on the Add button at the top of the page and wait for the new pane, to open:

![Figure 3.2: Creating a new NSG allow rule using the Azure portal](image-url)
3. In the new pane, we need to provide information for the **Source** (location and port range), **Destination** (location and port range), **Protocol**, **Action**, **Priority**, **Name**, and **Description** fields. If you want to allow traffic, make sure you select **Allow** for **Action**. An example of how to create a rule to allow traffic over port **443** (thus allowing traffic to the web server) is shown in Figure 3.3:

**Add inbound security rule**

![Add inbound security rule](image)

- **Source**: Any
- **Source port ranges**: *
- **Destination**: Any
- **Destination port ranges**: 443
- **Protocol**: Any, TCP, UDP, ICMP
- **Action**: Allow
- **Priority**: 100
- **Name**: Port_443
- **Description**: Allow HTTPS

**Figure 3.3: Creating a rule to allow traffic over port 443**
How it works...

By default, all traffic coming from Azure Load Balancer or Azure Virtual Network is allowed. All traffic coming over the internet is denied. To change this, we need to create additional rules. Make sure you set the right priority when creating rules. Rules with the highest priority (that is, those with the lower number) are processed first, so if you have two rules, one of which is denying traffic and one of which is allowing it, the rule with higher priority will take precedence, while the one with lower priority will be ignored.

In this recipe, you learned how to create a new rule to allow inbound traffic. In the next recipe, you will learn how to create a new rule in NSG to deny traffic.

Creating a new deny rule in an NSG

When a new NSG is created, only the default rules are present. The default rules allow all outbound traffic and block all inbound traffic. To change this, additional rules need to be created. Now, we are going to show you how to create a new outbound rule to deny traffic.

Getting ready

Before you start, open your browser and go to the Azure portal at https://portal.azure.com. Locate the previously created NSG.

How to do it...

To create a new NSG deny rule using the Azure portal, we must follow these steps:

1. In the NSG pane, locate the Outbound security rules option under Settings.
2. Click on the Add button at the top of the page and wait for the new pane to open:

![NSG1 | Outbound security rules](Figure 3.4: Creating a new NSG deny rule using the Azure portal)
3. In the new pane, we need to provide information for **Source** (location and port range), **Destination** (location and port range), **Protocol**, **Action**, **Priority**, **Name**, and **Description**. If you want to deny traffic, make sure you select **Deny** for **Action**. An example of how to create a rule to deny traffic over port **22** is shown in **Figure 3.5**:

![Add outbound security rule](image)

**Figure 3.5: Adding an outbound security rule**
How it works...

All outbound traffic is allowed by default, regardless of where it is going. If we want to explicitly deny traffic on a specific port, we need to create a rule to do so. Make sure you set the priority right when creating rules. Rules with the highest priority (those with the lowest numbers) are processed first, so if you have two rules where one is denying traffic and one is allowing it, the rule with higher priority will apply.

Let's move on to the next recipe, where you will learn how to create an NSG rule using PowerShell.

Creating a new NSG rule with PowerShell

Alternatively, we can create an NSG rule using PowerShell. This command can be executed right after the NSG has been created, allowing us to create and configure an NSG in a single script. This way, we can standardize deployment and have rules applied each time an NSG is created.

Getting ready

Open the PowerShell console and make sure you are connected to your Azure subscription.

How to do it...

To create a new NSG rule, execute the following command:

```powershell
$nsg = Get-AzNetworkSecurityGroup -Name 'nsg1' -ResourceGroupName 'Packt-Networking-Script'

$nsg | Add-AzNetworkSecurityRuleConfig -Name 'Allow_HTTPS' -Description 'Allow_HTTPS' -Access Allow -Protocol Tcp -Direction Inbound -Priority 100 -SourceAddressPrefix Internet -SourcePortRange * -DestinationAddressPrefix * -DestinationPortRange 443 | Set-AzNetworkSecurityGroup
```

How it works...

Using a script, creating an NSG rule is just a matter of parameters. The Access parameter, which can be either Allow or Deny, will determine whether we want to allow traffic or deny it. The Direction parameter, which can be Inbound or Outbound, determines whether the rule is for inbound or outbound traffic. All other parameters are the same, no matter what kind of rule we want to create. Again, priority plays a very important role, so we must make sure it’s chosen correctly.
There's more...

As mentioned in the Creating a new NSG with PowerShell recipe, we can create the NSG and the rules that are needed in a single script. The following script is an example of this:

```powershell
$nsg = New-AzNetworkSecurityGroup -Name 'nsg1' -ResourceGroupName 'Packt-Networking-Script' -Location "westeurope"
$nsg | Add-AzNetworkSecurityRuleConfig -Name 'Allow_HTTPS' -Description 'Allow_HTTPS' -Access Allow -Protocol Tcp -Direction Inbound -Priority 100 -SourceAddressPrefix Internet -SourcePortRange * -DestinationAddressPrefix * -DestinationPortRange 443 | Set-AzNetworkSecurityGroup
```

This recipe explained how to create a new NSG rule using PowerShell. In the next recipe, you will learn how to assign an NSG to a subnet.

**Assigning an NSG to a subnet**

The NSG and its rules must be assigned to a resource to have any impact. Here, you are going to see how to associate an NSG with a subnet.

**Getting ready**

Before you start, open your browser and go to the Azure portal at [https://portal.azure.com](https://portal.azure.com). Locate the previously created NSG.

**How to do it...**

To assign an NSG to a subnet, follow these steps:

1. In the NSG pane, locate the **Subnets** option under **Settings**.
2. Click on the **Associate** button at the top of the page and wait for the new pane to open:
3. In the new pane, first select the virtual network that contains the subnet you want to associate the NSG with, and then select the subnet, as seen in Figure 3.7:
4. After submitting the change, the subnet will appear in a list of associated subnets:

![Figure 3.8: A list of associated subnets](image)

**How it works...**

When an NSG is associated with a subnet, the rules in the NSG will apply to all of the resources in the subnet. Note that the subnet can be associated with more than one NSG, and the rules from all the NSGs will apply in that case. Priority is the most important factor when looking at a single NSG, but when the rules from more NSGs are observed, the **Deny** rule will prevail. So, if we have two NSGs on a subnet, one with **Allow** on port **443** and another one with the **Deny** rule on the same port, traffic on this port will be denied.

Let's move on to the next recipe and learn how to assign an NSG to a network interface.

**Assigning an NSG to a network interface**

Now, we are going to widen our scope and show you how to associate an NSG with a network interface.

**Getting ready**

Before you start, open your browser and go to the Azure portal at [https://portal.azure.com](https://portal.azure.com). Locate the previously created NSG.

**How to do it...**

To assign an NSG to a network interface, follow these steps:

1. In the NSG pane, locate the **Network interfaces** option under **Settings**.
2. Click on the **Associate** button at the top of the page and wait for the new pane to open:

![Network Interfaces](image)

**Figure 3.9: Assigning the NSG to a network interface**

3. Select the NIC you want to associate the NSG with from the list of those available:

![Associate Network Interface](image)

**Figure 3.10: Associating with the network interface**
How it works...

When an NSG is associated with an NIC, the NSG rules will apply only to a single NIC (or a VM associated with the NIC). The NIC can be associated with only one NSG directly, but a subnet associated with an NIC can have an association with another NSG (or even multiple NSGs). This is similar to when we have multiple NSGs assigned to a single subnet, and the **Deny** rule will take higher priority. If one of the NSGs allows traffic on a port, but another NSG is blocking it, traffic will be denied.

In this recipe, you learned how to assign an NSG to a network interface. Let's move on to the next recipe, where you will learn how to assign an NSG using PowerShell.

**Assigning an NSG to a subnet with PowerShell**

Alternatively, we can associate an NSG using Azure PowerShell. In this recipe, we are going to show you how to associate an NSG with a subnet.

**Getting ready**

Open the PowerShell console and make sure you are connected to your Azure subscription.

**How to do it...**

To associate an NSG with a subnet, execute the following command:

```
$vnet = Get-AzVirtualNetwork -Name 'Packt-Script' -ResourceGroupName 'Packt-Networking-Script'
subnet = Get-AzVirtualNetworkSubnetConfig -VirtualNetwork $vnet -Name BackEnd
$nsg = Get-AzNetworkSecurityGroup -ResourceGroupName 'Packt-Networking-Script' -Name 'nsg1'
subnet.NetworkSecurityGroup = $nsg
Set-AzVirtualNetwork -VirtualNetwork $vnet
```

**How it works...**

To assign an NSG using PowerShell, we need to collect information on the virtual network, subnet, and NSG. When all of the information is gathered, we can perform the association using the `Set-AzVirtualNetwork` command and apply the changes.

Let's move on to the next recipe and create an ASG using the Azure portal.
Creating an Application Security Group (ASG)

ASGs are an extension of NSGs, allowing us to create additional rules and take better control of traffic. Using only NSGs allows us to create rules that will allow or deny traffic only for a specific source, IP address, or subnet. ASGs allow us to create better filtering and create additional checks on what traffic is allowed based on ASGs. For example, with NSGs, we can create a rule that subnet A can communicate with subnet B. If we have the application structure for it and an associated ASG, we can add resources in application groups. By adding this element, we can create a rule that will allow communication between subnet A and subnet B, but only if the resources belong to the same application.

Getting ready


How to do it...

To create an ASG using the Azure portal, we must follow these steps:

1. In the Azure portal, select Create a resource and choose Application security group under Networking (or search for application security group in the search bar).

2. The parameters we need to define for deployment are Subscription, Resource group, Name, and Region. An example of the required parameters is shown in Figure 3.11:

Create an application security group

<table>
<thead>
<tr>
<th>Basics</th>
<th>Tags</th>
<th>Review + create</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project details</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subscription *</td>
<td></td>
<td>Microsoft Azure Sponsorship</td>
</tr>
<tr>
<td>Resource group *</td>
<td></td>
<td>Packt-Networking-Portal</td>
</tr>
</tbody>
</table>

Create new

<table>
<thead>
<tr>
<th>Instance details</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Name *</td>
<td></td>
</tr>
<tr>
<td>Region *</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name *</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ASG1</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Region *</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>(Europe) West Europe</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3.11: Creating an ASG using the Azure portal
How it works...

ASGs don't make much difference on their own and must be combined with NSGs to create NSG rules that will allow better control of traffic, applying additional checks before traffic flow is allowed.

Now that we have created an ASG, let's move on to a new recipe where we will associate the ASG with a VM.

**Associating an ASG with a VM**

After creating an ASG, we must associate it with a VM. After this is done, we can create rules with the NSG and ASG for traffic control.

**Getting ready**

Before you start, open your browser and go to the Azure portal at [https://portal.azure.com](https://portal.azure.com). Locate the previously created VM.

**How to do it...**

To associate an ASG with a VM, we must follow these steps:

1. In the VM pane, locate the **Networking** settings.
2. In the **Networking** settings, select the **Application security groups** tab, as shown in Figure 3.12:

![Figure 3.12: Associating an ASG with a VM](image)

---

46 | Network Security Groups
3. In the **Application security groups** settings, select **Configure the application security groups**, as shown in Figure 3.13:

![Figure 3.13: Configuring ASGs](image)

4. In the new pane from the list of available ASGs, select the ASG that you want to associate the VM with:

![Configure the application security groups](image)

5. After clicking **Save**, it takes a few seconds to apply the changes, after which the VM will be associated with the ASG.
How it works...

The VM must be associated with the ASG. We can associate more than one VM with each ASG. The ASG is then used in combination with the NSG to create new NSG rules. In the next recipe, we will create new rules using an NSG and an ASG.

Creating rules with an NSG and an ASG

As a final step, we can use NSGs and ASGs to create new rules with better control. This approach allows us to have better control of traffic, limiting incoming traffic not only to a specific subnet but also only based on whether or not the resource is part of the ASG.

Getting ready

Before you start, open your browser and go to the Azure portal at https://portal.azure.com. Locate the previously created NSG.

How to do it...

To create a rule using both an ASG and an NSG, we must follow these steps:

1. In the NSG pane, find **Inbound security rules**. Select **Add** to add a new rule.

2. For the source, select **Application Security Group**, and then select the ASG you want to use as the source. We also need to provide parameters for **Source**, **Source port ranges**, **Destination**, **Destination port ranges**, **Protocol**, **Action**, **Priority**, **Name**, and **Description**. An example is shown in Figure 3.15:
Creating rules with an NSG and an ASG

How it works...

Using only NSGs to create rules, we can allow or deny traffic only for a specific IP address or range. With an ASG, we can widen or narrow this as needed. For example, we can create a rule to allow VMs from a frontend subnet, but only if these VMs are in a specific ASG. Alternatively, we can allow access to a number of VMs from different virtual networks and subnets, but only if they belong to a specific ASG.