

# Microsoft Azure Network Security

Securing Your Cloud Workloads at the Network Level



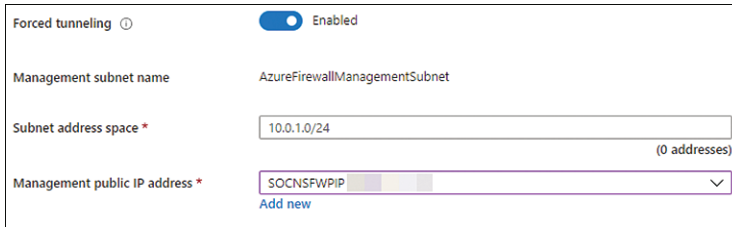
Nicholas DiCola  
Anthony Roman

Foreword by Jonathan Trull, General Manager, Security Solutions and Incident Response Business, Microsoft

# Microsoft Azure Network Security

Nicholas DiCola  
Anthony Roman

Forced tunneling can only be configured when an Azure Firewall is first deployed. If the Forced Tunneling option is set to Enabled during deployment, there are some extra configuration options that need to be set, as shown in Figure 3-9.



Forced tunneling ⓘ	<input checked="" type="checkbox"/> Enabled
Management subnet name	AzureFirewallManagementSubnet
Subnet address space *	10.0.1.0/24 (0 addresses)
Management public IP address *	SOCNSFWPIP <a href="#">Add new</a>

**FIGURE 3-9** Azure Firewall forced tunneling settings

The major change to take into account when deploying Azure Firewall in forced tunneling mode is that another subnet is created and a public IP address is assigned for management use. The reason for these additions is to ensure that Azure can still manage the Firewall even when outbound traffic is routed to another destination.

It is important to note that the actual routing of traffic to the required destination is not done on the Azure Firewall but rather by adding routes to the AzureFirewallSubnet that is created with Firewall. Without the Forced Tunneling option enabled and the management subnet configured, adding routes to AzureFirewallSubnet would not be permitted and would break the operation of the service.

## SNAT Control

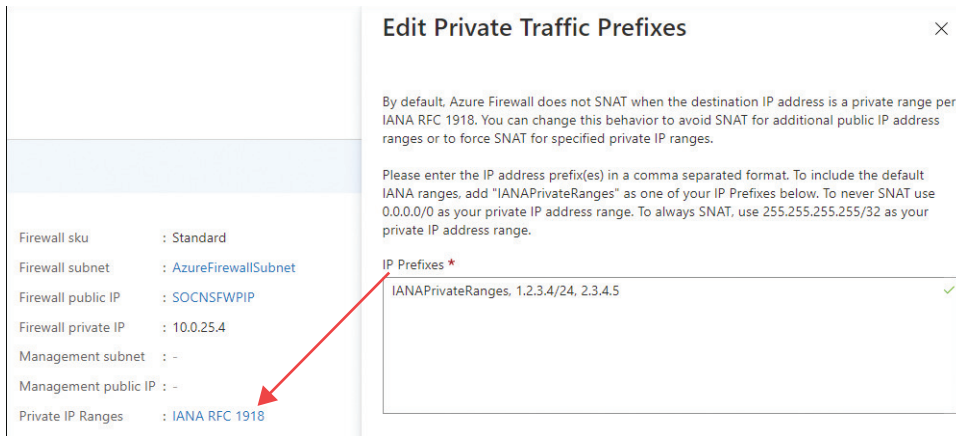
In the default configuration, Azure Firewall performs Source Network Address Translation (SNAT) on all traffic bound for any public destination, meaning that the source of the packet is changed to the public IP of the Firewall. This is standard behavior of edge devices, but there are scenarios where preserving the original source is desirable.

In the case of forced tunneling, if outbound traffic is forwarded to another appliance from Azure Firewall, that traffic is SNAT'd by Firewall, and the original source is unknown to the appliance. In this case, it is best to disable SNAT for the IP of the downstream appliance to preserve the original source.

Another scenario when SNAT would not be needed is if public IP ranges are used inside a private network alongside IANA RFC 1918 addresses. Although this may not be the most common scenario, it can be addressed by configuring Firewall not to SNAT for those destinations.

Configuration of private IP ranges (the destination ranges that Azure Firewall will not SNAT to) can be configured on the **Overview** blade of the Firewall, as shown in Figure 3-10.

Also note that if required, Firewall can be configured to SNAT all traffic by entering 255.255.255.255/32 as the range or never to SNAT by entering 0.0.0.0/0.



**FIGURE 3-10** Azure Firewall SNAT configuration (private IP ranges)

## Traffic inspection

This chapter focuses on using Azure Firewall to control traffic flow for security and connectivity, but it makes sense to briefly mention the inspection capabilities of Azure Firewall before covering them more thoroughly in Chapter 4 (along with inspection capabilities of other services).

Like any other network security appliance or service, Azure Firewall has capabilities to do more than just direct traffic that is forwarded to it. It can also apply various types of inspection to the traffic for security.

- **Threat intelligence** Integrated with Microsoft's internal threat feeds, Azure Firewall can block inbound traffic from known malicious IP addresses or outbound traffic to malicious FQDNs or IP addresses.
- **TLS termination** Azure Firewall can terminate outbound traffic to decrypt and analyze the full packets. This capability enables other inspection types, both those that are currently existing and planned for the future.
- **IDS/IPS** Signature-based intrusion detection and prevention can be applied to traffic for the purpose of blocking malicious traffic. While IDS/IPS can be applied to encrypted or clear-text traffic, decrypting traffic naturally enables more complete inspection.
- **Full URL filtering** The ability to decrypt traffic has the added benefit of being able to see the full URL rather than just the host that is readable from encrypted packets. Using the full URL in rules allows security teams to be more specific in what traffic is allowed and blocked and enriches log data generated by Azure Firewall.

This set of inspection capabilities is shorter than what some NVAs can provide, but this is continually being updated. Azure Firewall has only been generally available since 2018 and will continue to grow in capability and maturity as time passes.

# Rule types

When Azure Firewall is first deployed into the hub VNet and traffic starts to be routed to it, the default behavior is to block all traffic. It is possible to override this behavior by creating a rule that allows all traffic, but this is not advisable from a security perspective in nearly all cases. Rather, a positive security model should be followed to allow traffic only when the function of the systems involved directly requires it.

There are three different rule types that can be configured to allow traffic through Azure Firewall, including Network Rules, Application Rules, and DNAT Rules. These rules are organized into groups and can be created and managed either directly on individual Firewall instances via the Rules blade or using Firewall Policies with Firewall Manager.

## Network rules

Network rules are used to control traffic within the private network space or east-west traffic. To understand the components and options available, create a new rule collection on a Firewall from the portal as shown in Figure 3-11.

Add network rule collection

Name \*

MyRule

Priority \*

allowed numeric values between 100-65000

Action \*

Allow

Rules

IP Addresses

name	Protocol	Source type	Source	Destination type	Destination Addr...	Destination Ports
	0 selected	IP address	*, 192.168.10.1, 192...	IP address	*, 192.168.10.1, 192...	8080, 8080-8090, *

Service Tags

name	Protocol	Source type	Source	Service Tags	Destination Ports
	0 selected	IP address	*, 192.168.10.1, 192.168...	0 selected	8080, 8080-8090, *

FQDNs (preview)

name	Protocol	Source type	Source	Destination FQDNs	Destination Ports
	0 selected	IP address	*, 192.168.10.1, 192.168...	time.windows.com	8080, 8080-8090, *

FIGURE 3-11 Azure Firewall network rules

Network rule collections have the following general options:

- **Name** This is a unique name for the collection, which can be up to 80 characters long and can be used to include relevant information used for rule auditing over time.
- **Priority** Number from 100 to 65,000 that specifies in which order the collection is processed. Lower numbers are processed first. It is advised that you space out rule collection priority so new collections can be inserted between existing ones as needed.
- **Action** This is the result when traffic matches a rule: Allow or Deny.

A network rule collection can have multiple individual rules associated, which inherit settings from the collection, such as Priority and Action. The rules are grouped by the destination type, which can be IP Address, Service Tags, or FQDNs. The options for individual rules include:

- **Name** This is a unique name within the rule.
- **Protocol** The protocol can be set to TCP, UDP, ICMP, or Any. Multiselect is also allowed.
- **Source Type** The source type can be either IP Address or IP Group. IP Groups are extremely useful in this context because they can be managed separately from the rules that use them, enabling IP Groups to be dynamic without having to constantly change Firewall configuration.
- **Source** Sources can be a comma-separated list of IP addresses and CIDR blocks, \*, or a multiselect list of IP Groups, if that is the type.
- **Destination Type** This is only available for IP address rules and can be either IP Address or IP Group.
- **Destination Address** Destinations can be a comma-separated list of IP addresses and CIDR blocks, \*, or a multiselect list of IP Groups, if that is the type.
- **Service Tags** This is a multiselect list of available Service Tags, which correspond to Microsoft-managed lists of IP addresses and ranges associated with common services.
- **Destination FQDNs** This is a comma-separated list of FQDNs and requires DNS Proxy to be enabled to ensure that resolution is the same for clients and Azure Firewall.
- **Destination Ports** Ports can be specified in a comma-separated list of individual ports, ranges, or \*.

## Application rules

Application rules control outbound traffic from the network to the internet. In server environments, internet access should be restricted to only what is essential, and Application Rules can be used to manage a list of allowed destinations. Create an application rule collection by selecting **Add Application Rule** from the **Application Rule tab** of the **Rules** blade as shown in Figure 3-12.

The options available for the collection are

- **Name** This is a unique name for the collection—up to 80 characters long—which can be used to include relevant information used for rule auditing over time.
- **Priority** Number from 100 to 65,000 that specifies in which order the collection is processed. Lower numbers are processed first. It is advised that you space out rule collection priority so new collections can be inserted between existing ones as needed.
- **Action** This is the result when traffic matches a rule: Allow or Deny.

**FIGURE 3-12** Azure Firewall DNS settings

Individual rules are grouped by the destination type, either FQDN tags or target FQDNs, and have the following options:

- **Name** This is a unique name within the rule.
- **Source Type** The source type can be either IP Address or IP Group. IP Groups are extremely useful in this context because they can be managed separately from the rules that use them, enabling IP Groups to be dynamic without having to constantly change Firewall configuration.
- **Source** Sources can be a comma-separated list of IP addresses and CIDR blocks, \*, or a multiselect list of IP Groups, if that is the type.
- **Protocol:Port** Protocol and optional port can be specified here in a comma-separated list of protocols (HTTP, HTTPS, or MSSQL). If a nonstandard port is used, it can be specified after the protocol—for example, HTTP:8080.
- **FQDN Tags** Supported FQDN Tags include MicrosoftActiveProtectionService, WindowsDiagnostics, WindowsUpdate, AppServiceEnvironment, AzureBackup, AzureKubernetesService, HdInsight, and WindowsVirtualDesktop.
- **Target FQDNs** These are represented in a comma-separated list, which can include wildcards such as \*.microsoft.com.

## DNAT rules

Destination network address translation (DNAT) rules can be used to allow traffic into the network from the internet. DNAT rules forward specified external traffic to internal destinations, which is useful for centrally managing access to internal resources without having to assign public IP addresses to each service. This is commonly used for allowing management access

to servers or to make an internal web application available outside the network. You create a DNAT rule collection in the same way that you create application or network rule collections. This is shown in Figure 3-13.

**Add NAT rule collection**

Name \* MyinboundRule ✓

Priority \* allowed numeric values between 100-65000

Action Destination Network Address Translation (DNAT) ✓

Rules

name	Protocol	Source type	Source	Destination Addr...	Destination Ports	Translated address	Translated port
	0 selected	IP address	*, 192.168.10.1, 192...	192.168.10.0	8080	192.168.10.0	8080

**FIGURE 3-13** Azure Firewall DNAT Rule Collection

DNAT rule collections have the following options:

- **Name** This is a unique name for the collection—up to 80 characters long—which can be used to include relevant information used for rule auditing over time.
- **Priority** Number from 100 to 65,000 that specifies which order the collection is processed. Lower numbers are processed first. It is advised that you space out rule collection priority so new collections can be inserted between existing ones as needed.

Individual DNAT rules have the following properties:

- **Name** This is a unique name within the rule.
- **Protocol** The protocol can be set to TCP or UDP. Multiselect is also allowed.
- **Source Type** The source type can be either IP Address or IP Group. IP Groups are extremely useful in this context because they can be managed separately from the rules that use them, enabling IP Groups to be dynamic without having to constantly change Firewall configuration.
- **Source** Sources can be a comma-separated list of IP addresses and CIDR blocks, \*, or a multiselect list of IP Groups, if that is the type.
- **Destination Address** This is a single Azure Firewall associated public IP address that will listen for incoming requests.
- **Destination Port** A single port to open on the public IP address.
- **Translated Address** This will be the internal IP address of the resource on the network that will be accessible from the internet.
- **Translated Port** This will be the destination port on the internal address, which can be different from the public port that is open.

DNAT rules serve a valid purpose, but they should be used with caution. Even though Azure Firewall stands between potential attackers and the resources advertised by the rules, those resources are still vulnerable to attack from the allowed destinations. Whenever possible, restrict the allowed sources to trusted IP ranges and always use strong access control on the destination service.