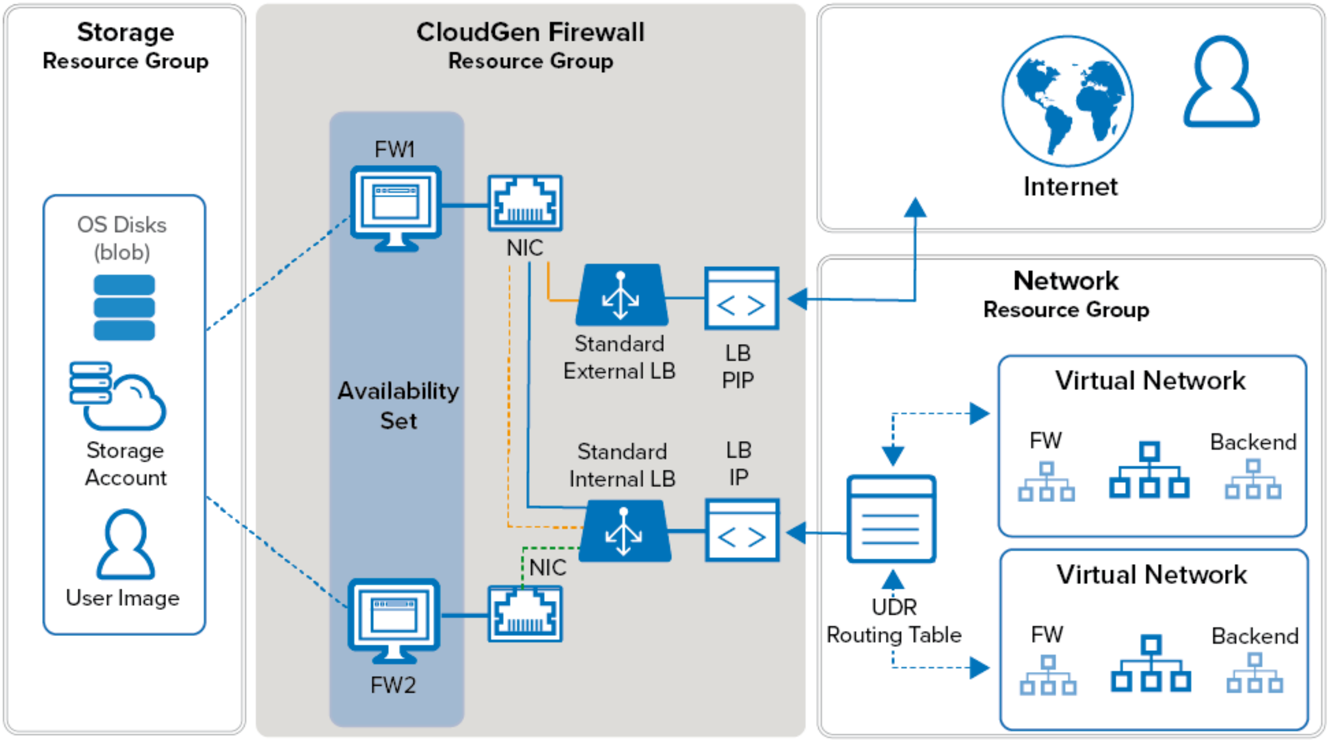
How to Configure a High Availability Cluster in Azure with the Standard Load Balancer

https://campus.barracuda.com/doc/96026373/

Configure a high availability cluster to ensure that the services running on the Barracuda CloudGen Firewall VMs are always available even if one unit is unavailable due to maintenance or a hardware issue. To be able to configure an HA cluster, both firewalls VMs must be deployed to the same subnet and be placed in either an Availability Set or Availability Zone (where available). This ensures that the VMs are placed in different fault and update domains inside the Azure data center. Incoming connections are forwarded to the active firewall by the Azure Load Balancer. The load balancer actively monitors the services on the firewall and, when an HA failover takes place, redirects the traffic to the other, now-active firewall. You must create load balancer rules and health probes for each service for the load balancer to know which ports to forward and how to monitor them. The load balancer does not fail over immediately after the service has failed over, since it requires at least two probes to fail before reacting. Combined with the minimum poll time of 5 seconds, this means that failover will take at least 10 seconds during which no traffic can be forwarded.

The standard load balancer allows stateful sessions to remain as there are no IP address changes with this method. The backend VMs are configured to use the firewall as the default gateway and, if needed, access control between the backend subnets using Azure user-defined routing. Because only one IP address can be configured as the destination, a Standard type internal load balancer IP address is used, and this load balancer directs traffic to the active firewall. Now, the backend VMs can connect via the active firewall to the Internet.



Step 1. Deploy Two CloudGen Firewall VMs

To configure an HA cluster, deploy two CloudGen VMs. The public IPs attached to the NICs are removed after configuring client-to-site VPN access via the load balancer. To be able to use them in an HA cluster, the deployment must meet the following requirements:

- Static private (internal) IP addresses must be used.
- The SKU of the public IP of each firewall must match the SKU of the load balancer. In this case, the public IP must have a standard SKU since a standard load balancer will be used.
- The same instance size for both VMs must be used.
- Both firewalls must be the same Barracuda CloudGen Firewall for Azure model.
- Both VMs must be deployed in one Availability Set or across Availability Zones.

For more information, see How to Deploy a CloudGen Firewall from the Microsoft Azure Marketplace or How to Deploy a CloudGen Firewall in Microsoft Azure Using PowerShell and ARM.

Official templates are available to assist you to deploy quicker. These can be found in our GitHub: https://github.com/barracudanetworks/ngf-azure-templates . If you are using the GitHub template, provisioning may take a while. Until it completes, you will get the error message "access denied" if you try to connect via Barracuda Firewall Admin. If boot diagnostics are enabled, you can view the log. Further deployment examples can be found in the contrib folder.

To test, you can deploy a stand-alone proof of concept environment by following this guide: https://app.barracuda.com/resource/ref_architectures/azure_high_availability_cluster

Step 2. Change the Firewall Network Configuration to Use the Static Private IP Addresses

On both firewall VMs, change the network configuration to use a static network interface. Use the static private IP address you assigned to the NIC during deployment.

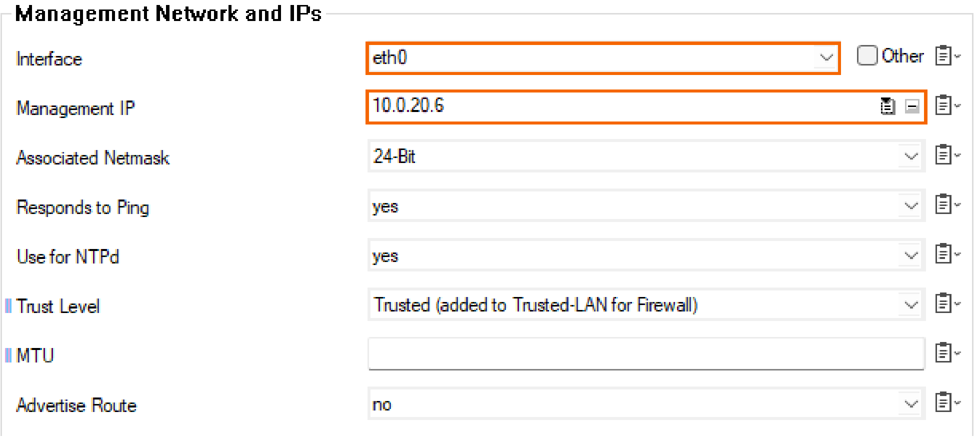
Step 2.1 Reconfigure the Network Interface

Change the network interface type from dynamic to static.

> You can skip this when deploying clustered templates as these pre-complete these steps.

1. Go to **CONFIGURATION > Configuration Tree > Box > Network**.

2. In the left menu, click **xDSL/DHCP**.
3. Click **Lock**.
4. If present, delete the **DHCP01** entry in the **DHCPv4 Links** list.
5. Expand the **DHCPv4 Enabled** drop-down list and select **No**.
6. Click **Send Changes**.
7. In the left menu, click **IP Configuration**.
8. In the **Management Network and IPs** section, select **eth0** from the **Interface** list.
9. Enter the static internal IP address from Step 1 as the **Management IP**. E.g., `10.0.20.6`



Management Network and IPs

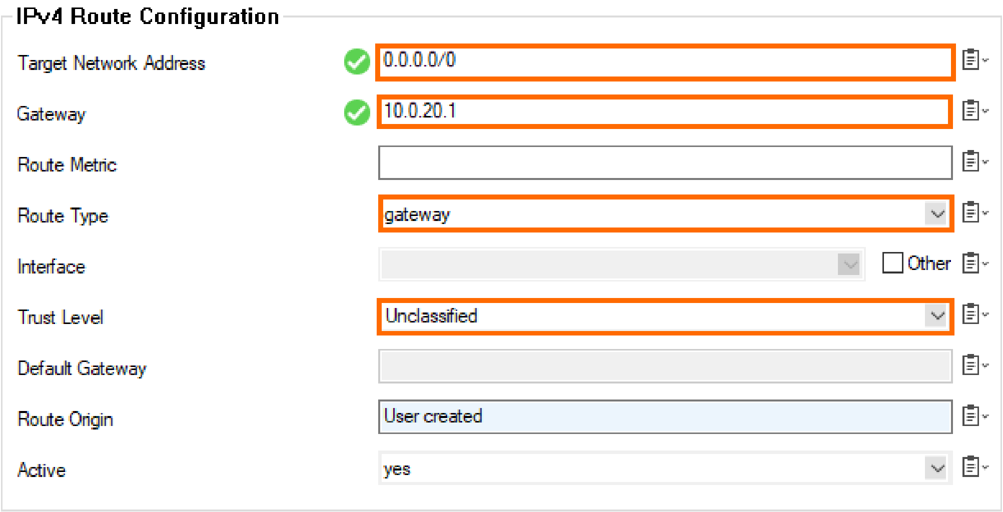
Interface	eth0 ☐ Other
Management IP	10.0.20.6
Associated Netmask	24-Bit
Responds to Ping	yes
Use for NTPd	yes
Trust Level	Trusted (added to Trusted-LAN for Firewall)
MTU	
Advertise Route	no

Step 2.2 Create the Default Route

Add the default route. The default gateway in Azure subnets is always the first IP in the subnet. E.g., 10.0.20.1 if the subnet is 10.0.20.0/24.

You can skip this when deploying clustered templates as these pre-complete these steps.

1. In the left menu, click **Advanced Routing**.
2. Click **+** in the **IPv4 Routing Table** and configure the following settings:
   - **Name** – Enter a descriptive name for the route and click **OK**.
   - **Target Network Address** – Enter `0.0.0.0/0`
   - **Route Type** – Select **gateway**.
   - **Gateway** – Enter the first IP address of the subnet the firewalls reside in. E.g., `10.0.20.1` if the IP addresses of the firewalls are 10.0.20.6 and 10.0.20.7.
   - **Trust Level** – Select **Unclassified**.



IPv4 Route Configuration

Target Network Address: 0.0.0.0/0

Gateway: 10.0.20.1

Route Metric:

Route Type: gateway

Interface: ☐ Other

Trust Level: Unclassified

Default Gateway:

Route Origin: User created

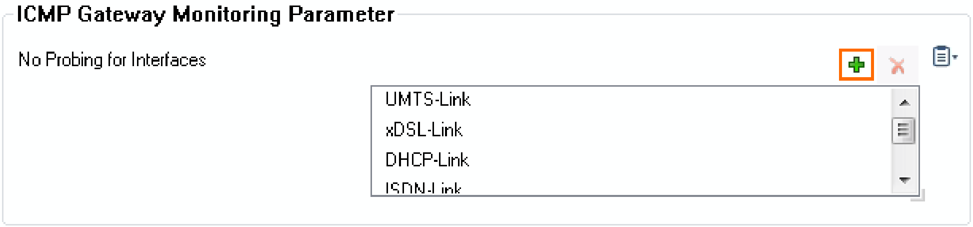
Active: yes

3. Click **OK**.
4. Click **Send Changes** and **Activate**.

Step 2.3 Disable ICMP Monitoring of the Gateway

ICMP probing must be disabled for the interface.

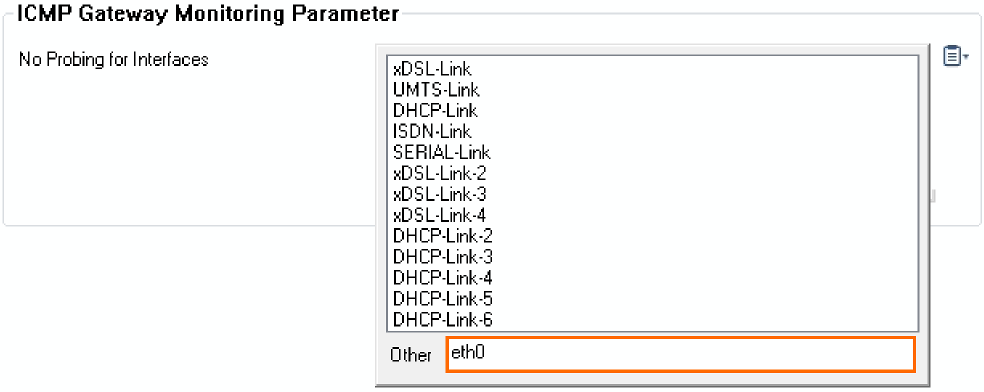
1. Go to **CONFIGURATION** > **Configuration Tree** > **Infrastructure Services** > **Control**.
2. Click **Lock**.
3. In the **ICMP Gateway Monitoring Parameter** section, click **+** to add an entry to the **No Probing for Interface** table.



ICMP Gateway Monitoring Parameter

No Probing for Interfaces

4. In the **Other** field, enter `eth0`.



ICMP Gateway Monitoring Parameter

No Probing for Interfaces

Other: eth0

5. Click **Send Changes** and **Activate**.

Step 2.4 Activate the Network Changes

Activate the changes to the network configuration.

1. Go to **CONTROL > Box**.
2. In the **Network** section of the left menu, click **Activate new network configuration**.
3. Click **Failsafe**.

Open the **CONTROL > Network** page. Your interface and IP address are now static.

Step 3. (PAYG only) Import PAYG Licenses from the Secondary Firewall

Step 3.1 Export the PAYG license from the Secondary Firewall

1. Log into the secondary firewall.
2. Go to **CONFIGURATION > Configuration Tree > Box > Box Licenses**.
3. Click **Lock**.
4. Select the license file, click **Export**, and select **Export to File**.
5. Click **Unlock**.

Step 3.2 Import the PAYG License on the Primary Firewall

1. Log into the primary firewall.
2. Go to **CONFIGURATION > Configuration Tree > Box > Box Licenses**.
3. Click **Lock**.
4. Click **+** and select **Import from Files**.
5. Select the license file exported from the secondary firewall.
6. Click **OK**.
7. Select **I agree** to accept the terms and conditions.
8. Click **OK**.
9. Click **Send Changes** and **Activate**.

The primary firewall now has both PAYG licenses listed in the **Licenses** list.

Step 4. Configure an HA Cluster on the CloudGen Firewall VMs

Configure the two firewalls to synchronize session and configuration information. Use **Application Redirect** access rules to redirect incoming traffic from the eth0 interface to the services. Use the internal IP address of the primary and secondary firewall as the destination of the rule to ensure that it matches without regard to which firewall VM the service is currently running on.

For more information, see How to Set Up a High Availability Cluster.

Step 5. (BYOL only) Activate and License the two Firewall VMs

Activate the license on the secondary firewall, then on the primary firewall. If the primary unit is activated prior to the secondary unit, the licenses for the secondary cannot be downloaded. In this case, reboot the primary firewall, perform a complete manual HA sync, and update to download and install the licenses correctly.

For more information, see How to Activate and License a Standalone High Availability Cluster.

Step 6. Disassociate the Public IP Addresses

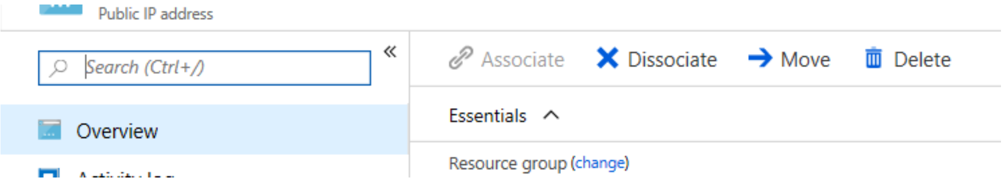
When both a load balancer and a public IP are available for the firewall VM, the public IP is used as the default source IP address for the VM. This means that outgoing connections use different source IP addresses depending on which firewall is active.

If the portal deployed the box using a basic public IP on the network interface, you must remove it for the external standard load balancer to work correctly. You can replace it with standard SKU public IPs directly on the NIC.

Using the Azure Web Portal

For each firewall VM, remove the public IP address from the network interface.

1. Go to https://portal.azure.com.
2. Locate the **Network Interface** attached to your primary firewall VM.
3. Click **Public IP Address**. The **Public IP address** column opens.
4. Click **Disassociate**.



5. Repeat for the secondary firewall VM.

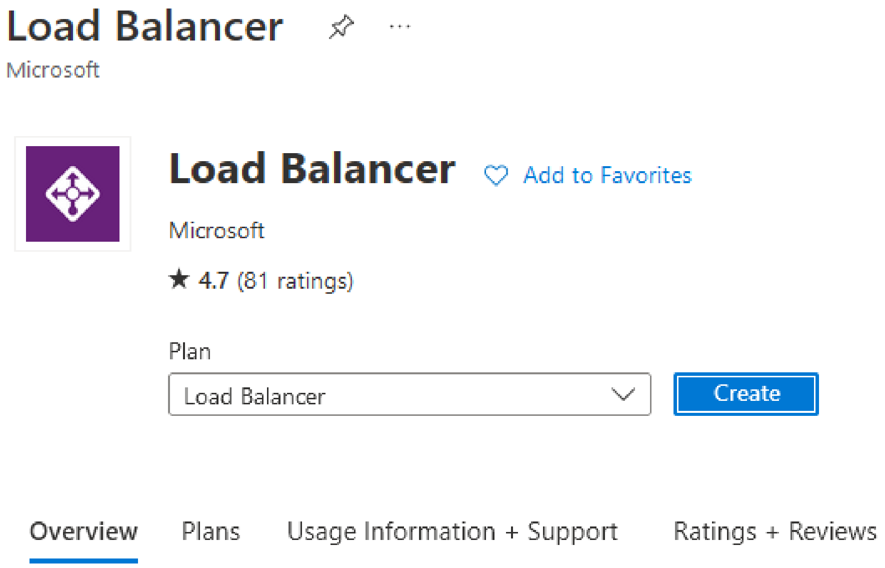
Step 7. Create the External Load Balancer

After you have deployed the two firewalls, you can create the load balancers that will direct traffic as required. You will need to create a new external and internal load balancer to handle all potential traffic flows.

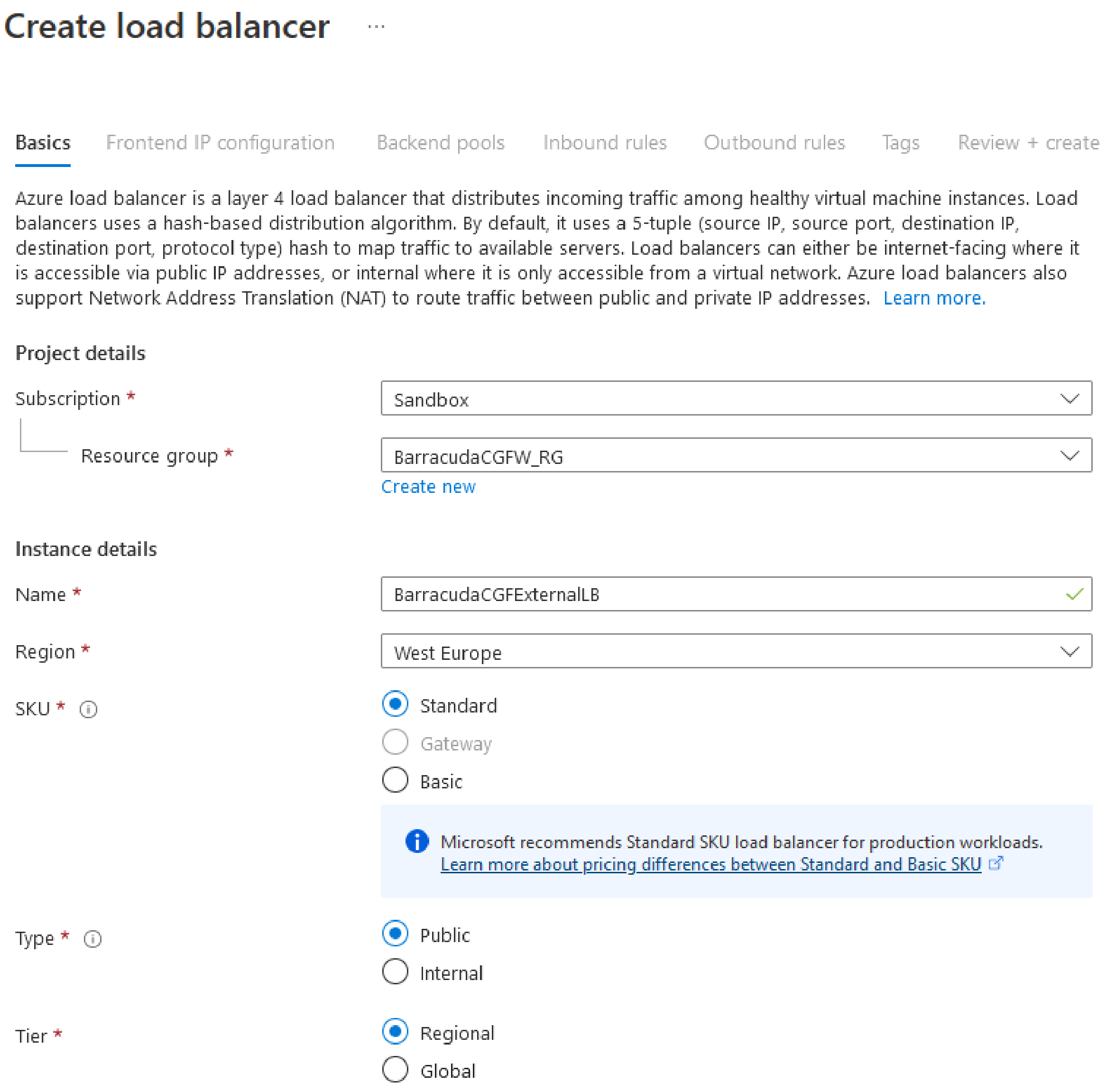
Step 7.1 Configure the Basic Settings

From the Azure portal:

1. Under **Azure services**, click the **+** symbol to **Create a resource**.
2. Type in `Load Balancer` and select the resource from the list. The **Load Balancer** page opens.



3. Click **Create**.
4. On the next page, configure the following settings:
   - **Name** – Enter the name of the External Load Balancer.
   - **Region** – Select your region.
   - **SKU** – Select the **Standard** SKU.
   - **Type** – Select the **Public** type.
   - **Tier** – Select your tier.



Create load balancer ...

Azure load balancer is a layer 4 load balancer that distributes incoming traffic among healthy virtual machine instances. Load balancers uses a hash-based distribution algorithm. By default, it uses a 5-tuple (source IP, source port, destination IP, destination port, protocol type) hash to map traffic to available servers. Load balancers can either be internet-facing where it is accessible via public IP addresses, or internal where it is only accessible from a virtual network. Azure load balancers also support Network Address Translation (NAT) to route traffic between public and private IP addresses. Learn more.

Project details

Subscription * Sandbox

Resource group * BarracudaCGFW_RG

Create new

Instance details

Name * BarracudaCGFExternalLB ✓

Region * West Europe

SKU * ⓘ
- ● Standard
- ○ Gateway
- ○ Basic

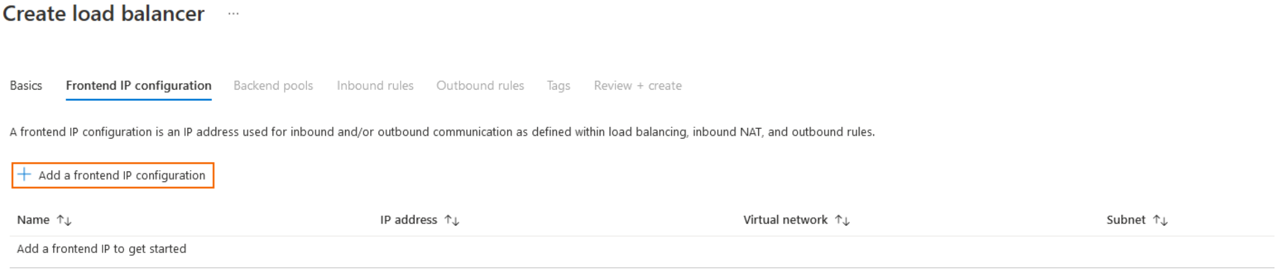
Microsoft recommends Standard SKU load balancer for production workloads.
Learn more about pricing differences between Standard and Basic SKU

Type * ⓘ
- ● Public
- ○ Internal

Tier *
- ● Regional
- ○ Global

Step 7.2 Add a Frontend IP Configuration

1. Click **Next: Frontend IP configuration**.
2. Click **+** to **Add a frontend IP configuration**.

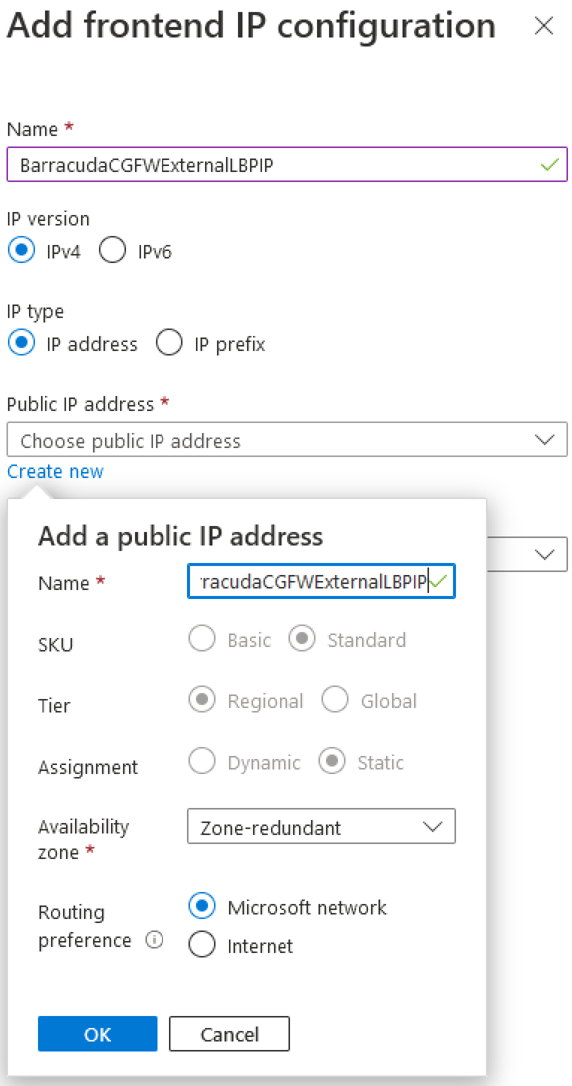


Create load balancer ...

A frontend IP configuration is an IP address used for inbound and/or outbound communication as defined within load balancing, inbound NAT, and outbound rules.

Name ↑↓	IP address ↑↓	Virtual network ↑↓	Subnet ↑↓
Add a frontend IP to get started			

3. On the next page, enter the **Name** of the frontend IP.
4. Select an existing **Public IP address** or create a new public IP address.



Add frontend IP configuration

Name *: BarracudaCGFWExternalLBPIP

IP version: ● IPv4 ○ IPv6

IP type: ● IP address ○ IP prefix

Public IP address *: Choose public IP address

Create new

Add a public IP address

Name *: racudaCGFWExternalLBPIP

SKU: ○ Basic ● Standard

Tier: ● Regional ○ Global

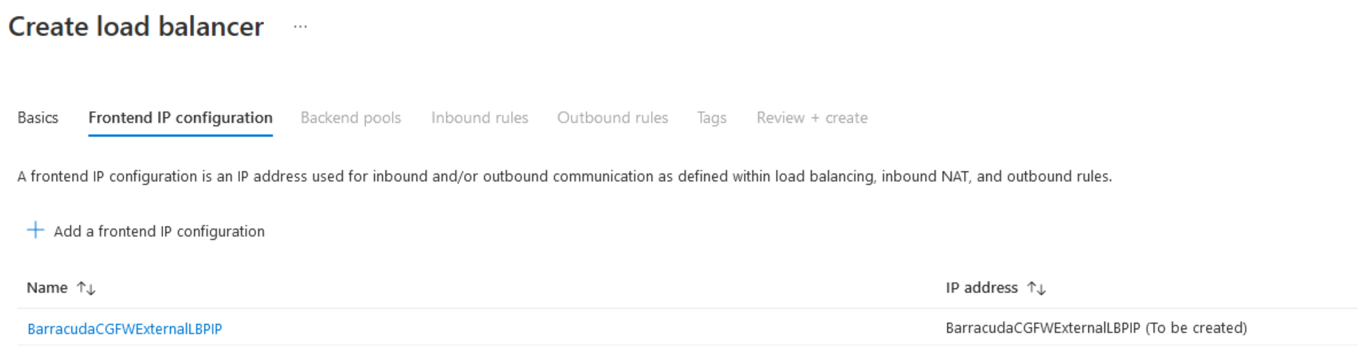
Assignment: ○ Dynamic ● Static

Availability zone *: Zone-redundant

Routing preference: ● Microsoft network ○ Internet

5. Click **Add**.

The entry is now displayed in the list.



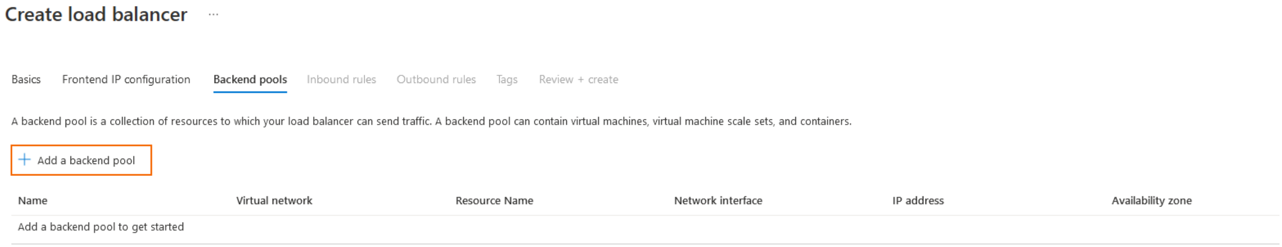
Create load balancer

A frontend IP configuration is an IP address used for inbound and/or outbound communication as defined within load balancing, inbound NAT, and outbound rules.

Name	IP address
BarracudaCGFWExternalLBPIP	BarracudaCGFWExternalLBPIP (To be created)

Step 7.3 Add a Backend Pool

1. Click **Next: Backend pools**.
2. Click **+** to **Add a backend pool**.

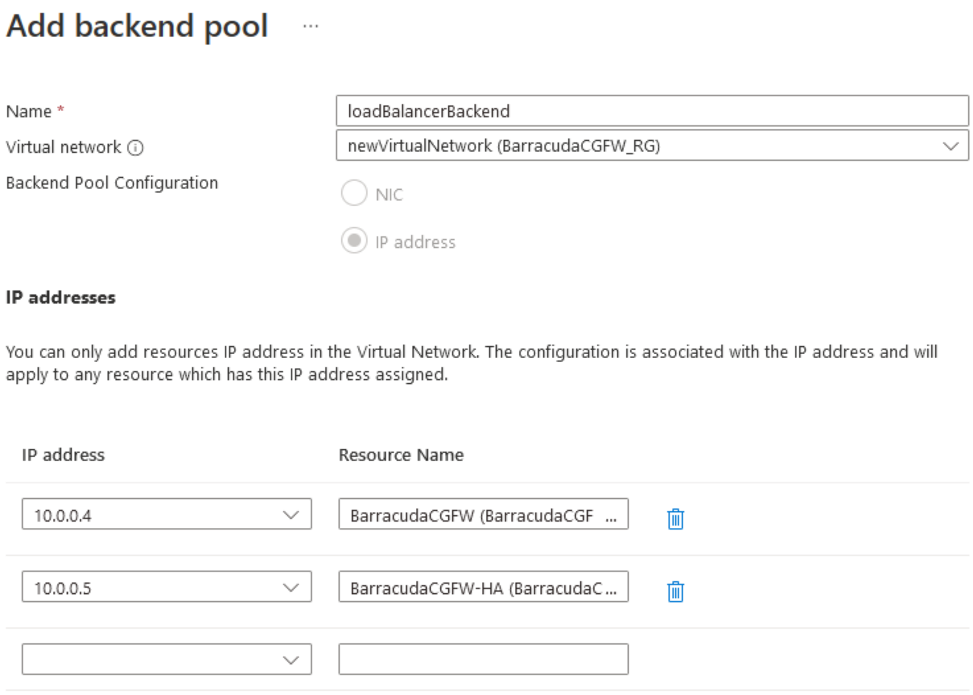


Create load balancer ···

A backend pool is a collection of resources to which your load balancer can send traffic. A backend pool can contain virtual machines, virtual machine scale sets, and containers.

Name	Virtual network	Resource Name	Network interface	IP address	Availability zone
Add a backend pool to get started					

3. Complete the fields as below:
   - **Name** – Enter your desired name for the backend pool.
   - **Virtual Network** – Select the virtual network you built your firewalls in.
   - **Backend Pool Configuration** – Select **IP address**.
   - **IP address** – Select the CGF IP addresses where you wish the LB to send traffic to.



Add backend pool ···

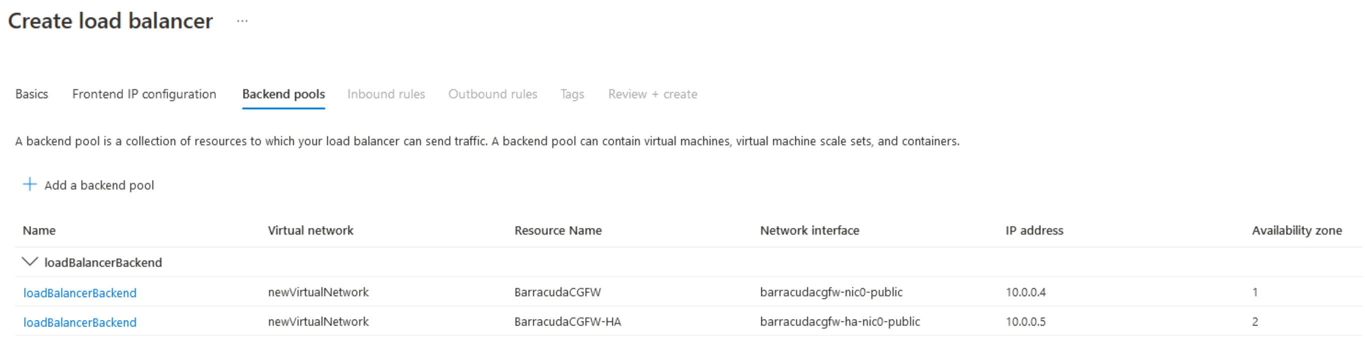
Name * loadBalancerBackend
Virtual network ⓘ newVirtualNetwork (BarracudaCGFW_RG)
Backend Pool Configuration ○ NIC ● IP address

IP addresses

You can only add resources IP address in the Virtual Network. The configuration is associated with the IP address and will apply to any resource which has this IP address assigned.

IP address	Resource Name
10.0.0.4	BarracudaCGFW (BarracudaCGF ...
10.0.0.5	BarracudaCGFW-HA (BarracudaC ...

4. Repeat for the second firewall VM you built.
5. Click **Add**.



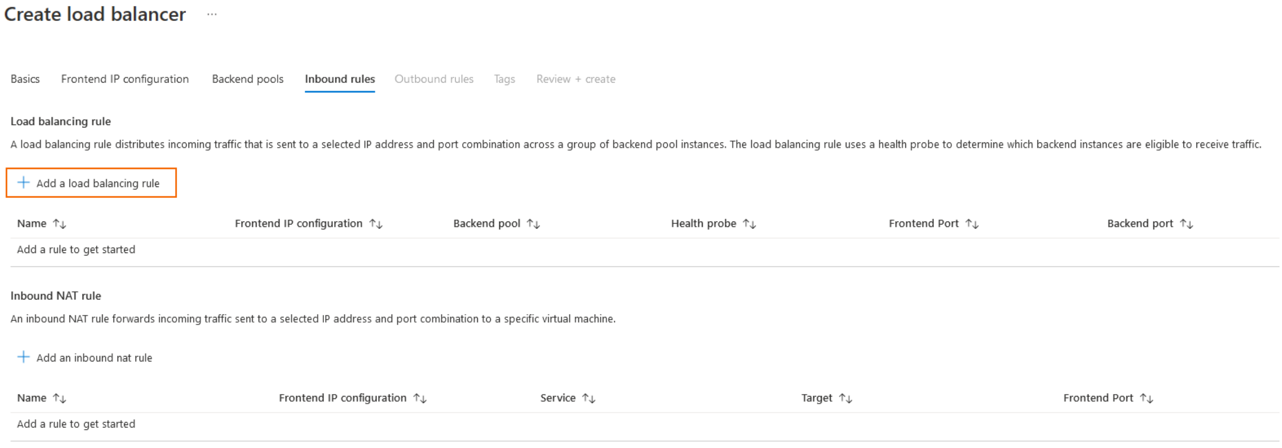
A backend pool is a collection of resources to which your load balancer can send traffic. A backend pool can contain virtual machines, virtual machine scale sets, and containers.

Name	Virtual network	Resource Name	Network interface	IP address	Availability zone
loadBalancerBackend					
loadBalancerBackend	newVirtualNetwork	BarracudaCGFW	barracudacgfw-nic0-public	10.0.0.4	1
loadBalancerBackend	newVirtualNetwork	BarracudaCGFW-HA	barracudacgfw-ha-nic0-public	10.0.0.5	2

Step 7.4 Add an Inbound Rule

To create the load balancing rules for incoming traffic, you must create one for TCP and one for UDP so that the firewalls can forward traffic over these ports from the Internet. This instruction creates rules for inbound client VPNs that meet this requirement.

1. Click **Next: Inbound rules**.
2. Click **+** to **Add a load balancing rule** to create a new rule.



Load balancing rule

A load balancing rule distributes incoming traffic that is sent to a selected IP address and port combination across a group of backend pool instances. The load balancing rule uses a health probe to determine which backend instances are eligible to receive traffic.

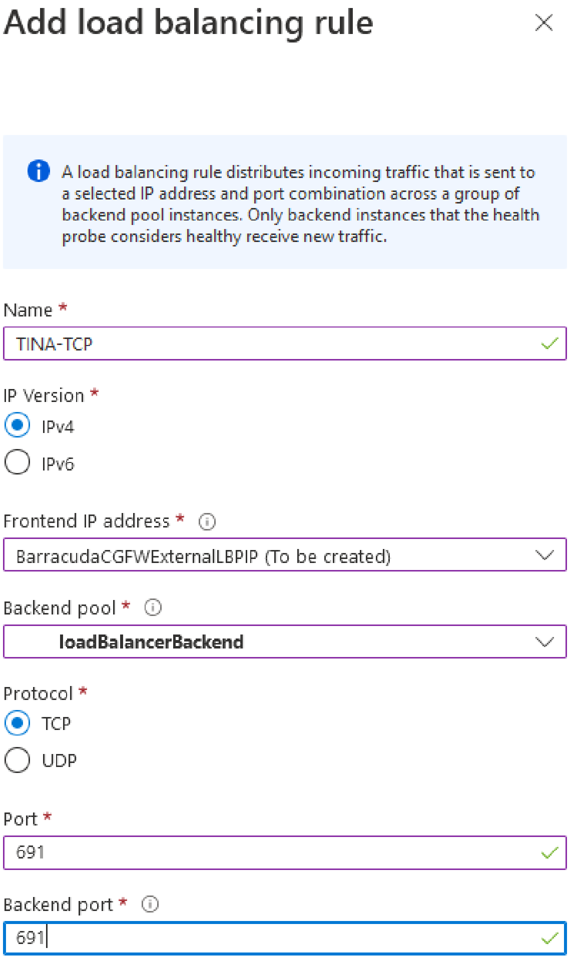
Name ↑↓	Frontend IP configuration ↑↓	Backend pool ↑↓	Health probe ↑↓	Frontend Port ↑↓	Backend port ↑↓
Add a rule to get started					

Inbound NAT rule

An inbound NAT rule forwards incoming traffic sent to a selected IP address and port combination to a specific virtual machine.

Name ↑↓	Frontend IP configuration ↑↓	Service ↑↓	Target ↑↓	Frontend Port ↑↓
Add a rule to get started				

3. Configure the settings as below:
    - **Name** – Suggested name: `TINA-TCP`
    - **IP Version** – **IPv4**
    - **Frontend IP address** – Select the front-end IP created at build time.
    - **Backend pool** – Select the backend pool just created.
    - **Protocol** – **TCP**
    - **Port** – 691
    - **Backend port** – 691



Add load balancing rule

A load balancing rule distributes incoming traffic that is sent to a selected IP address and port combination across a group of backend pool instances. Only backend instances that the health probe considers healthy receive new traffic.

Name * TINA-TCP

IP Version *
● IPv4
○ IPv6

Frontend IP address * BarracudaCGFWExternalLBPIP (To be created)

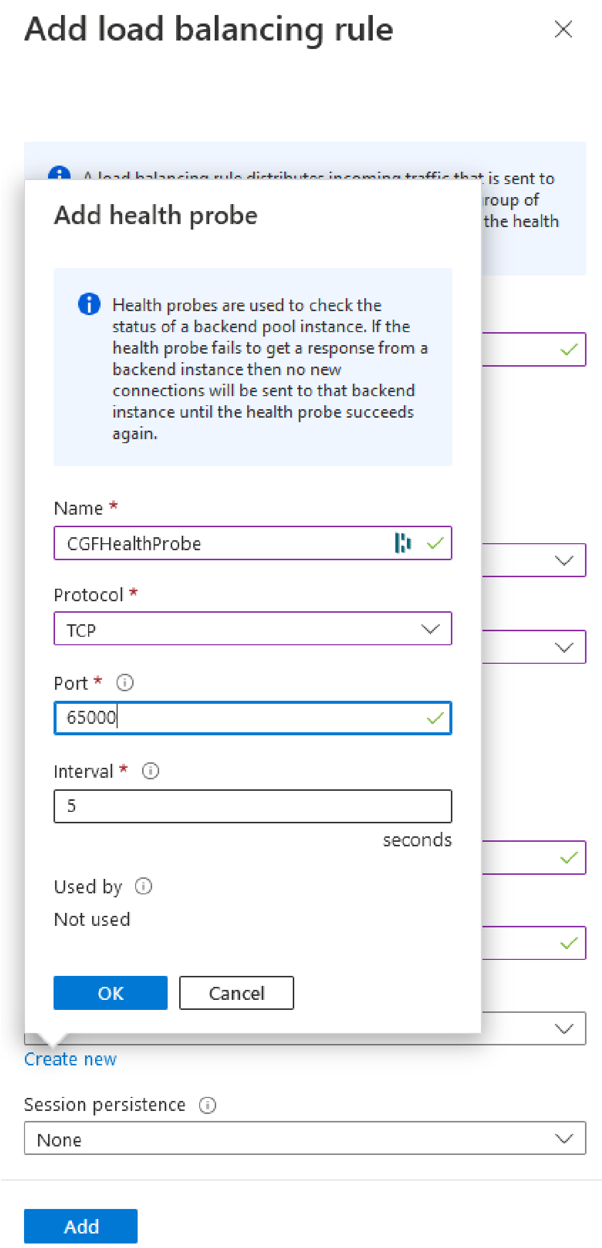
Backend pool * loadBalancerBackend

Protocol *
● TCP
○ UDP

Port * 691

Backend port * 691

- **Health probe** – Select a probe created previously, or click **Create new** to create a new one:
    - **Name** – Suggested name: `CGFHealthProbe`
    - **Protocol** – **TCP**
    - **Port** – `65000`
    - **Interval** – Leave as default.



Add health probe

Health probes are used to check the status of a backend pool instance. If the health probe fails to get a response from a backend instance then no new connections will be sent to that backend instance until the health probe succeeds again.

Name * CGFHealthProbe

Protocol * TCP

Port * 65000

Interval * 5 seconds

Used by: Not used

Session persistence: None

- Click **OK** to create the health probe.

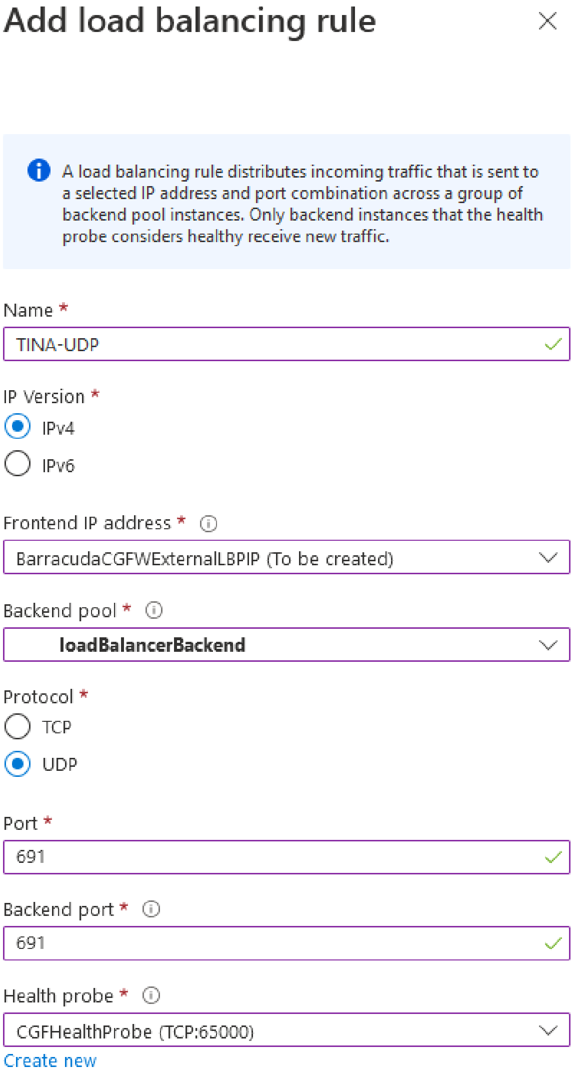
- **Session persistence** – Leave as default (**None**).
- **Idle timeout (minutes)** – Leave as default (4).
- **TCP reset** – Leave as default (Disabled).
- **Floating IP** – Leave as default (Disabled).
- **Outbound source network address translation (SNAT)** – Leave as default (**(Recommended) Use outbound rules to provide backend pool members access to the Internet.**).

4. Click **Add** to create the load balancing rule.

Repeat the steps above to create a second rule for UDP, but change the following settings:

- **Name** – Suggested name: `TINA-UDP`

- **Protocol** – **UDP**
- **Port** – 691
- **Backend port** – 691



Add load balancing rule

A load balancing rule distributes incoming traffic that is sent to a selected IP address and port combination across a group of backend pool instances. Only backend instances that the health probe considers healthy receive new traffic.

Name * TINA-UDP

IP Version *
● IPv4
○ IPv6

Frontend IP address * BarracudaCGFWExternalLBPIP (To be created)

Backend pool * loadBalancerBackend

Protocol *
○ TCP
● UDP

Port * 691

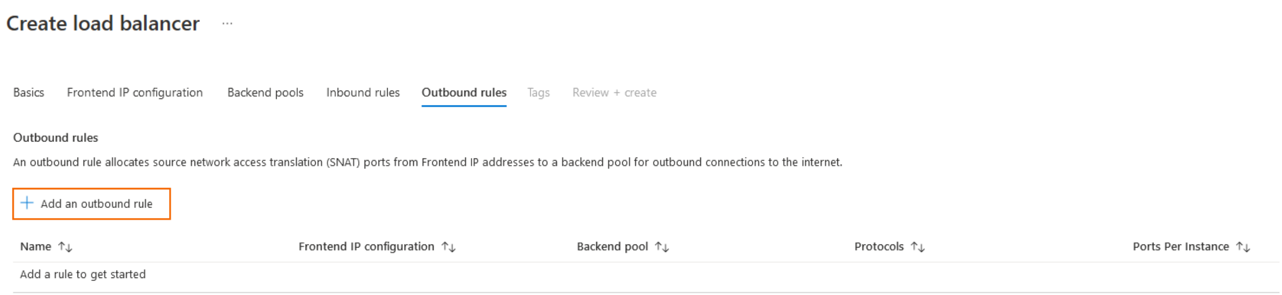
Backend port * 691

Health probe * CGFHealthProbe (TCP:65000)

Step 7.5 Add an Outbound Rule

To create the load balancing rules for outbound traffic, you must have for minimum one rule in order for the firewalls to pass traffic out to the Internet. This instruction creates an **Any** rule for outbound traffic that meet this requirement.

1. Click **Next: Outbound rules**.
2. Click **+** to **Add an outbound rule**.

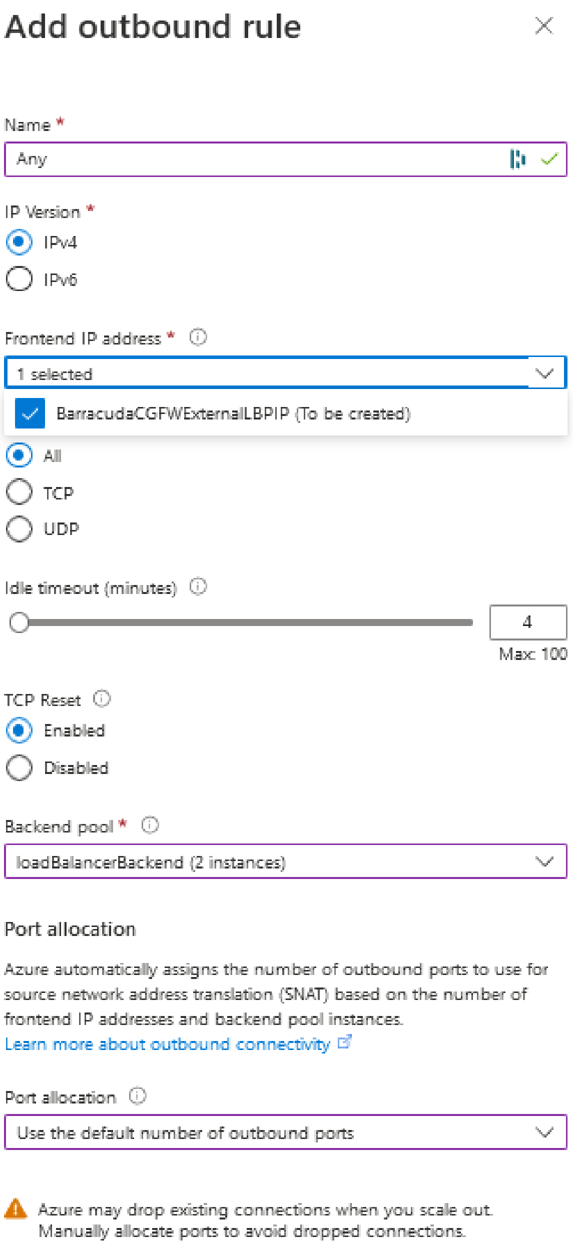


Outbound rules

An outbound rule allocates source network access translation (SNAT) ports from Frontend IP addresses to a backend pool for outbound connections to the internet.

Name ↑↓	Frontend IP configuration ↑↓	Backend pool ↑↓	Protocols ↑↓	Ports Per Instance ↑↓
Add a rule to get started				

3. Configure the settings as below:
    - **Name** – Suggested name: Any
    - **IP Version – IPv4**
    - **Frontend IP address** – Select the front-end IP created at build time.
    - **Protocol – All**
    - **Idle timeout (minutes)** – Leave as default (4).
    - **TCP reset** – Leave as default (Enabled).
    - **Backend pool** – Select the backend pool just created.
    - **Port allocation** – Use the default number of outbound ports.



Add outbound rule

Name * Any

IP Version *
● IPv4
○ IPv6

Frontend IP address * 1 selected
☑ BarracudaCGFWExternalLBPIP (To be created)

● All
○ TCP
○ UDP

Idle timeout (minutes) 4
Max: 100

TCP Reset
● Enabled
○ Disabled

Backend pool * loadBalancerBackend (2 instances)

Port allocation

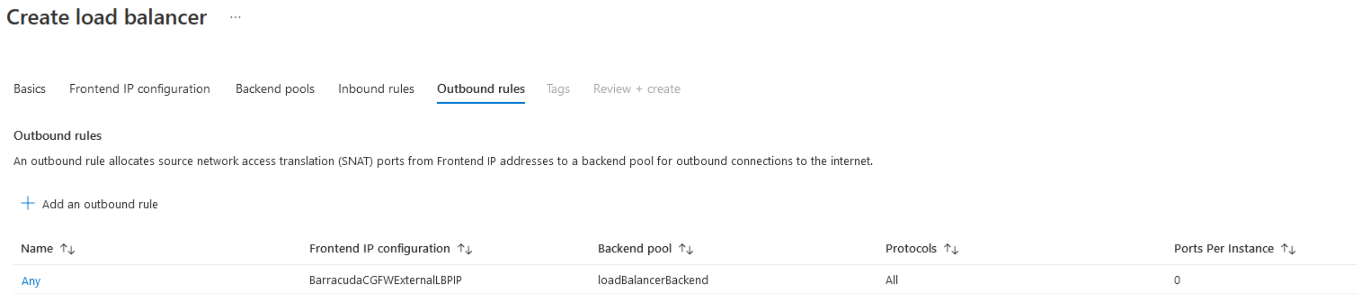
Azure automatically assigns the number of outbound ports to use for source network address translation (SNAT) based on the number of frontend IP addresses and backend pool instances.
Learn more about outbound connectivity

Port allocation: Use the default number of outbound ports

Azure may drop existing connections when you scale out. Manually allocate ports to avoid dropped connections.

4. Click **Add**.

The rule is now displayed in the list.



Create load balancer

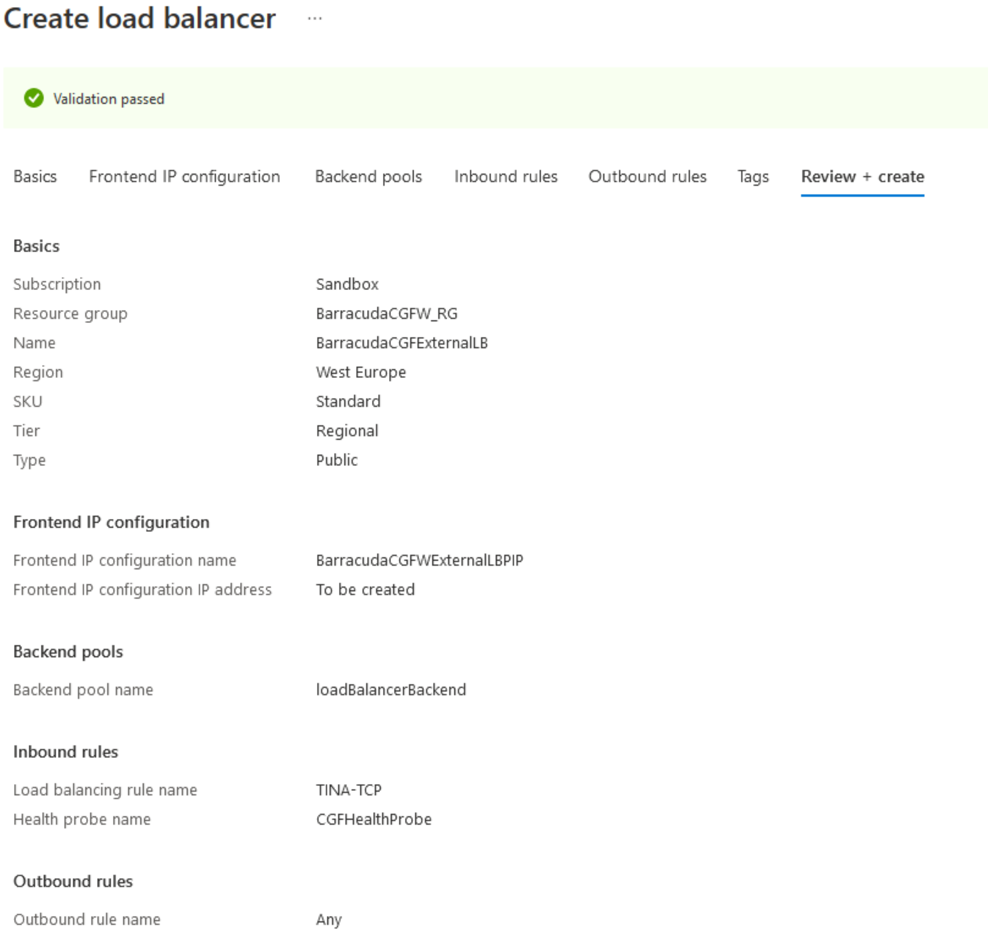
Outbound rules

An outbound rule allocates source network access translation (SNAT) ports from Frontend IP addresses to a backend pool for outbound connections to the internet.

Name	Frontend IP configuration	Backend pool	Protocols	Ports Per Instance
Any	BarracudaCGFWExternalLBPIP	loadBalancerBackend	All	0

Step 7.6 Create the Load Balancer

1. Click **Next: Tags**.
2. Click **Next: Review + create**.



Create load balancer

Basics

Subscription	Sandbox
Resource group	BarracudaCGFW_RG
Name	BarracudaCGFExternalLB
Region	West Europe
SKU	Standard
Tier	Regional
Type	Public

Frontend IP configuration

Frontend IP configuration name	BarracudaCGFWExternalLBPIP
Frontend IP configuration IP address	To be created

Backend pools

Backend pool name	loadBalancerBackend

Inbound rules

Load balancing rule name	TINA-TCP
Health probe name	CGFHealthProbe

Outbound rules

Outbound rule name	Any

3. Review your settings and click **Create**.

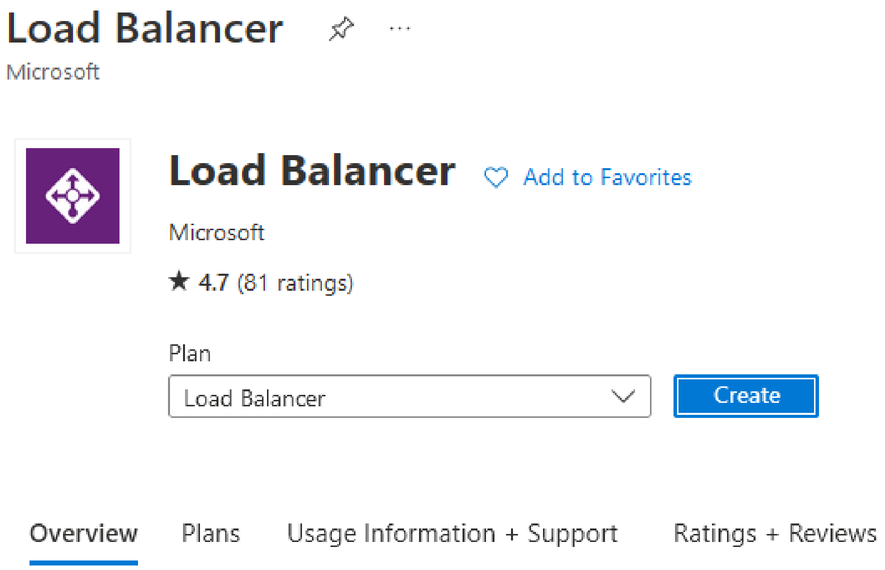
Step 8. Create the Internal Load Balancer

The internal load balancer is essential for a standard load balancer HA design because it is the destination for all user-defined routes.

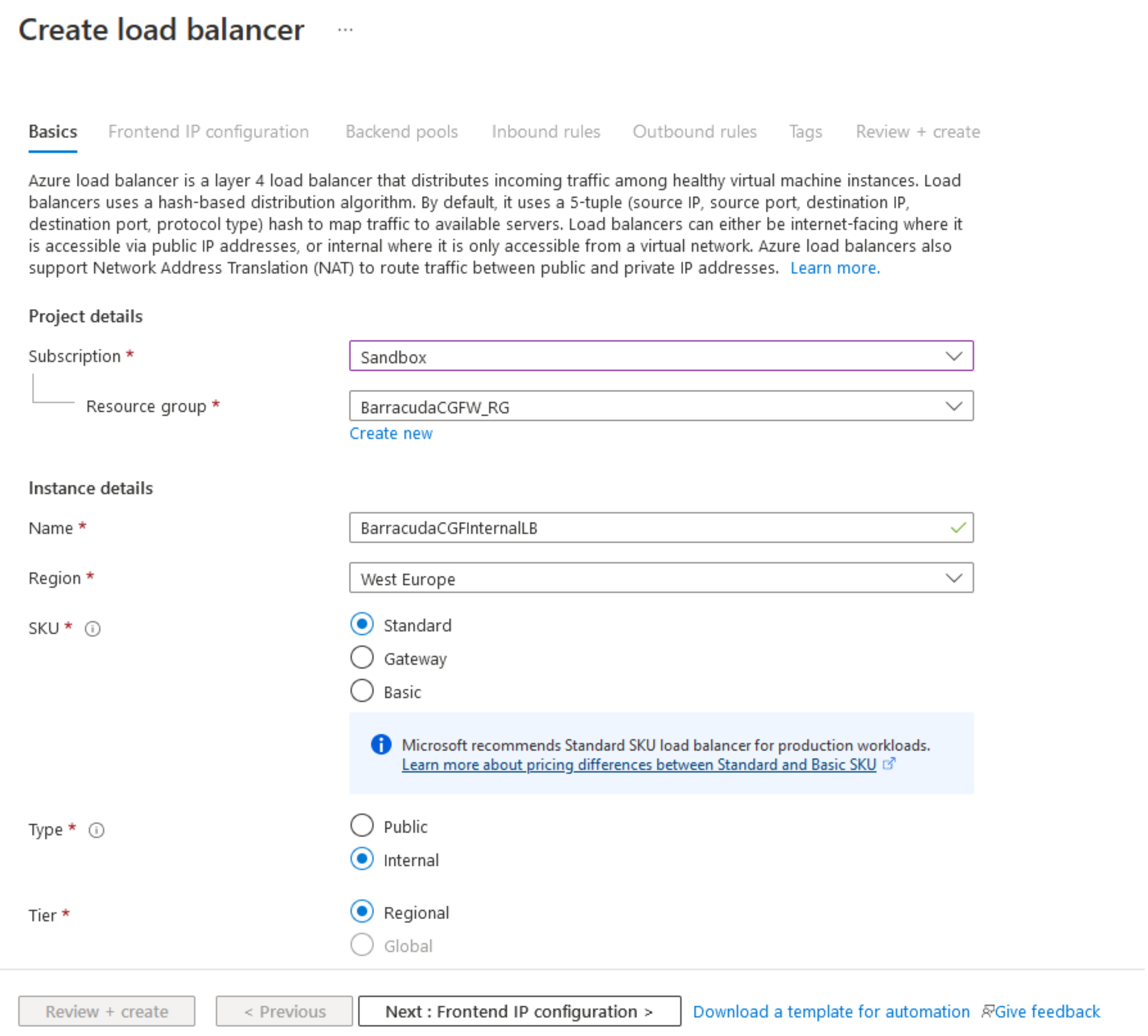
Step 8.1 Configure the Basic Settings

From the Azure portal:

1. Under **Azure services**, click the **+** symbol to **Create a resource**.
2. Type in `Load Balancer` and select the resource from the list. The **Load Balancer** page opens.



3. Click **Create**.
4. On the next page, configure the following settings:
    - **Name** – Enter the name of the Internal Load Balancer.
    - **Region** – Select your region.
    - **SKU** – Select the **Standard** SKU.
    - **Type** – Select the **Internal** type.
    - **Tier** – Select you tier.



Create load balancer ···

Basics | Frontend IP configuration | Backend pools | Inbound rules | Outbound rules | Tags | Review + create

Azure load balancer is a layer 4 load balancer that distributes incoming traffic among healthy virtual machine instances. Load balancers uses a hash-based distribution algorithm. By default, it uses a 5-tuple (source IP, source port, destination IP, destination port, protocol type) hash to map traffic to available servers. Load balancers can either be internet-facing where it is accessible via public IP addresses, or internal where it is only accessible from a virtual network. Azure load balancers also support Network Address Translation (NAT) to route traffic between public and private IP addresses. Learn more.

Project details

Subscription * – Sandbox

Resource group * – BarracudaCGFW_RG

Create new

Instance details

Name * – BarracudaCGFInternalLB

Region * – West Europe

SKU * – ● Standard ○ Gateway ○ Basic

Microsoft recommends Standard SKU load balancer for production workloads.
Learn more about pricing differences between Standard and Basic SKU

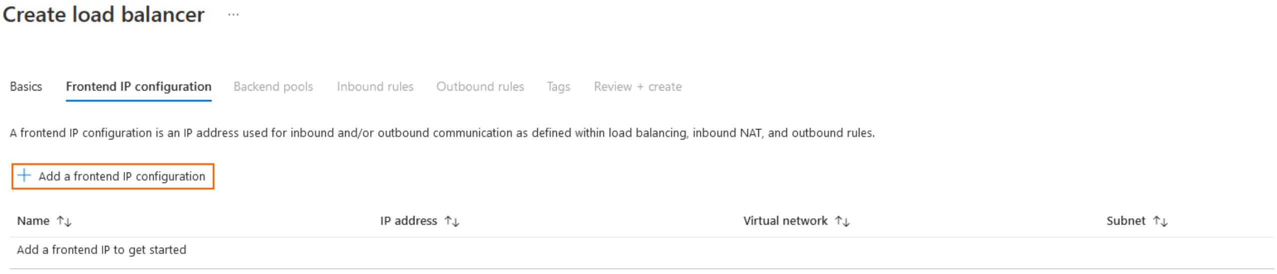
Type * – ○ Public ● Internal

Tier * – ● Regional ○ Global

Review + create
< Previous
Next : Frontend IP configuration >
Download a template for automation
Give feedback

Step 8.2 Add a Frontend IP Configuration

1. Click **Next: Frontend IP configuration**.
2. Click **+** to **Add a frontend IP configuration**.



Create load balancer ···

Basics | Frontend IP configuration | Backend pools | Inbound rules | Outbound rules | Tags | Review + create

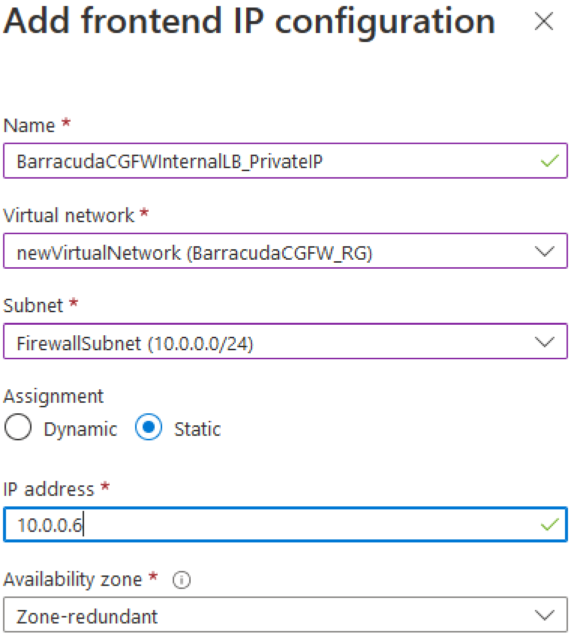
A frontend IP configuration is an IP address used for inbound and/or outbound communication as defined within load balancing, inbound NAT, and outbound rules.

+ Add a frontend IP configuration

Name ↑↓	IP address ↑↓	Virtual network ↑↓	Subnet ↑↓
Add a frontend IP to get started			

3. On the next page, configure the following settings:
    - **Name** – Enter the name of the frontend IP.
    - **Virtual Network** – Select the virtual network your firewalls are in.

  - **Subnet** – Select the subnet the firewalls are in.
  - **IP Address assignment** – Select **Static**.
  - **Private IP address** – Enter a private IP in that subnet for the load balancer to use.
  - **Availability Zone**– Select **Zone-redundant**.



Add frontend IP configuration

Name *
BarracudaCGFWInternalLB_PrivateIP

Virtual network *
newVirtualNetwork (BarracudaCGFW_RG)

Subnet *
FirewallSubnet (10.0.0.0/24)

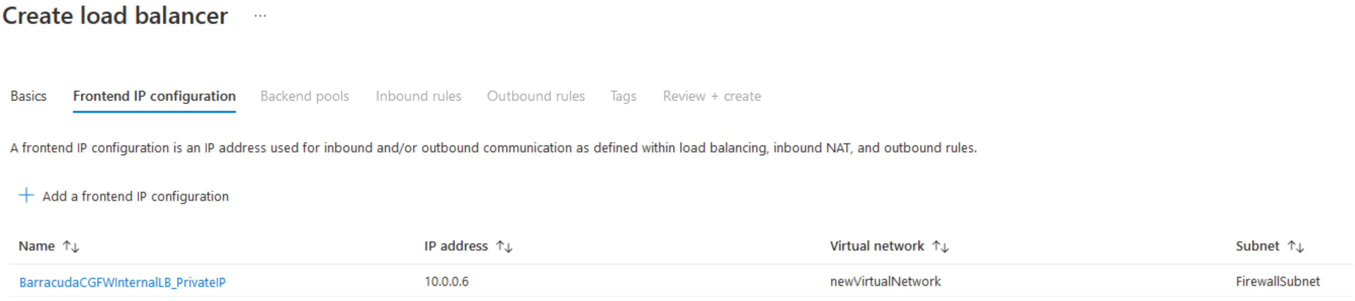
Assignment
○ Dynamic ● Static

IP address *
10.0.0.6

Availability zone *
Zone-redundant

4. Click **Add**.

The entry is now displayed in the list.



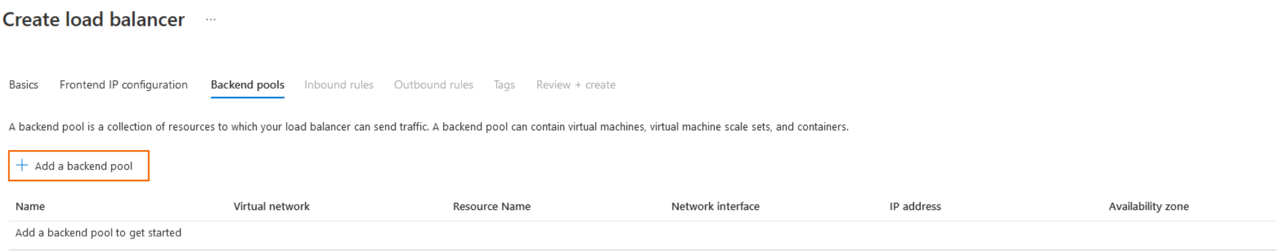
Create load balancer

A frontend IP configuration is an IP address used for inbound and/or outbound communication as defined within load balancing, inbound NAT, and outbound rules.

Name	IP address	Virtual network	Subnet
BarracudaCGFWInternalLB_PrivateIP	10.0.0.6	newVirtualNetwork	FirewallSubnet

Step 8.3 Add a Backend Pool

1. Click **Next: Backend pools**.
2. Click **+** to **Add a backend pool**.

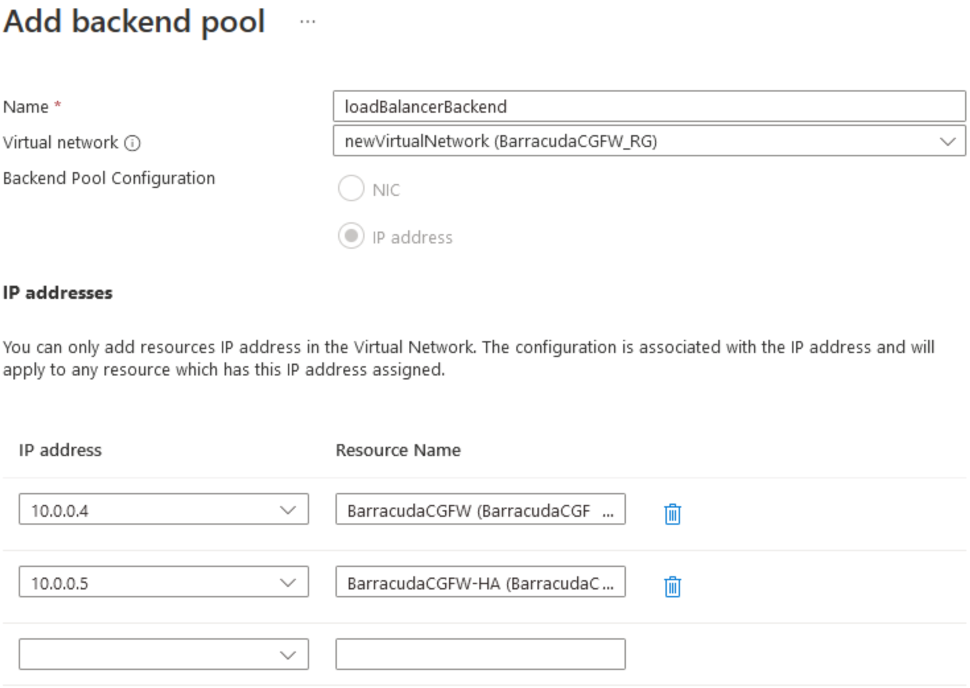


Create load balancer

A backend pool is a collection of resources to which your load balancer can send traffic. A backend pool can contain virtual machines, virtual machine scale sets, and containers.

Name	Virtual network	Resource Name	Network interface	IP address	Availability zone
Add a backend pool to get started					

3. Complete the fields as below:
    - **Name** – Enter your desired name for the backend pool.
    - **Virtual Network** – Select the virtual network you built your firewalls in.
    - **Backend Pool Configuration** – Select **IP address**.
    - **IP address** – Select the CGF IP addresses where you wish the LB to send traffic to.



Add backend pool

Name * loadBalancerBackend

Virtual network newVirtualNetwork (BarracudaCGFW_RG)

Backend Pool Configuration ○ NIC ● IP address

IP addresses

You can only add resources IP address in the Virtual Network. The configuration is associated with the IP address and will apply to any resource which has this IP address assigned.

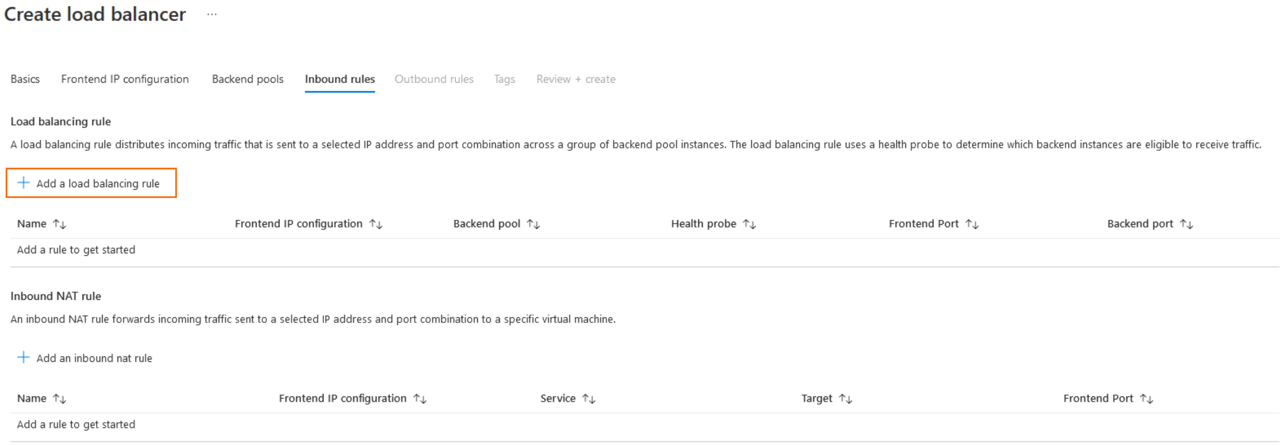
IP address	Resource Name
10.0.0.4	BarracudaCGFW (BarracudaCGF ...
10.0.0.5	BarracudaCGFW-HA (BarracudaC...

4. Repeat the steps for the second firewall VM you built.
5. Click **Add**.

Step 8.4 Add an Inbound Rule

To create the load balancing rules for incoming traffic, you need to enable load balancing for all ports so that the firewalls can check all traffic.

1. Click **Next: Inbound rules**.
2. Click **+** to **Add a load balancing rule**.



Load balancing rule

A load balancing rule distributes incoming traffic that is sent to a selected IP address and port combination across a group of backend pool instances. The load balancing rule uses a health probe to determine which backend instances are eligible to receive traffic.

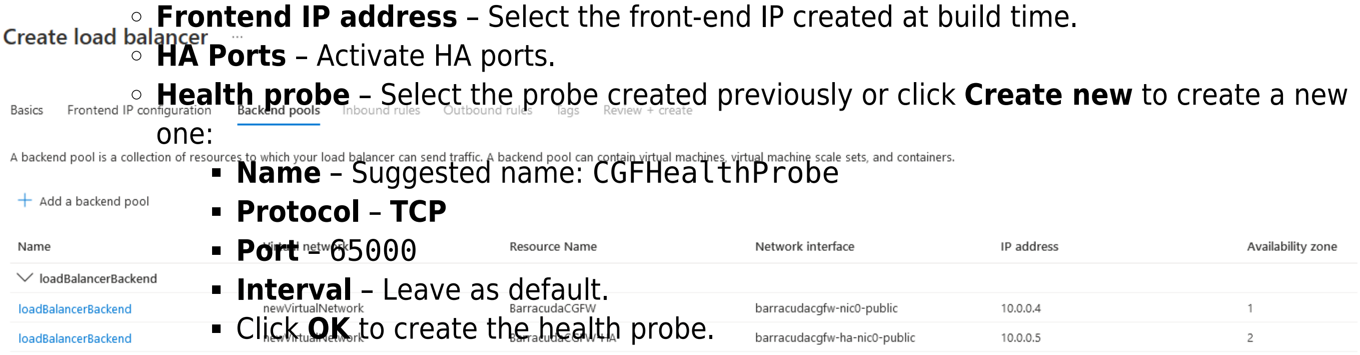
Name ↑↓	Frontend IP configuration ↑↓	Backend pool ↑↓	Health probe ↑↓	Frontend Port ↑↓	Backend port ↑↓
Add a rule to get started					

Inbound NAT rule

An inbound NAT rule forwards incoming traffic sent to a selected IP address and port combination to a specific virtual machine.

Name ↑↓	Frontend IP configuration ↑↓	Service ↑↓	Target ↑↓	Frontend Port ↑↓
Add a rule to get started				

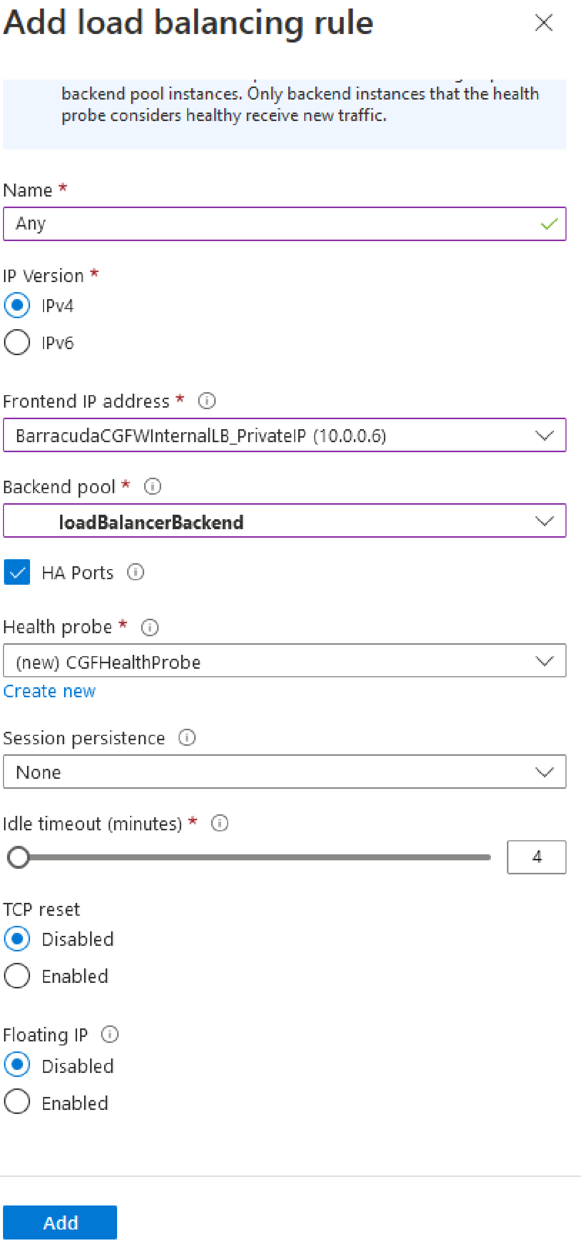
3. Configure the settings as below:
    - **Name** – Suggested name: Any
    - **IP Version** – **IPv4**
    - **Frontend IP address** – Select the front-end IP created at build time.
    - **HA Ports** – Activate HA ports.
    - **Health probe** – Select the probe created previously or click **Create new** to create a new one:
        - **Name** – Suggested name: `CGFHealthProbe`
        - **Protocol** – **TCP**
        - **Port** – `65000`
        - **Interval** – Leave as default.
        - Click **OK** to create the health probe.
    - **Session persistence** – Leave as default (**None**).
    - **Idle timeout (minutes)** – Leave as default (4).
    - **TCP reset** – Leave as default (Disabled).



A backend pool is a collection of resources to which your load balancer can send traffic. A backend pool can contain virtual machines, virtual machine scale sets, and containers.

Name	Virtual network	Resource Name	Network interface	IP address	Availability zone
loadBalancerBackend					
loadBalancerBackend	newVirtualNetwork	BarracudaCGFW	barracudacgfw-nic0-public	10.0.0.4	1
loadBalancerBackend	newVirtualNetwork	BarracudaCGFW-HA	barracudacgfw-ha-nic0-public	10.0.0.5	2

- ◦ **Floating IP** – Leave as default (Disabled).



Add load balancing rule

backend pool instances. Only backend instances that the health probe considers healthy receive new traffic.

Name * Any

IP Version *
● IPv4
○ IPv6

Frontend IP address * BarracudaCGFWInternalLB_PrivateIP (10.0.0.6)

Backend pool * loadBalancerBackend

☑ HA Ports

Health probe * (new) CGFHealthProbe

Session persistence None

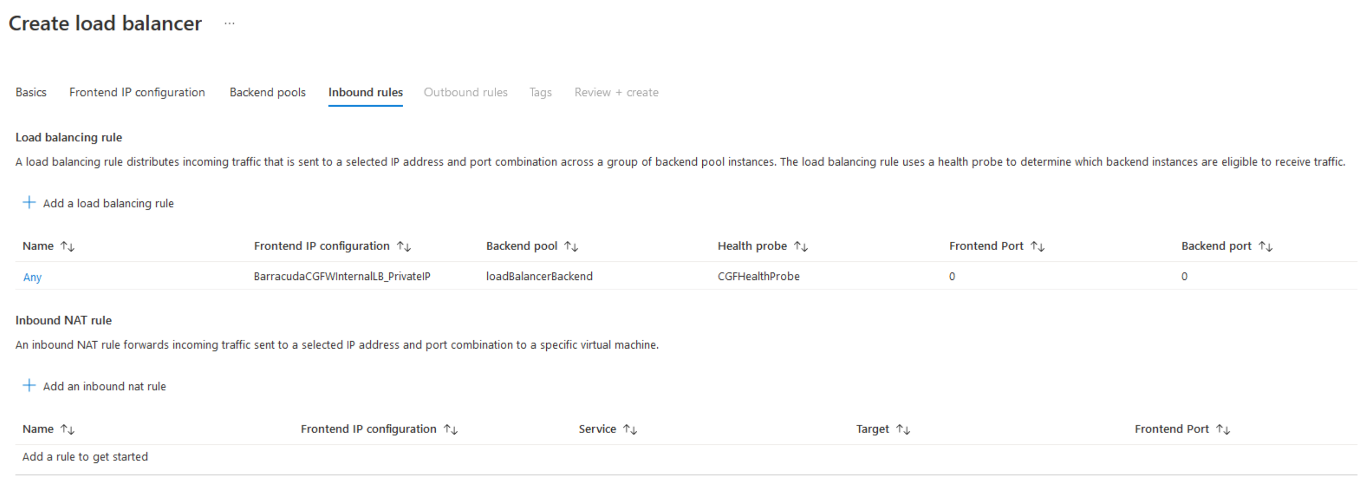
Idle timeout (minutes) * 4

TCP reset
● Disabled
○ Enabled

Floating IP
● Disabled
○ Enabled

4. Click **Add**.

The rule is now displayed in the list.



Load balancing rule

A load balancing rule distributes incoming traffic that is sent to a selected IP address and port combination across a group of backend pool instances. The load balancing rule uses a health probe to determine which backend instances are eligible to receive traffic.

Name ↑↓	Frontend IP configuration ↑↓	Backend pool ↑↓	Health probe ↑↓	Frontend Port ↑↓	Backend port ↑↓
Any	BarracudaCGFWInternalLB_PrivateIP	loadBalancerBackend	CGFHealthProbe	0	0

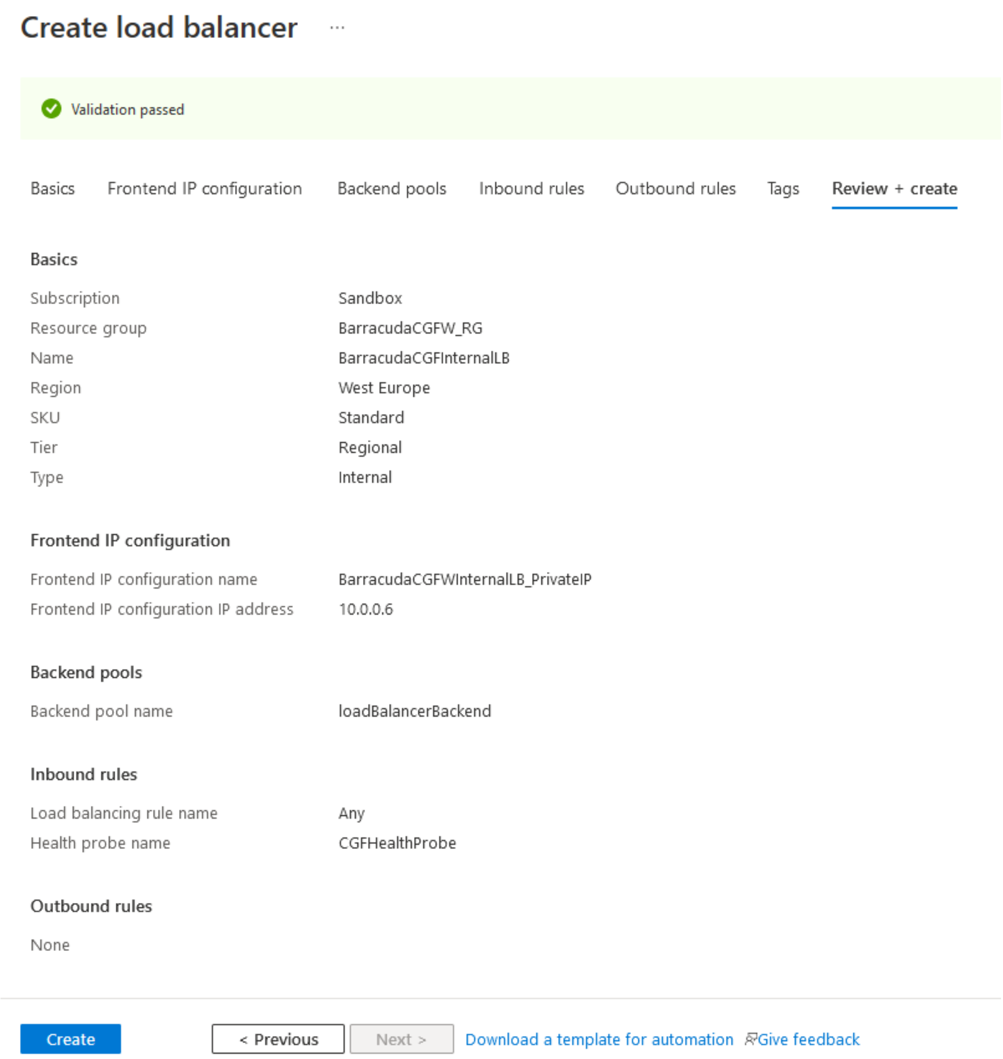
Inbound NAT rule

An inbound NAT rule forwards incoming traffic sent to a selected IP address and port combination to a specific virtual machine.

Name ↑↓	Frontend IP configuration ↑↓	Service ↑↓	Target ↑↓	Frontend Port ↑↓
Add a rule to get started				

Step 8.5 Create the Load Balancer

1. Click **Next: Outbound rules**.
2. Click **Next: Tags**.
3. Click **Next: Review + create**.



Create load balancer

Validation passed

Basics

Subscription	Sandbox
Resource group	BarracudaCGFW_RG
Name	BarracudaCGFInternalLB
Region	West Europe
SKU	Standard
Tier	Regional
Type	Internal

Frontend IP configuration

Frontend IP configuration name	BarracudaCGFWInternalLB_PrivateIP
Frontend IP configuration IP address	10.0.0.6

Backend pools

Backend pool name	loadBalancerBackend

Inbound rules

Load balancing rule name	Any
Health probe name	CGFHealthProbe

Outbound rules

None

4. Review your settings and click **Create**.

Now you have completed the setup of the load balancers.

Step 9. Enable IP Forwarding

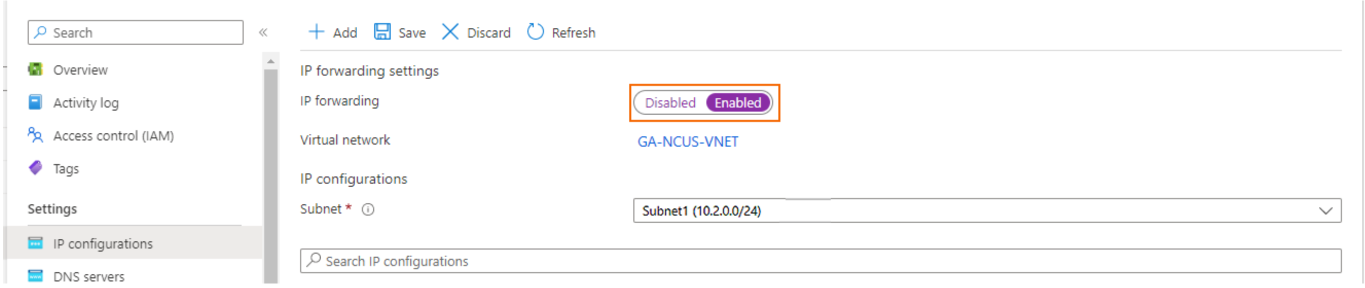
To allow the firewall to pass traffic not intended for itself, you must update the network interface.

In the Azure portal:

1. Go to the virtual machine.
2. Go to **Networking**, and locate the Network Interface attached to the firewall.
3. In **IP configurations**, make sure that **IP forwarding** is enabled.

If not already done, make the ipconfig static by clicking on it and setting the assignment to

static.



IP forwarding settings

IP forwarding: Disabled **Enabled**

Virtual network: GA-NCUS-VNET

IP configurations

Subnet *: Subnet1 (10.2.0.0/24)

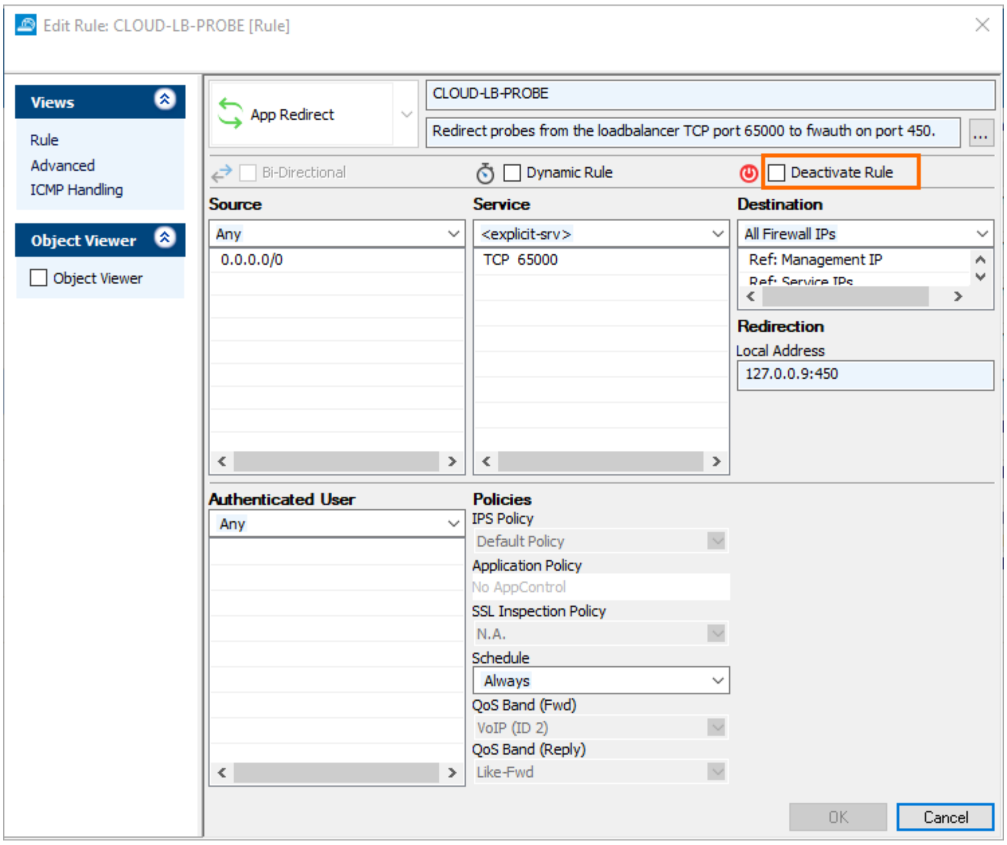
Search IP configurations

For more information, see [How to Configure a Client-to-Site VPN Group Policy]() or [How to Configure a Client-to-Site TINA VPN with Personal Licenses]() .

Step 10. Allow the Load Balancer Health Probes to Succeed

Activate the preconfigured firewall forwarding rule to allow load balancer health probes to succeed. The connection will use the port you indicated in Steps 2 & 3 above. It will originate from 168.63.129.16 and can be redirected to any service running locally on the firewall (e.g., 127.0.0.9:450 for firewall authentication service, or 127.0.0.9:691 for FW TINA VPN).

1. Log into the Barracuda CloudGen Firewall with Barracuda Firewall Admin.
2. Go to **CONFIGURATION > Configuration Tree > Assigned Services > Firewall > Forwarding Rules**.
3. Click **Lock**.
4. Open the rule `CLOUD-LB-PROBE`.
5. To activate the rule, clear the check box next to **Deactivate Rule**.



Edit Rule: CLOUD-LB-PROBE [Rule]

App Redirect

CLOUD-LB-PROBE

Redirect probes from the loadbalancer TCP port 65000 to fwauth on port 450.

☐ Bi-Directional ☐ Dynamic Rule ☐ Deactivate Rule

Source: Any — 0.0.0.0/0

Service: <explicit-srv> — TCP 65000

Destination: All Firewall IPs — Ref: Management IP, Ref: Service IPs

Redirection — Local Address: 127.0.0.9:450

Authenticated User: Any

Policies

IPS Policy: Default Policy

Application Policy: No AppControl

SSL Inspection Policy: N.A.

Schedule: Always

QoS Band (Fwd): VoIP (ID 2)

QoS Band (Reply): Like-Fwd

6. Click **OK**.
7. Click **Send Changes** and **Activate**, then click **Activate**.

Step 11. Configure User-Defined Routing in Azure

Configure UDR for the backend VMs to use the internal load balancer's IP as their default gateways for all connections to the Internet. 0.0.0.0/0 will only impact traffic that does not have a route already present in Azure. E.g., Internet.

To affect traffic within the VNET, subnet, or peered VNET, introduce routes for a matching destination network. (Check the effective routing of a VM if uncertain what routes are present already).

Step 12. Configure a Client-to-Site VPN for Management Access

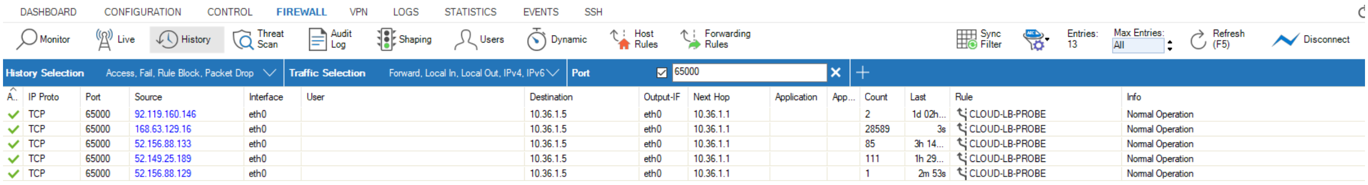
Configure a TINA client-to-site VPN that will be used for management access. Connect via the load balancer public IP address.

For more information, see How to Configure a Client-to-Site VPN Group Policy or How to Configure a Client-to-Site TINA VPN with Personal Licenses.

Check the Connection

Use a client-to-site VPN connection to manage both Barracuda CloudGen Firewall VMs via the internal IP addresses. For more information, see Client-to-Site VPN.
Go to the **Firewall > History** view and confirm you can see the health probes succeeding. Traffic should be passing through the firewall correctly. If you see timeouts, confirm NSGs on the interfaces permit traffic and that **IP Forwarding** is enabled.

Example of Successful Monitoring Polls on port 65000



Entries: 13
Max Entries: All

History Selection: Access, Fail, Rule Block, Packet Drop
Traffic Selection: Forward, Local In, Local Out, IPv4, IPv6
Port ☑ 65000

A..	IP Proto	Port	Source	Interface	User	Destination	Output-IF	Next Hop	Application	App...	Count	Last	Rule	Info
✓	TCP	65000	92.119.160.146	eth0		10.36.1.5	eth0	10.36.1.1			2	1d 02h...	CLOUD-LB-PROBE	Normal Operation
✓	TCP	65000	168.63.129.16	eth0		10.36.1.5	eth0	10.36.1.1			28589	3s	CLOUD-LB-PROBE	Normal Operation
✓	TCP	65000	52.156.88.133	eth0		10.36.1.5	eth0	10.36.1.1			85	3h 14...	CLOUD-LB-PROBE	Normal Operation
✓	TCP	65000	52.149.25.189	eth0		10.36.1.5	eth0	10.36.1.1			111	1h 29...	CLOUD-LB-PROBE	Normal Operation
✓	TCP	65000	52.156.88.129	eth0		10.36.1.5	eth0	10.36.1.1			1	2m 53s	CLOUD-LB-PROBE	Normal Operation

Figures