

Exposing OCI Cache via a Network Load Balancer (NLB)

Pre-requisites:

- OCI Cache instance created in a Public Subnet

Creating and Configuring the NLB

Create network load balancer

A network load balancer provides automated traffic distribution from one entry point to multiple servers in a backend set. The network load balancer ensures that your services remain available by directing traffic only to healthy servers in the backend set.

Load balancer name
Demo_PK_NLB

Choose visibility type

Public
You can use the assigned public IP address as a front end for incoming traffic. ✓

Private
You can use the assigned private IP address as a front end for internal incoming VCN traffic.

☐ Allow IPv6 address assignment
Enables a dual-stack IPv4/IPv6 implementation for your load balancer. Learn more about [IPv6 addresses](#).

Assign a public IP address

Ephemeral IPv4 address
Automatically assign an IPv4 address from the Oracle pool. ✓

Reserved IPv4 address
Select an existing reserved IPv4 address or create a new one from one of your IP pools.

Choose networking

Virtual cloud network in PiotrKurzynoga [\(Change compartment\)](#)
DemoNetwork

Subnet in PiotrKurzynoga [\(Change compartment\)](#)
Pub_sub

☒ Use network security groups to control traffic ⓘ

Network security group in PiotrKurzynoga [\(Change compartment\)](#)
PK-nsg

+ Another network security group

Show advanced options

Select public if you want to expose your traffic to the public internet.

Note: Use Network Security Groups (NSGs) to ensure that unauthorized parties are not able to connect to your cluster by creating appropriate Security Rules.

Create network load balancer

✓ Add details
2 Configure listener
3 Choose backends
4 Review and create

A listener is a logical entity that checks for incoming traffic on the network load balancer's IP address. You must configure at least one listener for each traffic type in order to handle UDP, TCP, or UDP/TCP traffic. You can configure additional listeners after you create your network load balancer.

Listener name
Demo_Listener_PK

Specify the type of traffic your listener handles
UDP TCP UDP/TCP

Ingress traffic port
☐ Use any port
☒ Specify the port 6379

This port will be used for the backend ports, if any, selected in the following step.

Proxy protocol
☐ Enable proxy protocol V2
Sends additional connection information, such as the source and destination IP addresses and ports, to their backends across layers of network address translation (NAT) or TCP proxies.

Locking down to port 6379 so that only Redis specific requests will be allowed through.

Create network load balancer

✓ Add details
✓ Configure listener
3 Choose backends
4 Review and create

A network load balancer distributes traffic to backends within a backend set. A backend set is a logical entity defined by a load balancing policy, a health check policy, and a list of backends (compute instances).

Backend set name
NLB_Backend_PK

Select backends

Add by compute instance or IP addresses *Optional*
No backends selected. Click Add backends to select resources from a list of available compute instances or add IP addresses. You can also add backends after you create the network load balancer.

Add backends

☐ Preserve source IP ⓘ

Specify health check policy

A health check is a test to confirm the availability of backends. A health check can be a request or a connection attempt. Based on a time interval you specify, the network load balancer applies the health check policy to continuously monitor backends.

Protocol TCP Port 6379

Interval in MS *Optional* 10000 Timeout in MS *Optional* 3000

Number of retries *Optional* 3

Request data *Optional*

This will be base64 encoded.

Response data *Optional*

This will be base64 encoded.

☒ Fail open
If selected, the network load balancer continues to move traffic to backends in this backend set using the current configuration, even if all the backends' state becomes unhealthy.

☐ Enable instant failover
When enabled existing flows will be reshaped to a healthy backend when the existing backend is marked as unhealthy, except in cases where fail open is enabled and all backends are marked as unhealthy.

Add the back-end (OCI Cache primary endpoint IP – this can be retrieved from within the DNS Management by finding the appropriate zone entry) as demonstrated below by selecting the IP Address option, then configure the health check policy for your applicable port.

Backend IPv4 addresses

IP addresses can be added if client header preservation on the backend set is disabled.

IP address	Port ⓘ	Weight ⓘ
192.168.0	6379	1

+ Another backend



Update security list rules

After adding IP address backends, manually configure your security list rules to ensure proper traffic flow. [Learn more about configuring security lists.](#)

This is what the creation summary should look like.

Create network load balancer

- ✓ Add details
- ✓ Configure listener
- ✓ Choose backends
- 4 Review and create

Load balancer details

[Edit](#)

Name: Demo_PK_NLB
Visibility: Public
Address assignment: IPv4

VCN: DemoNetwork
Subnet: Pub_sub
NSG: PK-nsg

Listener details

[Edit](#)

Name: Demo_Listener_PK
IP protocol version: IPv4
Attached backend: NLB_Backend_PK
Traffic type: TCP

Port: 6379
Proxy protocol version 2: Disabled

Backend set details

[Edit](#)

Name: NLB_Backend_PK
IP protocol version: IPv4
Attached listener: Demo_Listener_PK

Source IP preservation: Disabled
Load balancing policy: Five tuple hash
Security lists: manual

Backends

Name	IP address	Availability domain	Compartment	Port	Weight
-	192.168.0	-	-	6379	1

Health check policy

[Edit](#)

Protocol: TCP
Interval in MS: 10000
Number of retries: 3
Fail open: Enabled

Port: 6379
Timeout in MS: 3000
Request data: -
Response data: -
Instant failover: Disabled

Testing

We can now connect to our OCI Cache with Redis instance from anywhere.

Networking > Load balancers > Network load balancer > Network load balancer details

Demo_PK_NLB

Edit Move resource Add tags Create path analysis Delete

Load balancer information Tags

Load balancer information

OCID: ...jfourng Show Copy

Created: Tue Aug 13, 2024, 13:39:27 UTC

IP address: 130.61.227.133 (public)

IP address: 192.168.0.173 (private)

Subnet: Pub_sub

Network security groups: PK-nsg Edit

Overall health

OK

Backend sets health

0 Critical

0 Warning

0 Unknown

1 OK

```
phantompete@...$ redis-cli --tls -h 130.61.227.133:6379>
```