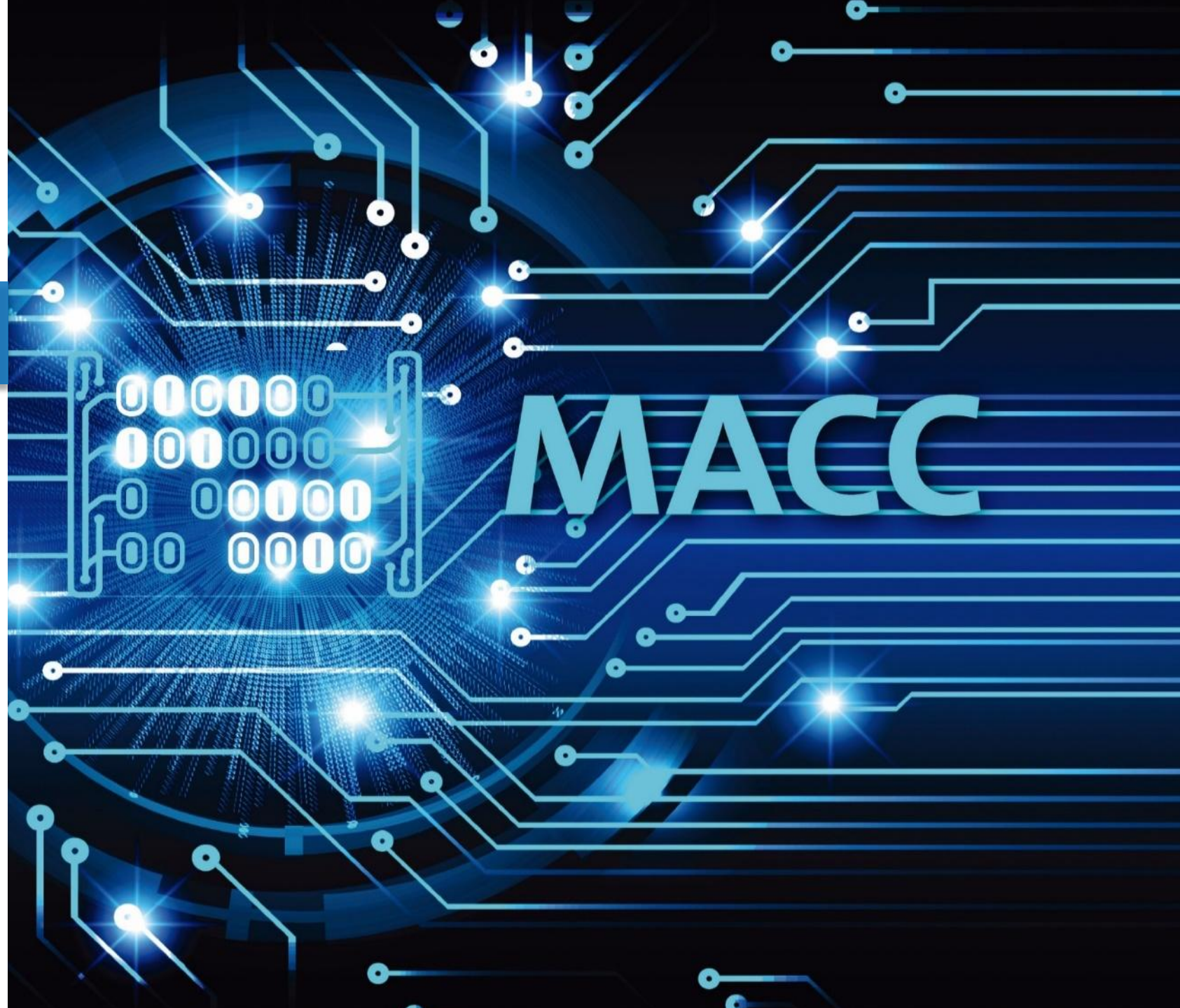
Universidad del Rosario

MACC
Matemáticas Aplicadas y Ciencias de la Computación

AWS Load Balancing

Ciberseguridad



Daniel Orlando Díaz López, PhD

Profesor principal
Departamento MACC
Universidad del Rosario
danielo.diaz@urosario.edu.co

Elastic Load Balancing

Elastic Load Balancing automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, and IP addresses. It can handle the varying load of your application traffic in a single Availability Zone or across multiple Availability Zones. Elastic Load Balancing offers three types of load balancers: Application Load Balancer, Network Load Balancer and Classic Load Balancer.

Using Elastic Load Balancing in Your Amazon Virtual Private Cloud (VPC)

Elastic Load Balancing makes it easy to create an internet-facing entry point into your Amazon Virtual Private Cloud (VPC). You can assign security groups to your load balancer to control which ports are open to a list of allowed sources. Together, Elastic Load Balancing works with Amazon Virtual Private Cloud (VPC) to provide you the flexibility to centrally manage SSL settings and offload CPU intensive workloads from your applications.

Architecture of the Elastic Load Balancing Service and How It Works

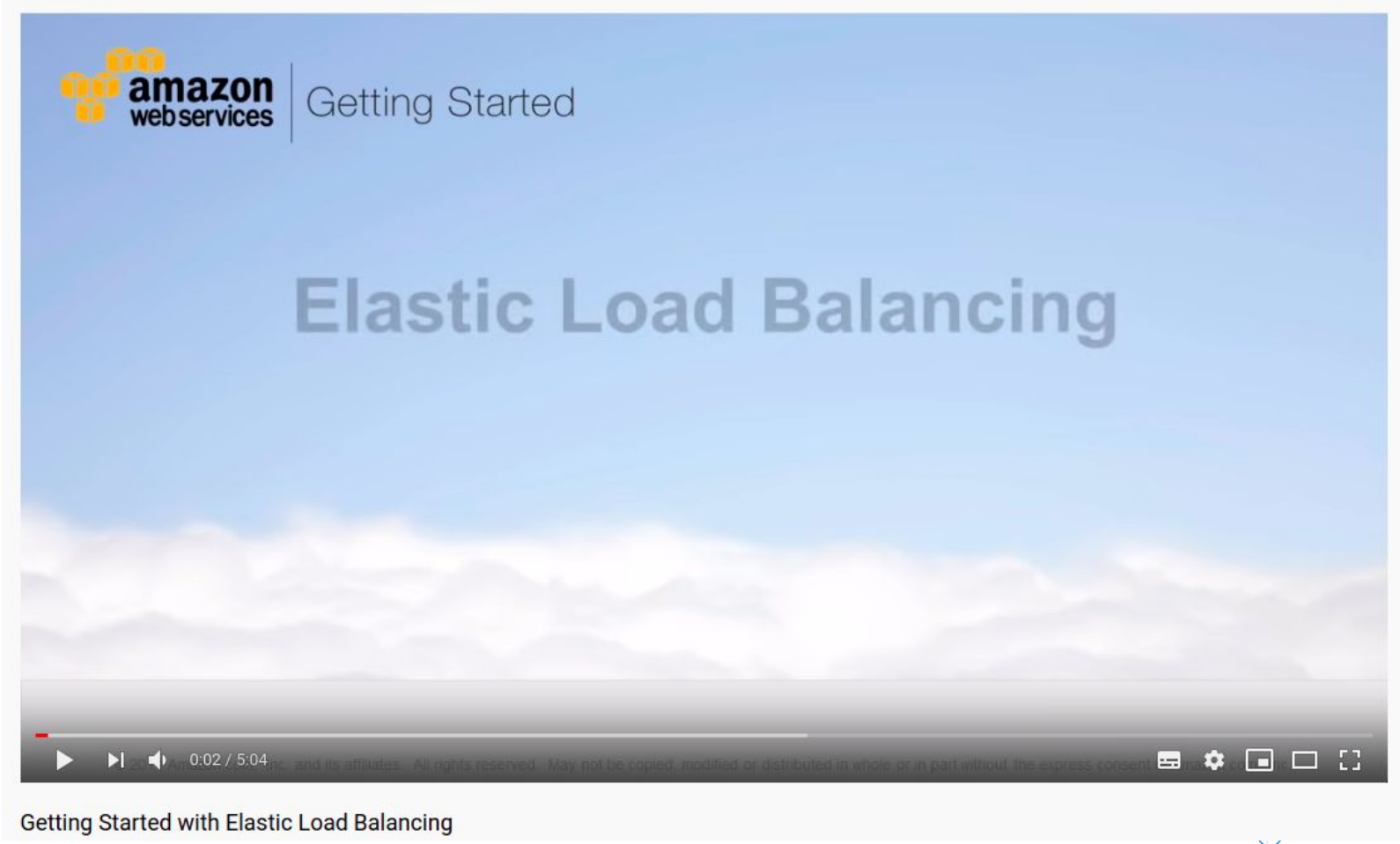
There are two logical components in the Elastic Load Balancing service architecture: load balancers and a controller service. The load balancers are resources that monitor traffic and handle requests that come in through the Internet. The controller service monitors the load balancers, adds and removes capacity as needed, and verifies that load balancers are behaving properly. See figure below to observe the architecture of the AWS Elastic Load Balancing Service.

AWS Load Balancing

Universidad del Rosario | MACC Matemáticas Aplicadas y Ciencias de la Computación

Introductory video: Getting Started with Elastic Load Balancing

https://www.youtube.com/watch?v=8KQ8aLoxVi0&feature=youtu.be



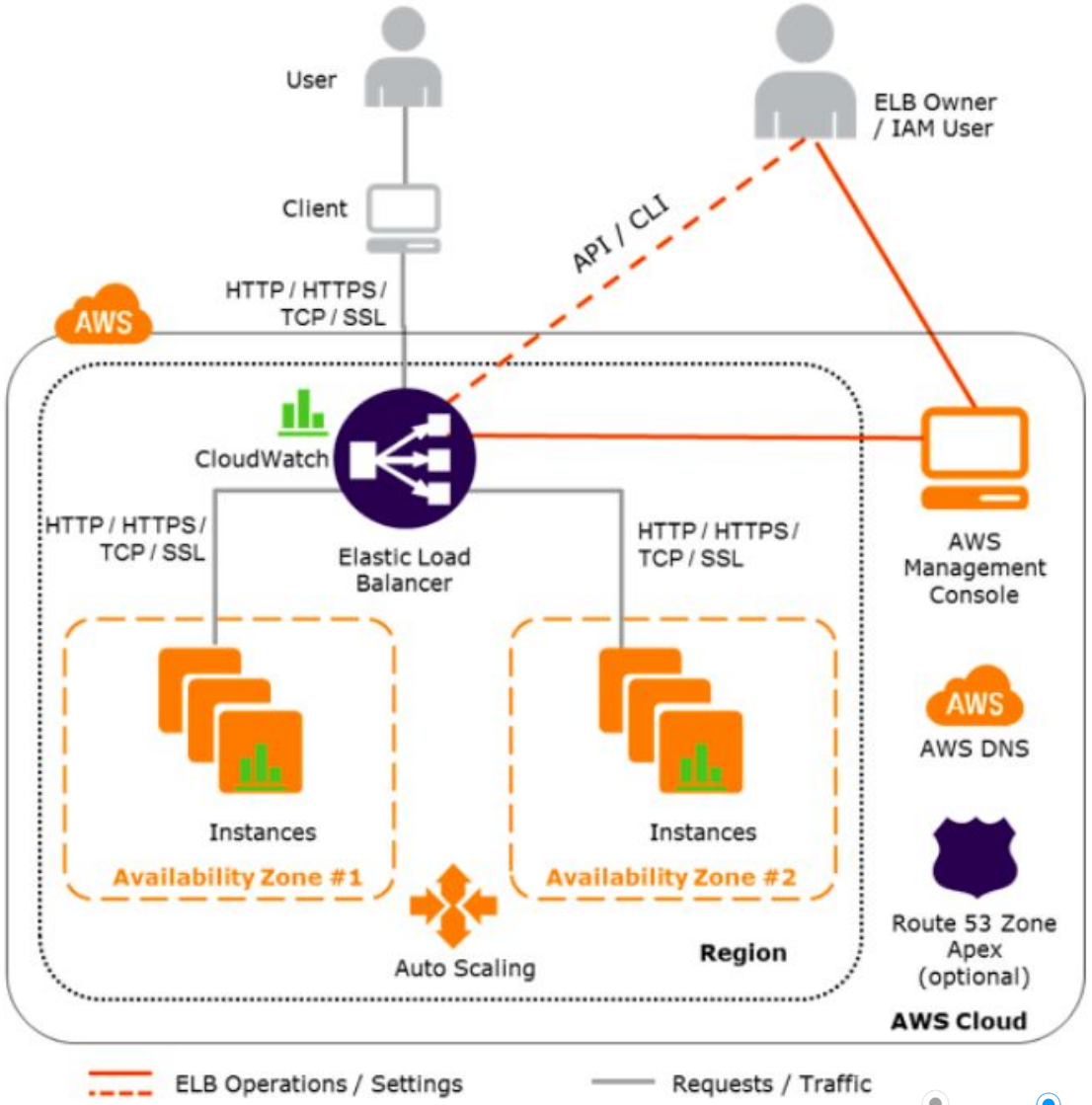
Getting Started with Elastic Load Balancing

AWS Load Balancing

Universidad del Rosario | MACC Matemáticas Aplicadas y Ciencias de la Computación

Steps:

1. Create a load balancer (Application Load Balancer or Network Load Balancer)
2. Specify a unique name and a network
3. Create listeners for your load balancer
   a. HTTP or HTTPS for Application Load Balancer
   b. TCP or UDP for Network Load Balancer
4. Configure health checks for your load balancer



AWS Load Balancer benefits:

- Handle of load in one or multiple Availability Zones
- 1 additional security layer before the servers
- Control traffic for all servers
- Avoid DoS
- High availability services

Availability zones vs Regions

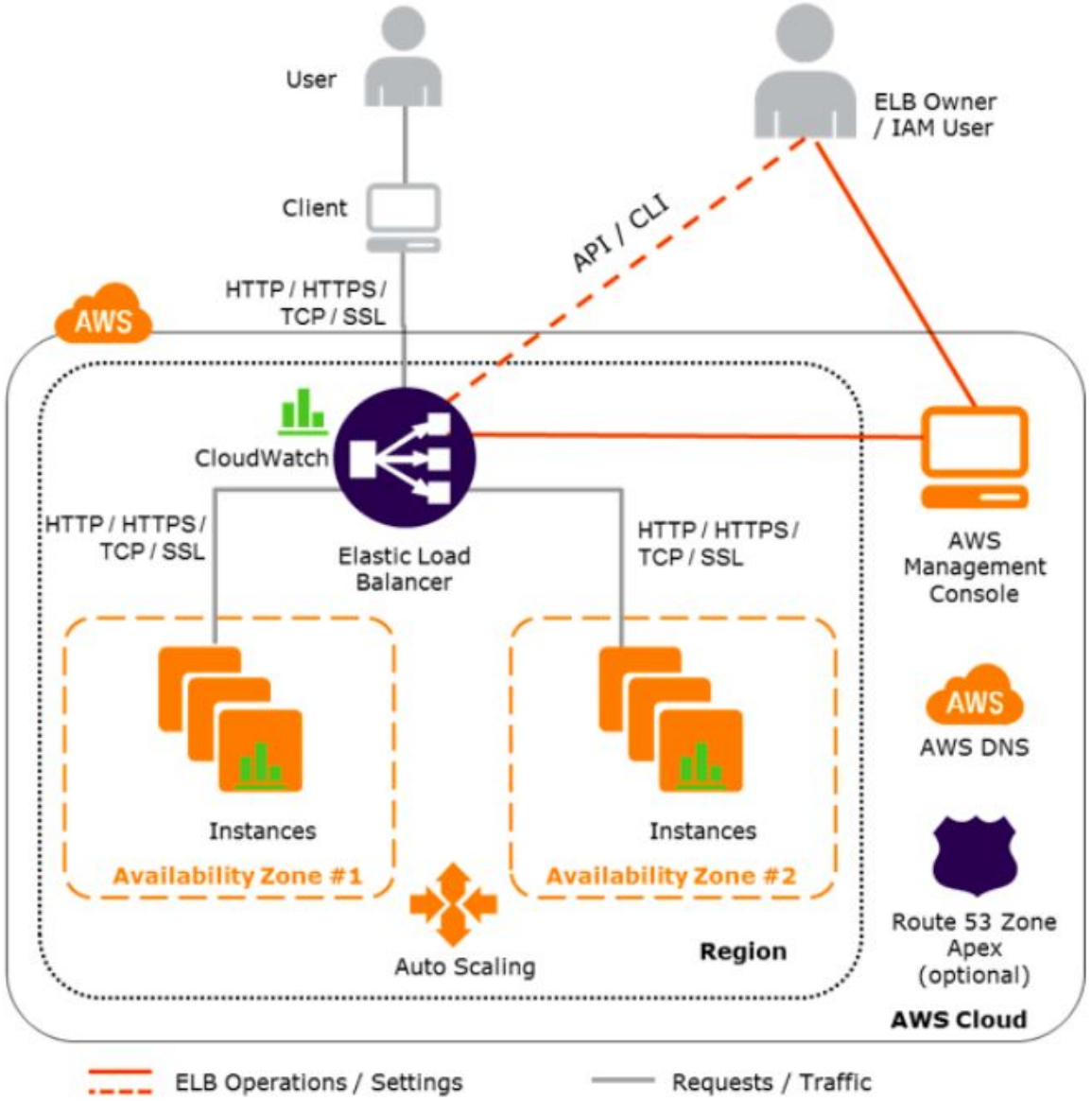
- There are 31 AWS regions around the world.
- Each regions may have one or more availability zones. In total there are 99 availability zones.
- Forecast of 5 more Availability Zones and 5 more AWS Regions in Canada, Israel, Malaysia, New Zealand, and Thailand.

https://aws.amazon.com/about-aws/global-infrastructure/

AWS Load Balancing

Universidad del Rosario | MACC Matemáticas Aplicadas y Ciencias de la Computación

Follow the steps of the presentation about VPC to create a VPC with two instances as represented in the next topology:



Document everything and deliver two reports in e-aulas:
i. Creation of VPC (First Report)
ii. Implementation of the Load Balancer (Second Report)

Universidad del Rosario | MACC Matemáticas Aplicadas y Ciencias de la Computación

Create a VPC

Create VPC | Actions

Filter by tags and attributes or search by keyword | 1 to 3 of 3

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP options set	Main Route table	Main Network ACL
VPC-Comp...	vpc-04c7932410c4c0947	available	10.0.0.0/16	-	dopt-f476808e	rtb-09164cdd4ba5bc58a	acl-01d73d8065a2deef2
Test-DD	vpc-04f2ccbd1e732f6cf	available	10.0.0.0/16	-	dopt-f476808e	rtb-0a3026a7a7d6bcbbd	acl-08fa98b6f3a9624bf

VPC: vpc-04c7932410c4c0947

Description | CIDR Blocks | Flow Logs | Tags

VPC ID	vpc-04c7932410c4c0947	**Tenancy**	default
State	available	**Default VPC**	No
IPv4 CIDR	10.0.0.0/16	**Classic link**	Disabled
IPv6 CIDR	-	**IPv6 Pool**	-
DNS resolution	Enabled	**Network ACL**	acl-01d73d8065a2deef2
DNS hostnames	Enabled	**DHCP options set**	dopt-f476808e
ClassicLink DNS Support	Disabled	**Route table**	rtb-09164cdd4ba5bc58a
Owner	094295054279		

AWS Load Balancing

Configure a Security Group

Requirement:

- The security group should allow servers connect to http/https servers in internet to download software or updates.
- The security group should allow servers connect to Email Servers through protocols POP3 and IMAPS
- The security group should allow servers connect to DNS servers so web pages can be requested by name and not only by IP
- The security group should allow servers connect to make connectivity test through ICMP (Ping tool)

Inbound rules | Outbound rules | Tags

Outbound rules

Type	Protocol	Port range	Destination	Description - optional
HTTP	TCP	80	0.0.0.0/0	Allow to download software
HTTP	TCP	80	::/0	Allow to download software
POP3	TCP	110	0.0.0.0/0	To allow that our servers contact Email servers
POP3	TCP	110	::/0	To allow that our servers contact Email servers
POP3S	TCP	995	0.0.0.0/0	To allow that our servers contact Email servers
POP3S	TCP	995	::/0	To allow that our servers contact Email servers
IMAPS	TCP	993	0.0.0.0/0	To allow that our servers contact Email servers
IMAPS	TCP	993	::/0	To allow that our servers contact Email servers
IMAP	TCP	143	0.0.0.0/0	To allow that our servers contact Email servers
IMAP	TCP	143	::/0	To allow that our servers contact Email servers
DNS (UDP)	UDP	53	0.0.0.0/0	To allow that our servers contact DNS servers
DNS (UDP)	UDP	53	::/0	To allow that our servers contact DNS servers
DNS (TCP)	TCP	53	0.0.0.0/0	To allow that our servers contact DNS servers
DNS (TCP)	TCP	53	::/0	To allow that our servers contact DNS servers
HTTPS	TCP	443	0.0.0.0/0	Allow to download software
HTTPS	TCP	443	::/0	Allow to download software
All ICMP - IPv4	ICMP	All	0.0.0.0/0	To allow that our servers validate connectivity
	ICMP	All	::/0	To allow that our servers validate connectivity

Inbound rules | Outbound rules | Tags

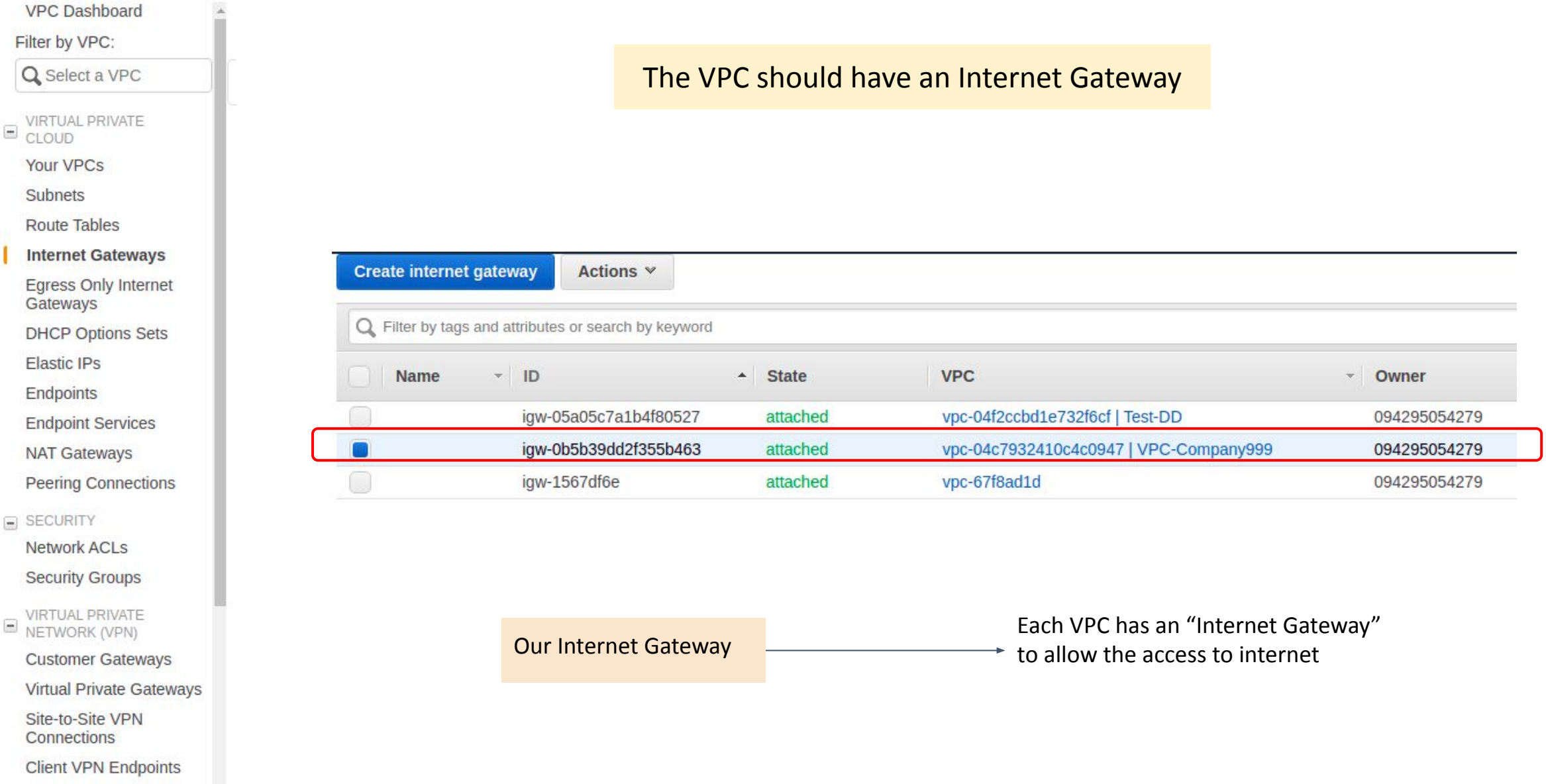
Inbound rules Edit inbound rules

Type	Protocol	Port range	Source	Description - optional
HTTP	TCP	80	0.0.0.0/0	Web Access
HTTP	TCP	80	::/0	Web Access
SSH	TCP	22	0.0.0.0/0	SSH for administration
SSH	TCP	22	::/0	SSH for administration
HTTPS	TCP	443	0.0.0.0/0	Web Access
HTTPS	TCP	443	::/0	Web Access

Requirement:

- The security group should allow servers be reachable (Inbound Rules) from internet through protocol HTTP/HTTPS and, SSH

The VPC should have an Internet Gateway



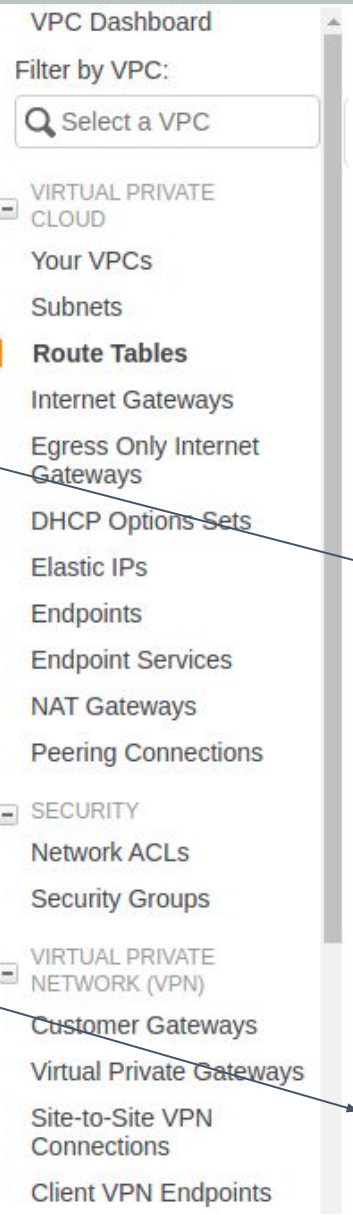
Our Internet Gateway → Each VPC has an “Internet Gateway” to allow the access to internet

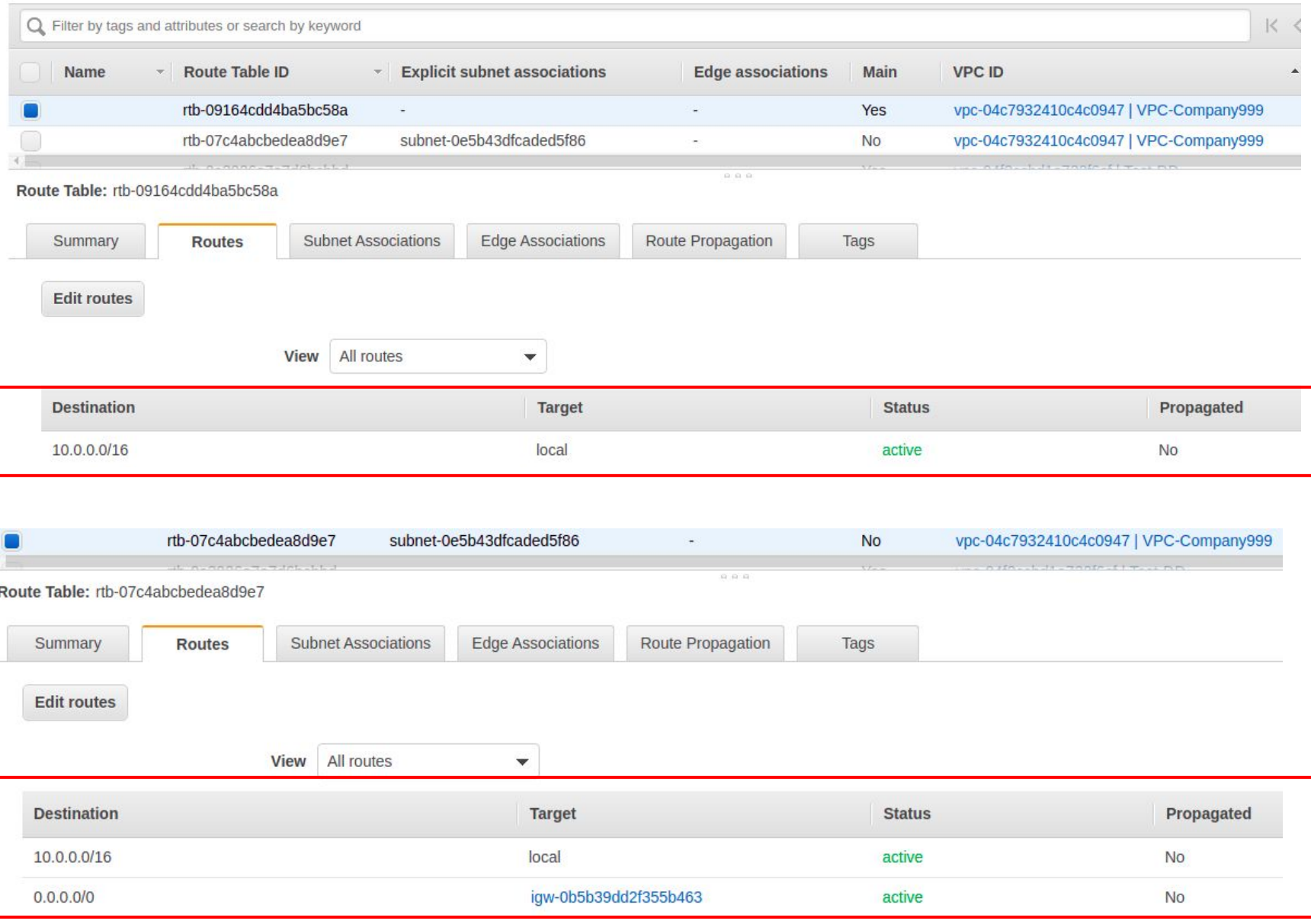
AWS Load Balancing

The VPC should have Route Tables

This route allow that instances of the subnets may communicate with other instances in the VPC

Thess routes allow that instances of the subnets may communicate with Internet



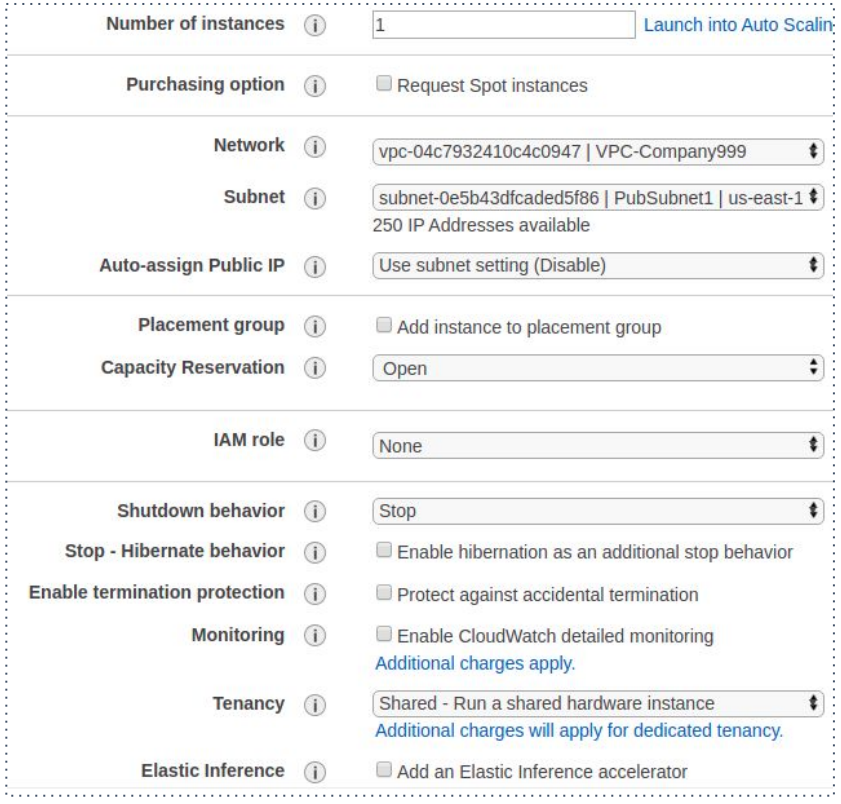


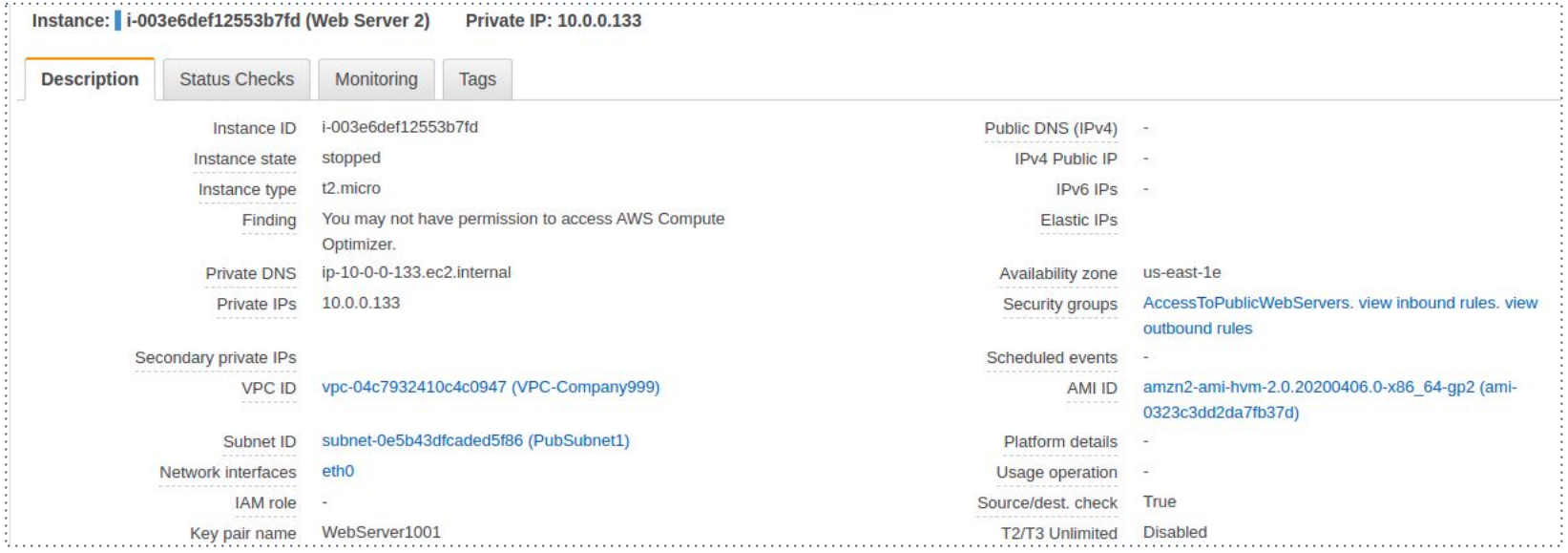
AWS Load Balancing

Create the Web servers (2)

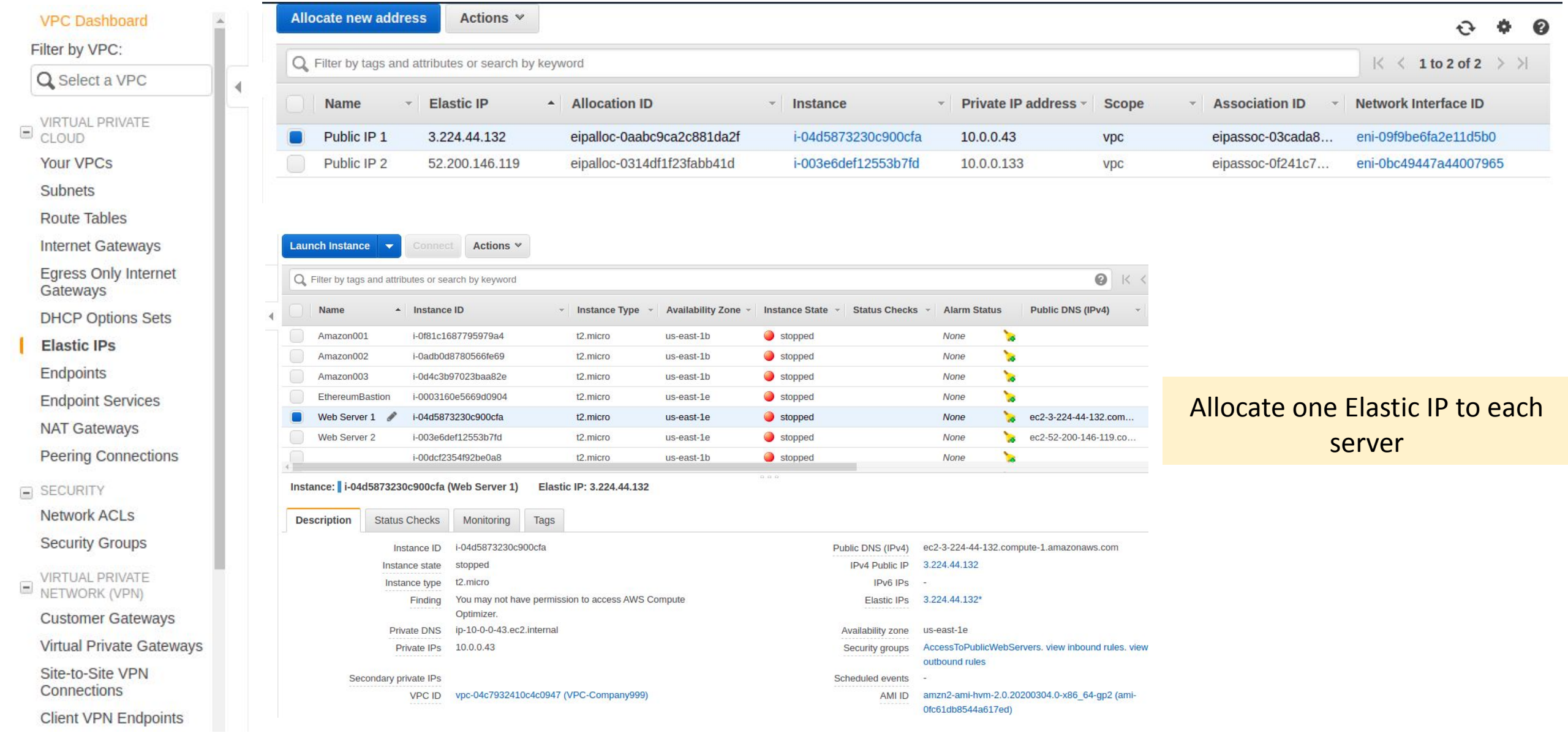
Requirements:

- Instances must be created in the same public subnet
- Both instances must be in the same security group





AWS Load Balancing



Allocate one Elastic IP to each server

Connect to each Server

```
(base) daniel@probook-440:~/Dropbox/UR/Ciberseguridad/AWS$ ssh -i "WebServer1000.pem" ec2-user@ec2-3-224-44-132.compute-1.amazonaws.com
The authenticity of host 'ec2-3-224-44-132.compute-1.amazonaws.com (3.224.44.132)' can't be established.
ECDSA key fingerprint is SHA256:J3SWkvTLNEqncer958lmO7hxx05wrw0EKwT7YSpPs+s.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-3-224-44-132.compute-1.amazonaws.com,3.224.44.132' (ECDSA) to the list of known hosts.
Last login: Thu Apr  2 21:11:01 2020 from static-190-24-131-147.static.etb.net.co

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
5 package(s) needed for security, out of 11 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-10-0-0-43 ~]$
```

```
(base) daniel@probook-440:~/Dropbox/UR/Ciberseguridad/AWS$ ssh -i "WebServer1001.pem" ec2-user@ec2-52-200-146-119.compute-1.amazonaws.com
The authenticity of host 'ec2-52-200-146-119.compute-1.amazonaws.com (52.200.146.119)' can't be established.
ECDSA key fingerprint is SHA256:gLfD+/mkZWI2HIryBUvNf6Qy/0Y6vhglV6Rj23p/Tso.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-52-200-146-119.compute-1.amazonaws.com' (ECDSA) to the list of known hosts.
Warning: the ECDSA host key for 'ec2-52-200-146-119.compute-1.amazonaws.com' differs from the key for the IP address '52.200.146.119'
Offending key for IP in /home/daniel/.ssh/known_hosts:3
Are you sure you want to continue connecting (yes/no)? yes

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-0-0-133 ~]$
```

Install a web server on each Server

Install Apache Web Server with the following commands:

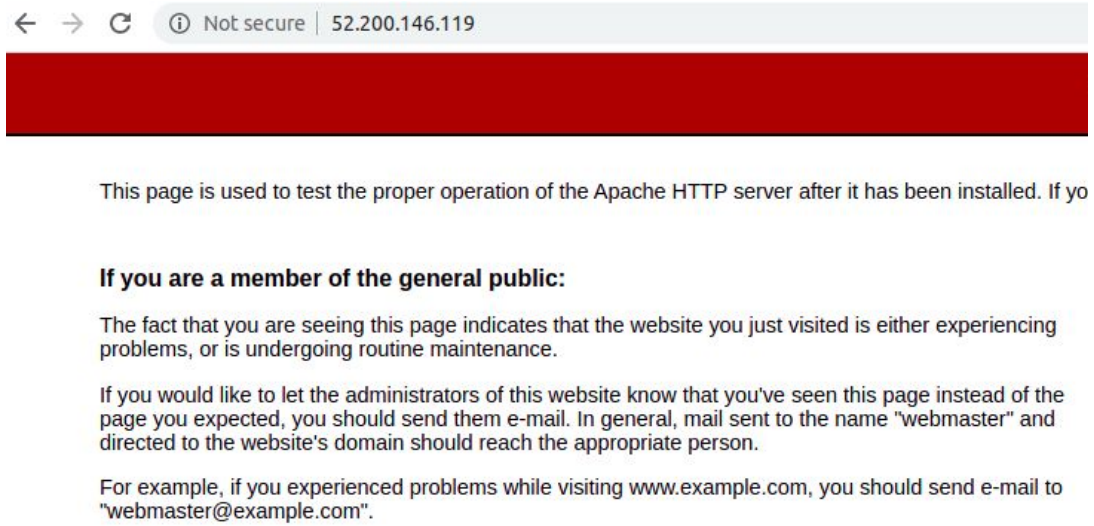
a. Connect to your EC2 instance and install the Apache web server.

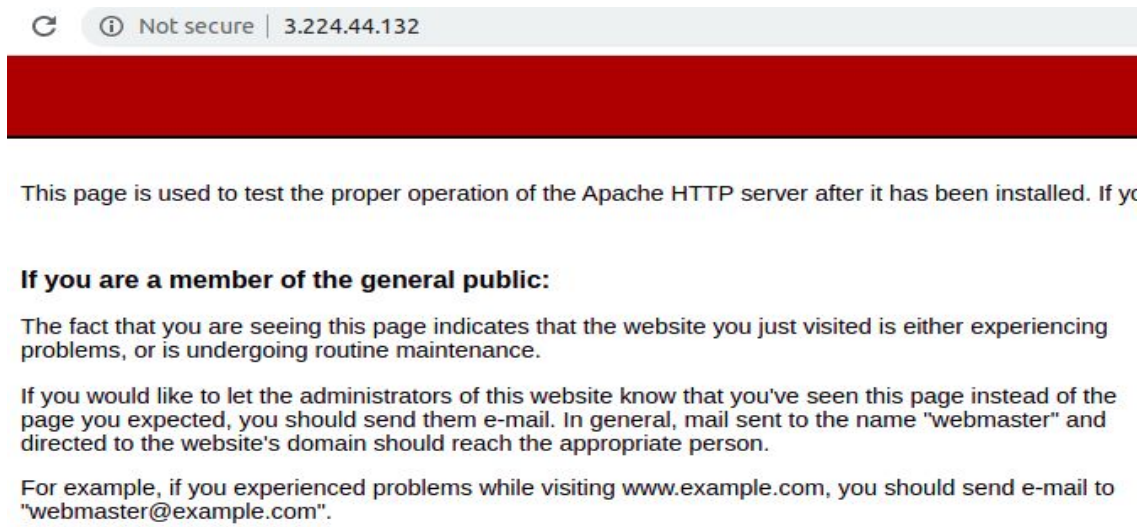
```
$ sudo yum -y install httpd
```

b. Start the service.

```
$ sudo service httpd start
```

Validate that the Web servers are working:





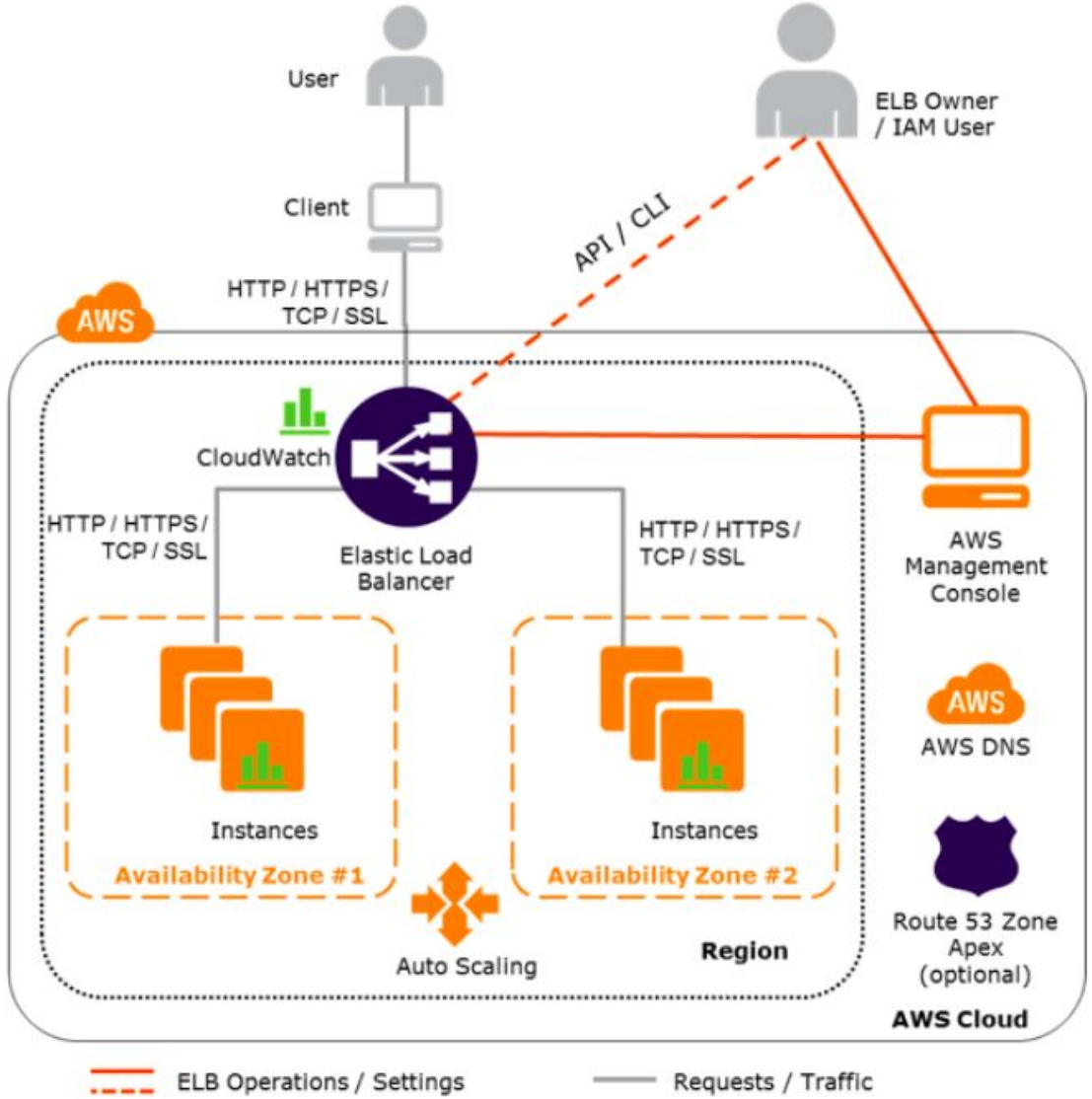
AWS Load Balancing

Universidad del Rosario | MACC Matemáticas Aplicadas y Ciencias de la Computación

Now we have our topology and we are ready to configure our Load Balancer!
(Second report)

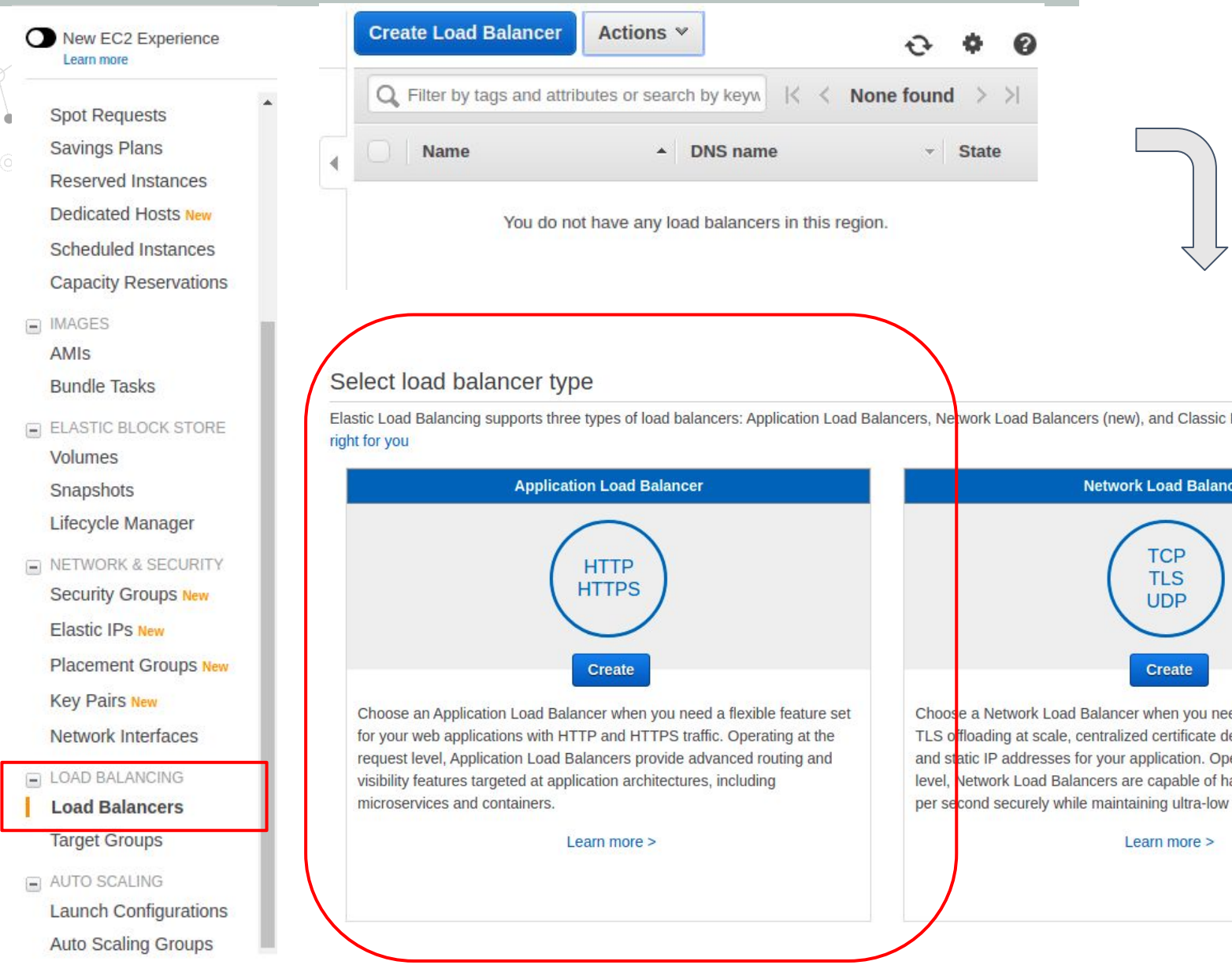
Steps:

1. Create a load balancer (Application Load Balancer or Network Load Balancer)
2. Specify a unique name and a network
3. Create listeners for your load balancer
   a. HTTP or HTTPS for Application Load Balancer
   b. TCP or UDP for Network Load Balancer
4. Configure health checks for your load balancer



AWS Load Balancing

Universidad del Rosario | MACC Matemáticas Aplicadas y Ciencias de la Computación



Basic configuration

Load balancer name

Name must be unique within your AWS account and can't be changed after the load balancer is created.

LB1

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme | Info

Scheme can't be changed after the load balancer is created.

- (•) Internet-facing
  An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. Learn more
- ( ) Internal
  An internal load balancer routes requests from clients to targets using private IP addresses.

IP address type | Info

Select the type of IP addresses that your subnets use.

- (•) IPv4
  Recommended for internal load balancers.
- ( ) Dualstack
  Includes IPv4 and IPv6 addresses.

Network mapping Info

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC Info

Select the virtual private cloud (VPC) for your targets or you can create a new VPC. Only VPCs with an internet gateway are enabled for selection. The selected VPC can't be changed after the load balancer is created. To confirm the VPC for your targets, view your target groups.

-
vpc-0129cca7353dc8539
IPv4: 172.31.0.0/16

Create new VPC

Mappings Info

Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

- [x] **us-east-1a (use1-az2)**

Subnet

subnet-0f6ab1652698fbbab

IPv4 address

Assigned by AWS

- [x] **us-east-1b (use1-az4)**

Subnet

subnet-08d07230a64d52c68

IPv4 address

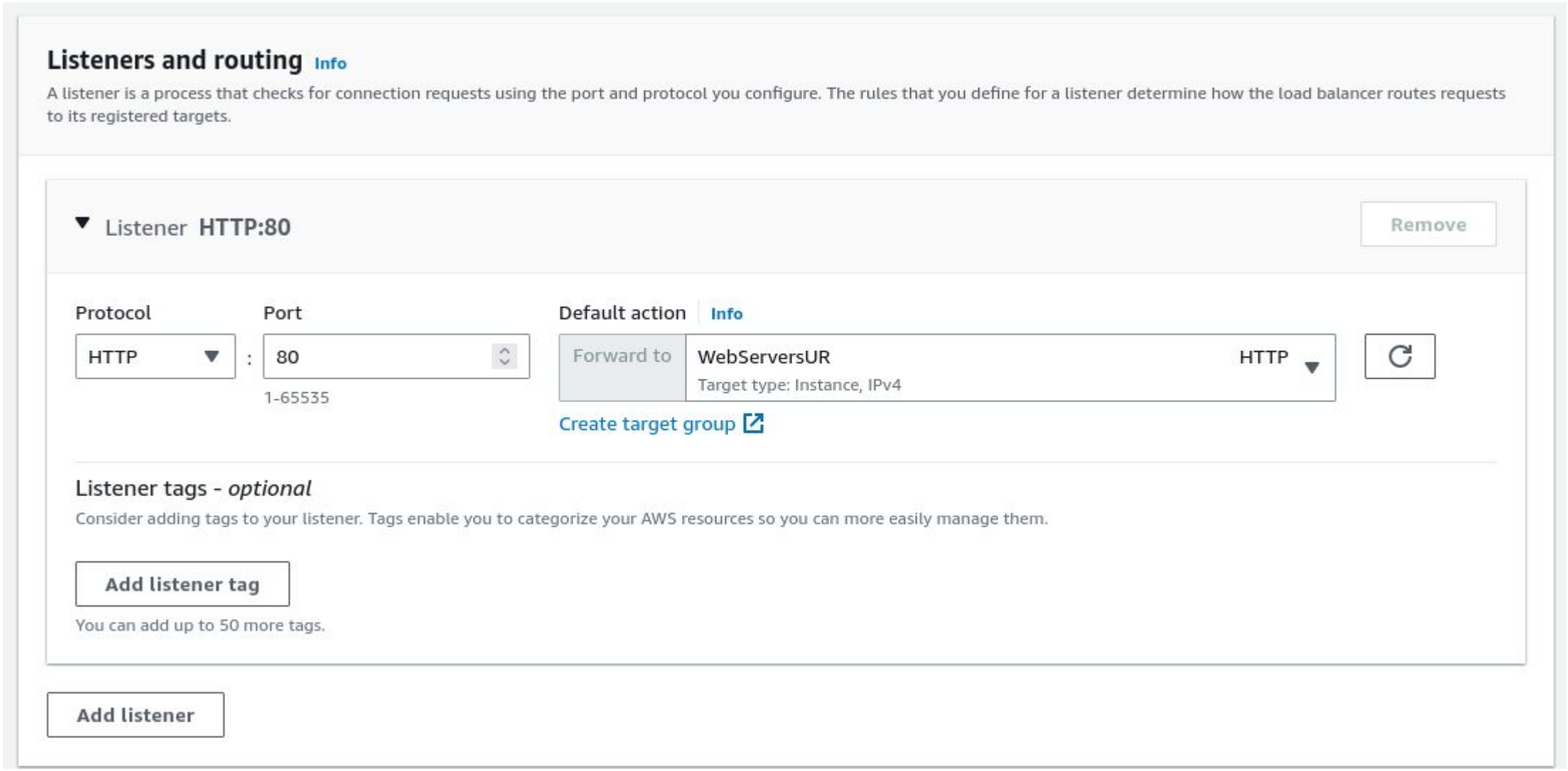
Assigned by AWS

Basic configuration

Settings in this section can't be changed after the target group is created.

Choose a target type

- (•) **Instances**
  - Supports load balancing to instances within a specific VPC.
  - Facilitates the use of Amazon EC2 Auto Scaling to manage and scale your EC2 capacity.
- ( ) **IP addresses**
  - Supports load balancing to VPC and on-premises resources.
  - Facilitates routing to multiple IP addresses and network interfaces on the same instance.
  - Offers flexibility with microservice based architectures, simplifying inter-application communication.
  - Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6 NAT.
- ( ) **Lambda function**
  - Facilitates routing to a single Lambda function.
  - Accessible to Application Load Balancers only.
- ( ) **Application Load Balancer**
  - Offers the flexibility for a Network Load Balancer to accept and route TCP requests within a specific VPC.
  - Facilitates using static IP addresses and PrivateLink with an Application Load Balancer.


Listeners and routing Info
A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.
Listener HTTP:80
Remove
Protocol
Port
Default action Info
HTTP
80
1-65535
Forward to
WebServersUR
HTTP
Target type: Instance, IPv4
Create target group
Listener tags - optional
Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.
Add listener tag
You can add up to 50 more tags.
Add listener

Target group name

WebServersUR

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Protocol: HTTP : Port: 80

1-65535

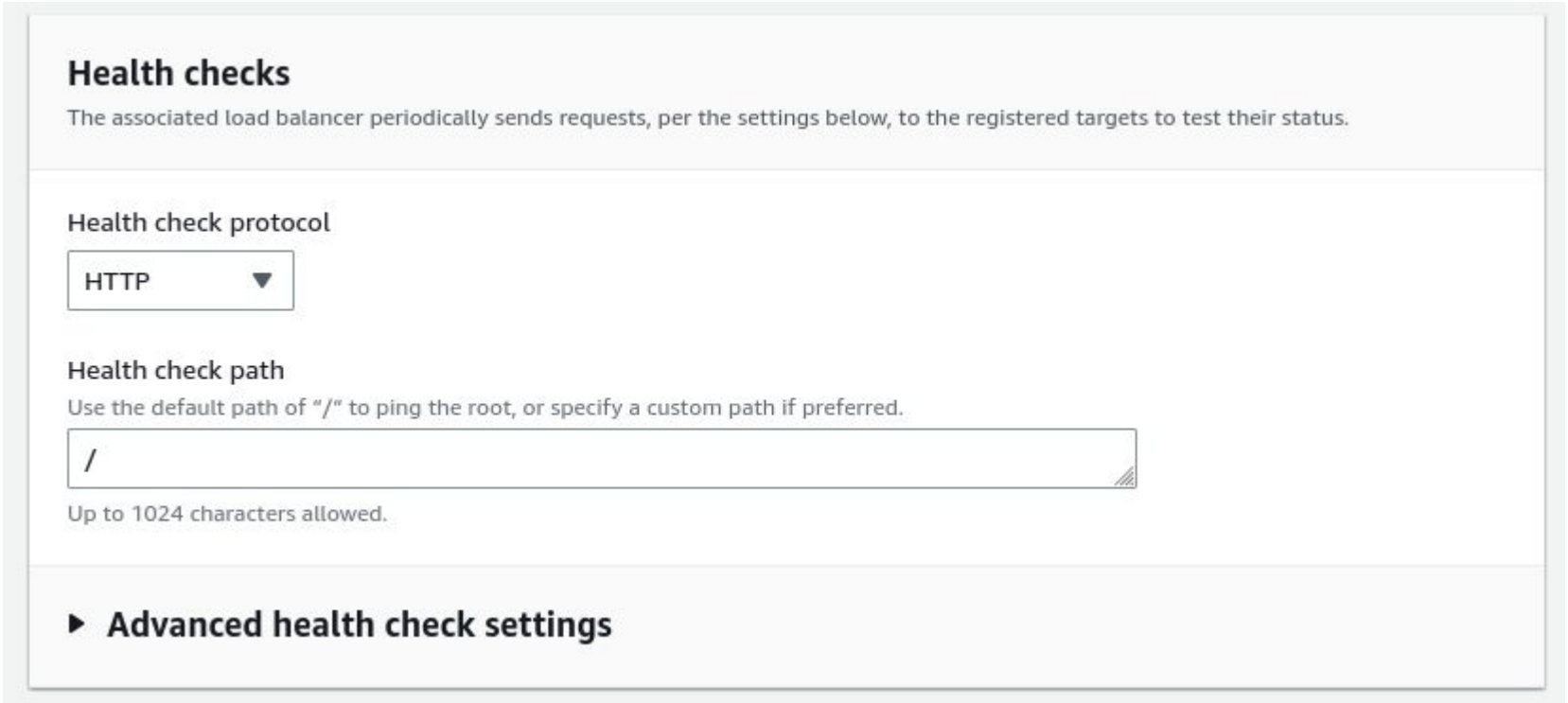
VPC

Select the VPC with the instances that you want to include in the target group.

-
vpc-0129cca7353dc8539
IPv4: 172.31.0.0/16

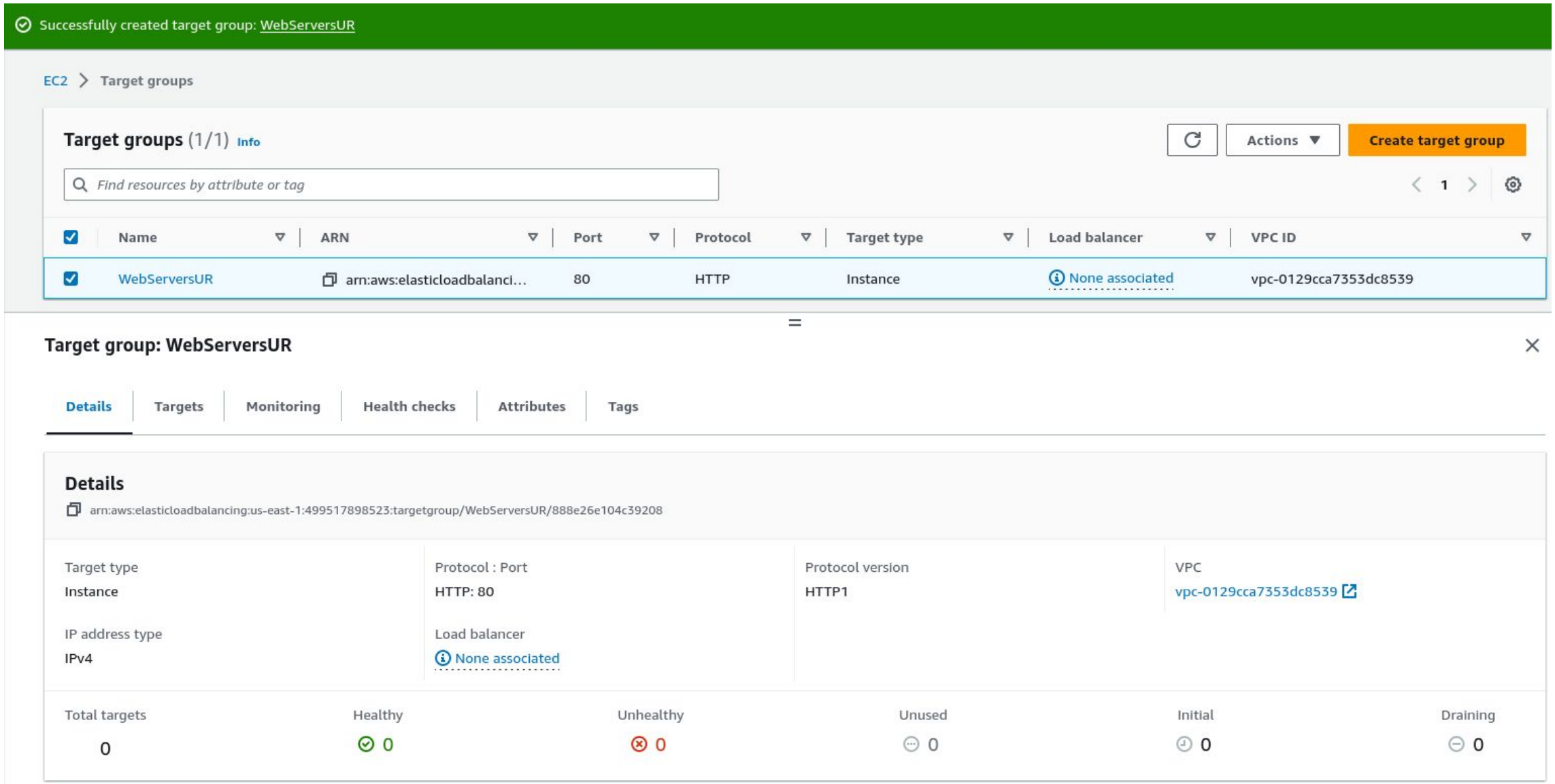
Protocol version

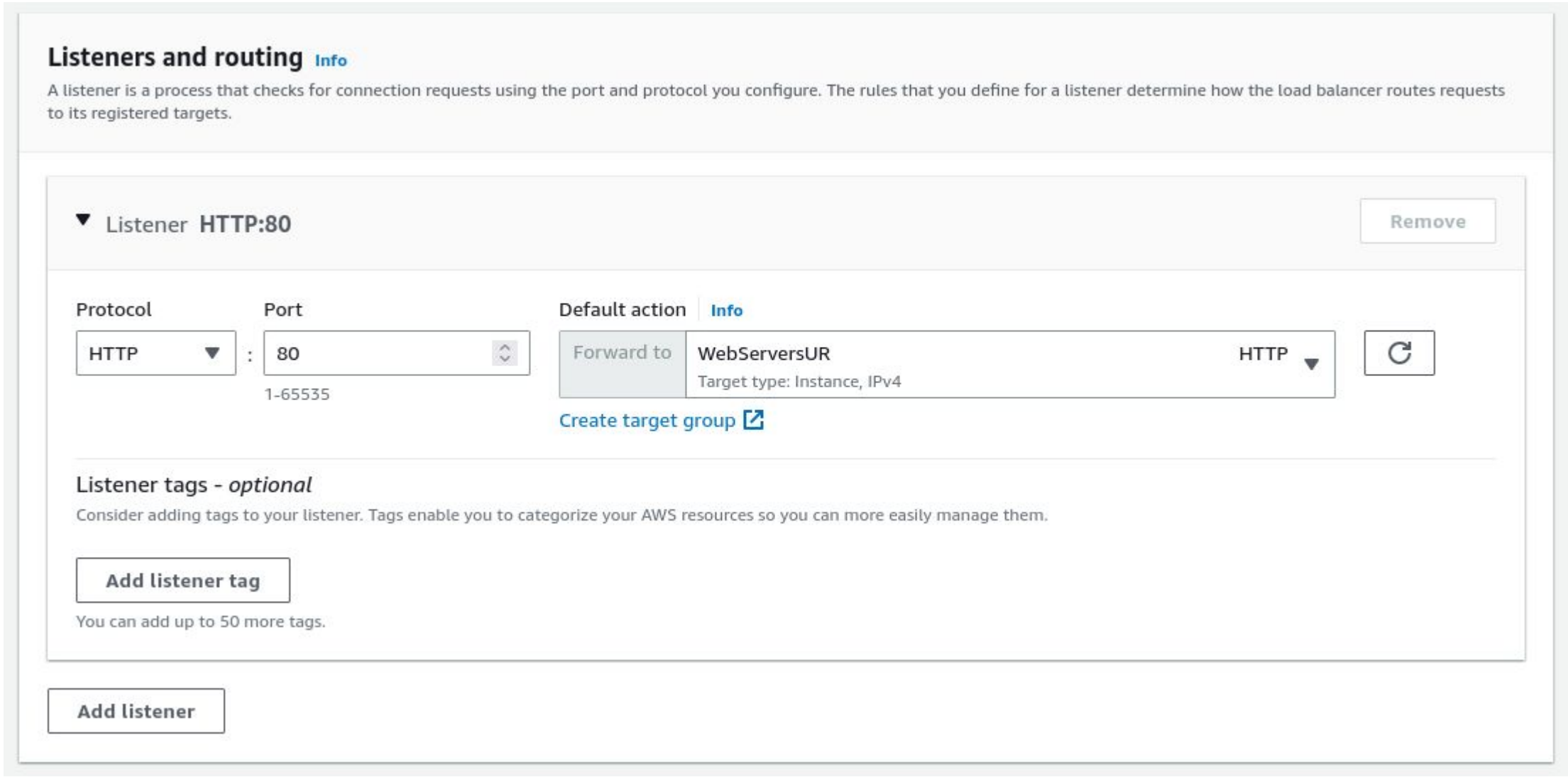
- (•) HTTP1
  Send requests to targets using HTTP/1.1. Supported when the request protocol is HTTP/1.1 or HTTP/2.
- ( ) HTTP2
  Send requests to targets using HTTP/2. Supported when the request protocol is HTTP/2 or gRPC, but gRPC-specific features are not available.
- ( ) gRPC
  Send requests to targets using gRPC. Supported when the request protocol is gRPC.


Health checks
The associated load balancer periodically sends requests, per the settings below, to the registered targets to test their status.
Health check protocol
HTTP
Health check path
Use the default path of "/" to ping the root, or specify a custom path if preferred.
/
Up to 1024 characters allowed.
Advanced health check settings

AWS Load Balancing

Universidad del Rosario | MACC Matemáticas Aplicadas y Ciencias de la Computación




Listeners and routing Info
A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.
Listener HTTP:80
Remove
Protocol
Port
HTTP
:
80
1-65535
Default action Info
Forward to
WebServersUR
Target type: Instance, IPv4
HTTP
Create target group
Listener tags - optional
Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.
Add listener tag
You can add up to 50 more tags.
Add listener

Summary

Review and confirm your configurations. Estimate cost

Basic configuration Edit

LB1

- Internet-facing
- IPv4

Security groups Edit

- default
  sg-01572dc9cb02ad66f

Network mapping Edit

VPC vpc-0129cca7353dc8539

- us-east-1a
  subnet-0f6ab1652698fbbab
- us-east-1b
  subnet-08d07230a64d52c68

Listeners and routing Edit

- HTTP:80 defaults to
  WebServersUR

Add-on services Edit

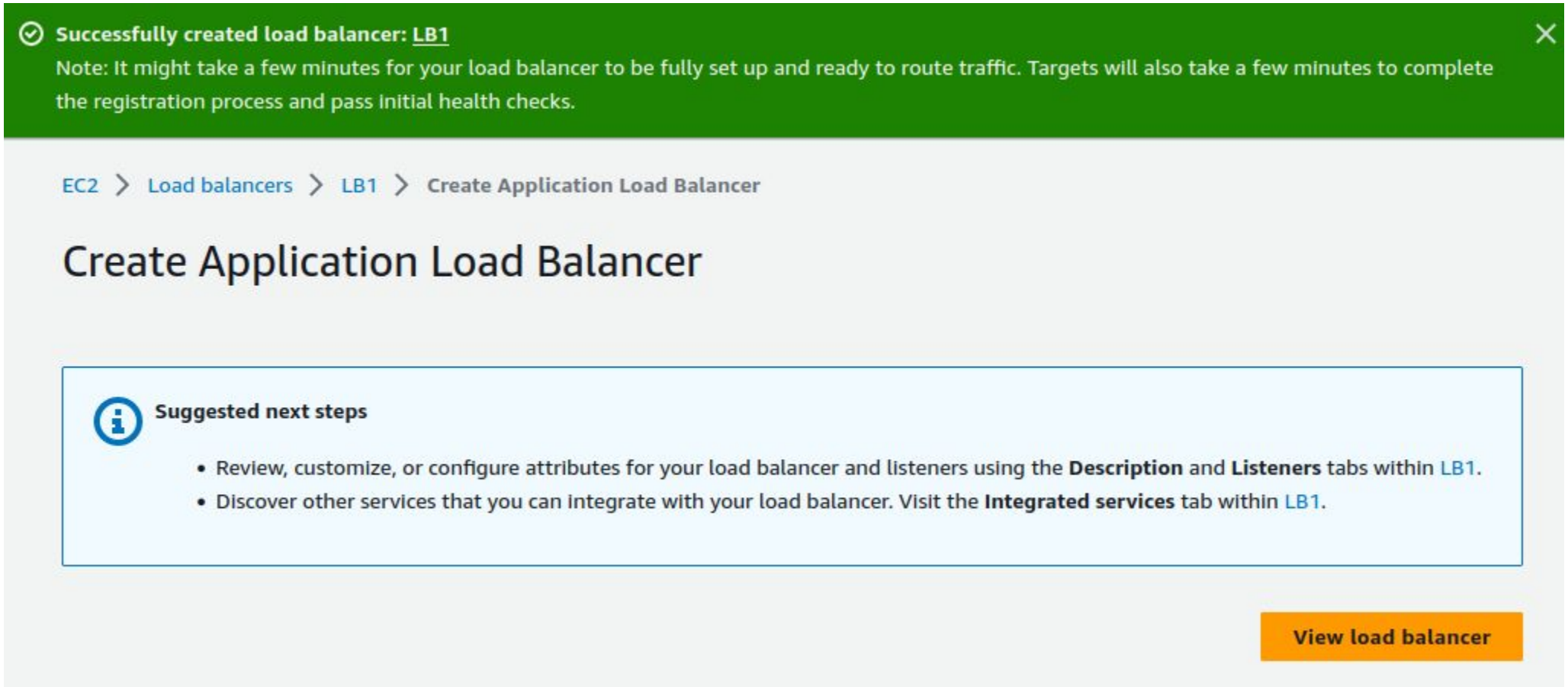
None

Tags Edit

None

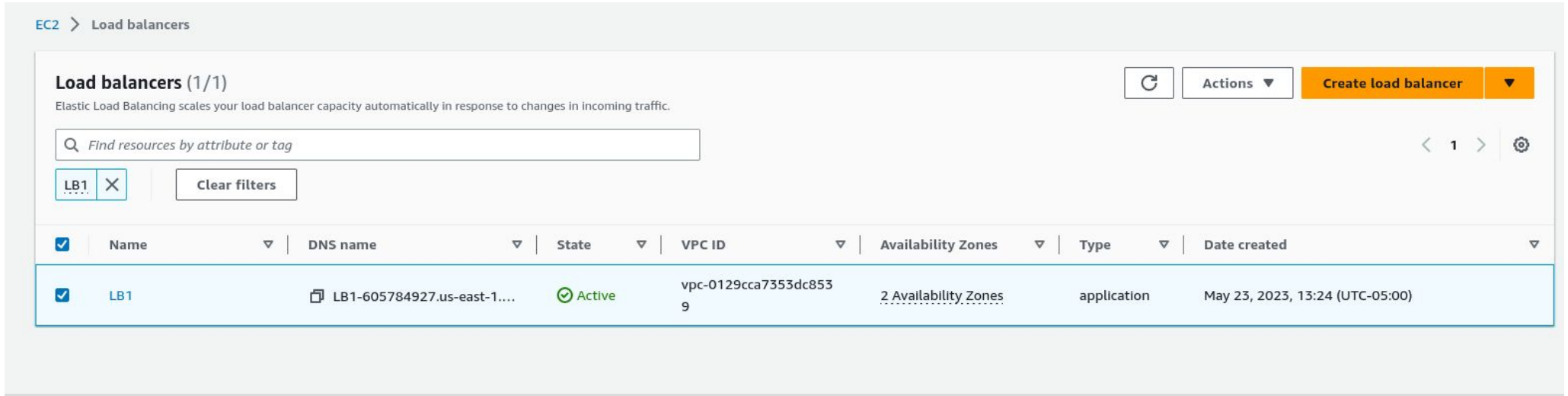
Attributes

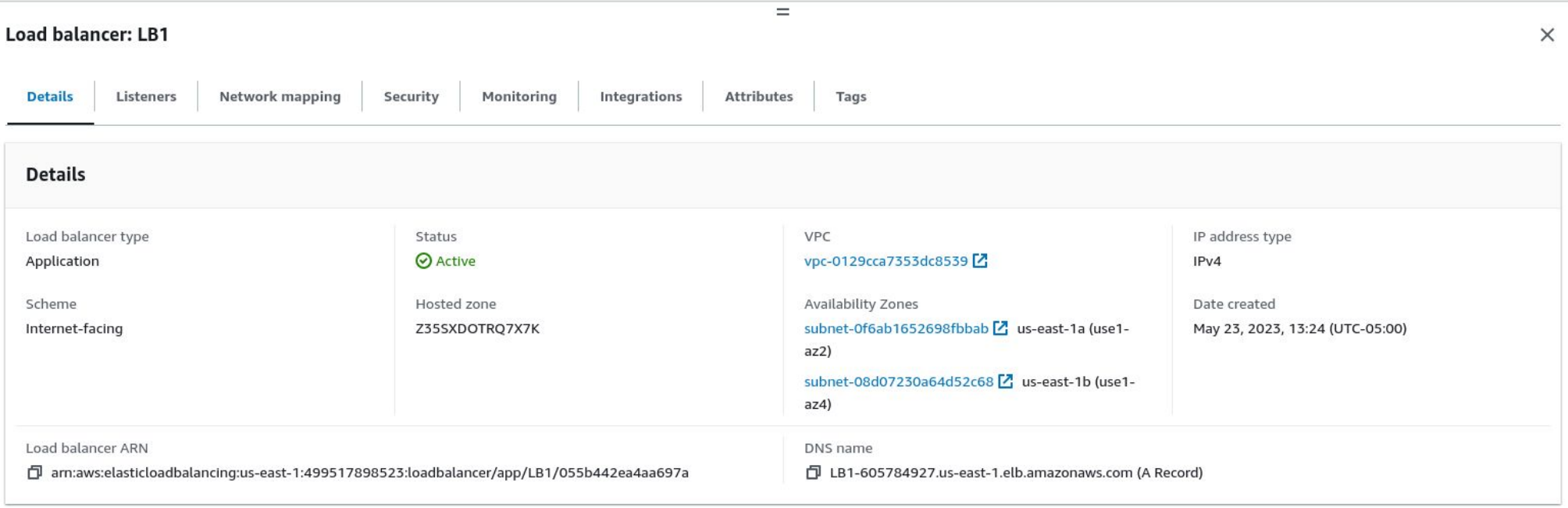
Certain default attributes will be applied to your load balancer. You can view and edit them after creating the load balancer.

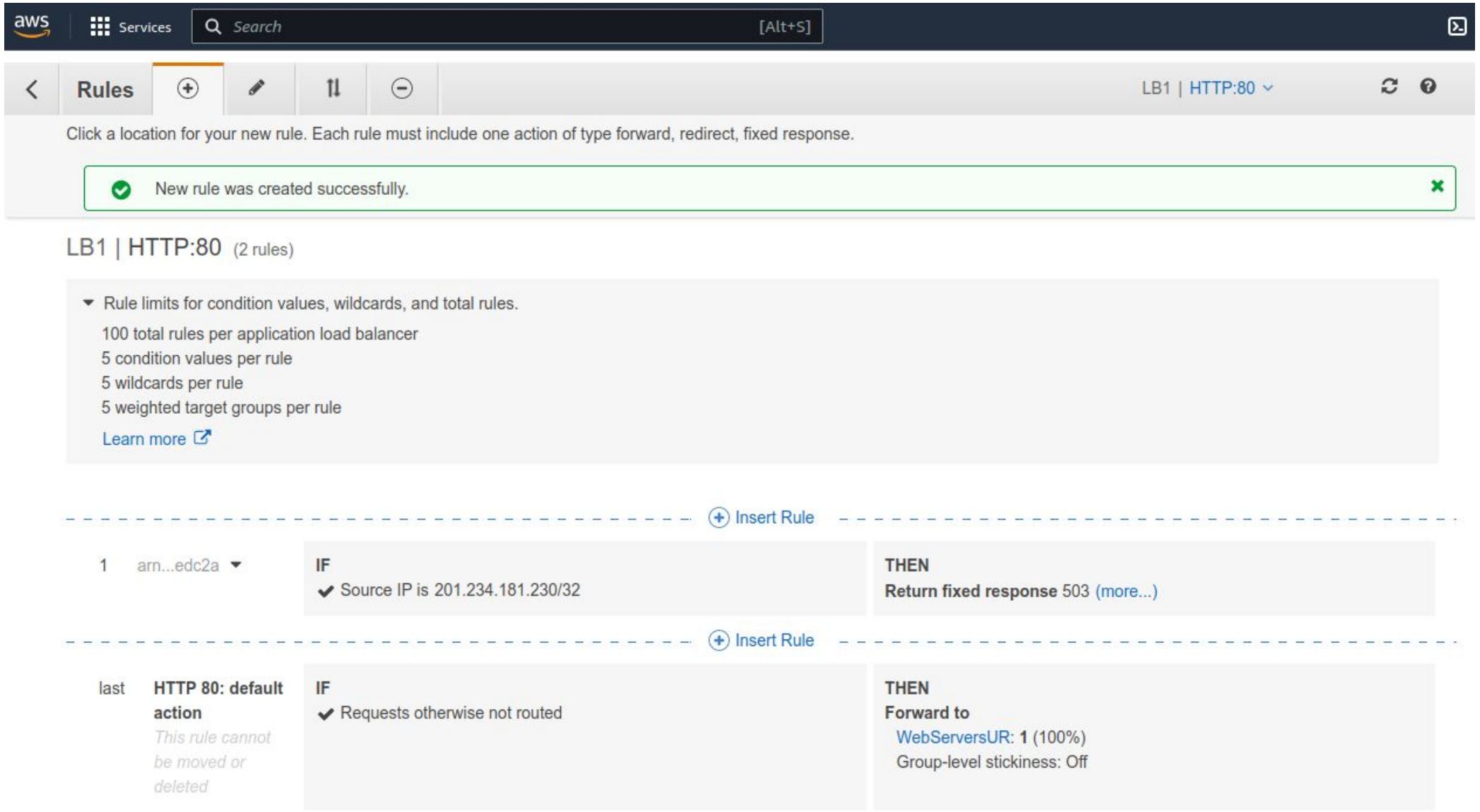

Successfully created load balancer: LB1
Note: It might take a few minutes for your load balancer to be fully set up and ready to route traffic. Targets will also take a few minutes to complete the registration process and pass initial health checks.
EC2 > Load balancers > LB1 > Create Application Load Balancer
Create Application Load Balancer
Suggested next steps
• Review, customize, or configure attributes for your load balancer and listeners using the Description and Listeners tabs within LB1.
• Discover other services that you can integrate with your load balancer. Visit the Integrated services tab within LB1.
View load balancer

AWS Load Balancing

Universidad del Rosario | MACC Matemáticas Aplicadas y Ciencias de la Computación






aws
Services
Search
[Alt+S]
Rules
LB1 | HTTP:80
Click a location for your new rule. Each rule must include one action of type forward, redirect, fixed response.
New rule was created successfully.
LB1 | HTTP:80 (2 rules)
Rule limits for condition values, wildcards, and total rules.
100 total rules per application load balancer
5 condition values per rule
5 wildcards per rule
5 weighted target groups per rule
Learn more
Insert Rule
1
arn...edc2a
IF
Source IP is 201.234.181.230/32
THEN
Return fixed response 503 (more...)
Insert Rule
last
HTTP 80: default action
This rule cannot be moved or deleted
IF
Requests otherwise not routed
THEN
Forward to
WebServersUR: 1 (100%)
Group-level stickiness: Off

Universidad del Rosario | MACC | HINNT

¡Gracias!