46°

Lab - AWS re/Start

Uso de escalado automático en AWS (Linux)

Amazon EC2 Auto Scaling

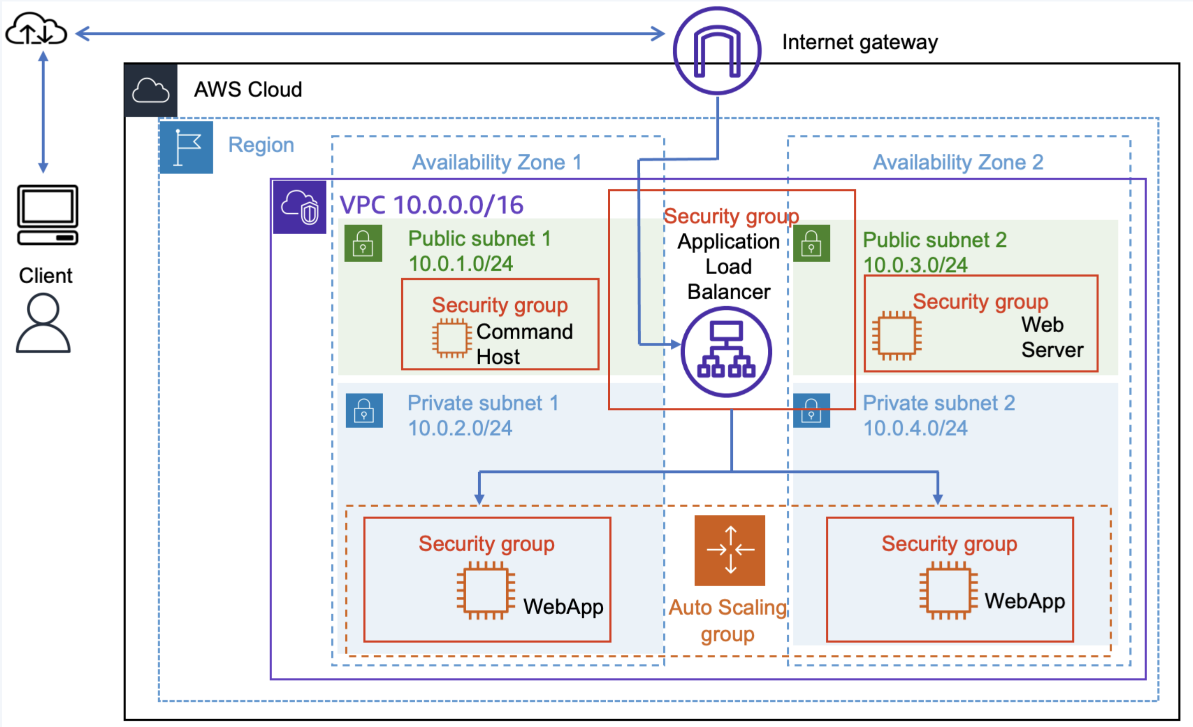
SWIPE

Tarea 01

Interactuando con Arquitecturas

Los objetivos son:

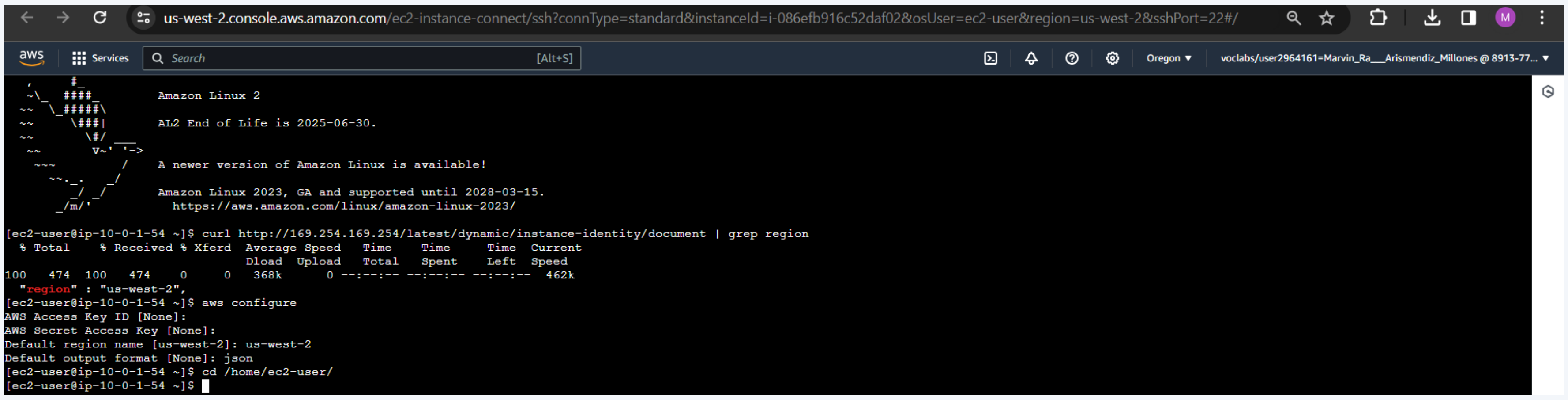
- Crear una instancia EC2 mediante un comando de la CLI de AWS.
- Crear una nueva AMI mediante la CLI de AWS.
- Crear una plantilla de lanzamiento de Amazon EC2.
- Cree una configuración de lanzamiento de Amazon EC2 Auto Scaling.
- Configure las políticas de escalado y cree un grupo de Auto Scaling para aumentar y reducir el número de servidores en función de una carga variable.



SWIPE

Tarea 01

Empezamos creando la AMI de la instancia *Command Host* desde la CLI de AWS

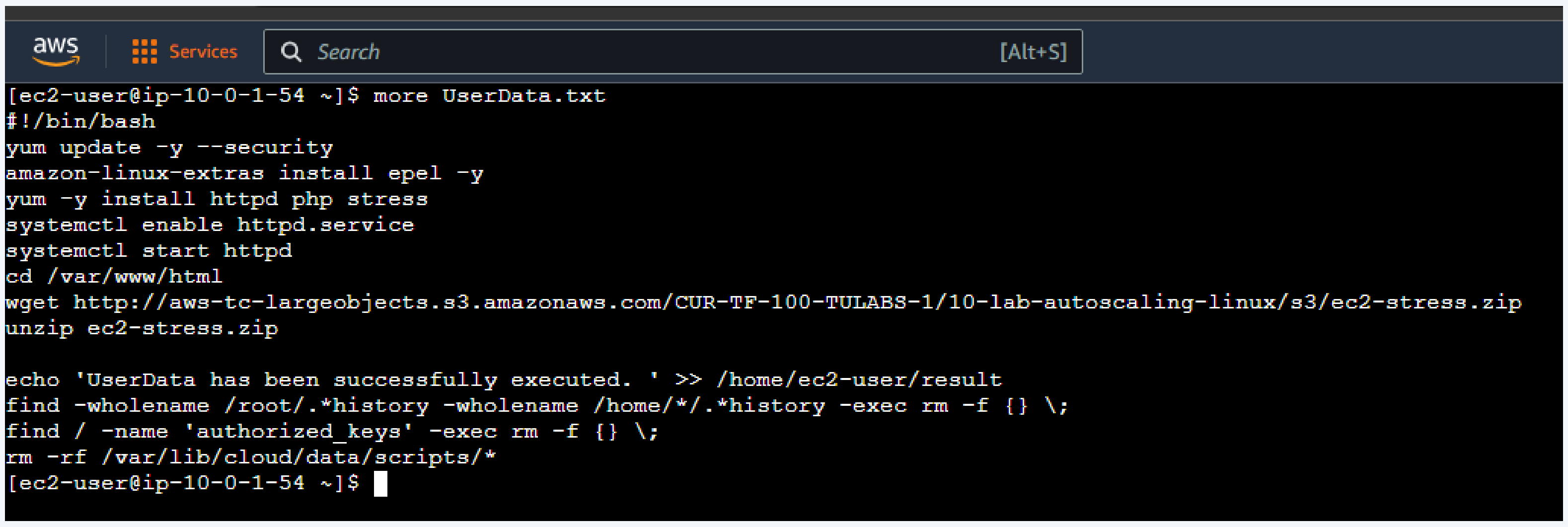


us-west-2.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-086efb916c52daf02&osUser=ec2-user&region=us-west-2&sshPort=22#/
aws
Services
Search
[Alt+S]
Oregon ▾
voclabs/user2964161=Marvin_Ra__Arismendiz_Millones @ 8913-77... ▾

```
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2025-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-10-0-1-54 ~]$ curl http://169.254.169.254/latest/dynamic/instance-identity/document | grep region
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   474  100   474    0     0   368k      0 --:--:-- --:--:-- --:--:--  462k
  "region" : "us-west-2",
[ec2-user@ip-10-0-1-54 ~]$ aws configure
AWS Access Key ID [None]:
AWS Secret Access Key [None]:
Default region name [us-west-2]: us-west-2
Default output format [None]: json
[ec2-user@ip-10-0-1-54 ~]$ cd /home/ec2-user/
[ec2-user@ip-10-0-1-54 ~]$ 
```

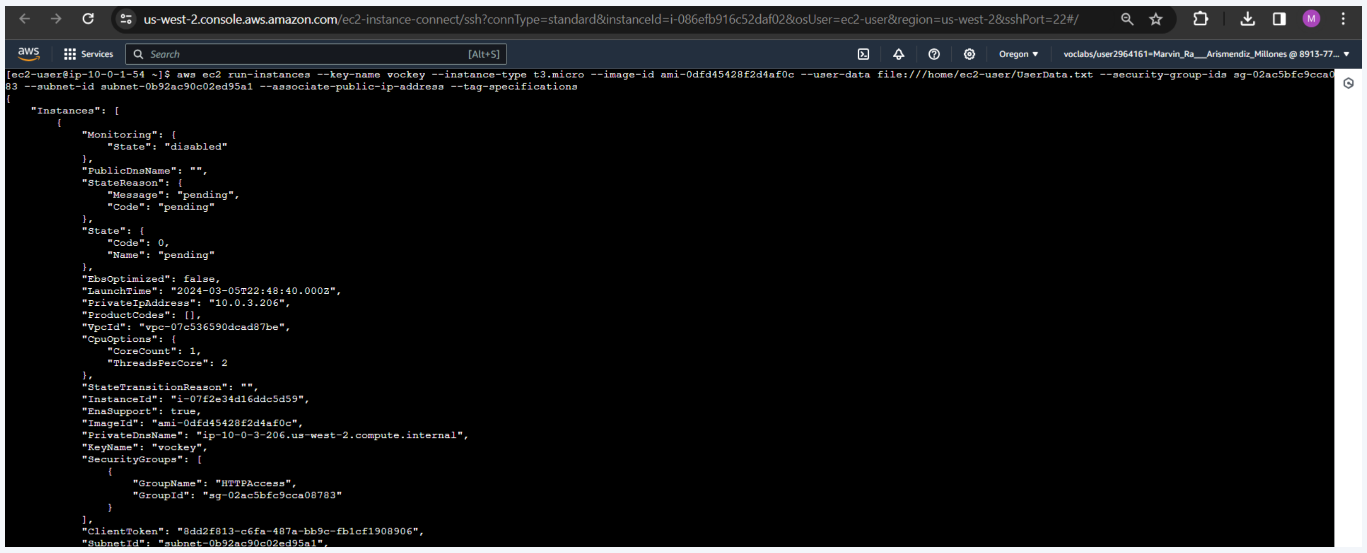
Primero, crearemos una nueva instancia, a continuación se muestra la User Data para esta instancia a crear



```
[ec2-user@ip-10-0-1-54 ~]$ more UserData.txt
#!/bin/bash
yum update -y --security
amazon-linux-extras install epel -y
yum -y install httpd php stress
systemctl enable httpd.service
systemctl start httpd
cd /var/www/html
wget http://aws-tc-largeobjects.s3.amazonaws.com/CUR-TF-100-TULABS-1/10-lab-autoscaling-linux/s3/ec2-stress.zip
unzip ec2-stress.zip

echo 'UserData has been successfully executed. ' >> /home/ec2-user/result
find -wholename /root/.*history -wholename /home/*/.*history -exec rm -f {} \;
find / -name 'authorized_keys' -exec rm -f {} \;
rm -rf /var/lib/cloud/data/scripts/*
[ec2-user@ip-10-0-1-54 ~]$ 
```

Lanzamos la nueva instancia



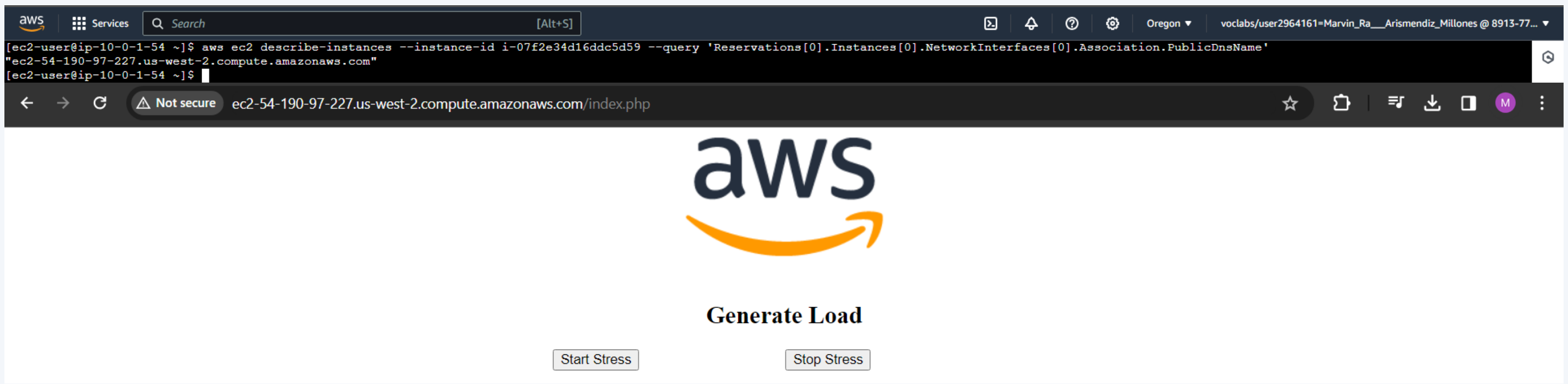
us-west-2.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-086efb916c52daf02&osUser=ec2-user&region=us-west-2&sshPort=22#/
aws
Services
Search
[Alt+S]
Oregon ▾
voclabs/user2964161=Marvin_Ra__Arismendiz_Millones @ 8913-77... ▾

```
[ec2-user@ip-10-0-1-54 ~]$ aws ec2 run-instances --key-name vockey --instance-type t3.micro --image-id ami-0dfd45428f2d4af0c --user-data file:///home/ec2-user/UserData.txt --security-group-ids sg-02ac5bfc9cca0
83 --subnet-id subnet-0b92ac90c02ed95a1 --associate-public-ip-address --tag-specifications
{
    "Instances": [
        {
            "Monitoring": {
                "State": "disabled"
            },
            "PublicDnsName": "",
            "StateReason": {
                "Message": "pending",
                "Code": "pending"
            },
            "State": {
                "Code": 0,
                "Name": "pending"
            },
            "EbsOptimized": false,
            "LaunchTime": "2024-03-05T22:48:40.000Z",
            "PrivateIpAddress": "10.0.3.206",
            "ProductCodes": [],
            "VpcId": "vpc-07c536590dcad87be",
            "CpuOptions": {
                "CoreCount": 1,
                "ThreadsPerCore": 2
            },
            "StateTransitionReason": "",
            "InstanceId": "i-07f2e34d16ddc5d59",
            "EnaSupport": true,
            "ImageId": "ami-0dfd45428f2d4af0c",
            "PrivateDnsName": "ip-10-0-3-206.us-west-2.compute.internal",
            "KeyName": "vockey",
            "SecurityGroups": [
                {
                    "GroupName": "HTTPAccess",
                    "GroupId": "sg-02ac5bfc9cca08783"
                }
            ],
            "ClientToken": "8dd2f813-c6fa-487a-bb9c-fb1cf1908906",
            "SubnetId": "subnet-0b92ac90c02ed95a1",
```

SWIPE

Tarea 01

Ahora, veamos esta nueva instancia:



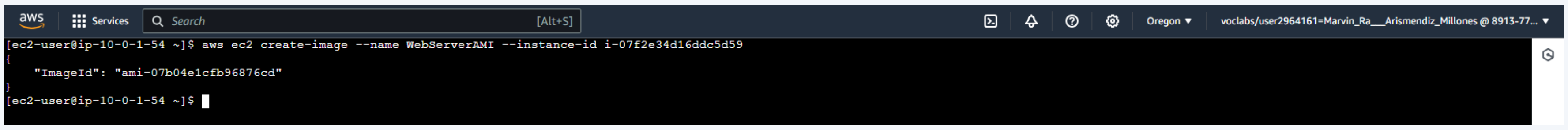
```
[ec2-user@ip-10-0-1-54 ~]$ aws ec2 describe-instances --instance-id i-07f2e34d16ddc5d59 --query 'Reservations[0].Instances[0].NetworkInterfaces[0].Association.PublicDnsName'
"ec2-54-190-97-227.us-west-2.compute.amazonaws.com"
[ec2-user@ip-10-0-1-54 ~]$
```

Services
Search
[Alt+S]
Oregon
voclabs/user2964161=Marvin_Ra___Arismendiz_Millones @ 8913-77...
Not secure
ec2-54-190-97-227.us-west-2.compute.amazonaws.com/index.php

Generate Load

Start Stress | Stop Stress

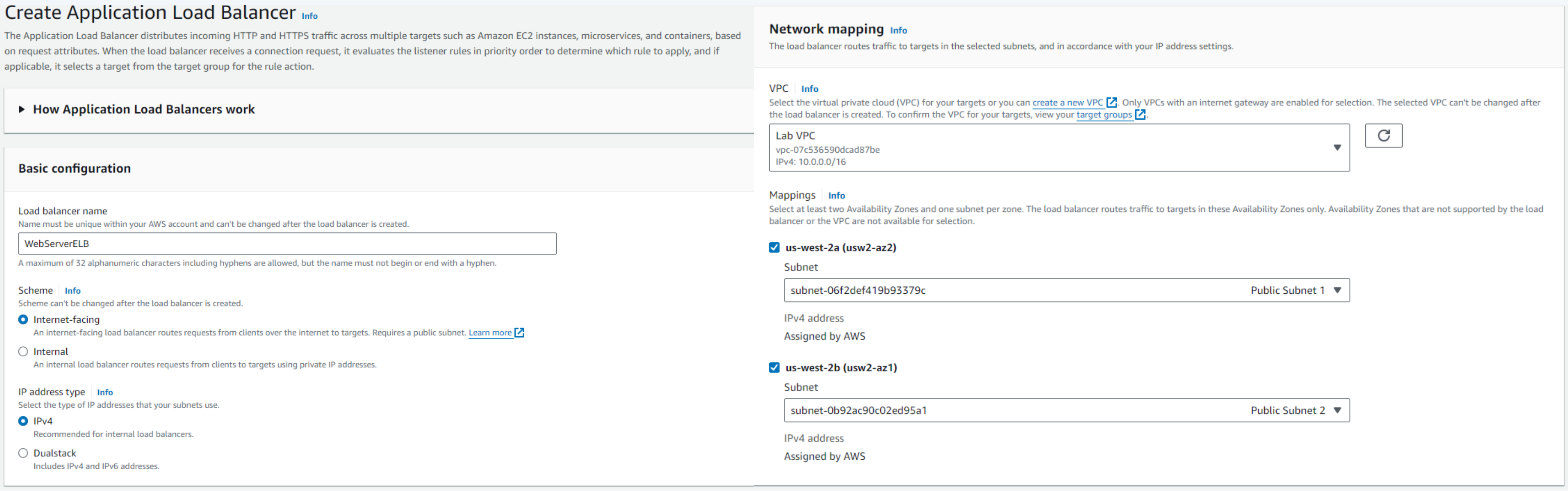
Luego, procedemos a crear una AMI de esta instancia



```
[ec2-user@ip-10-0-1-54 ~]$ aws ec2 create-image --name WebServerAMI --instance-id i-07f2e34d16ddc5d59
{
    "ImageId": "ami-07b04e1cfb96876cd"
}
[ec2-user@ip-10-0-1-54 ~]$
```

Services
Search
[Alt+S]
Oregon
voclabs/user2964161=Marvin_Ra___Arismendiz_Millones @ 8913-77...

Después de ello, procedemos a crear un ELB:



Create Application Load Balancer Info

The Application Load Balancer distributes incoming HTTP and HTTPS traffic across multiple targets such as Amazon EC2 instances, microservices, and containers, based on request attributes. When the load balancer receives a connection request, it evaluates the listener rules in priority order to determine which rule to apply, and if applicable, it selects a target from the target group for the rule action.

▸ How Application Load Balancers work

Basic configuration

Load balancer name
Name must be unique within your AWS account and can't be changed after the load balancer is created.
WebServerELB
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.
Lab VPC
vpc-07c536590dcad87be
IPv4: 10.0.0.0/16

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.

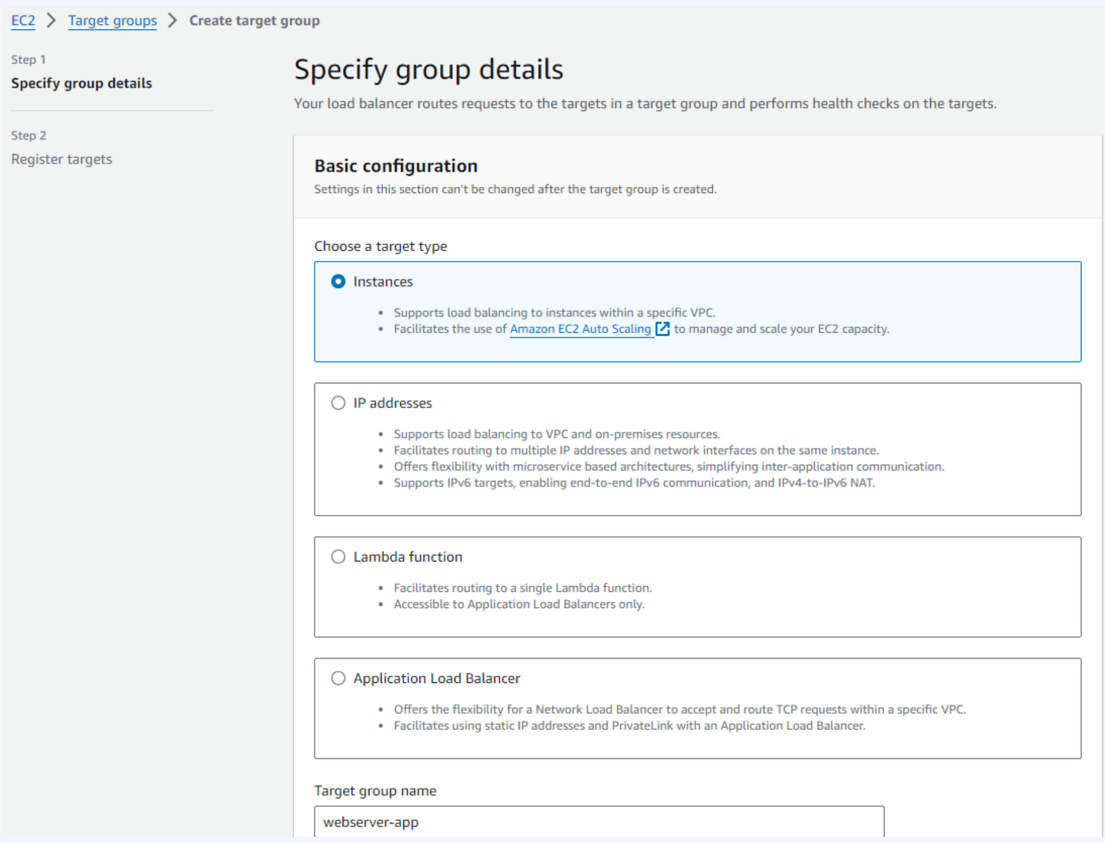
☑ us-west-2a (usw2-az2)
Subnet
subnet-06f2def419b93379c Public Subnet 1
IPv4 address
Assigned by AWS

☑ us-west-2b (usw2-az1)
Subnet
subnet-0b92ac90c02ed95a1 Public Subnet 2
IPv4 address
Assigned by AWS

SWIPE

Tarea 01

Más de la configuración del ALB (Application Load Balancer)



EC2 > Target groups > Create target group

Step 1
Specify group details

Step 2
Register targets

Specify group details

Your load balancer routes requests to the targets in a target group and performs health checks on the targets.

Basic configuration

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

Choose a target type

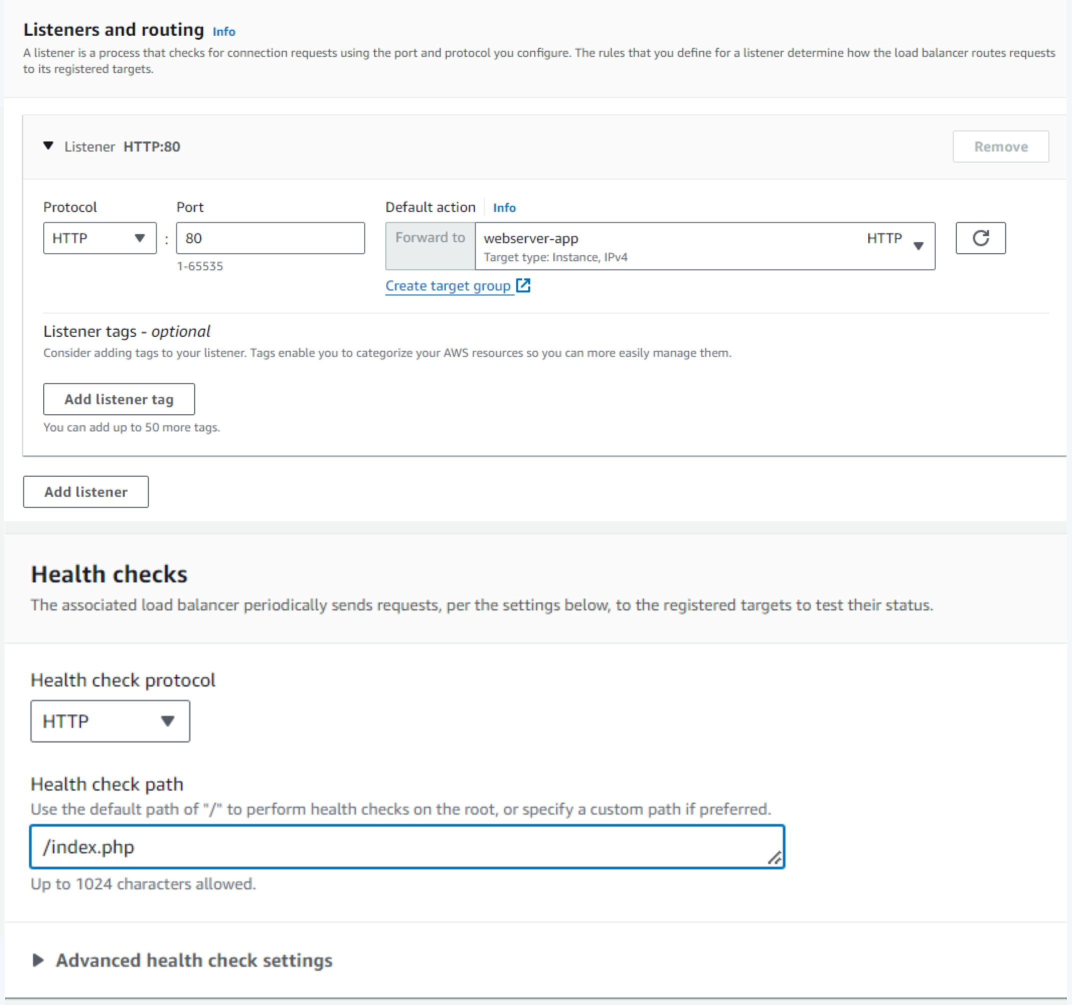
● Instances
- Supports load balancing to instances within a specific VPC.
- Facilitates the use of Amazon EC2 Auto Scaling to manage and scale your EC2 capacity.

○ 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.

Target group name: webserver-app



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**

Protocol: HTTP
Port: 80 (1-65535)
Default action: Forward to webserver-app HTTP
Target type: Instance, IPv4

Listener tags - *optional*

Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

You can add up to 50 more tags.

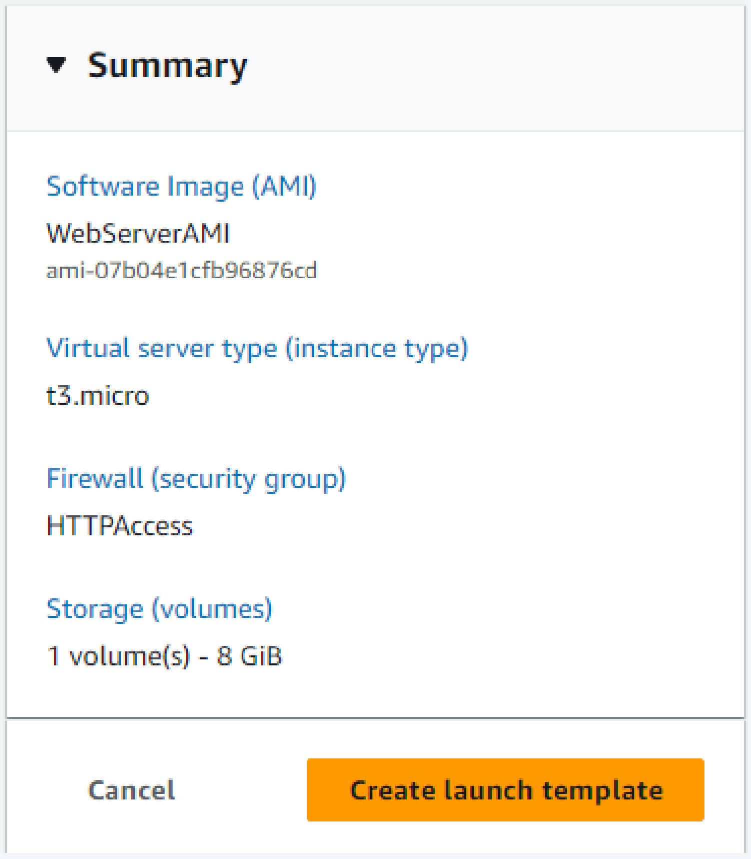
Health checks

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 perform health checks on the root, or specify a custom path if preferred.
/index.php
Up to 1024 characters allowed.

Luego, procedemos a crear la plantilla de lanzamiento



Summary

Software Image (AMI)
WebServerAMI
ami-07b04e1cfb96876cd

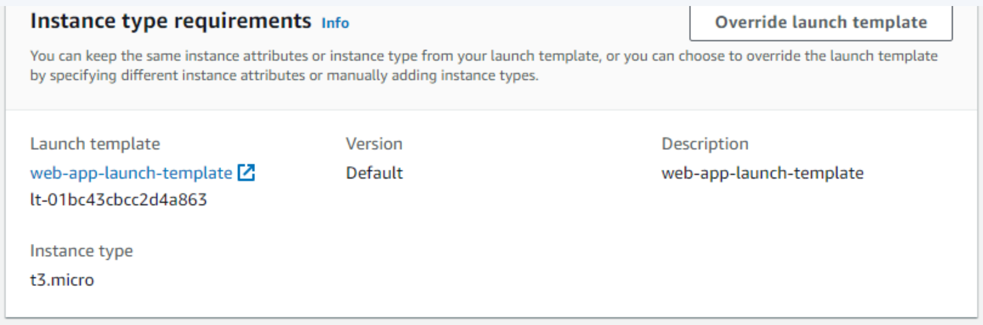
Virtual server type (instance type)
t3.micro

Firewall (security group)
HTTPAccess

Storage (volumes)
1 volume(s) - 8 GiB

Tarea 01

Luego creamos el ASG (Auto Scaling Group)



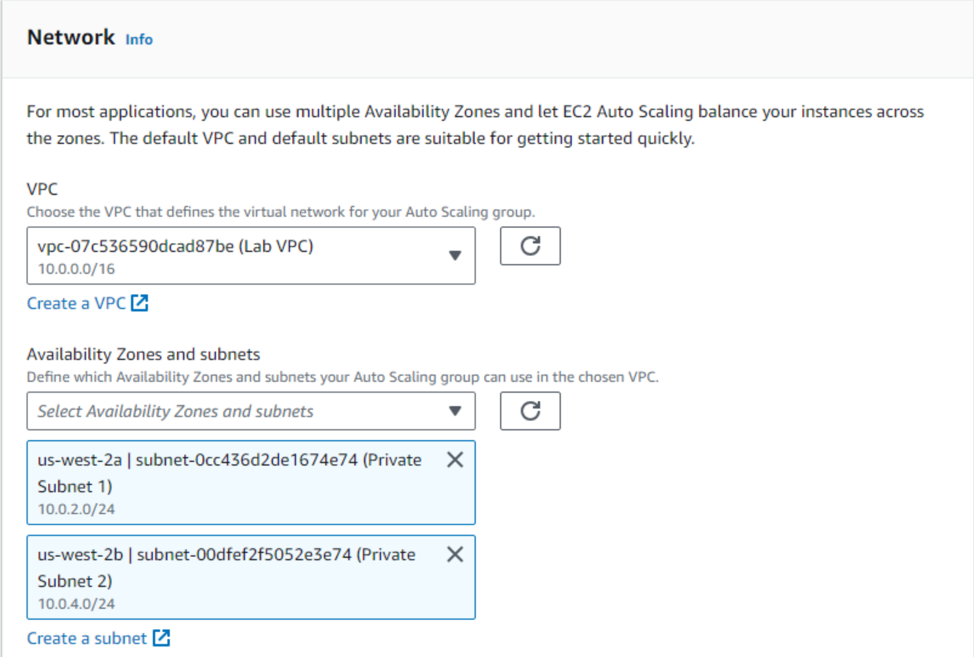
Instance type requirements Info

Override launch template

You can keep the same instance attributes or instance type from your launch template, or you can choose to override the launch template by specifying different instance attributes or manually adding instance types.

Launch template	Version	Description
web-app-launch-template lt-01bc43cbcc2d4a863	Default	web-app-launch-template

Instance type
t3.micro



Network Info

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

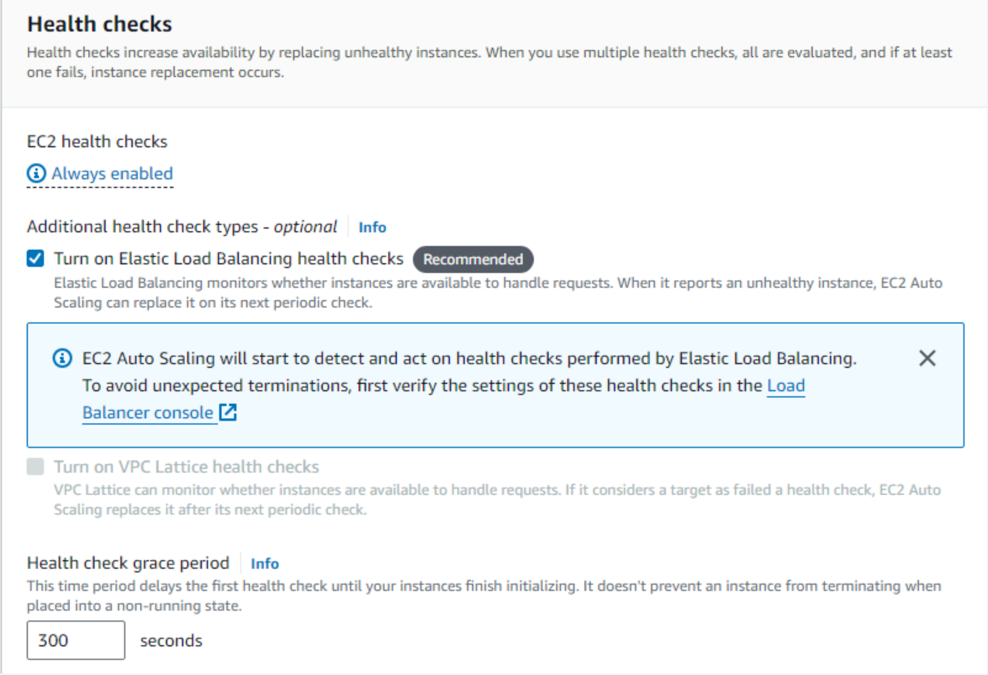
VPC
Choose the VPC that defines the virtual network for your Auto Scaling group.
vpc-07c536590dcad87be (Lab VPC)
10.0.0.0/16
Create a VPC

Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.
Select Availability Zones and subnets

us-west-2a | subnet-0cc436d2de1674e74 (Private Subnet 1)
10.0.2.0/24

us-west-2b | subnet-00dfef2f5052e3e74 (Private Subnet 2)
10.0.4.0/24

Create a subnet



Health checks

Health checks increase availability by replacing unhealthy instances. When you use multiple health checks, all are evaluated, and if at least one fails, instance replacement occurs.

EC2 health checks
Always enabled

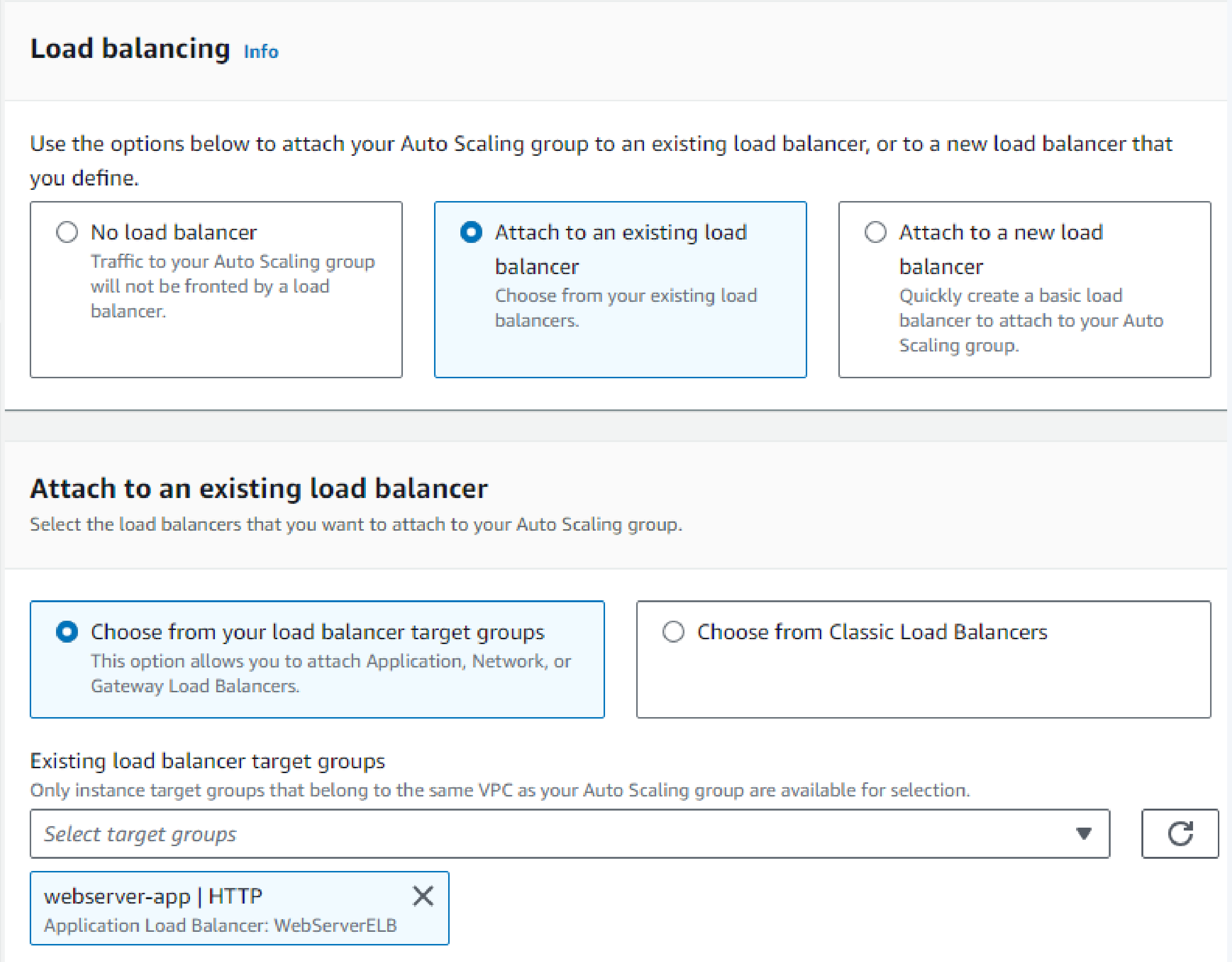
Additional health check types - *optional* Info

☑ Turn on Elastic Load Balancing health checks Recommended
Elastic Load Balancing monitors whether instances are available to handle requests. When it reports an unhealthy instance, EC2 Auto Scaling can replace it on its next periodic check.

EC2 Auto Scaling will start to detect and act on health checks performed by Elastic Load Balancing. To avoid unexpected terminations, first verify the settings of these health checks in the Load Balancer console

□ Turn on VPC Lattice health checks
VPC Lattice can monitor whether instances are available to handle requests. If it considers a target as failed a health check, EC2 Auto Scaling replaces it after its next periodic check.

Health check grace period Info
This time period delays the first health check until your instances finish initializing. It doesn't prevent an instance from terminating when placed into a non-running state.
300 seconds



Load balancing Info

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

○ No load balancer
Traffic to your Auto Scaling group will not be fronted by a load balancer.

● Attach to an existing load balancer
Choose from your existing load balancers.

○ Attach to a new load balancer
Quickly create a basic load balancer to attach to your Auto Scaling group.

Attach to an existing load balancer

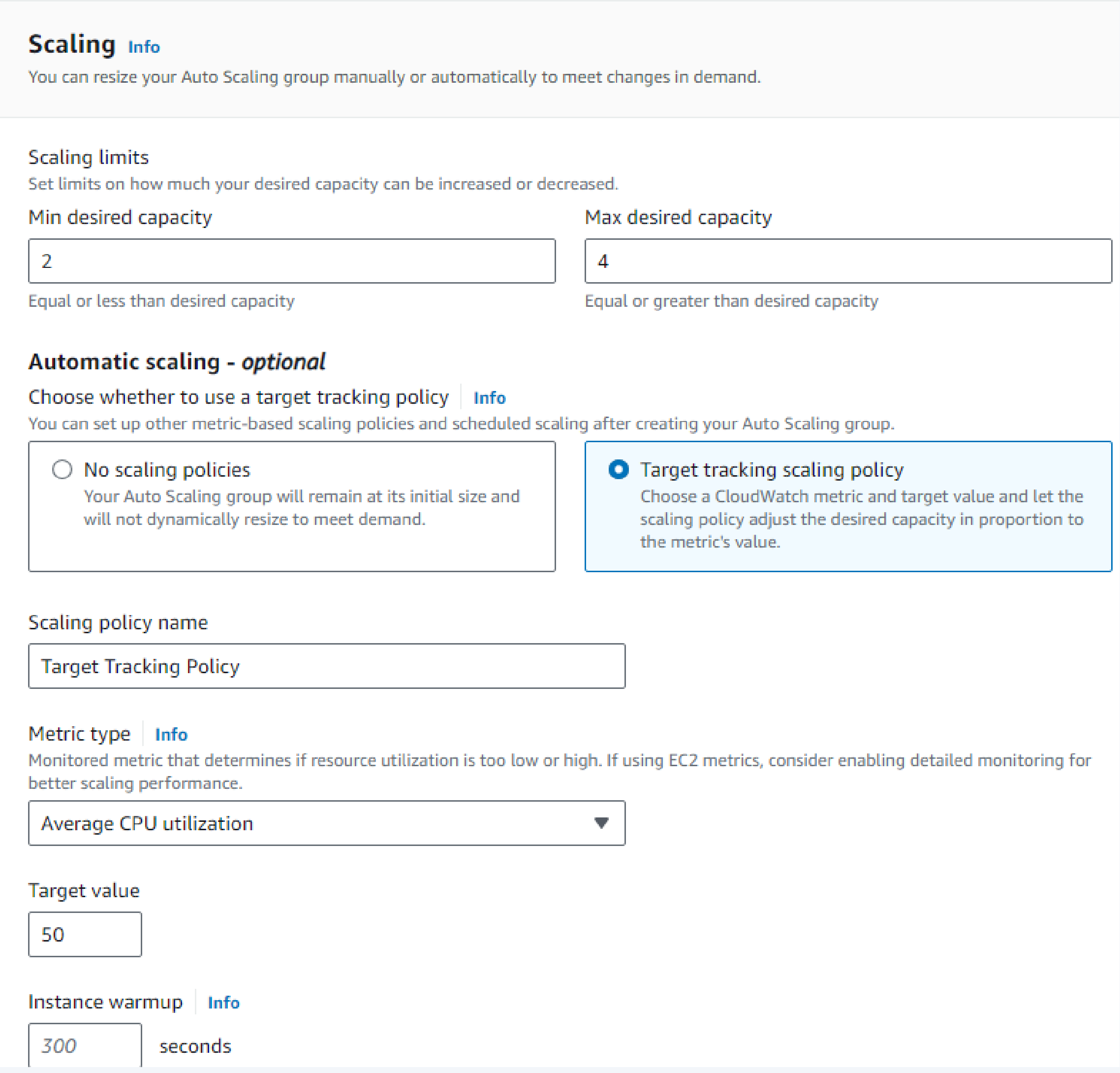
Select the load balancers that you want to attach to your Auto Scaling group.

● Choose from your load balancer target groups
This option allows you to attach Application, Network, or Gateway Load Balancers.

○ Choose from Classic Load Balancers

Existing load balancer target groups
Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.
Select target groups

webserver-app | HTTP
Application Load Balancer: WebServerELB



Scaling Info

You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits
Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity	Max desired capacity
2	4
Equal or less than desired capacity	Equal or greater than desired capacity

Automatic scaling - *optional*

Choose whether to use a target tracking policy Info
You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

○ No scaling policies
Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.

● Target tracking scaling policy
Choose a CloudWatch metric and target value and let the scaling policy adjust the desired capacity in proportion to the metric's value.

Scaling policy name
Target Tracking Policy

Metric type Info
Monitored metric that determines if resource utilization is too low or high. If using EC2 metrics, consider enabling detailed monitoring for better scaling performance.
Average CPU utilization

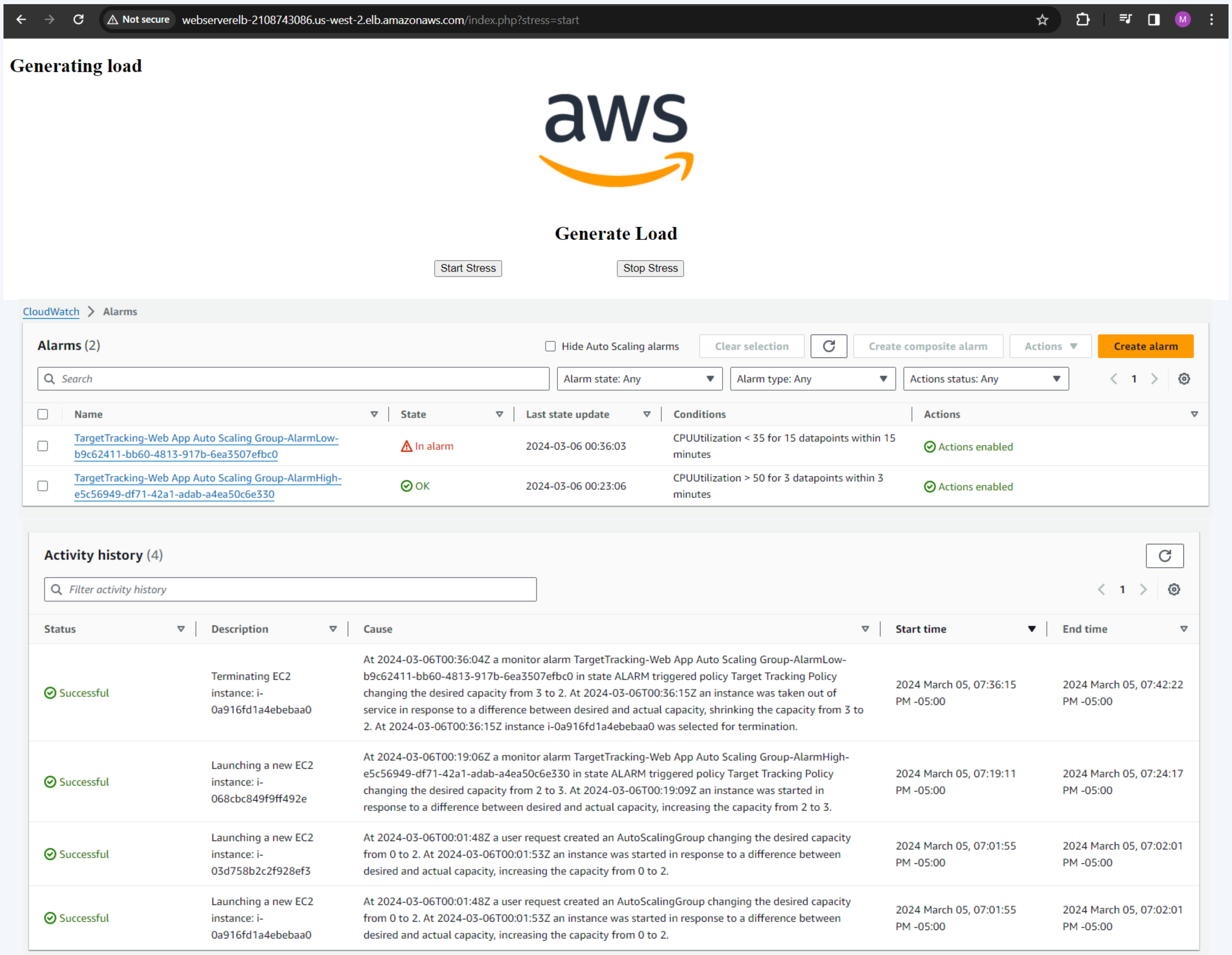
Target value
50

Instance warmup Info
300 seconds

SWIPE

Tarea 01

Y lo probamos, funciona correctamente



⚠ Not secure
webserverelb-2108743086.us-west-2.elb.amazonaws.com/index.php?stress=start

Generating load

Generate Load

Alarms (2)

□	Name	State	Last state update	Conditions	Actions
□	TargetTracking-Web App Auto Scaling Group-AlarmLow-b9c62411-bb60-4813-917b-6ea3507efbc0	⚠ In alarm	2024-03-06 00:36:03	CPUUtilization < 35 for 15 datapoints within 15 minutes	✓ Actions enabled
□	TargetTracking-Web App Auto Scaling Group-AlarmHigh-e5c56949-df71-42a1-adab-a4ea50c6e330	✓ OK	2024-03-06 00:23:06	CPUUtilization > 50 for 3 datapoints within 3 minutes	✓ Actions enabled

Activity history (4)

Status	Description	Cause	Start time	End time
✓ Successful	Terminating EC2 instance: i-0a916fd1a4ebebaa0	At 2024-03-06T00:36:04Z a monitor alarm TargetTracking-Web App Auto Scaling Group-AlarmLow-b9c62411-bb60-4813-917b-6ea3507efbc0 in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 3 to 2. At 2024-03-06T00:36:15Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 3 to 2. At 2024-03-06T00:36:15Z instance i-0a916fd1a4ebebaa0 was selected for termination.	2024 March 05, 07:36:15 PM -05:00	2024 March 05, 07:42:22 PM -05:00
✓ Successful	Launching a new EC2 instance: i-068cbc849f9ff492e	At 2024-03-06T00:19:06Z a monitor alarm TargetTracking-Web App Auto Scaling Group-AlarmHigh-e5c56949-df71-42a1-adab-a4ea50c6e330 in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 2 to 3. At 2024-03-06T00:19:09Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 2 to 3.	2024 March 05, 07:19:11 PM -05:00	2024 March 05, 07:24:17 PM -05:00
✓ Successful	Launching a new EC2 instance: i-03d758b2c2f928ef3	At 2024-03-06T00:01:48Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2024-03-06T00:01:53Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.	2024 March 05, 07:01:55 PM -05:00	2024 March 05, 07:02:01 PM -05:00
✓ Successful	Launching a new EC2 instance: i-0a916fd1a4ebebaa0	At 2024-03-06T00:01:48Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2024-03-06T00:01:53Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.	2024 March 05, 07:01:55 PM -05:00	2024 March 05, 07:02:01 PM -05:00

SWIPE