



42°

Lab - AWS re/Start

Creación de instancias de Amazon EC2



EC2



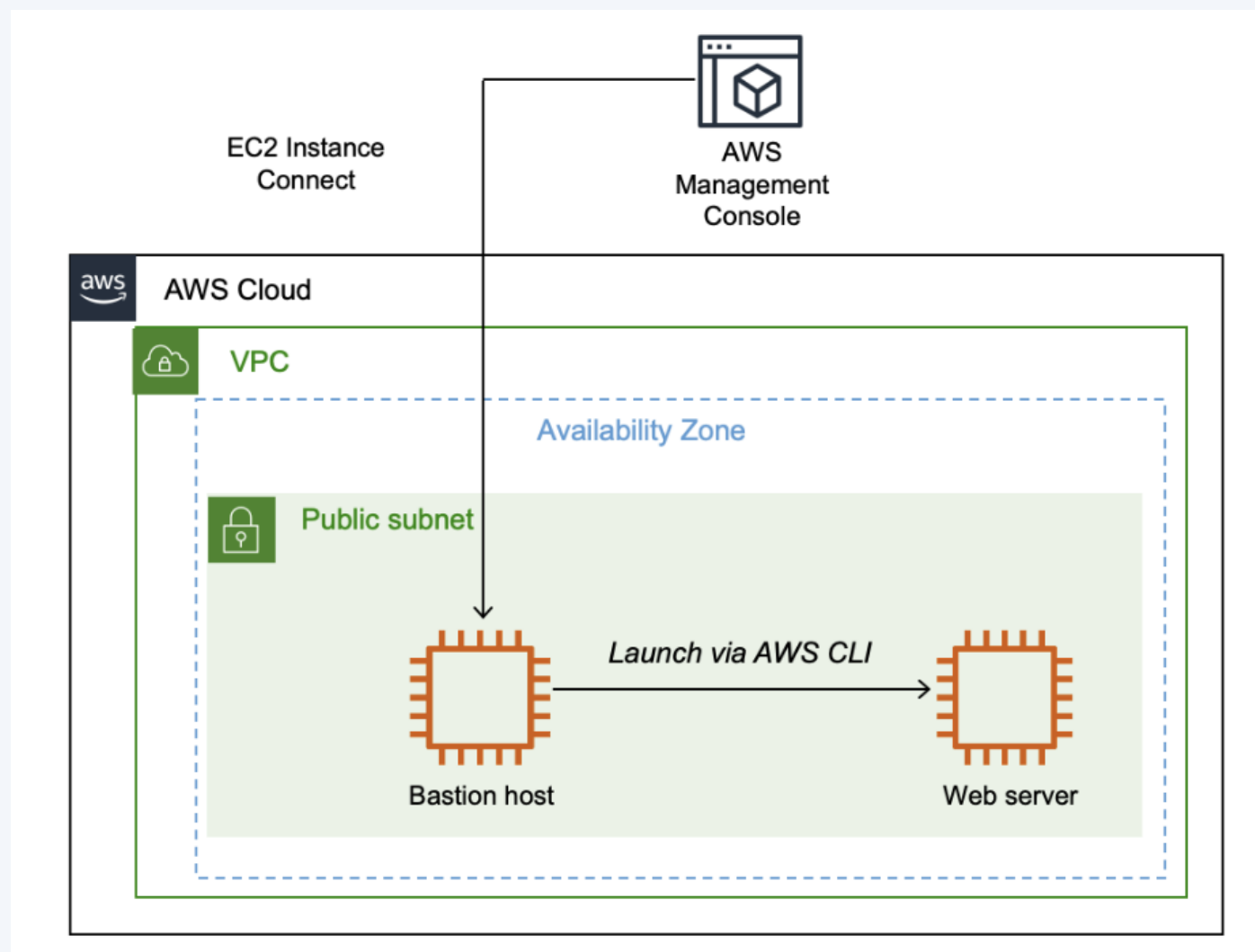
Tarea 01



Interactuando con Amazon EC2

A continuación, se muestra los objetivos del laboratorio:

- Inicie una instancia EC2 mediante la consola de AWS.
- Conéctese a la instancia EC2 mediante EC2 Instance Connect.
- Inicie una instancia EC2 mediante la CLI de AWS.



Tarea 01



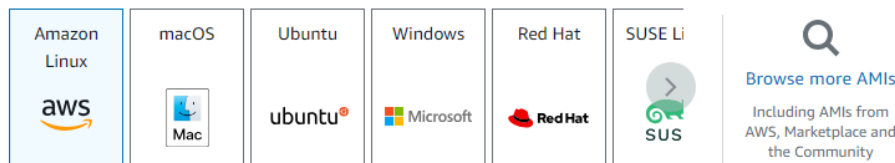
Lo primero, empezamos creando una instancia EC2 en la consola de *AWS* con el nombre *Bastion Host*.

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Recents [Quick Start](#)



Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type
ami-0895022f3dac85884 (64-bit (x86)) / ami-09ebdd80c5c138f65 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs [Free tier eligible](#)

Description

Amazon Linux 2 Kernel 5.10 AMI 2.0.20240223.0 x86_64 HVM gp2

Architecture

64-bit (x86)

AMI ID

ami-0895022f3dac85884

Verified provider

VPC - required [Info](#)

vpc-07afd39380c28a104 (Lab VPC)
10.0.0.0/16

Subnet [Info](#)

subnet-0aa9216d1cec95b0d Public Subnet
VPC: vpc-07afd39380c28a104 Owner: 767397882364
Availability Zone: us-west-2a IP addresses available: 250 CIDR: 10.0.0.0/24

Auto-assign public IP [Info](#)

Enable

Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group ☐ Select existing security group

Security group name - required

Bastion security group

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _-:/()#,@!+=&;!\$*

Description - required [Info](#)

Permit SSH connections

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0)

Remove

Type Info	Protocol Info	Port range Info
ssh	TCP	22
Source type Info	Source Info	Description - optional Info
Anywhere	<input type="text"/> Add CIDR, prefix list or security	e.g. SSH for admin desktop
	0.0.0.0/0	

▼ Instance type [Info](#) | [Get advice](#)

Instance type

t3.micro
Family: t3 2 vCPU 1 GiB Memory Current generation: true
On-Demand SUSE base pricing: 0.0104 USD per Hour
On-Demand Windows base pricing: 0.0196 USD per Hour
On-Demand RHEL base pricing: 0.0704 USD per Hour
On-Demand Linux base pricing: 0.0104 USD per Hour

☒ All generations

[Compare instance types](#)

Additional costs apply for AMIs with pre-installed software

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Proceed without a key pair (Not recommended) Default value

[Create new key pair](#)

▼ Configure storage [Info](#)

Advanced

1x 8 GiB gp2 Root volume (Not encrypted)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

Add new volume

Click refresh to view backup information

The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

0 x File systems

Edit

▼ Advanced details [Info](#)

Domain join directory [Info](#)

Select

[Create new directory](#)

IAM instance profile [Info](#)

Bastion-Role
arn:aws:iam::767397882364:instance-profile/Bastion-Role

[Create new IAM profile](#)

Tarea 01



También podemos lanzar instancias desde la interfaz de línea de comandos:

```
us-west-2.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-094bd4e4a1bc1edbc&osUser=ec2-user&region=us-west-2&sshPort=22#/  
[ec2-user@ip-10-0-0-71 ~]$ aws configure  
AWS Access Key ID [None]:  
AWS Secret Access Key [None]:  
Default region name [None]: us-west-2  
Default output format [None]: json  
[ec2-user@ip-10-0-0-71 ~]$ #Set the Region  
[ec2-user@ip-10-0-0-71 ~]$ AZ=$(curl -s http://169.254.169.254/latest/meta-data/placement/availability-zone)  
[ec2-user@ip-10-0-0-71 ~]$ export AWS_DEFAULT_REGION=${AZ::-1}  
--bash: -1: substring expression < 0  
[ec2-user@ip-10-0-0-71 ~]$ #Retrieve latest Linux AMI  
[ec2-user@ip-10-0-0-71 ~]$ AMI=$(aws ssm get-parameters --names /aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-gp2 --query 'Parameters[0].Value' --output text)  
[ec2-user@ip-10-0-0-71 ~]$ echo $AMI  
ami-0dfd45428f2d4a40c  
[ec2-user@ip-10-0-0-71 ~]$ SUBNET=$(aws ec2 describe-subnets --filters 'Name=tag:Name,Values=Public Subnet' --query Subnets[0].SubnetId --output text)  
[ec2-user@ip-10-0-0-71 ~]$ echo $SUBNET  
subnet-0a9216d1cec95b0d  
[ec2-user@ip-10-0-0-71 ~]$ SG=$(aws ec2 describe-security-groups --filters Name=group-name,Values=WebSecurityGroup --query SecurityGroups[0].GroupId --output text)  
[ec2-user@ip-10-0-0-71 ~]$ echo $SG  
sg-029eaae8ba5efc62  
[ec2-user@ip-10-0-0-71 ~]$ wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-100-RESTR1-1-23732/171-lab-JAWS-create-ec2/s3/UserData.txt  
--2024-03-04 00:31:45-- https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-100-RESTR1-1-23732/171-lab-JAWS-create-ec2/s3/UserData.txt  
Resolving aws-tc-largeobjects.s3.us-west-2.amazonaws.com (aws-tc-largeobjects.s3.us-west-2.amazonaws.com)... 52.92.209.202, 52.218.235.1, 52.92.201.42, ...  
Connecting to aws-tc-largeobjects.s3.us-west-2.amazonaws.com (aws-tc-largeobjects.s3.us-west-2.amazonaws.com)|52.92.209.202|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 327 [text/plain]  
Saving to: 'UserData.txt.1'  
  
0% [=====] 0  
100%[=====] 327  
2024-03-04 00:31:45 (13.9 MB/s) - 'UserData.txt.1' saved [327/327]  
  
[ec2-user@ip-10-0-0-71 ~]$ cat UserData.txt.1  
#!/bin/bash  
# Install Apache Web Server  
yum install -y httpd  
  
# Turn on web server  
systemctl enable httpd.service  
systemctl start httpd.service  
  
# Download App files  
wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-100-RESTR1-1/171-lab-$5BJAWS$5D-create-ec2/dashboard-app.zip  
unzip dashboard-app.zip -d /var/www/html/  
[ec2-user@ip-10-0-0-71 ~]$
```

Y completamos el lanzamiento de la instancia:

```
[ec2-user@ip-10-0-0-71 ~]$ INSTANCE=$(  
> aws ec2 run-instances \  
> --image-id $AMI \  
> --subnet-id $SUBNET \  
> --security-group-ids $SG \  
> --user-data file:///home/ec2-user/UserData.txt \  
> --instance-type t3.micro \  
> --tag-specifications 'ResourceType=instance,Tags=[{Key=Name,Value=Web Server}]' \  
> --query 'Instances[*].InstanceId' \  
> --output text \  
> )  
[ec2-user@ip-10-0-0-71 ~]$ echo $INSTANCE  
i-0c66d451976c98b15  
[ec2-user@ip-10-0-0-71 ~]$
```

Después de unos minutos la instancia ya está corriendo, y podemos verificar el servidor yendo a su dirección IPv4 DNS

