

# 22°

**Lab - AWS re/Start**

## **Creación de una VPC y lanzamiento de un servidor web**

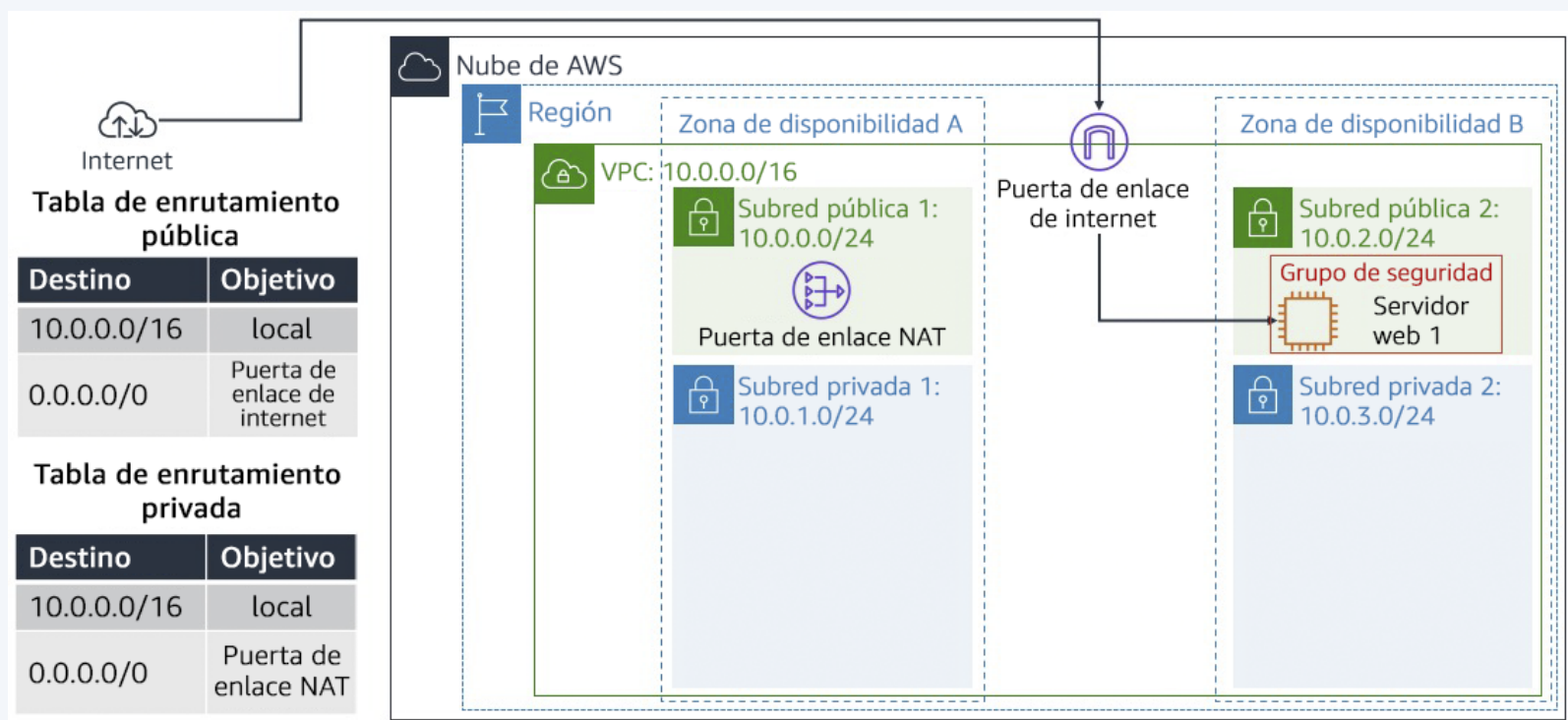


## Tarea 01



# Resolviendo el Ticket del cliente

En este laboratorio, deberá utilizar Amazon Virtual Private Cloud (VPC) para **crear su propia VPC** y agregar componentes adicionales con el fin de producir una red personalizada para una gran empresa. Además, creará **grupos de seguridad** para la instancia de EC2. Deberá configurar y personalizar **una instancia de EC2 para ejecutar un servidor web** y lanzarla en la VPC que se parece al siguiente diagrama del cliente:



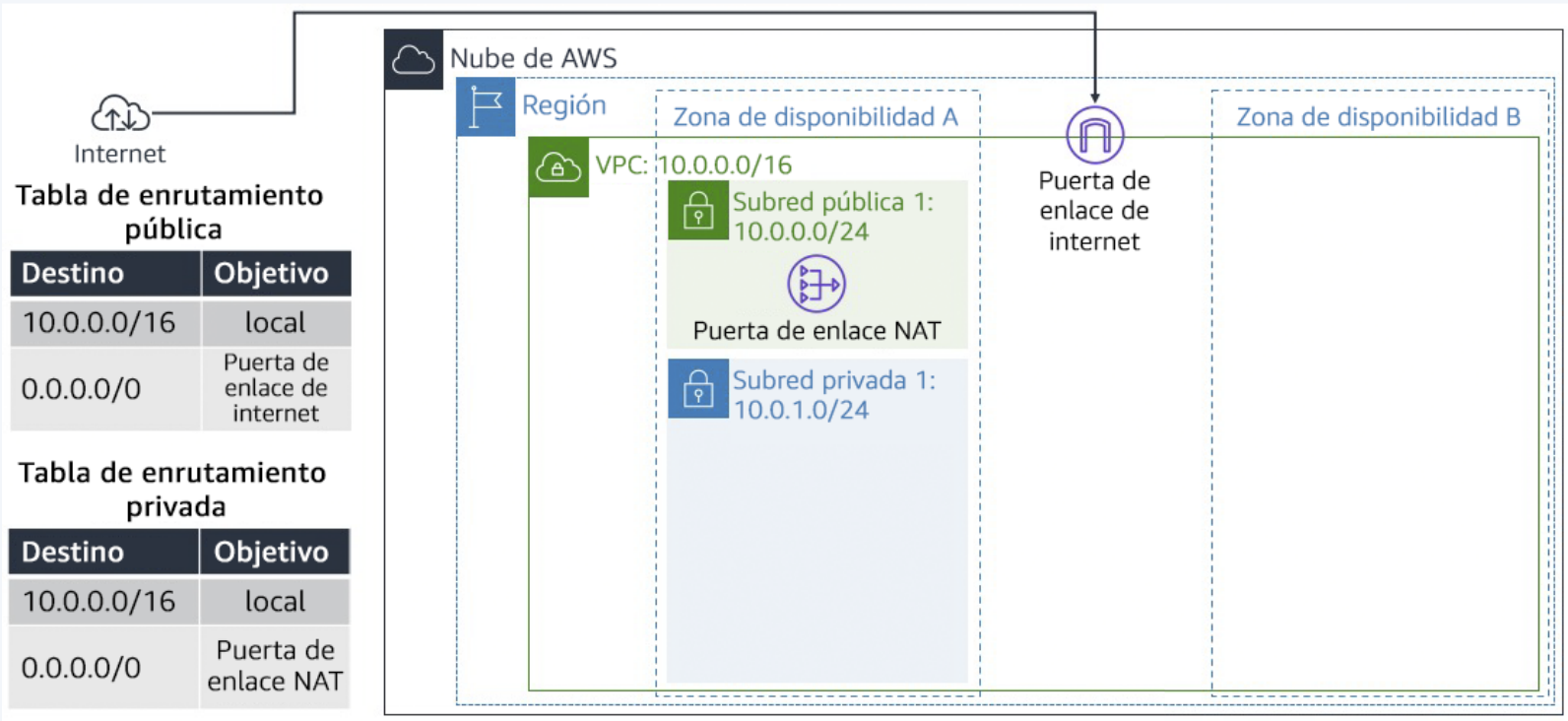
# Tarea 01



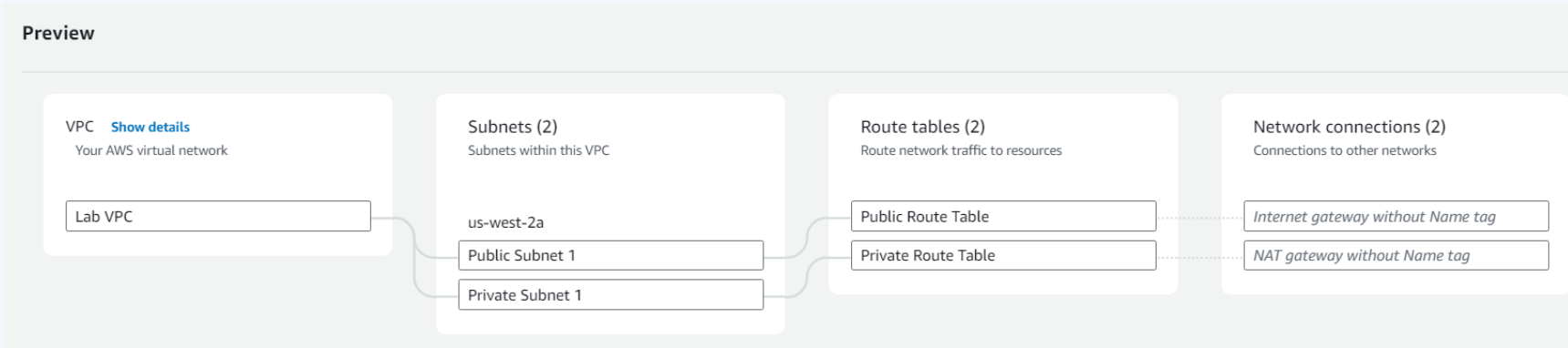
Empezamos por la creación de la VPC, y la subred pública y privada de una AZ. Me refiero al siguiente diagrama, pues adicionalmente asignaremos las tablas de enrutamiento a cada subred.

OJO:

- Cada subred necesita de una tabla de enrutamiento. Y esta tabla de enrutamiento puede ser común entre subredes.



Así la vista previa es:



# Tarea 01



Y la configuración de la VPC fue la siguiente, donde también señalamos que **se creará una puerta de enlace NAT en una zona de disponibilidad**. Recuerda que las subredes se crean a nivel de AZs:

**VPC settings**

Resources to create [Info](#)

Create only the VPC resource or the VPC and other networking resources.

☐ VPC only ☒ VPC and more

Name tag auto-generation [Info](#)

Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.

☐ Auto-generate

IPv4 CIDR block [Info](#)

Determine the starting IP and the size of your VPC using CIDR notation.

65,536 IPs

CIDR block size must be between /16 and /28.

IPv6 CIDR block [Info](#)

☒ No IPv6 CIDR block ☐ Amazon-provided IPv6 CIDR block

Tenancy [Info](#)

Number of Availability Zones (AZs) [Info](#)

Choose the number of AZs in which to provision subnets. We recommend at least two AZs for high availability.

☐ 1 ☒ 2 ☐ 3

Number of public subnets [Info](#)

The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the internet.

☐ 0 ☒ 1

Number of private subnets [Info](#)

The number of private subnets to add to your VPC. Use private subnets to secure backend resources that don't need public access.

☐ 0 ☒ 1 ☐ 2

▼ Customize subnets CIDR blocks

Public subnet CIDR block in us-west-2a

256 IPs

Private subnet CIDR block in us-west-2a

256 IPs

NAT gateways (\$) [Info](#)

Choose the number of Availability Zones (AZs) in which to create NAT gateways. Note that there is a charge for each NAT gateway

☐ None ☒ In 1 AZ ☐ 1 per AZ

VPC endpoints [Info](#)

Endpoints can help reduce NAT gateway charges and improve security by accessing S3 directly from the VPC. By default, full access policy is used. You can customize this policy at any time.

☐ None ☒ S3 Gateway

Luego procedemos a crear las subredes restantes del diagrama, estas se encuentran **en otra zona de disponibilidad** (en este caso **usw2-az2**).

# Tarea 01



Así, logramos crear ambas subredes dentro de la otra AZ, estas son sus configuraciones:

subnet-0f6c16645130a36a9 / Public Subnet 2				Actions
Details				
Subnet ID subnet-0f6c16645130a36a9	Subnet ARN arn:aws:ec2:us-west-2:252526466788:subnet/subnet-0f6c16645130a36a9	State Available	IPv4 CIDR 10.0.2.0/24	
Available IPv4 addresses 251	IPv6 CIDR -	Availability Zone us-west-2b	Availability Zone ID usw2-az2	
Network border group us-west-2	VPC vpc-0dedfe04e09eb78e1   Lab VPC	Route table rtb-022a47b56d4a11aa5	Network ACL acl-08aaf1cfc3f5d6435	
Default subnet No		Auto-assign IPv6 address No	Auto-assign customer-owned IPv4 address No	

subnet-0330b37e6c70b4e88 / Private Subnet 2				Actions
Details				
Subnet ID subnet-0330b37e6c70b4e88	Subnet ARN arn:aws:ec2:us-west-2:252526466788:subnet/subnet-0330b37e6c70b4e88	State Available	IPv4 CIDR 10.0.3.0/24	
Available IPv4 addresses 251	IPv6 CIDR -	Availability Zone us-west-2b	Availability Zone ID usw2-az2	
Network border group us-west-2	VPC vpc-0dedfe04e09eb78e1   Lab VPC	Route table rtb-022a47b56d4a11aa5	Network ACL acl-08aaf1cfc3f5d6435	
Default subnet No		Auto-assign IPv6 address No	Auto-assign customer-owned IPv4 address No	

Ahora, lo que debemos hacer es asignar una tabla de enrutamiento a cada una de estas subredes. Así, la *Public route table* está asociada a las subredes públicas

Routes	Subnet associations	Edge associations	Route propagation	Tags
Explicit subnet associations (2)				
Find subnet association				
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	
Public Subnet 1	subnet-0ca438b9eee5ad2d8	10.0.0.0/24	-	
Public Subnet 2	subnet-0f6c16645130a36a9	10.0.2.0/24	-	

# Tarea 01



Y la tabla de enrutamiento privada (*Private route table*) estará asociada a las subredes privadas.

Explicit subnet associations (2)				Edit subnet associations
<input type="text" value="Find subnet association"/>				< 1 > ⚙
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	
Private Subnet 2	<a href="#">subnet-0330b37e6c70b4e88</a>	10.0.3.0/24	–	
Private Subnet 1	<a href="#">subnet-03af3ea21fd0c80f2</a>	10.0.1.0/24	–	

OJO: Notar que el tráfico que quiere **acceder a internet desde algún servicio de la subred privada** tiene como **target a la puerta de enlace NAT**. La cual es el medio para que los servicios en una subred privada puedan conectarse a internet tras pasar también por la puerta de enlace de internet (IGW)

Routes (2)				Both ▼	Edit routes
<input type="text" value="Filter routes"/>				< 1 > ⚙	
Destination	Target	Status	Propagated		
0.0.0.0/0	<a href="#">nat-002851562661cb98b</a>	✔ Active	No		
10.0.0.0/16	local	✔ Active	No		

Ahora, procedemos a crear el grupo de seguridad para la instancia de EC2, donde se desplegará el servidor web.



# Tarea 01



En este caso permite el acceso a la instancia desde cualquier origen (0.0.0.0/0) via el protocolo HTTP.

VPC > Security Groups > sg-0571c798d5e2aa7b0 - Web Security Group

sg-0571c798d5e2aa7b0 - Web Security Group Actions

**Details**

Security group name Web Security Group	Security group ID sg-0571c798d5e2aa7b0	Description Enable HTTP access	VPC ID vpc-0dedfe04e09eb78e1
Owner 252526466788	Inbound rules count 1 Permission entry	Outbound rules count 1 Permission entry	

**Inbound rules** | Outbound rules | Tags

**Inbound rules (1)** Manage tags Edit inbound rules

Search

	Name	Security group rule...	IP version	Type	Protocol	Port range	Source
<input type="checkbox"/>	-	sgr-0015790efb81aa608	IPv4	HTTP	TCP	80	0.0.0.0/0

Finalmente, creamos la instancia de EC2. Acá podemos ver su configuración de gestión de redes (*networking*)

▼ **Network settings** [Info](#)

VPC - required [Info](#)

vpc-0dedfe04e09eb78e1 (Lab VPC)  
10.0.0.0/16 Refresh

Subnet [Info](#)

subnet-0f6c16645130a36a9 Public Subnet 2  
VPC: vpc-0dedfe04e09eb78e1 Owner: 252526466788  
Availability Zone: us-west-2b IP addresses available: 251 CIDR: 10.0.2.0/24 Refresh [Create new subnet](#)

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

Common security groups [Info](#)

Select security groups

Web Security Group sg-0571c798d5e2aa7b0 X  
VPC: vpc-0dedfe04e09eb78e1 Refresh [Compare security group rules](#)

Security groups that you add or remove here will be added to or removed from all your network interfaces.

► **Advanced network configuration**

# Tarea 01

---



Y también tenemos los *Datos de Usuario* utilizados para que la máquina virtual sea el servidor web

```
#!/bin/bash
#Install Apache Web Server and PHP
dnf install -y httpd mariadb105-server php
#Download Lab files
wget https://us-east-1-tcprod.s3.amazonaws.com/courses/CUR-TF-100-RSNETK/v3.0.0.prod-ea58589a/267-lab-NF-build-vpc-web-server/scripts/lab-app.zip
unzip lab-app.zip -d /var/www/html/
#Turn on web server
systemctl enable httpd
systemctl start httpd
```

A continuación se muestra la página web, lo que indica que el servidor tiene las correctas configuraciones de red. Cabe recalcar que pudimos verificarlos gracias a la **dirección pública IPv4 DNS**

