

### **Caso Práctico: Aplicación Inventario en 2 EC2 (con ALB) y Base de datos MySQL en RDS.**

La arquitectura debe cumplir los siguientes requisitos:

- a) Crea dos grupos de seguridad
  - a. SGEC2 con el puerto 80 abierto a todo el mundo en entrada y todo abierto de salida
  - b. SGBBDD con el puerto 3306 abierto para el SGEC2 de entrada y todo abierto de salida
- b) Instalación de una máquina EC2 una aplicación de inventario con el siguiente user-data y aplica el SGEC2.

```
#!/bin/bash
# Install Apache Web Server and PHP
yum install -y httpd mysql
amazon-linux-extras install -y php7.2
# Download Lab files
wget
https://us-west-2-tcprod.s3.amazonaws.com/courses/ILT-TF-300-ACARCH/v1
.0.23/lab-2- webapp/scripts/inventory-app.zip
unzip inventory-app.zip -d /var/www/html/
# Download and install the AWS SDK for PHP
wget https://github.com/aws/aws-sdk-php/releases/download/3.62.3/aws.zip
unzip aws -d /var/www/html
# Turn on web server
chkconfig httpd on
service httpd start
```

Comprueba que la página web funciona. Pero te dará un error de asignación de role. Es un error normal por la propia aplicación de inventario. Documenta con un pantallazo la salida de la web

b) Agrega el Role: LabInstanceProfile a la máquina creada. (AYUDA: Este role está creado en el Learning Lab, no hay que crearlo solo asignarlo a la máquina virtual)

c) Carga otra vez la página y comprueba que el error desaparece y pide configurar la base de datos en el menú Settings. Documenta con un pantallazo la salida de la web

d) Crea una base de datos en el servicio RDS, elegimos una de la capa gratuita de AWS RDS MySQL con Free Tier y aplica el SGBBDD. d) La base de datos no puede tener exposición pública.

e) Los datos de creación de la BBDD MySQL deberán de ser los que se indican a continuación (buscad los correspondientes menús en AWS RDS al crear la BBDD): El resto de los parámetros los que ponga AWS. No es necesario cambiar ninguno más. Documenta con un pantallazo el tablero de la bbdd creada

For **Settings**, configure:

- **DB instance identifier:** `inventory-db`
- **Master username:** `dbadmin`
- **Master password:** `lab-password`
- **Confirm password:** `lab-password`

For **DB instance size**, configure:

- **DB instance class:** Select ☒ **Burstable classes (includes t classes)** and then select `db.t2.micro`

Note: If `db.t2.micro` is not visible in the drop-down then enable the **Include previous generation classes** button.

Expand ► **Additional configuration**, then configure:

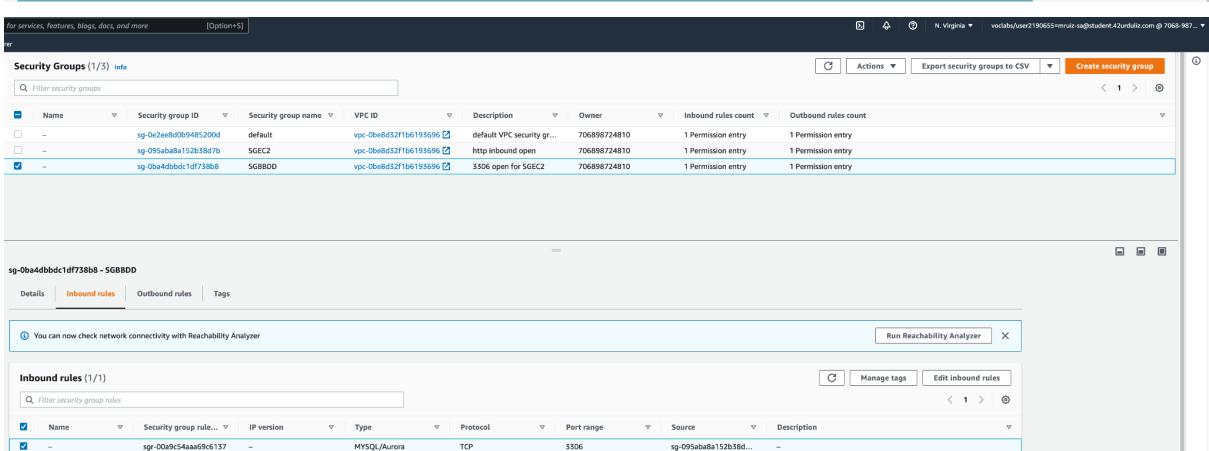
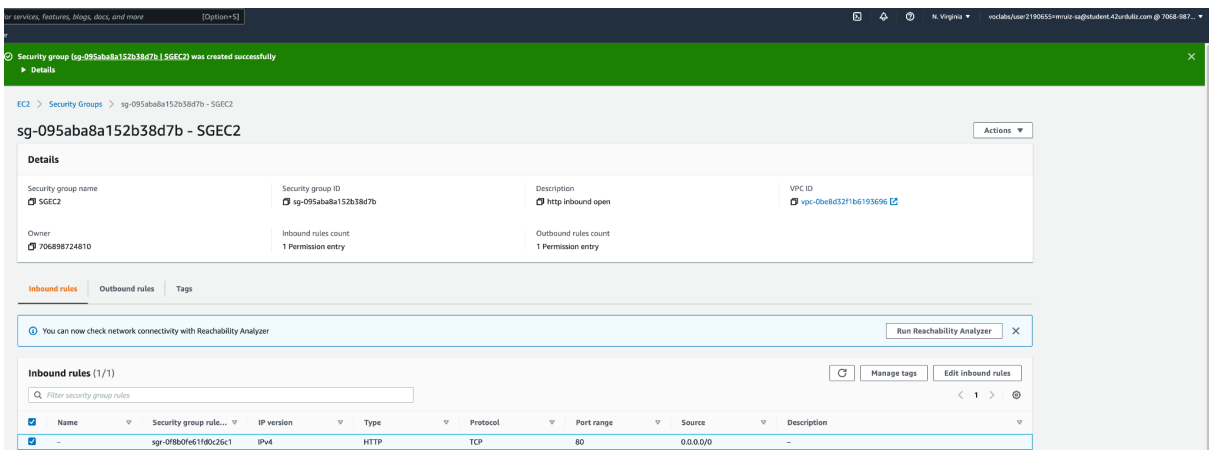
- **Initial database name:** `inventory`
- Uncheck **Enable Enhanced monitoring**

f) Configura la web en la pestaña settings. NOTA: cuidado con los datos. Piensa bien que tienes que poner. Una vez conectada la base de datos tienen que salir 3 elementos de inventario. Agrega un cuarto elemento y documenta con un pantallazo los 4 elementos.

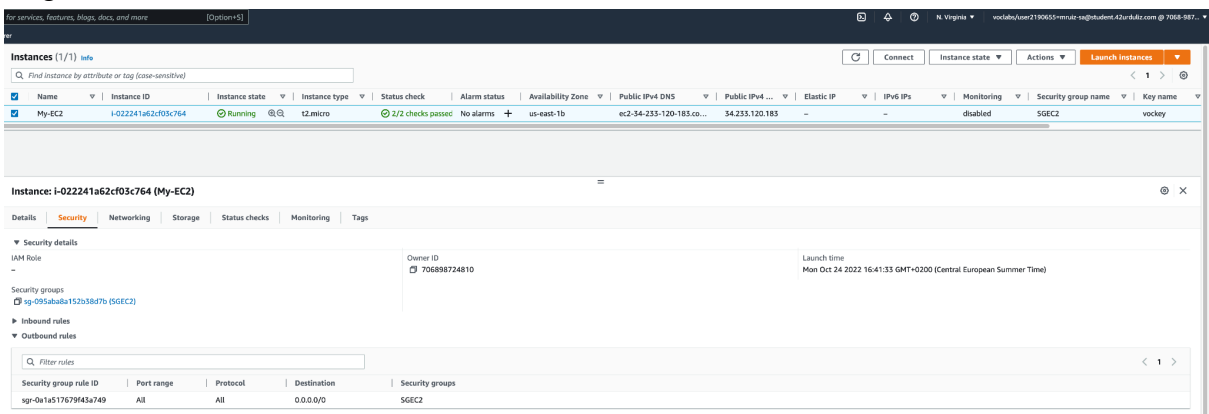
g) Crea una segunda EC2 igual que la anterior en una zona de disponibilidad diferente

h) Crea un balanceador de aplicaciones ALB con un grupo de destino que contenga las dos EC2. Visita la URL del balanceador. Documenta con dos pantallazos el ALB y el Target Group (grupo destino).

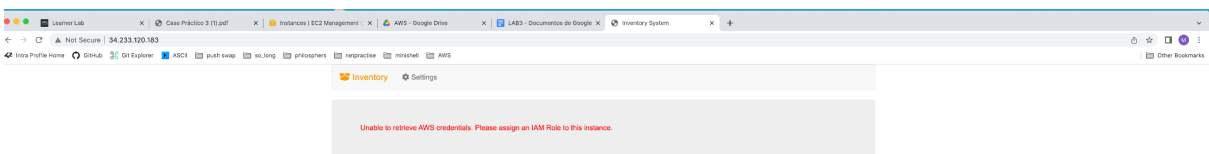
## Creamos los dos SG:



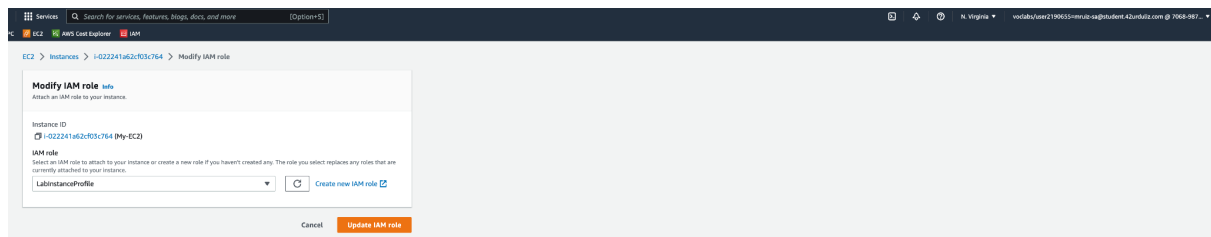
## Creamos la instancia EC2 con el User-Data que nos dan en el enunciado y asignamos al SGE2:



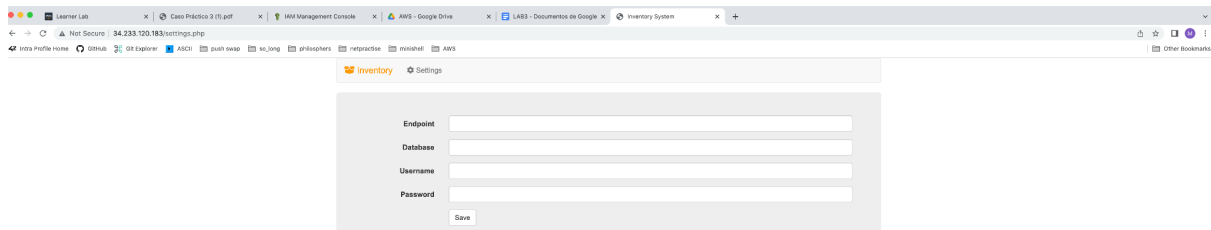
## Probamos a abrir la página y sí que nos conecta pero con el error que menciona el enunciado:



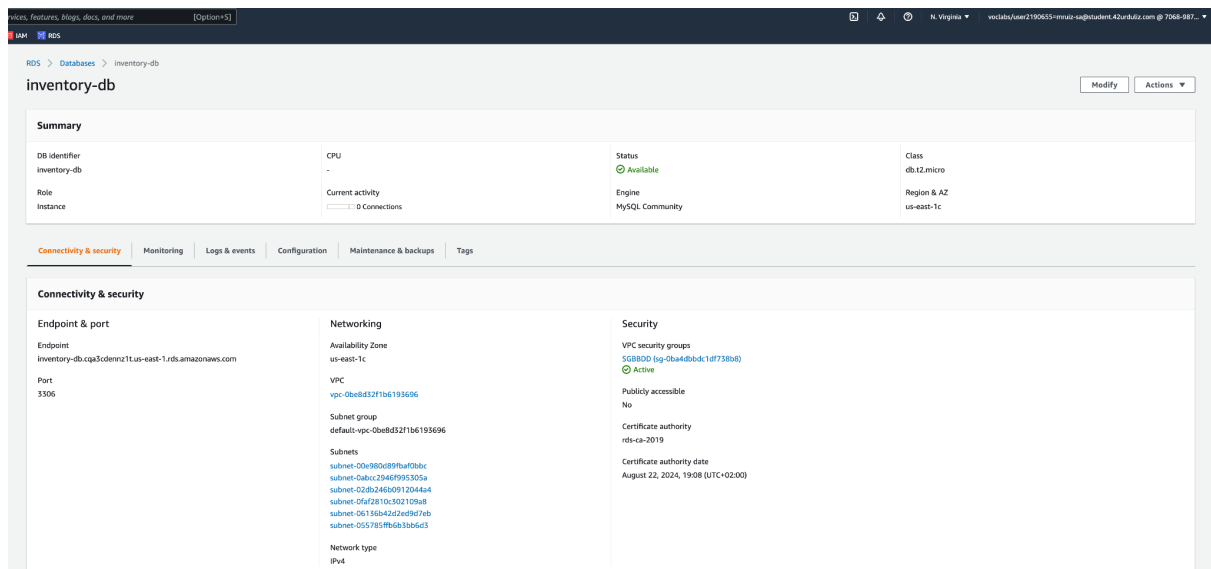
Asignamos el LabInstanceProfile a la instancia:



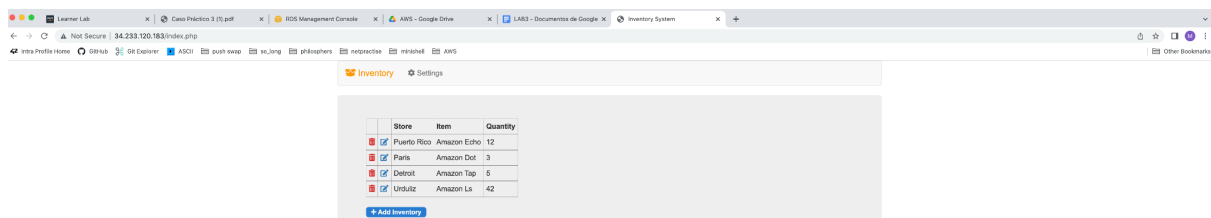
Una vez asignado, recargamos la página y el error desaparece y nos deja entrar en la pestaña settings:



Creamos la Base de datos con los parámetros que nos indican.



Configuramos la pestaña Setting de la página web poniendo los datos de la DB creada y nos aparecen 3 ítems en la DB, creamos un cuarto y el resultado es el siguiente:



Creamos una segunda EC2 igual que la primera pero en una AZ diferente.

**Instances (1/2)** Info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs	Monitoring	Security group name	Key name
My-EC2	i-022241a62cf03c764	Running	t2.micro	2/2 checks passed	No alarms	us-east-1b	ec2-34-233-120-183.co...	34.233.120.183	-	-	disabled	SGEC2	vokey
My-second-EC2	i-0c59d1f4e4c82800b	Running	t2.micro	Initializing	No alarms	us-east-1a	ec2-54-204-194-31.co...	54.204.194.31	-	-	disabled	SGEC2	vokey

**Instance: i-0c59d1f4e4c82800b (My-second-EC2)**

**Instance summary** info

Instance ID: i-0c59d1f4e4c82800b (My-second-EC2)

IPv6 address: -

Hostname type: IP name: ip-172-31-80-162.ec2.internal

Answer private resource DNS name: IPv4 (A)

Auto-assigned IP address: 54.204.194.31 (Public IP)

IAM role: -

Public IPv4 address: 54.204.194.31 | [open address](#)

Instance state: Running

Private IP DNS name (IPv4 only): ip-172-31-80-162.ec2.internal

Instance type: t2.micro

VPC ID: vpc-0be8d32f1b6193696

Subnet ID: subnet-05578b5fb6b3b6d3

Private IPv4 addresses: 172.31.80.162

Public IPv4 DNS: ec2-54-204-194-31.compute-1.amazonaws.com | [open address](#)

Elastic IP addresses: -

AWS Compute Optimizer finding: Opt-in to AWS Compute Optimizer for recommendations. | [Learn more](#)

Auto Scaling Group name: -

Creamos el Target group con las 2 EC2 que hemos creado previamente.:

**Successfully created target group: EC2-Group**

**EC2 > Target groups**

**Target groups (1/1)** Info

Name	ARN	Port	Protocol	Target type	Load balancer	VPC ID
EC2-Group	arn:aws:elasticloadbalancing:us-east-1:706898724810:loadbalancer/targetgroup/EC2-Group	80	HTTP	Instance	None associated	vpc-0be8d32f1b6193696

**Target group: EC2-Group**

**Registered targets (2)**

Instance ID	Name	Port	Zone	Health status	Health status details
i-0c59d1f4e4c82800b	My-second-EC2	80	us-east-1a	unused	Target group is not configured to receive traffic from the load balancer
i-022241a62cf03c764	My-EC2	80	us-east-1b	unused	Target group is not configured to receive traffic from the load balancer

Creamos el Application Load Balancer

**Create Load Balancer** Actions

search: ALB-For-My-DB Add filter

Name	DNS name	State	VPC ID	Availability Zones	Type	Created At	Monitoring
ALB-For-My-DB	ALB-For-My-DB-5286678.us-east-1.elb.amazonaws.com	Active	vpc-0be8d32f1b6193696	us-east-1a, us-east-1b	application	October 24, 2022 at 5:33:18 PM UTC-2	

**Load balancer: ALB-For-My-DB**

**Description** Listeners Monitoring Integrated services Tags

**Basic Configuration**

Name: ALB-For-My-DB

ARN: arn:aws:elasticloadbalancing:us-east-1:706898724810:loadbalancer/app/ALB-For-My-DB:ad706234b6899122

DNS name: ALB-For-My-DB-5286678.us-east-1.elb.amazonaws.com (A Record)

State: Active

Type: application

Scheme: internet-facing

IP address type: ipv4

VPC: vpc-0be8d32f1b6193696

Availability Zones: subnet-05578b5fb6b3b6d3 - us-east-1a, subnet-0fa2810c302109a8 - us-east-1b

Hosted zone: Z35XD0TRO7X7K

Creation time: October 24, 2022 at 5:33:18 PM UTC-2

**Security**

Security groups: sg-095abaf15b2b387b, SGEC2

Y si abrimos la URL del “DNS Name” del ALB vemos que nos carga la página de la DB:

