

Obligatorio

Amazon EKS

10 / 11 / 2024

ADMINISTRACIÓN DE
INFRAESTRUCTURAS



Joaquin Jozami
Salvador Vanoli
Valentin Veintemilla



Israel Bellizzi



UTEc

Introducción.....	3
Consigna.....	3
Procedimiento para realizar el obligatorio.....	5
1. Configurar el entorno.....	5
Instalar el CLI de AWS en la consola del SO.....	5
Acceder a la consola de AWS del learner lab.....	7
Instalar kubectl.....	13
Instalar eksctl.....	15
2. Crear un clúster de EKS utilizando la consola de AWS o la AWS CLI.....	17
Creación del clúster y comunicación con la terminal.....	17
3. Crear un Dockerfile para la aplicación.....	22
Crear las imágenes.....	22
Subir las imágenes a los repositorios.....	27
4. Desplegar la aplicación.....	30
Creación del archivo configuración (yaml) de la base de datos.....	30
Creación del archivo configuración (yaml) del servicio.....	30
Creación del archivo configuración (yaml) del despliegue.....	32
5. Exponer la aplicación.....	36
Creación de un grupo de nodos.....	36
Activación de la base de datos.....	39
6. Prueba de la aplicación.....	40
Prueba de funcionamiento de la aplicación.....	40
7. Certificado SSL/TLS.....	42
Creación del DNS.....	42
Creación y aplicación del certificado SSL.....	46
Subir el certificado SSL obtenido a AWS ACM.....	51
Registrar el DNS creado previamente en AWS Route 53.....	52
Actualizar los NS de ClouDNS por los de Route 53.....	53
Registrar el certificado en los listeners HTTPS.....	54
Configurar backend para que sea seguro.....	56
Comprobación final.....	61
8. Seguridad.....	63
Buenas prácticas.....	63
Aplicación de buenas prácticas.....	63
Escanear las imágenes docker con trivy.....	63
Utilizar AWS Secrets Manager.....	63
CloudTrail y IAM.....	63
Uso de HTTPS y certificados SSL.....	63
Fuzz testing tools.....	64
AFL (American Fuzzy Lop).....	64

ZZUF.....	64
LibFuzzer.....	64
Burp Suite (Intruder Module).....	64
Radamsa.....	64
Peach Fuzzer.....	65
SAST (Static Application Security Testing).....	65
SonarQube.....	65
Checkmarx CxSAST.....	65
Veracode Static Analysis.....	65
Nmap.....	66
Nessus Pro.....	66
OWASP ZAP (Zed Attack Proxy).....	66
Pasos para llevar a cabo los tests de seguridad.....	67
SonarQube.....	67
Nmap.....	73
Nessus Pro.....	76
OWASP ZAP.....	81
Resultados de los tests en la página.....	83
Diagrama de Red de nuestra Aplicación Web.....	84
Video demostrativo.....	84
Conclusión.....	85
Fuentes.....	85

Introducción

En este documento quedará explícito el procedimiento realizado para el Obligatorio de la asignatura Administración de Infraestructuras de la carrera Tecnólogo en Informática de UTEC.

Quedarán documentados los pasos llevados a cabo para dar con el resultado esperado asignado por la consigna. Se incluirán imágenes demostrativas, así como de los resultados obtenidos.

Consigna

Objetivo: Implementar una aplicación en contenedores en un clúster de Kubernetes utilizando el servicio Amazon Elastic Kubernetes Service (EKS).

Pasos:

1. Configurar el Entorno:

- Configurar la interfaz de línea de comandos de AWS (AWS CLI).
- Configurar kubectl, la interfaz de línea de comandos de Kubernetes.
- Crear un Clúster EKS:

2. Crear un clúster de EKS utilizando la consola de AWS o la AWS CLI.

- Configurar kubectl para que se comunique con el clúster de EKS.
- Se le proporcionará un repositorio en github con una aplicación de prueba para desplegar, que contiene un “Docker-compose.yml” el cual genera 2 imágenes personalizadas (frontend y backend) y utiliza una imagen oficial de postgresSQL para la BD.

3. Crear un Dockerfile para la aplicación.

- Crear las imágenes de Docker a raíz del repositorio.
- Subir las imágenes a Amazon Elastic Container Registry (ECR).

4. Desplegar la Aplicación:

- Crear un archivo de configuración de Kubernetes (deployment.yaml) para desplegar la aplicación.
- Usar kubectl apply -f deployment.yaml para crear los recursos necesarios en el clúster de EKS.

5. Exponer la Aplicación:

- Crear un archivo de configuración de Kubernetes (service.yaml) para exponer la aplicación a través de un servicio.

- Usar kubectl apply -f service.yaml para crear el servicio en el clúster de EKS.

6. Prueba de la Aplicación:

- Acceder a la aplicación utilizando la IP o el nombre de dominio asociado al servicio.

7. Certificado SSL/TLS:

- Instale un certificado digital de sitio en el front-end, y solo acepte peticiones https (redirija las peticiones http al puerto 443).
- Investigue Hosting de DNS y emisores de certificados, que le permita solicitar un certificado válido de prueba (ejemplo, 90 días) de forma gratuita. Ej.: <https://www.cloudns.net> (hosting) <https://zeross.com/> (certificado de prueba).
- Utilizar AWS Route 53 para asociar el dominio de prueba con el EKS.

8. Seguridad:

- Investigue todas las recomendaciones y buenas prácticas de seguridad relacionadas con el proyecto.
- Liste lo investigado, incluyendo una breve descripción de cada una.
- Implemente en su proyecto todas las recomendaciones que pueda.
- Herramientas de hacking:
 - Analice el código de la aplicación con una herramienta SAST en busca de vulnerabilidades.
 - Escanee la IP externa pública de la aplicación, con Nmap y con Nessus Pro, para ver que puertos y servicios están expuestos y si hay vulnerabilidades conocidas.
 - Escanee la aplicación web en busca de vulnerabilidades basadas en OWASP (por ejemplo, con ZAP).

9. Entrega:

- URL de la imagen en ECR.
- Archivos de configuración de Kubernetes (deployment.yaml y service.yaml).
- Capturas de pantalla que muestren la aplicación en funcionamiento.
- (Opcional) Archivo de configuración para un Ingress Controller para gestionar el acceso externo al servicio.
- Documento detallado de paso a paso. Este documento tiene como objetivo que los grupos con otro proyecto puedan probar instalar el de ustedes, sin necesidad de preguntar o consultar nada (debe tener un nivel de detalle muy alto).
- Este ejercicio permitirá obtener experiencia práctica con Kubernetes y Amazon EKS, y les proporcionará una comprensión de cómo desplegar, exponer y gestionar aplicaciones en contenedores en un entorno de producción.

Procedimiento para realizar el obligatorio

1. Configurar el entorno

Instalar el CLI de AWS en la consola del SO

Llevamos a cabo la guía de instalación de la web de amazon para instalar el CLI.

Web: <https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html>

Se debe abrir la consola y ejecutar los siguientes comandos en orden:

- Para la instalación:
 - curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip" unzip awscliv2.zip sudo ./aws/install

```
creating: aws/dist/docutils/writers/html5_polyglot/
creating: aws/dist/docutils/writers/latex2e/
creating: aws/dist/docutils/writers/odf_odt/
creating: aws/dist/docutils/writers/pep_html/
creating: aws/dist/docutils/writers/ss_html/
creating: aws/dist/docutils/writers/ss_html/themes/
creating: aws/dist/docutils/writers/ss_html/themes/big-black/
creating: aws/dist/docutils/writers/ss_html/themes/big-white/
creating: aws/dist/docutils/writers/ss_html/themes/default/
creating: aws/dist/docutils/writers/ss_html/themes/medium-black/
creating: aws/dist/docutils/writers/ss_html/themes/medium-white/
creating: aws/dist/docutils/writers/ss_html/themes/small-black/
creating: aws/dist/docutils/writers/ss_html/themes/small-white/
inflating: aws/dist/docutils/writers/ss_html/themes/README.txt
inflating: aws/dist/docutils/writers/ss_html/themes/big-black/_base_.css
inflating: aws/dist/docutils/writers/ss_html/themes/big-black/framing.css
inflating: aws/dist/docutils/writers/ss_html/themes/big-black/pretty.css
inflating: aws/dist/docutils/writers/ss_html/themes/default/print.css
inflating: aws/dist/docutils/writers/ss_html/themes/default/slides.js
inflating: aws/dist/docutils/writers/ss_html/themes/default/outline.css
inflating: aws/dist/docutils/writers/ss_html/themes/default/opera.css
inflating: aws/dist/docutils/writers/ss_html/themes/default/s5-core.css
inflating: aws/dist/docutils/writers/ss_html/themes/default/slides.css
inflating: aws/dist/docutils/writers/ss_html/themes/default/pretty.css
inflating: aws/dist/docutils/writers/ss_html/themes/default/framing.css
inflating: aws/dist/docutils/writers/ss_html/themes/medium-black/pretty.css
inflating: aws/dist/docutils/writers/ss_html/themes/medium-black/_base_.css
inflating: aws/dist/docutils/writers/ss_html/themes/small-white/framing.css
inflating: aws/dist/docutils/writers/ss_html/themes/small-white/pretty.css
inflating: aws/dist/docutils/writers/ss_html/themes/small-black/_base_.css
inflating: aws/dist/docutils/writers/ss_html/themes/small-black/pretty.css
inflating: aws/dist/docutils/writers/ss_html/themes/big-white/framing.css
inflating: aws/dist/docutils/writers/ss_html/themes/big-white/pretty.css
inflating: aws/dist/docutils/writers/ss_html/themes/medium-white/pretty.css
inflating: aws/dist/docutils/writers/ss_html/themes/medium-white/framing.css
inflating: aws/dist/docutils/writers/odf_odt/styles.odt
inflating: aws/dist/docutils/writers/html5_polyglot/minimal.css
inflating: aws/dist/docutils/writers/html5_polyglot/math.css
inflating: aws/dist/docutils/writers/html5_polyglot/template.txt
inflating: aws/dist/docutils/writers/html5_polyglot/tuftig.css
inflating: aws/dist/docutils/writers/html5_polyglot/plain.css
inflating: aws/dist/docutils/writers/html5_polyglot/responsive.css
inflating: aws/dist/docutils/writers/pep_html/pep.css
inflating: aws/dist/docutils/writers/pep_html/template.txt
inflating: aws/dist/docutils/writers/html4css1/html4css1.css
inflating: aws/dist/docutils/writers/html4css1/template.txt
inflating: aws/dist/docutils/writers/latex2e/titlepage.tex
inflating: aws/dist/docutils/writers/latex2e/xelatex.tex
inflating: aws/dist/docutils/writers/latex2e/default.tex
inflating: aws/dist/docutils/writers/latex2e/docutils.sty
inflating: aws/dist/docutils/writers/latex2e/titlingpage.tex
you can now run: /usr/local/bin/aws --version
[enter]veintemilla@utec-19248:~$ /usr/local/bin/aws --version
awscli/2.18.4 Python/3.12.6 Linux/6.5.0-45-generic exe/x86_64.glibc.2.32
[enter]veintemilla@utec-19248:~$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
```

- Para la actualización:

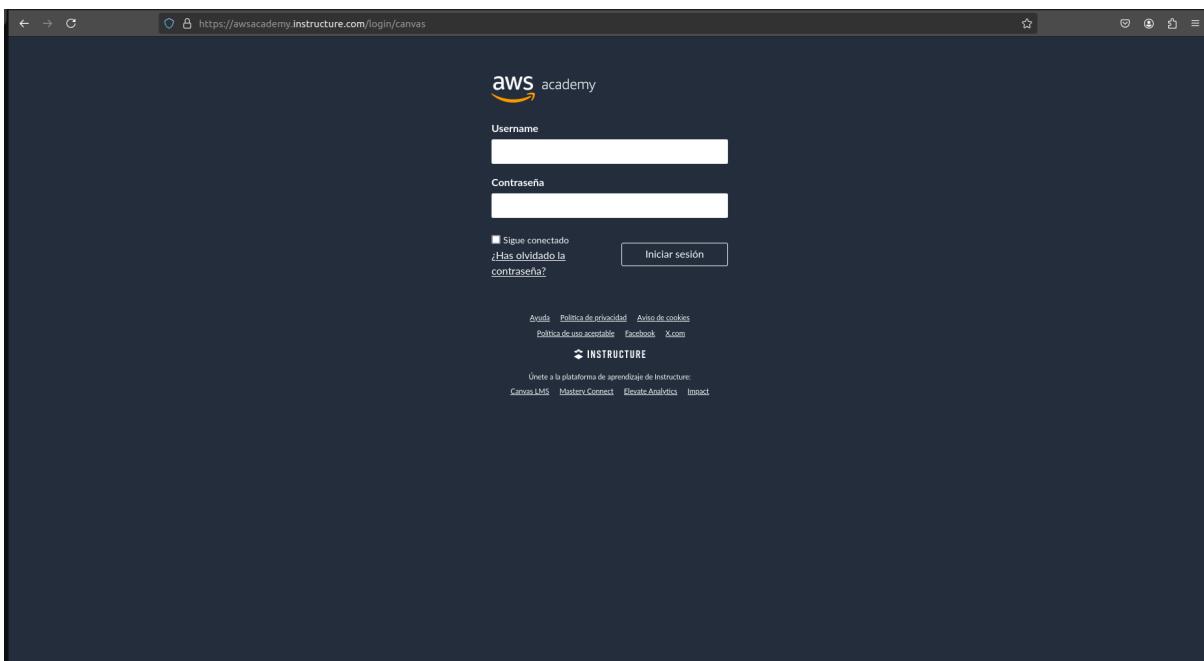
- curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
- unzip awscliv2.zip
- sudo ./aws/install --bin-dir /usr/local/bin --install-dir /usr/local/aws-cli --update

```

inflating: aws/dist/docutils/parsers/rst/include/isonum.txt
inflating: aws/dist/docutils/parsers/rst/include/isoamsb.txt
inflating: aws/dist/docutils/parsers/rst/include/xhtml1-special.txt
inflating: aws/dist/docutils/parsers/rst/include/isomscr-wide.txt
inflating: aws/dist/docutils/parsers/rst/include/isomfrk-wide.txt
inflating: aws/dist/docutils/parsers/rst/include/isopub.txt
inflating: aws/dist/docutils/parsers/rst/include/isoamsa.txt
inflating: aws/dist/docutils/parsers/rst/include/isocyr1.txt
inflating: aws/dist/docutils/parsers/rst/include/isodia.txt
inflating: aws/dist/docutils/parsers/rst/include/isomscr.txt
inflating: aws/dist/docutils/parsers/rst/include/isobox.txt
inflating: aws/dist/docutils/parsers/rst/include/README.txt
inflating: aws/dist/docutils/parsers/rst/include/mmlalias.txt
inflating: aws/dist/docutils/parsers/rst/include/isotech.txt
inflating: aws/dist/docutils/parsers/rst/include/isompf-wide.txt
inflating: aws/dist/docutils/writers/s5_html/themes/README.txt
inflating: aws/dist/docutils/writers/s5_html/themes/big-black/__base__
inflating: aws/dist/docutils/writers/s5_html/themes/big-black/framing.css
inflating: aws/dist/docutils/writers/s5_html/themes/big-black/pretty.css
inflating: aws/dist/docutils/writers/s5_html/themes/default/print.css
inflating: aws/dist/docutils/writers/s5_html/themes/default/slides.js
inflating: aws/dist/docutils/writers/s5_html/themes/default/outline.css
inflating: aws/dist/docutils/writers/s5_html/themes/default/opera.css
inflating: aws/dist/docutils/writers/s5_html/themes/default/s5-core.css
inflating: aws/dist/docutils/writers/s5_html/themes/default/slides.css
inflating: aws/dist/docutils/writers/s5_html/themes/default/pretty.css
inflating: aws/dist/docutils/writers/s5_html/themes/default/framing.css
inflating: aws/dist/docutils/writers/s5_html/themes/medium-black/pretty.css
inflating: aws/dist/docutils/writers/s5_html/themes/medium-black/__base__
inflating: aws/dist/docutils/writers/s5_html/themes/small-white/framing.css
inflating: aws/dist/docutils/writers/s5_html/themes/small-white/pretty.css
inflating: aws/dist/docutils/writers/s5_html/themes/small-black/__base__
inflating: aws/dist/docutils/writers/s5_html/themes/small-black/pretty.css
inflating: aws/dist/docutils/writers/s5_html/themes/big-white/framing.css
inflating: aws/dist/docutils/writers/s5_html/themes/big-white/pretty.css
inflating: aws/dist/docutils/writers/s5_html/themes/medium-white/pretty.css
inflating: aws/dist/docutils/writers/s5_html/themes/medium-white/framing.css
inflating: aws/dist/docutils/writers/odf_odt/styles.odt
inflating: aws/dist/docutils/writers/html5_polyglot/minimal.css
inflating: aws/dist/docutils/writers/html5_polyglot/math.css
inflating: aws/dist/docutils/writers/html5_polyglot/template.txt
inflating: aws/dist/docutils/writers/html5_polyglot/tuftig.css
inflating: aws/dist/docutils/writers/html5_polyglot/plain.css
inflating: aws/dist/docutils/writers/html5_polyglot/responsive.css
inflating: aws/dist/docutils/writers/pep_html/pep.css
inflating: aws/dist/docutils/writers/pep_html/template.txt
inflating: aws/dist/docutils/writers/html4css1/html4css1.css
inflating: aws/dist/docutils/writers/html4css1/template.txt
inflating: aws/dist/docutils/writers/latex2e/titlepage.tex
inflating: aws/dist/docutils/writers/latex2e/xelatex.tex
inflating: aws/dist/docutils/writers/latex2e/default.tex
inflating: aws/dist/docutils/writers/latex2e/docutils.sty
inflating: aws/dist/docutils/writers/latex2e/titlingpage.tex
Found same AWS CLI version: /usr/local/aws-cli/v2/2.18.4. Skipping install.
valentin@temilla:~$ 
```

Acceder a la consola de AWS del learner lab

- Inicia sesión en https://www.awsacademy.com/vforcesite/LMS_Login con el correo de UTEC y tu contraseña. Si aún no la estableces, tendrás que hacerlo.



- Una vez iniciada la sesión, serás redirigido a la página principal, desde ahí tendrás que acceder a Panel de control → Learner lab → Modules.

The screenshot shows the AWS Academy control panel. On the left sidebar, under 'Panel de control', there are several cards: 'AWS Academy Cloud Foundations...' (ACFv2ES-LT13-89879) and 'AWS Academy Learner Lab [89878]' (ALLv2ES-LA-LT13-89878). The 'AWS Academy Learner Lab' card is circled in red.

- Acceder al laboratorio a través del enlace que se indica en la imagen a continuación.

The screenshot shows the course page for 'AWS Academy Learner Lab [89878]'. The left sidebar lists 'Cursos' and 'AWS Academy Learner Lab [89878]'. The main content area shows course modules: 'Bienvenida e información general sobre el curso', 'Módulos', 'Foros de discusión', 'Calificaciones', and 'Lucid (Whiteboard)'. Under 'Módulos', the 'Laboratorio de aprendizaje de AWS Academy' module is expanded, showing 'Iniciar el Laboratorio de aprendizaje de AWS Academy'. This button is circled in red.

- Deberá iniciar el laboratorio con el botón “>Start Lab”. El estado de ejecución se indicará a través de una “luz” que se tornará de roja a amarilla y de amarilla a verde.

The screenshot shows the AWS Academy Learner Lab interface. On the left is a sidebar with icons for Cuenta, Tablero, Cursos, Calendario, Bandeja de entrada, Historial, and Ayuda. The main area has a top navigation bar with 'https://awsacademy.instructure.com/courses/89878/modules/items/8229630'. Below this is a header with 'ALLv2ES-LA...' and 'Iniciar el Laboratorio de aprendizaje de AWS Academy'. A yellow circle highlights the 'AWS' status indicator. Another yellow circle highlights the 'Start Lab' button. The central workspace shows a terminal window with the command 'eee_w_3697096@runweb139626:~\$'. To the right is a 'Learner Lab' panel with sections like 'Environment Overview', 'Environment Navigation', and 'Access the AWS Management Console'. At the bottom are 'Anterior' and 'Siguiente' buttons.

En amarillo, se debe esperar.

This screenshot is identical to the one above, but the 'AWS' status indicator is now green, indicating the session has started. The 'Start Lab' button remains highlighted with a red circle. The rest of the interface, including the terminal output and the 'Learner Lab' panel, is the same.

Al estar en verde ya estará iniciado.

The screenshot shows the AWS Academy Learner Lab interface. On the left is a sidebar with navigation links: Cuenta, Tablero, Cursos, Calendario, Bandeja de entrada, Historial, and Ayuda. The main area has a header with the course name, module, and current step. A terminal window shows a command prompt. To the right is a panel titled "Learner Lab" containing "Environment Overview" and "Environment Navigation". Below these are links for "Access the AWS Management Console", "Region restriction", "Service usage and other restrictions", "Using the terminal in the browser", "Running AWS CLI commands", "Using the AWS SDK for Python", "Preserving your budget", "Accessing EC2 Instances", "SSH Access to EC2 Instances", "SSH Access from Windows", and "SSH Access from a Mac". At the bottom of the panel is a note about instructions being updated. The top right of the interface includes buttons for "Start Lab", "End Lab", "AWS Details", "Readme", and "Reset". Navigation arrows at the bottom allow switching between previous and next steps.

- A continuación, con el laboratorio esté iniciado, se deberán obtener 3 datos para terminar de configurar el CLI:
 - AWS Access Key ID
 - AWS Secret Access Key
 - Default region name
- Estos deberán ser ingresados a petición al utilizar el comando “aws configure” en la consola del SO.

This screenshot is identical to the one above, showing the AWS Academy Learner Lab interface. However, the "AWS Details" button in the top right corner of the main toolbar is highlighted with a red circle. The rest of the interface, including the sidebar, terminal window, and "Learner Lab" panel, remains the same.

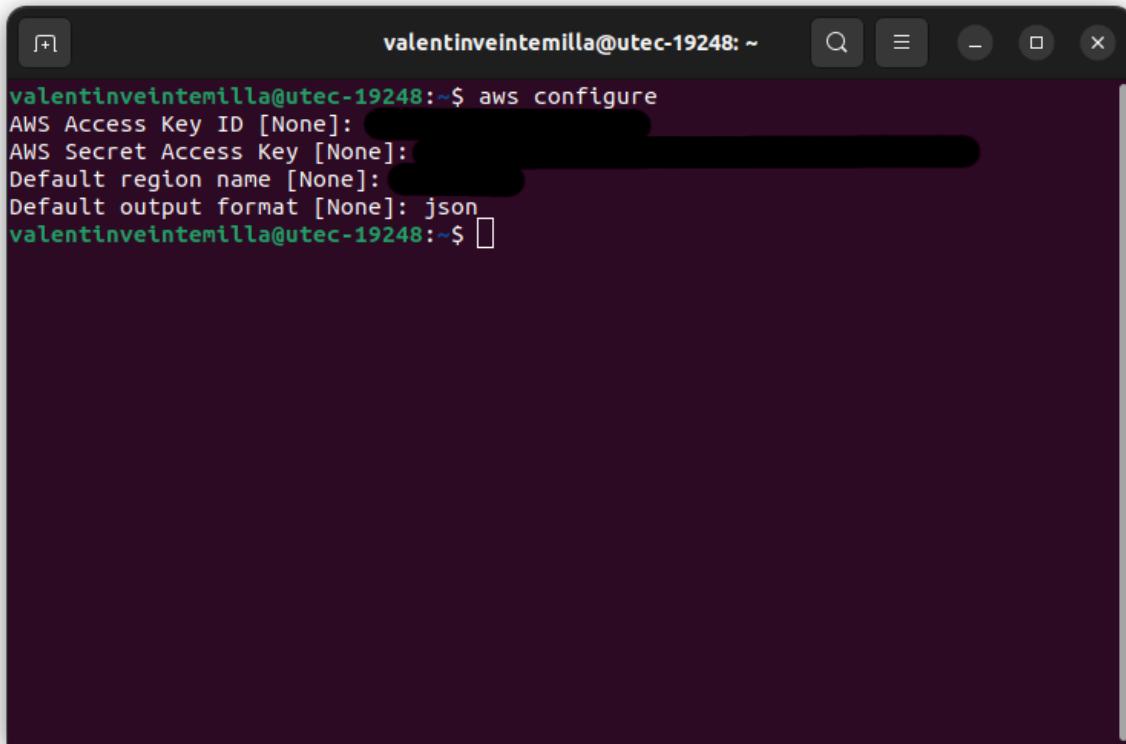
- Aquí, se debe presionar el botón “AWS Details”, permitiendo visualizar datos de importancia sobre la instancia de laboratorio lanzada (como la región y demás datos).

The screenshot shows the AWS Academy interface with the URL <https://awsacademy.instructure.com/courses/89878/modules/items/8229630>. The left sidebar includes links for Cuenta, Tablero, Cursos, Calendario, Bandeja de entrada, Historial, and Ayuda. The main content area shows a terminal session with the command `eee_w_3697996@runweb139626:~$`. On the right, a 'Cloud Access' panel is open, containing sections for AWS CLI (with a 'Show' button circled in red), Cloud Labs (showing session details like start and end times), and AWS Account (with fields for AWSAccountid and Region, where 'Region' is circled in red). Navigation buttons 'Anterior' and 'Siguiente' are at the bottom.

- Para obtener los datos necesarios para configurar el CLI, se debe presionar el botón “Show” señalado en la imagen anterior.

This screenshot shows the same AWS Academy interface as the previous one, but the 'Cloud Access' panel has been updated after pressing the 'Show' button. The AWS CLI section now displays the contents of the `~/.aws/credentials` file, specifically the `[default]` section with `aws_access_key_id`, `aws_secret_access_key`, and `aws_session_token`. The rest of the panel remains the same, showing Cloud Labs session details and navigation buttons.

- A continuación se debe ingresar el comando “aws configure” en la consola e ingresar los datos a medida que se piden.

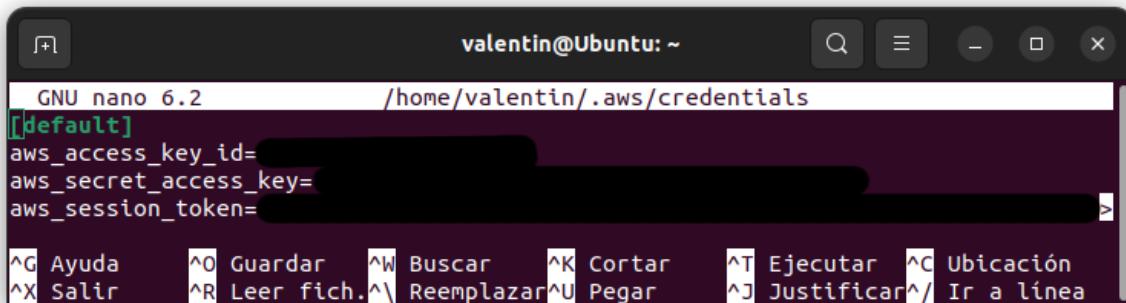


```
valentinveintemilla@utec-19248:~$ aws configure
AWS Access Key ID [None]:
AWS Secret Access Key [None]:
Default region name [None]:
Default output format [None]: json
valentinveintemilla@utec-19248:~$
```

- Una vez ingresados, para asegurar que funcione correctamente, se debe ingresar “nano ~/.aws/credentials”, pegando los siguientes datos dentro del archivo:



- Debería resultar en un archivo así:



The screenshot shows a terminal window titled "valentin@Ubuntu: ~". The command "nano .aws/credentials" is running. The file contains the following content:

```
GNU nano 6.2          /home/valentin/.aws/credentials
[default]
aws_access_key_id=REDACTED
aws_secret_access_key=REDACTED
aws_session_token=REDACTED
```

At the bottom, there are nano editor key bindings:

- ^G** Ayuda
- ^O** Guardar
- ^W** Buscar
- ^K** Cortar
- ^T** Ejecutar
- ^C** Ubicación
- ^X** Salir
- ^R** Leer fich.
- ^A** Reemplazar
- ^U** Pegar
- ^J** Justificar
- ^/** Ir a línea

Instalar kubectl

Llevamos a cabo la guía de instalación de kubectl.

Web:

https://docs.aws.amazon.com/eks/latest/userguide/install-kubectl.html#linux_amd64_kubectl

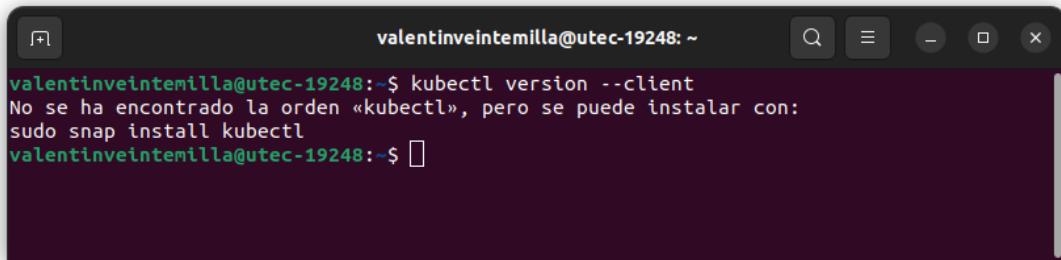
Se debe abrir la consola y ejecutar los siguientes comandos en orden:

- `kubectl version --client`
 - Esto es para saber si ya está instalado, en caso de que lo esté no se deben realizar los pasos de este apartado.

Step 1: Check if kubectl is installed

Determine whether you already have `kubectl` installed on your device.

```
kubectl version --client
```



The screenshot shows a terminal window titled "valentinveintemilla@utec-19248: ~". The command "kubectl version --client" is run, resulting in the following output:

```
valentinveintemilla@utec-19248:~$ kubectl version --client
No se ha encontrado la orden «kubectl», pero se puede instalar con:
sudo snap install kubectl
valentinveintemilla@utec-19248:~$ [ ]
```

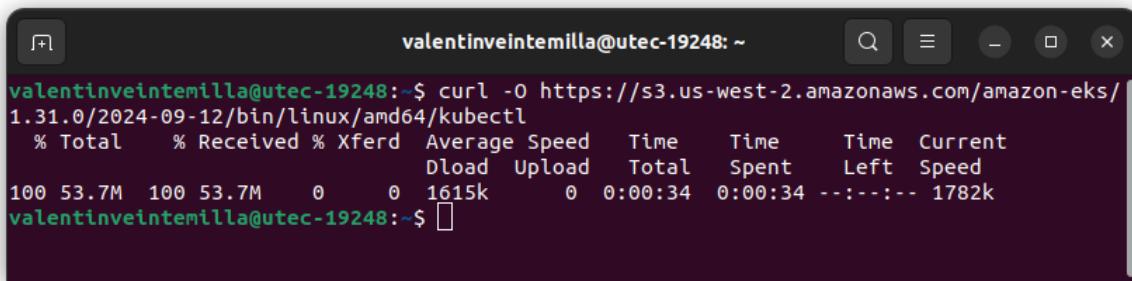
- Se debe elegir la versión respectiva del sistema operativo desde las opciones disponibles. Nosotros utilizamos Linux (amd64).

Step 2: Install or update kubectl

Install or update `kubectl` on one of the following operating systems:

- [macOS](#)
- [Linux \(amd64\)](#)
- [Linux \(arm64\)](#)
- [Windows](#)

- A continuación se ejecuta un comando para instalar kubectl en la última versión disponible, en nuestro caso fue:
 - curl -O
<https://s3.us-west-2.amazonaws.com/amazon-eks/1.31.0/2024-09-12/bin/linux/amd64/kubectl>



```
valentinveintemilla@utec-19248:~$ curl -O https://s3.us-west-2.amazonaws.com/amazon-eks/1.31.0/2024-09-12/bin/linux/amd64/kubectl
% Total    % Received % Xferd  Average Speed   Time   Time     Time  Current
          Dload  Upload   Total   Spent    Left  Speed
100 53.7M  100 53.7M    0     0  1615k      0  0:00:34  0:00:34  --:--:-- 1782k
valentinveintemilla@utec-19248:~$
```

- Se agrega permisos de ejecución al archivo descargado.
 - chmod +x ./kubectl
- Ahora se copian los descargados al archivo PATH.
 - mkdir -p \$HOME/bin && cp ./kubectl \$HOME/bin/kubectl && export PATH=\$HOME/bin:\$PATH
- Una vez realizados los pasos anteriores, chequea que kubectl se haya instalado correctamente, con el comando “kubectl version --client”.

```
valentinveintemilla@utec-19248:~$ kubectl version --client
Client Version: v1.31.0-eks-a737599
Kustomize Version: v5.4.2
valentinveintemilla@utec-19248:~$ 
```

Instalar eksctl

Llevamos a cabo la guía de instalación de eksctl.

Web:

<https://eksctl.io/installation/>

- Crear un script con el siguiente contenido y ejecutarlo. Esperar a que la ejecución termine correctamente.

```
#!/bin/bash
```

```
# for ARM systems, set ARCH to: `arm64`, `armv6` or `armv7`
ARCH=amd64
PLATFORM=$(uname -s)_$ARCH

curl -sLO
"https://github.com/eksctl-io/eksctl/releases/latest/download/eksctl_${PLATFORM}.tar.gz"

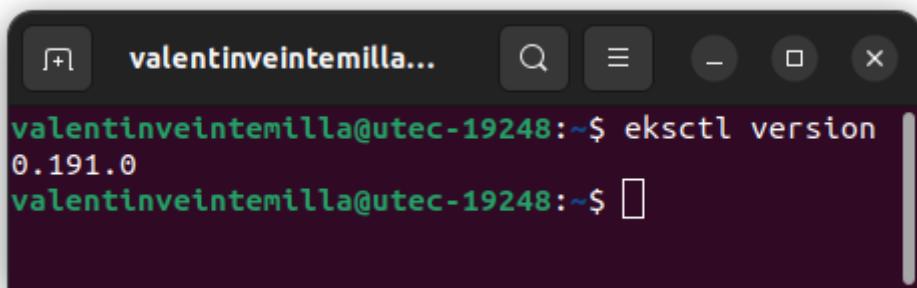
# (Optional) Verify checksum
curl -sL
"https://github.com/eksctl-io/eksctl/releases/latest/download/eksctl_checksums.txt" |
grep $PLATFORM | sha256sum --check

tar -xzf eksctl_${PLATFORM}.tar.gz -C /tmp && rm eksctl_${PLATFORM}.tar.gz

sudo mv /tmp/eksctl /usr/local/bin
```

```
valentinveintemilla@utec-19248:~$ nano eksctlinstall.sh
valentinveintemilla@utec-19248:~$ bash eksctlinstall.sh
eksctl_Linux_amd64.tar.gz: La suma coincide
[sudo] contraseña para valentinveintemilla:
valentinveintemilla@utec-19248:~$ 
```

- Chequear que eksctl se haya instalado correctamente.



```
valentinveintemilla@utec-19248:~$ eksctl version
0.191.0
valentinveintemilla@utec-19248:~$ 
```

2. Crear un clúster de EKS utilizando la consola de AWS o la AWS CLI.

Creación del clúster y comunicación con la terminal

- Una vez instalado todo lo anterior, procedemos a crear un Clúster EKS a través de la interfaz.
- Acceder a la instancia de laboratorio previamente iniciada.
- Apretar el botón verde, que abrirá la interfaz de aws.
- En el buscador, ingresar EKS y seleccionar “Elastic Kubernetes Service”.

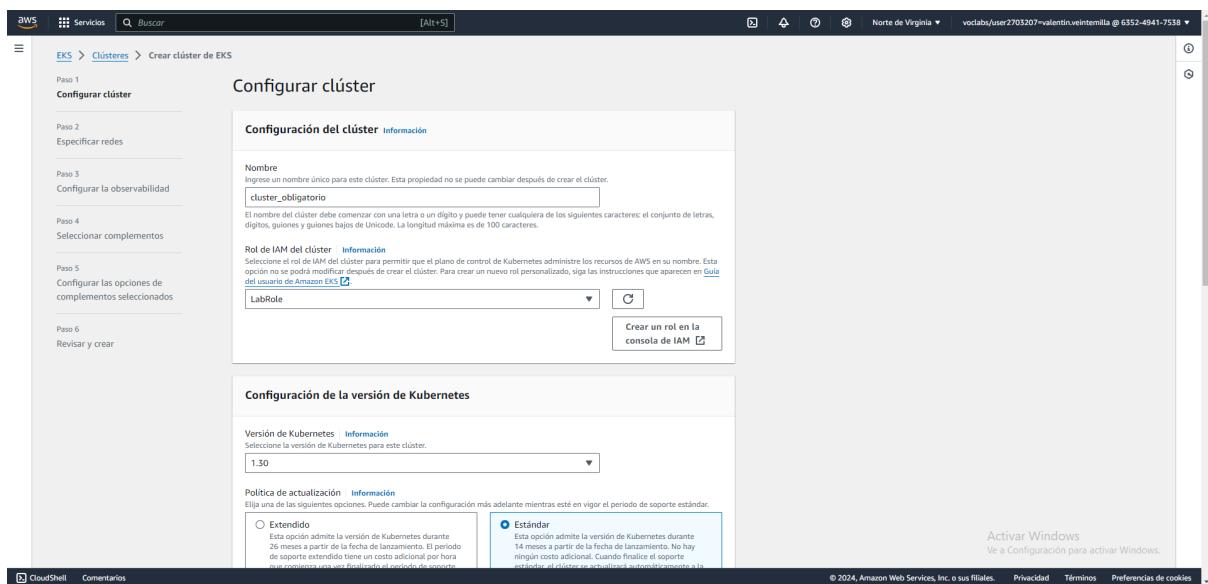
The screenshot shows the AWS search interface with the query 'EKS' entered. The results are categorized under 'Servicios' (Services) and 'Características' (Features). In the 'Servicios' section, 'Elastic Kubernetes Service' is highlighted with a yellow box. It is described as 'La forma más fiable de iniciar, ejecutar y escalar Kubernetes'. Other services listed include AWS FIS, EFS, and MediaStore. In the 'Características' section, there are categories like Clústeres, Puntos de acceso, Sistemas de archivo, and Instantáneas. To the right of the search results, there is a sidebar with a 'Información' section, a 'Buscar aplicaciones' search bar, and a chart showing costs over time.

- Clickear el botón de “Aregar clúster” → “Crear”

The screenshot shows the AWS EKS service console with the URL 'EKS > Clústeres'. The page displays a table for managing clusters, with a single row indicating 'No hay clústeres'. At the top right of the table, there is a 'Agregar clúster' button, which is highlighted with a yellow box. Other buttons in the header include 'Eliminar', 'Crear', 'Registrar', and 'Nuevo'. The left sidebar contains links for 'Amazon Elastic Kubernetes Service', 'Clústeres', 'Amazon EKS Anywhere', 'Servicios relacionados', and 'Configuración de la consola'.

- Llenar los datos

- Nombre: “**cluster Obligatorio**”.
- Dejar el Rol de IAM como el predeterminado.
- Versión de Kubernetes predeterminada (1.30 en nuestro caso).
- Política de actualización: **Estándar**
- Acceso de administrador para arrancar el cluster: Permitir
- Modo de autenticación de clústeres: **API DE EKS**



- Presionar Siguiente.
- VPC predeterminada.
- Subredes predeterminadas, quitando la **us-east-1e**
- Grupo de seguridad: **default**
- Familia de direcciones IP del clúster: **IPv4**
- Acceso al punto de enlace del cluster: **Público y privado**

Especificar redes

VPC **Información**
Seleccione la VPC que desea utilizar para los recursos del clúster de EKS. Para crear una nueva VPC, vaya a la Consola de VPC.

vpc-031ee2462578c9622 | Predeterminado

Subesredes
Elegir las subredes de la VPC donde el plano de control puede colocar interfaces de red elásticas (ENI) para facilitar la comunicación con el clúster. Para crear una nueva subred, vaya a la página correspondiente en la Consola de VPC.

subnet-06721929ba4888f7b us-east-1f 172.31.64.0/20
subnet-0bf9a004ba47f1fb3 us-east-1e 172.31.48.0/20
subnet-0c9efab5e560f97f5 us-east-1a 172.31.80.0/20
subnet-08aa93b2c899fab2 us-east-1d 172.31.16.0/20
subnet-01a0aa315fc05dc91 us-east-1c 172.31.32.0/20
subnet-0ac910e100d5d9f51 us-east-1d 172.31.0.0/20

Grupos de seguridad **Información**
Seleccione los grupos de seguridad que se van a aplicar a las interfaces de red elásticas administradas por EKS creadas en las subredes de los planes de control. Para crear un nuevo grupo de seguridad, vaya a la página correspondiente en la Consola de VPC.

sg-02ed8bc7243f07813 | default default VPC security group

Activar Windows
Ve a Configuración para activar Windows.

- Presionar Siguiente
- Configuración de la observabilidad: **Todo predeterminado**

Configurar la observabilidad

Métricas

Prometheus **Información**
 Envíe las métricas de Prometheus a Amazon Managed Service para Prometheus.
Supervise las métricas de sus aplicaciones e infraestructuras con Amazon Managed Service para Prometheus. Estas métricas incluyen datos sobre el estado y el rendimiento del sistema.

CloudWatch **Información**
 Puede activar Observabilidad de CloudWatch en sus clústeres mediante el complemento Observabilidad de CloudWatch. Una vez creado el clúster, diríjase a la pestaña de complementos e instale el complemento Observabilidad de CloudWatch para activar CloudWatch Application Signals e Información de contenedores y comenzar ainger telemetría en CloudWatch.

Registro del plano de control **Información**
Envíe registros de auditoría y diagnóstico desde el plano de control de Amazon EKS a CloudWatch Logs.

Servidor de la API
Registros que pertenecen a las solicitudes de la API al clúster.

Auditoria
Registros que pertenecen al acceso al clúster a través de la API de Kubernetes.

Autenticador
Registros que pertenecen a las solicitudes de autenticación en el clúster.

Activar Windows
Ve a Configuración para activar Windows.

- Presionar Siguiente
- Selección de complementos: **Todo predeterminado**

The screenshot shows the 'Seleccionar complementos' (Select Add-ons) step of the EKS cluster creation wizard. On the left, a sidebar lists steps from 1 to 6. Step 4, 'Seleccionar complementos', is selected. The main area displays a grid of add-ons categorized by provider:

- Complementos de Amazon EKS (11)**: Includes CoreDNS, kube-proxy, and Agent de Amazon EKS Pod Identity.
- CNI de Amazon VPC**: Includes GuardDuty.
- Supervisión en tiempo de ejecución de EKS en Amazon**: Includes CloudWatch Metrics and CloudWatch Metrics Insights.
- Agente de Amazon EKS Pod Identity**: Includes IAM Role for EKS Pod Identity.

Each add-on card includes a checkbox, a 'Información' link, and a 'Categoría' field. The 'CoreDNS' and 'kube-proxy' add-ons have their checkboxes checked. The 'Activar Windows' link is visible at the bottom right.

- Presionar siguiente
- Configuración de opciones de complementos seleccionados: **Todo predeterminado**

The screenshot shows the 'Configurar las opciones de complementos seleccionados' (Configure selected add-on options) screen. It displays three add-ons with their configuration options:

- CNI de Amazon VPC**: Version v1.18.1-eksbuild.3, Status: Listo para instalar (Ready to install).
- CoreDNS**: Version v1.11.1-eksbuild.8, Status: Listo para instalar (Ready to install).
- kube-proxy**: Version v1.30.0-eksbuild.3, Status: Listo para instalar (Ready to install).

Each add-on card includes an 'Eliminar complemento' (Delete add-on) button. The 'Activar Windows' link is visible at the bottom right.

- Presionar Siguiente
- Revisión y creación: Todo predeterminado

- Presionar **Crear**
- Por último, esperar a que el estado esté en “Activo”.

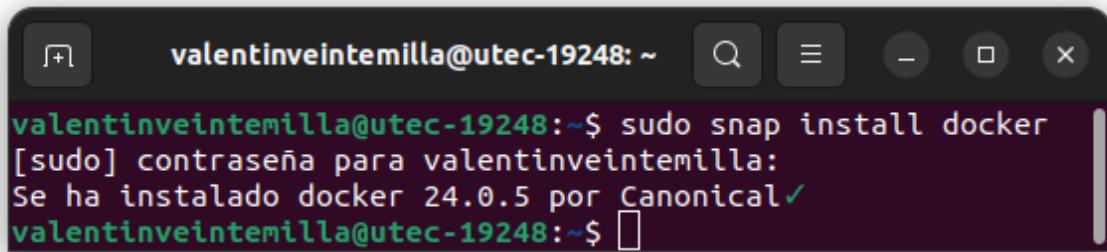
- Configurar kubectl para que interactúe con el clúster usando el siguiente comando
 - aws eks --region us-east-1 update-kubeconfig --name cluster_obligatorio

```
valentin@Ubuntu:~$ aws eks --region us-east-1 update-kubeconfig --name cluster_obligatorio
Added new context arn:aws:eks:us-east-1:635249417538:cluster/cluster_obligatorio to /home/valentin/.kube/config
valentin@Ubuntu:~$
```

3. Crear un Dockerfile para la aplicación.

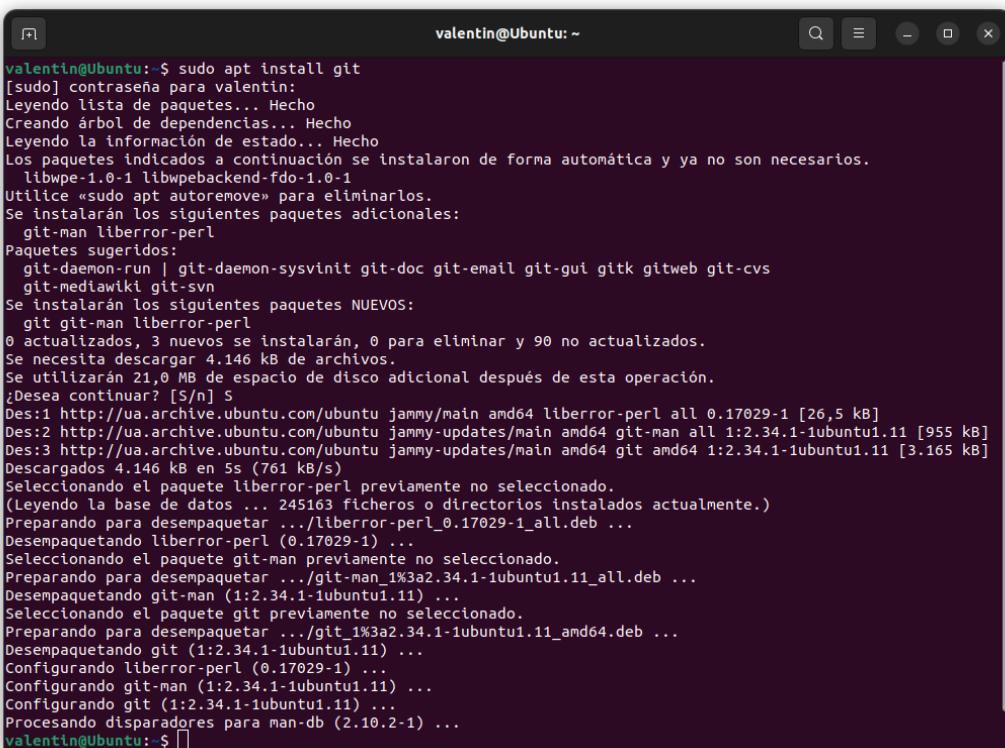
Crear las imágenes

- Instalar docker en la consola
 - sudo snap install docker



```
valentinveintemilla@utec-19248:~$ sudo snap install docker
[sudo] contraseña para valentinveintemilla:
Se ha instalado docker 24.0.5 por Canonical✓
valentinveintemilla@utec-19248:~$ 
```

- Clonar el repositorio con el .yml
 - Instalar git si aún no lo está con “sudo apt install git”



```
valentin@Ubuntu:~$ sudo apt install git
[sudo] contraseña para valentin:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  libwpe-1.0-1 libwpebackend-fdo-1.0-1
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes adicionales:
  git-man liberror-perl
Paquetes sugeridos:
  git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb git-cvs
  git-mediawiki git-svn
Se instalarán los siguientes paquetes NUEVOS:
  git git-man liberror-perl
0 actualizados, 3 nuevos se instalarán, 0 para eliminar y 90 no actualizados.
Se necesita descargar 4.146 kB de archivos.
Se utilizarán 21,0 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] S
Des:1 http://ua.archive.ubuntu.com/ubuntu jammy/main amd64 liberror-perl all 0.17029-1 [26,5 kB]
Des:2 http://ua.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git-man all 1:2.34.1-1ubuntu1.11 [955 kB]
Des:3 http://ua.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git amd64 1:2.34.1-1ubuntu1.11 [3.165 kB]
Descargados 4.146 kB en 5s (761 kB/s)
Seleccionando el paquete liberror-perl previamente no seleccionado.
(Leyendo la base de datos ... 245163 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../liberror-perl_0.17029-1_all.deb ...
Desempaquetando liberror-perl (0.17029-1) ...
Seleccionando el paquete git-man previamente no seleccionado.
Preparando para desempaquetar .../git-man_1%3a2.34.1-1ubuntu1.11_all.deb ...
Desempaquetando git-man (1:2.34.1-1ubuntu1.11) ...
Seleccionando el paquete git previamente no seleccionado.
Preparando para desempaquetar .../git_1%3a2.34.1-1ubuntu1.11_amd64.deb ...
Desempaquetando git (1:2.34.1-1ubuntu1.11) ...
Configurando liberror-perl (0.17029-1) ...
Configurando git-man (1:2.34.1-1ubuntu1.11) ...
Configurando git (1:2.34.1-1ubuntu1.11) ...
Procesando disparadores para man-db (2.10.2-1) ...
valentin@Ubuntu:~$ 
```

- git clone <https://github.com/Leonardosellanes/EKSApp.git>; cd EKSApp

- Ir al directorio database, dentro de la carpeta del backend:
 - cd backend/database
- Crear el archivo referente a la base de datos:
 - touch database.sqlite
- Si sqlite3 no está instalado (chequear mediante sqlite3 --version), ejecutar el siguiente comando:
 - sudo apt install sqlite3
- Accedemos a la base de datos creada mediante el comando:
 - sqlite3 database.sqlite
- Crear cada una de las tablas necesarias para el funcionamiento de la aplicación:

```
CREATE TABLE notes (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    data TEXT NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

```
CREATE TABLE jobs (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    queue TEXT NOT NULL,
    payload TEXT NOT NULL,
    attempts TINYINT UNSIGNED NOT NULL,
    reserved_at INTEGER UNSIGNED,
    available_at INTEGER UNSIGNED NOT NULL,
    created_at INTEGER UNSIGNED NOT NULL
);
```

```
CREATE TABLE job_batches (
    id TEXT PRIMARY KEY,
    name TEXT NOT NULL,
    total_jobs INTEGER NOT NULL,
    pending_jobs INTEGER NOT NULL,
    failed_jobs INTEGER NOT NULL,
    failed_job_ids TEXT NOT NULL,
    options TEXT,
```

```

cancelled_at INTEGER,
created_at INTEGER NOT NULL,
finished_at INTEGER
);

```

```

CREATE TABLE failed_jobs (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    uuid TEXT UNIQUE NOT NULL,
    connection TEXT NOT NULL,
    queue TEXT NOT NULL,
    payload TEXT NOT NULL,
    exception TEXT NOT NULL,
    failed_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

```

```

CREATE TABLE cache (
    key TEXT PRIMARY KEY,
    value TEXT NOT NULL,
    expiration INTEGER NOT NULL
);

```

```

CREATE TABLE cache_locks (
    key TEXT PRIMARY KEY,
    owner TEXT NOT NULL,
    expiration INTEGER NOT NULL
);

```

```

CREATE TABLE users (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    name TEXT NOT NULL,
    email TEXT NOT NULL UNIQUE,
    email_verified_at TIMESTAMP,
    password TEXT NOT NULL,
    remember_token TEXT,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

```

```

CREATE TABLE password_reset_tokens (
    email TEXT PRIMARY KEY,

```

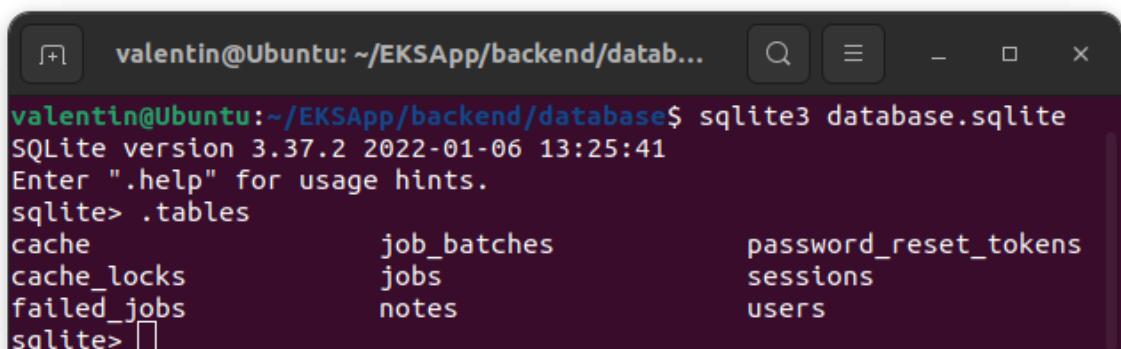
```

token TEXT NOT NULL,
created_at TIMESTAMP
);

CREATE TABLE sessions (
    id TEXT PRIMARY KEY,
    user_id INTEGER,
    ip_address TEXT(45),
    user_agent TEXT,
    payload TEXT,
    last_activity INTEGER,
    FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE SET NULL
);

```

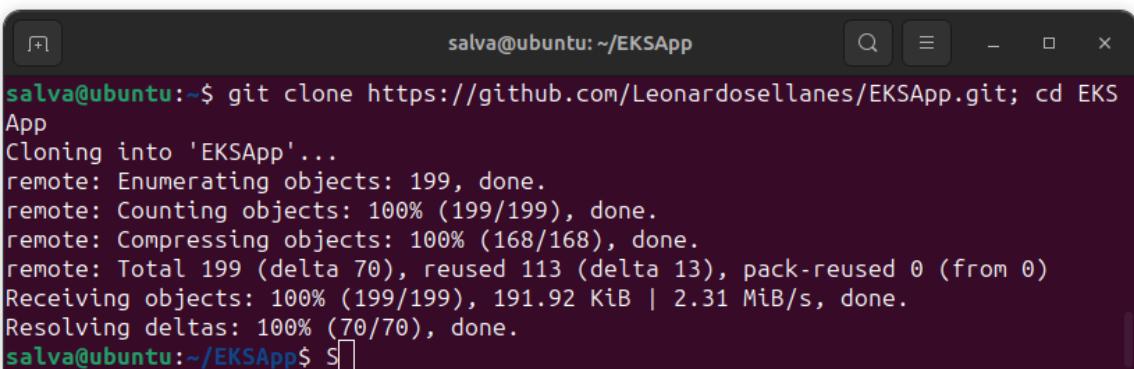
- Revisar la existencia de las tablas en la base de datos:
 - .tables



```

valentin@Ubuntu:~/EKSApp/backend/database$ sqlite3 database.sqlite
SQLite version 3.37.2 2022-01-06 13:25:41
Enter ".help" for usage hints.
sqlite> .tables
cache          job_batches        password_reset_tokens
cache_locks    jobs              sessions
failed_jobs   notes             users
sqlite> 

```



```

salva@ubuntu:~$ git clone https://github.com/Leonardosellanes/EKSApp.git; cd EKS
App
Cloning into 'EKSApp'...
remote: Enumerating objects: 199, done.
remote: Counting objects: 100% (199/199), done.
remote: Compressing objects: 100% (168/168), done.
remote: Total 199 (delta 70), reused 113 (delta 13), pack-reused 0 (from 0)
Receiving objects: 100% (199/199), 191.92 KiB | 2.31 MiB/s, done.
Resolving deltas: 100% (70/70), done.
salva@ubuntu:~/EKSApp$ 

```

- Antes de subir las imágenes, hay que instalar docker a través de los siguientes comandos
 - sudo apt update
 - sudo apt install -y docker.io

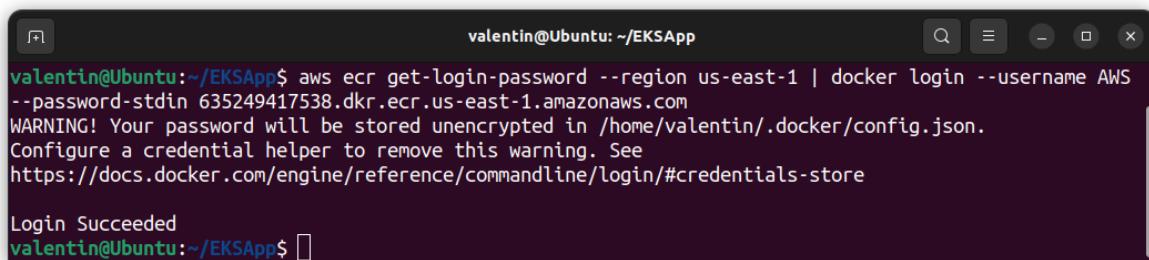
- Despues de instalarlo, hay que ejecutarlo y habilitarlo
 - sudo systemctl start docker
 - sudo systemctl enable docker
- Ya por ultimo, hay que agregar el usuario al grupo de Docker para que se le permita crear y subir imagenes
 - sudo usermod -aG docker \$USER
- Desde la raiz del repositorio, ejecutar el siguiente comando
 - sudo docker-compose build

```

valentin@Ubuntu:~/EKSApp$ sudo docker-compose build
[+] Building 29.2s (8/13)
=> => sha256:43c4264eed91be63b20e617d93e75256a6097070ce643c5e8f0379998b44f170 3.62MB / 3.62MB 3.1s
=> => sha256:bb15916673af8229814dcc44164d4616a6141bc5b58ae9362e741d64c6e4c91 3.28MB / 3.28MB 3.35
=> => extracting sha256:43c4264eed91be63b20e617d93e75256a6097070ce643c5e8f0379998b44f170 0.1s
=> => sha256:9feb3258c4c6a46b7e182fa372bfe03ff37fd0f7f6894ca97b564fc382d7d4 943B / 943B 3.35
=> => extracting sha256:bb15916673af8229814dcc44164d4616a6141bc5b58ae9362e741d64c6e4c91 0.1s
=> => sha256:32e436918c34ed4c536bdbaeace5213a8aaeb790ec65951fe681a10758ae2677 223B / 223B 3.55
=> => extracting sha256:9feb3258c4c6a46b7e182fa372bfe03ff37fd0f7f6894ca97b564fc382d7d4 0.0s
=> => sha256:e567f6a279d6230e9f2dfe3ce224ced757f36a350ad19c01a56514c379db40c 12.51MB / 12.51MB 4.95
=> => extracting sha256:32e436918c34ed4c536bdbaeace5213a8aaeb790ec65951fe681a10758ae2677 0.0s
=> => sha256:c5f15fd3b41bdxfc8b5af169f77fdd27b0b15cf69afe4e260f650cca4d3c80600 498B / 498B 3.75
=> => sha256:4eea53e5dd457c24459f7202e5d629033c0951eb5add7b1e823d763add96b0c8 17.68MB / 17.68MB 5.05
=> => extracting sha256:e567f6a279d6230e9f2dfe3ce224ced757f36a350ad19c01a56514c379db40c 0.1s
=> => sha256:1fc6bbab2995667b500e93d61b33f58e901d5b172577112233732918f53d5fa5b 30.37MB / 30.37MB 5.95
=> => extracting sha256:c5f15fd3b41bdxfc8b5af169f77fdd27b0b15cf69afe4e260f650cca4d3c80600 0.0s
=> => extracting sha256:4eea53e5dd457c24459f7202e5d629033c0951eb5add7b1e823d763add96b0c8 0.75
=> => sha256:ca23a67aeef6035b25a1778f5468293e5e07824ff2b98a23469cc3f592865840 19.71kB / 19.71kB 5.35
=> => sha256:417f9a398261bed58b52188f04ac3f78079c718c4acf213cb149f192c6b4a190 2.44kB / 2.44kB 5.45
=> => sha256:1706c1ce100af5f752296e1a14e35f225a106a961111d8319d948f868b49724b 259B / 259B 5.55
=> => sha256:6361fbcb0b7b59d84450c5dd80d634f85be27bdcc2a77b0d71db5fa429bbe71 1.64MB / 1.64MB 5.85
=> => sha256:fafb2a6d01c7190b7bec8dd92cd4bceafdf842a1b39c06918e64ef504a52c134e 420B / 420B 5.95
=> => sha256:085c4c1b11eac51aa21e64fd938052831127d0c03dac99d2927c86d7a2510659 93B / 93B 6.05
=> => extracting sha256:417f9a398261bed58b52188f04ac3f78079c718c4acf213cb149f192c6b4a190 0.0s
=> => extracting sha256:ca23a67aeef6035b25a1778f5468293e5e07824ff2b98a23469cc3f592865840 0.0s
=> => extracting sha256:1fc6bbab2995667b500e93d61b33f58e901d5b172577112233732918f53d5fa5b 1.35
=> => extracting sha256:1706c1ce100af5f752296e1a14e35f225a106a961111d8319d948f868b49724b 0.05
=> => extracting sha256:6361fbcb0b7b59d84450c5dd80d634f85be27bdcc2a77b0d71db5fa429bbe71 0.0s
=> => extracting sha256:fafb2a6d01c7190b7bec8dd92cd4bceafdf842a1b39c06918e64ef504a52c134e 0.0s
=> => extracting sha256:085c4c1b11eac51aa21e64fd938052831127d0c03dac99d2927c86d7a2510659 0.0s
=> [laravel stage-0 2/7] WORKDIR /var/www/html
=> [laravel stage-0 3/7] RUN apt-get update && apt-get install -y libpng-dev libjpeg-dev 14.3s
=> => # /usr/local/include/php/main -I/usr/local/include/php/TSRM -I/usr/local/include/php/Zend -I/usr/local
=> => # /include/php/ext -I/usr/local/include/php/ext/date/lib -I/usr/include/libpng16 -I/usr/include/freet
=> => # ype2 -fstack-protector-strong -fpic -fpie -O2 -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64 -DHAVE_CO
=> => # NFIG_H -fstack-protector-strong -fpic -fpie -O2 -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64 -D_GNU_
=> => # SOURCE -I/usr/src/php/ext/gd/libgd -DZEND_COMPILE_DL_EXT=1 -c /usr/src/php/ext/gd/libgd/gdfonts.c
=> => # -o libgd/gdfonts.lo -MMD -MF libgd/gdfonts.dep -MT libgd/gdfonts.lo

```

- Subir las imagenes a ECR
 - Primero, ejecutar este comando para autenticar Docker con Amazon ECR
 - aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin <account_id>.dkr.ecr.us-east-1.amazonaws.com
 - El <account_id> se obtiene desde la consola ejecutando el siguiente comando: aws sts get-caller-identity --query Account --output text

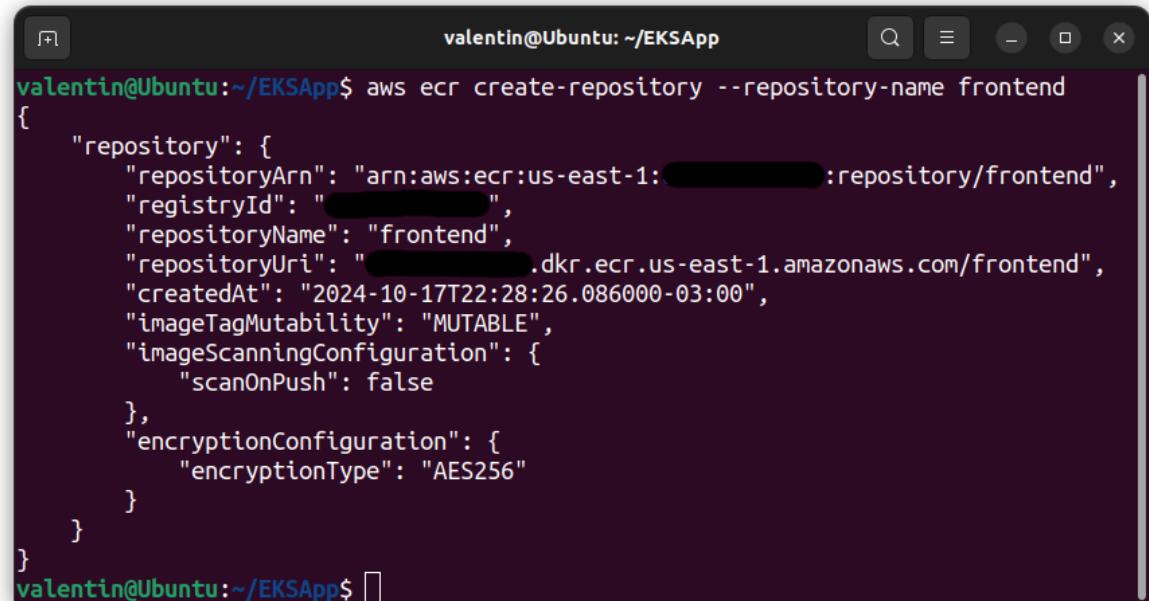


```
valentin@Ubuntu:~/EKSApp$ aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin 635249417538.dkr.ecr.us-east-1.amazonaws.com
WARNING! Your password will be stored unencrypted in /home/valentin/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
valentin@Ubuntu:~/EKSApp$
```

Subir las imágenes a los repositorios

- Crear repositorios para el front y el back
 - aws ecr create-repository --repository-name frontend
 - aws ecr create-repository --repository-name backend
 - Cabe aclarar que al ejecutar estos comandos la consola quedará esperando una entrada por si se quieren modificar los archivos de configuración del repositorio. Se deben dejar por defecto, para lo que se tendrá que presionar “Enter” y después “q”.



```
valentin@Ubuntu:~/EKSApp$ aws ecr create-repository --repository-name frontend
{
  "repository": {
    "repositoryArn": "arn:aws:ecr:us-east-1:████████:repository/frontend",
    "registryId": "████████",
    "repositoryName": "frontend",
    "repositoryUri": "████████.dkr.ecr.us-east-1.amazonaws.com/frontend",
    "createdAt": "2024-10-17T22:28:26.086000-03:00",
    "imageTagMutability": "MUTABLE",
    "imageScanningConfiguration": {
      "scanOnPush": false
    },
    "encryptionConfiguration": {
      "encryptionType": "AES256"
    }
  }
}
valentin@Ubuntu:~/EKSApp$
```

- A continuación, ejecutar los siguientes comandos para verificar y etiquetar las imágenes creadas
 - docker images

```
valentin@Ubuntu:~/EKSApp$ docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
eksapp-nuxt    latest    356742080c7c  27 seconds ago  380MB
eksapp-laravel latest    289dc777a70f  About a minute ago  603MB
valentin@Ubuntu:~/EKSApp$
```

Aclaración: “eksapp-nuxt” es la imagen del frontend, mientras que “eksapp-laravel” es la imagen del backend.

- docker tag eksapp-nuxt:latest
<account_id>.dkr.ecr.us-east-1.amazonaws.com/frontend:latest
- docker tag eksapp-laravel:latest
<account_id>.dkr.ecr.us-east-1.amazonaws.com/backend:latest

```
valentin@Ubuntu:~/EKSApp$ docker tag eksapp-nuxt:latest [REDACTED].dkr.ecr.us-east-1.amazonaws.com/frontend:latest
valentin@Ubuntu:~/EKSApp$ docker tag eksapp-laravel:latest [REDACTED].dkr.ecr.us-east-1.amazonaws.com/backend:latest
valentin@Ubuntu:~/EKSApp$
```

- Por último, subir las imágenes al repositorio con los siguientes comandos:
 - docker push <account_id>.dkr.ecr.us-east-1.amazonaws.com/frontend
 - docker push <account_id>.dkr.ecr.us-east-1.amazonaws.com/backend

```
valentin@Ubuntu:~/EKSApp$ docker push [REDACTED].dkr.ecr.us-east-1.amazonaws.com/frontend:latest
The push refers to repository [REDACTED].dkr.ecr.us-east-1.amazonaws.com/frontend]
e3eb88f2d6f1: Pushed
2eb32007d2a1: Pushed
a93612eb0e96: Pushed
0d98b3b633b2: Pushed
04f5f57834ba: Pushed
70736de621ab: Pushed
9e884bd72188: Pushed
63ca1fbb43ae: Pushed
latest: digest: sha256:9e62155b1f893f458b28a8ad8e96f21e51a244399234711e81ec91b93936542c size: 1998
valentin@Ubuntu:~/EKSApp$ docker push [REDACTED].dkr.ecr.us-east-1.amazonaws.com/backend:latest
The push refers to repository [REDACTED].dkr.ecr.us-east-1.amazonaws.com/backend]
869a8bcb131: Pushed
e47f49ecf366: Pushed
96c897668e09: Pushed
ed0444e21c9e: Pushed
186ddda5d0a7: Pushed
5f70bf18a086: Pushed
2f05cea538d0: Pushed
972e7c106c22: Pushed
74c97afc6571: Pushing 54.98MB/125.2MB
6b0a5b1180d2: Pushed
74c97afc6571: Pushing 125.7MB
74c97afc6571: Pushed
305a74163daa: Pushed
54634b9aecb1: Pushing 2.127MB/315.6MB
fa8de34729f6: Pushed
98b5f35ea9d3: Pushed
latest: digest: sha256:3f639f04b97a4aa8d74f848e0a834d51c24257176e22be2d6ea273c4d8e5bd47 size: 3460
valentin@Ubuntu:~/EKSApp$
```

- Para comprobar que los repositorios quedaron subidos correctamente, hay que acceder a la interfaz de AWS e ir al servicio “Elastic Container Registry”, donde deberían estar los repositorios “frontend” y “backend” con sus respectivas imágenes.

Nombre del repositorio	URI	Creado en	Immutabilidad de etiqueta	Tipo de cifrado
backend	635249417538.dkr.ecr.us-east-1.amazonaws.com/backend	17 de octubre de 2024, 22:36:21 (UTC-03)	Mutable	AES-256
frontend	635249417538.dkr.ecr.us-east-1.amazonaws.com/frontend	17 de octubre de 2024, 22:28:26 (UTC-03)	Mutable	AES-256

Imagen	Tipo de artefacto	Envío a	Tamaño (MB)	URI de imagen	Resumir
latest	Image	30 de octubre de 2024, 15:50:45 (UTC-03)	213.62	Copiar URI	sha256:795f42d0d72280d594f658a5d50b6...

Imagen	Tipo de artefacto	Envío a	Tamaño (MB)	URI de imagen	Resumir
latest	Image	30 de octubre de 2024, 15:57:38 (UTC-03)	47.84	Copiar URI	sha256:153a7428e7db725508bd90ef7d755...

4. Desplegar la aplicación

Creación del archivo configuración (yaml) de la base de datos

- Crear el archivo de despliegue de la base de datos
 - nano postgres-pv.yaml

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: postgres-pv
spec:
  capacity:
    storage: 1Gi # Cambia esto según tus necesidades
  accessModes:
    - ReadWriteOnce
  hostPath:
    path: /mnt/data/postgres # Cambia esto a una ruta válida en tu nodo
  persistentVolumeReclaimPolicy: Retain
  storageClassName: manual # Esto debe coincidir con tu PVC
```

- Aplicar el archivo postgres-pv.yaml
 - **ACLARACIÓN:** si se copia el contenido del postgres-pv.yaml directamente desde este documento puede quedar mal indexado y no ser reconocido.
 - Desplegar los recursos definidos en el archivo postgres-pv.yaml utilizando el siguiente comando: kubectl apply -f postgres-pv.yaml

Creación del archivo configuración (yaml) del servicio

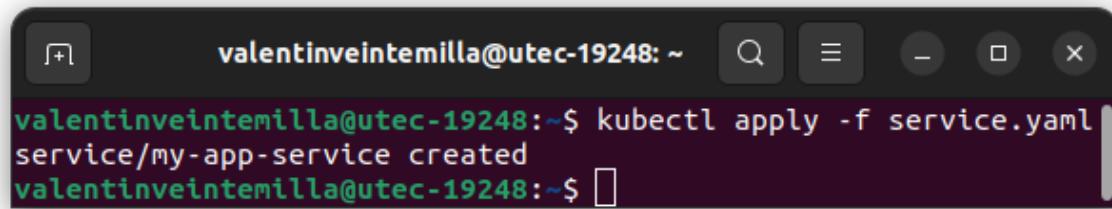
- Crear el archivo service.yaml
 - Crear el archivo utilizando el comando: nano service.yaml
 - Copiar y pegar el siguiente texto en el archivo donde:
 - apiVersion: v1: Define la versión de la API de Kubernetes que se usará.
 - ports.port: Es el puerto público donde la aplicación será accesible.
 - ports.targetPort: Es el puerto donde la aplicación escucha dentro del contenedor.

```

apiVersion: v1
kind: Service
metadata:
  name: laravel-service
spec:
  type: LoadBalancer
  ports:
    - port: 8000
      targetPort: 8000
  selector:
    app: laravel
---
apiVersion: v1
kind: Service
metadata:
  name: nuxt-service
spec:
  type: LoadBalancer
  ports:
    - port: 3000
      targetPort: 3000
  selector:
    app: nuxt
---
apiVersion: v1
kind: Service
metadata:
  name: postgres-service
spec:
  ports:
    - port: 5432
      targetPort: 5432
  selector:
    app: postgres

```

- Aplicar el archivo service.yaml
 - **ACLARACIÓN:** si se copia el contenido del service.yaml directamente desde este documento puede quedar mal indexado y no ser reconocido.
 - Ejecutar el siguiente comando: kubectl apply -f service.yaml



```
valentinveintemilla@utec-19248:~$ kubectl apply -f service.yaml
service/my-app-service created
valentinveintemilla@utec-19248:~$ 
```

Creación del archivo configuración (yaml) del despliegue

- Crear el archivo deployment.yaml:
 - Crear el archivo utilizando el comando: nano deployment.yaml
- Especificar el contenido del deployment:
 - Insertar el siguiente texto en el archivo, reemplazando <aws-account-id>, <external-ip-de-tu-servicio-laravel>, y ajustar los puertos. Las imágenes frontend y backend deben ser las mismas subidas previamente a ECR en el paso 3:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: laravel-deployment
spec:
  replicas: 1
  selector:
    matchLabels:
      app: laravel
  template:
    metadata:
      labels:
        app: laravel
    spec:
      containers:
        - name: laravel-app
          image: <aws-account-id>.dkr.ecr.us-east-1.amazonaws.com/backend:latest
          ports:
            - containerPort: 8000
          env:
            - name: APP_ENV
```

```

    value: "local"
  - name: APP_DEBUG
    value: "true"
  - name: DB_CONNECTION
    value: "sqlite"
  - name: DB_HOST
    value: "postgres-service"
  - name: DB_PORT
    value: "5432"
  - name: DB_DATABASE
    value: "eks_notes"
  - name: DB_USERNAME
    value: "root"
  - name: DB_PASSWORD
    value: "root"

---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nuxt-deployment
spec:
  replicas: 1
  selector:
    matchLabels:
      app: nuxt
  template:
    metadata:
      labels:
        app: nuxt
  spec:
    containers:
      - name: nuxt-app
        image: <aws-account-id>.dkr.ecr.us-east-1.amazonaws.com/frontend:latest
        ports:
          - containerPort: 3000
        env:
          - name: NUXT_PUBLIC_API_BASE
            value: "http://<external-ip-de-tu-servicio-laravel>:8000/api"
---
apiVersion: apps/v1

```

```

kind: Deployment
metadata:
  name: postgres-deployment
spec:
  replicas: 1
  selector:
    matchLabels:
      app: postgres
  template:
    metadata:
      labels:
        app: postgres
    spec:
      containers:
        - name: postgres-db
          image: postgres:13
          ports:
            - containerPort: 5432
      env:
        - name: POSTGRES_DB
          value: "eks_notes"
        - name: POSTGRES_USER
          value: "root"
        - name: POSTGRES_PASSWORD
          value: "root"
      volumeMounts:
        - name: postgres-storage
          mountPath: /var/lib/postgresql/data
  volumes:
    - name: postgres-storage
      persistentVolumeClaim:
        claimName: postgres-pvc

```

```

---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: postgres-pvc
spec:
  accessModes:
    - ReadWriteOnce

```

```

resources:
requests:
  storage: 1Gi # Debe coincidir con el tamaño del PV
  storageClassName: manual # Debe coincidir con el PV

```

- Datos:
 - Para obtener el <aws-account-id> se utiliza el comando previamente proporcionado (aws sts get-caller-identity --query Account --output text).
 - Para obtener el <external-ip-de-tu-servicio-laravel> se tiene que ejecutar “kubectl get services” y copiar el external ip del servicio de laravel.
 - **ACLARACIÓN:** si se copia el contenido del deployment.yaml directamente desde este documento puede quedar mal indexado y no ser reconocido.
- Aplicar el archivo deployment.yaml
 - Desplegar los recursos definidos en el archivo deployment.yaml utilizando el siguiente comando: kubectl apply -f deployment.yaml

```

valentinveintemilla@utec-19248:~$ kubectl apply -f deployment.yaml
deployment.apps/my-app created
valentinveintemilla@utec-19248:~$ 

```

- Verificar el estado del despliegue
 - Ejecutar el siguiente comando (los despliegues deberían ser visibles en la lista): kubectl get deployments

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
laravel-deployment	1/1	1	1	19m
nuxt-deployment	1/1	1	1	19m
postgres-deployment	1/1	1	1	19m

- Verificar los pods
 - Para ver los pods creados ejecute el siguiente comando: kubectl get pods

NAME	READY	STATUS	RESTARTS	AGE
laravel-deployment-554b4d5c48-2ljsb	0/1	Pending	0	100s
nuxt-deployment-74dc566c45-kblpf	0/1	Pending	0	100s

5. Exponer la aplicación

Creación de un grupo de nodos

- Crear un grupo de nodos desde la interfaz de AWS
 - Buscar EKS en la interfaz
 - Seleccionar EKS
 - Ingresamos a cluster_obligatorio
 - Ingresamos a informática
 - En la sección “grupos de nodos” seleccionar “agregar grupo de nodos”

The screenshot shows the AWS EKS console with the following details:

- General Cluster Information:** Estado: Activo, Versión de Kubernetes: 1.30, Período de soporte: Soporte estándar hasta el 28 de julio de 2025, Proveedor: EKS.
- Informática Tab:** Shows the 'Nodos' section with 0 nodes. A message says: "Este clúster no tiene ningún Nodos o usted no tiene permiso para verlos."
- Grupos de nodos (1) Tab:** Shows a single node group named 'nodos_obligatorio' with 3 desired nodes, AMI version 1.30.4-20241011, and status 'Creando'.
- Perfiles de Fargate Tab:** Shows 0 profiles, with a message: "Este clúster no tiene ningún perfil de Fargate." A button 'Agregar perfil de Fargate' is present.

- Establecer un nombre
- Presionar siguiente
- Tipo de instancia: t3.medium
- Configuración de escalado del grupo de nodos: 2 en mínimo y esperado, 4 en máximo
- El resto por defecto
- Presionar siguiente
- Dejar las subredes por defecto
- Presionar siguiente
- Presionar crear

Configurar grupo de nodos

Nombre: **nodos_obligatorio**

Role de IAM de nodo: **LabRole**

Plantilla de lanzamiento: **Utilizar la plantilla de lanzamiento**

Etiquetas de Kubernetes:

Especificar redes

Subredes:

- subnet-06721929ba4888f7b (us-east-1f) 172.31.64.0/20
- subnet-0c9defab5e46f097f5 (us-east-1a) 172.31.80.0/20
- subnet-0fa8a931b2ca8fb2 (us-east-1b) 172.31.16.0/20
- subnet-01daaa315fc0cd91 (us-east-1c) 172.31.32.0/20
- subnet-dac910e1dd5d8f51 (us-east-1d) 172.31.0.0/20

Configuración de escalado del grupo de nodos:

- Tamaño deseado: 3 nodos
- Tamaño mínimo: 3 nodos
- Tamaño máximo: 3 nodos

Configuración de red del grupo de nodos:

Subredes: **seleccionar subredes**

Configurar el acceso remoto a los nodos: **Configurar**

- Una vez creado el grupo de nodos, los nodos correspondientes a la aplicación se agregarán automáticamente. Es necesario esperar unos minutos para que los nodos correspondientes pasen de “no preparado” a “preparado”.

Información del clúster

Nombre del nodo	Tipo de instancia	Grupo de nodos	Creado	Estado
ip-172-31-28-11.ec2.internal	t1.micro	nodos_obligatorio	Creado hace unos segundos	No preparado
ip-172-31-76-88.ec2.internal	t1.micro	nodos_obligatorio	Creado hace unos segundos	No preparado
ip-172-31-95-236.ec2.internal	t1.micro	nodos_obligatorio	Creado hace unos segundos	No preparado

Grupos de nodos (1)

Nombre del grupo	Tamaño deseado	Versión de lanzamiento de la AMI	Plantilla de lanzamiento	Estado
nodos_obligatorio	3	1.30.4-20241011	-	Creando

Perfiles de Fargate (0)

Información del clúster

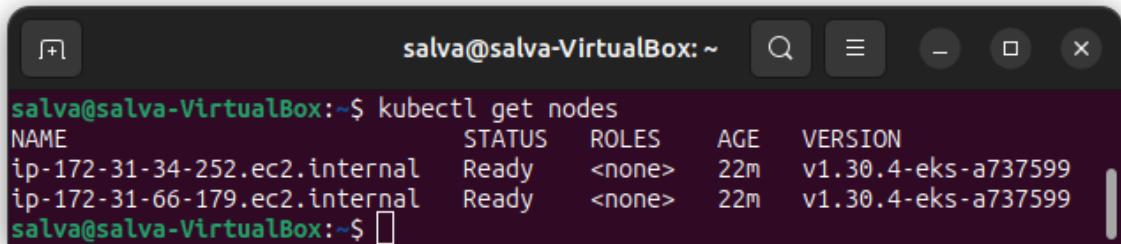
Nombre del nodo	Tipo de instancia	Grupo de nodos	Creado	Estado
ip-172-31-28-11.ec2.internal	t1.micro	nodos_obligatorio	Creado hace unos segundos	Preparado
ip-172-31-76-88.ec2.internal	t1.micro	nodos_obligatorio	Creado hace unos segundos	Preparado
ip-172-31-95-236.ec2.internal	t1.micro	nodos_obligatorio	Creado hace unos segundos	Preparado

Grupos de nodos (1)

Nombre del grupo	Tamaño deseado	Versión de lanzamiento de la AMI	Plantilla de lanzamiento	Estado
nodos_obligatorio	3	1.30.4-20241011	-	Activo

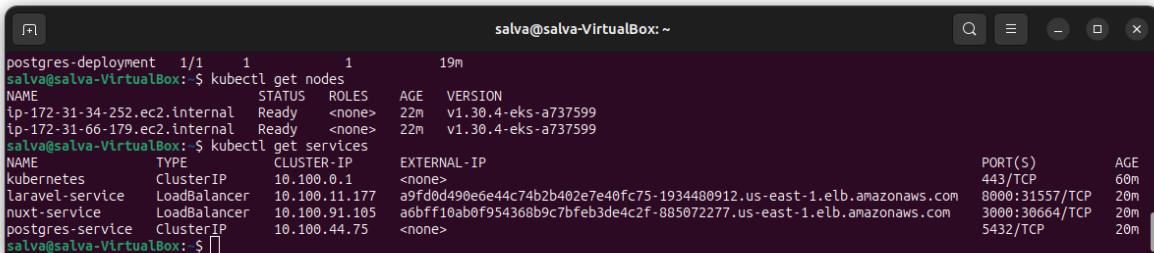
Perfiles de Fargate (0)

- Luego de ejecutar el deployment.yaml y el service.yaml, los nodos se comenzarán a construir en el grupo de nodos previamente creado. Para verificar el proceso se pueden ejecutar los comandos:
 - kubectl get nodes
 - kubectl get pods



```
salva@salva-VirtualBox:~$ kubectl get nodes
NAME           STATUS   ROLES      AGE   VERSION
ip-172-31-34-252.ec2.internal   Ready    <none>    22m   v1.30.4-eks-a737599
ip-172-31-66-179.ec2.internal   Ready    <none>    22m   v1.30.4-eks-a737599
salva@salva-VirtualBox:~$
```

- Verificar el estado del servicio
 - Ejecutar el comando: `kubectl get services`
 - El campo debajo de EXTERNAL-IP corresponde a la IP a utilizar para acceder a la aplicación.



```
postgres-deployment 1/1 1 1 19m
salva@salva-VirtualBox:~$ kubectl get nodes
NAME           STATUS   ROLES      AGE   VERSION
ip-172-31-34-252.ec2.internal   Ready    <none>    22m   v1.30.4-eks-a737599
ip-172-31-66-179.ec2.internal   Ready    <none>    22m   v1.30.4-eks-a737599
salva@salva-VirtualBox:~$ kubectl get services
NAME          TYPE        CLUSTER-IP   EXTERNAL-IP          PORT(S)        AGE
kubernetes    ClusterIP  10.100.0.1   <none>              443/TCP       60m
laravel-service LoadBalancer 10.100.11.177  a9fd0d490e6e44c74b2b402e7e40fc75-1934480912.us-east-1.elb.amazonaws.com  8006:31557/TCP  20m
nuxt-service   LoadBalancer 10.100.91.105  a6bff1ab0f954368b9c7bfeb3de4c2f-885072277.us-east-1.elb.amazonaws.com  3000:30664/TCP  20m
postgres-service ClusterIP  10.100.44.75  <none>              5432/TCP       20m
salva@salva-VirtualBox:~$
```

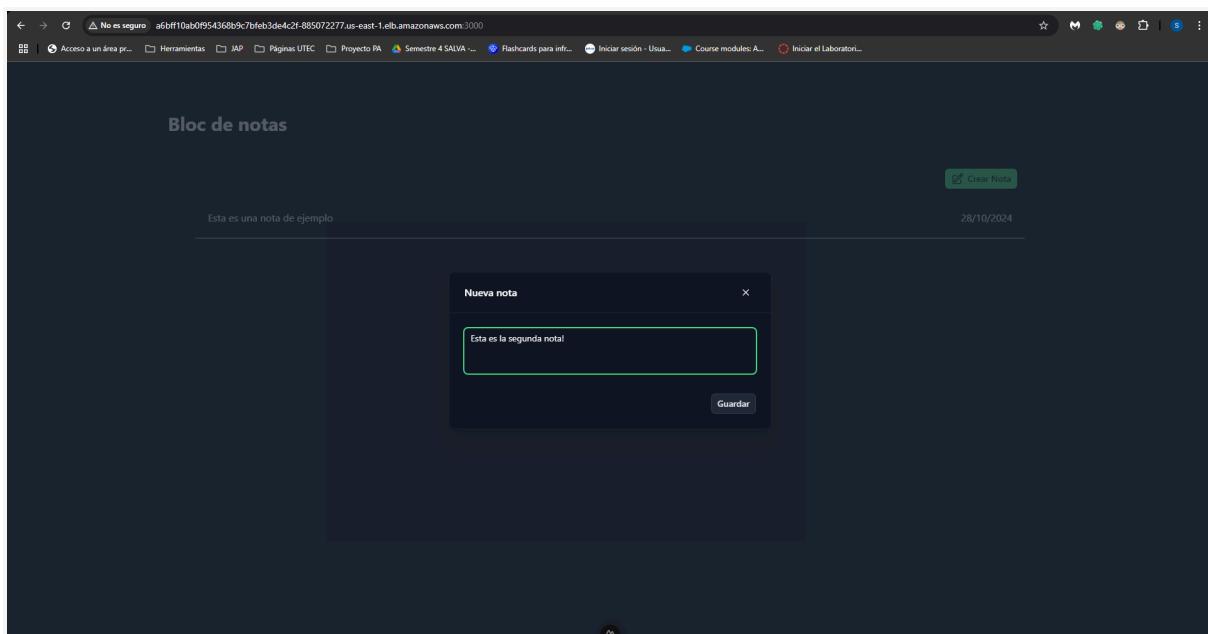
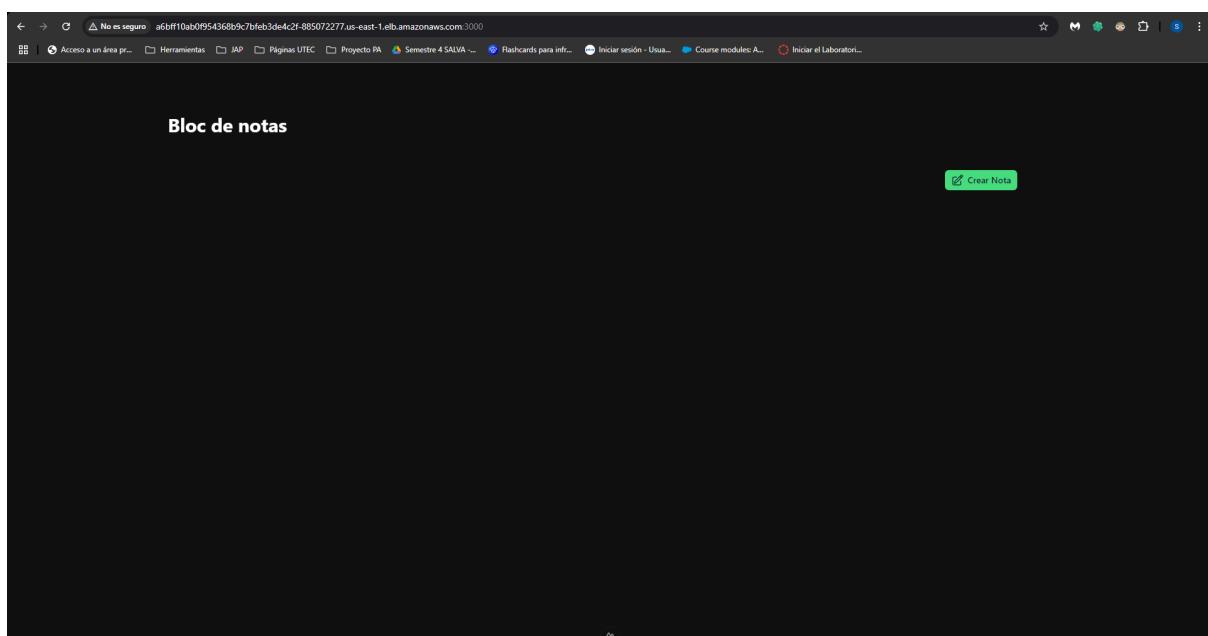
Activación de la base de datos

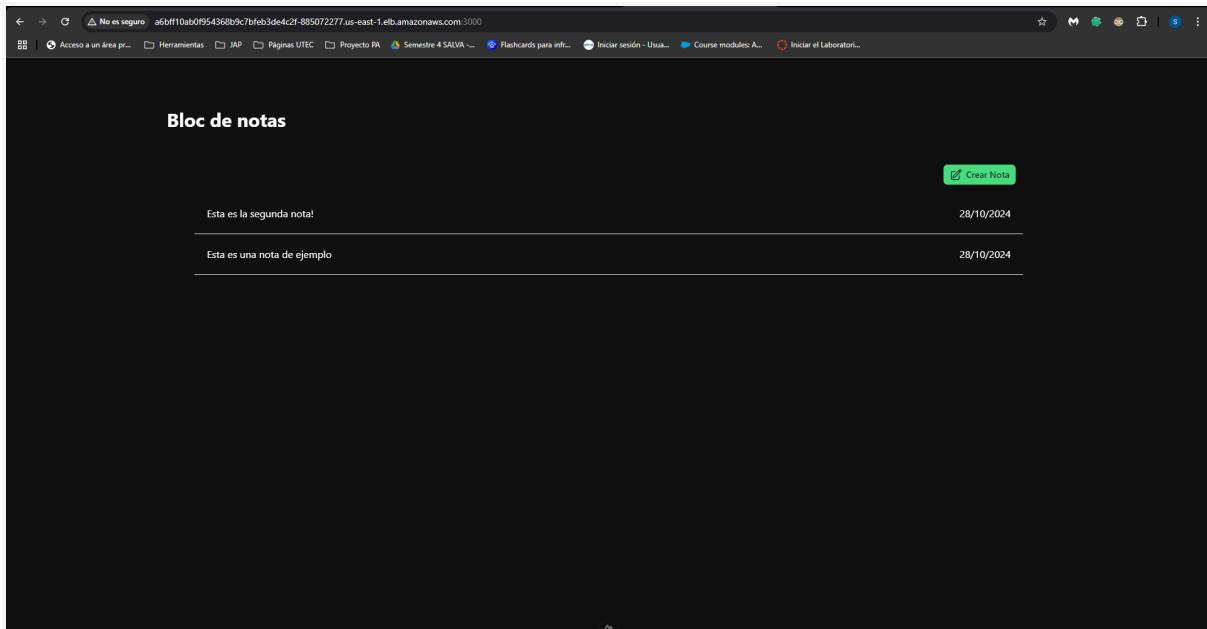
- Chequear la existencia de los pods, además de que los mismos se encuentren en estado Running:
 - `kubectl get pods`
- Ingresa mediante SSH al pod donde se encuentra el backend (contiene “laravel” en su nombre):
 - `kubectl exec -it <nombre-del-pod-de-laravel> -- /bin/bash`
- Luego, ingresa al directorio “database”:
 - `cd database`
- Aquí, chequea nuevamente que sqlite3 esté instalado (mediante `sqlite3 --version`), y en caso de no estarlo, ejecutar el siguiente comando:
 - `apt install sqlite3`
- Accedemos a la base de datos creada anteriormente:
 - `sqlite3 database.sqlite`
- Finalmente, revisamos la existencia de las tablas en la base de datos:
 - `.tables`

6. Prueba de la aplicación

Prueba de funcionamiento de la aplicación

- Acceder a la aplicación
 - Abrir el navegador web y escribir la **EXTERNAL-IP** en la barra de direcciones. Si todo está bien configurado, la aplicación se mostrará en funcionamiento.



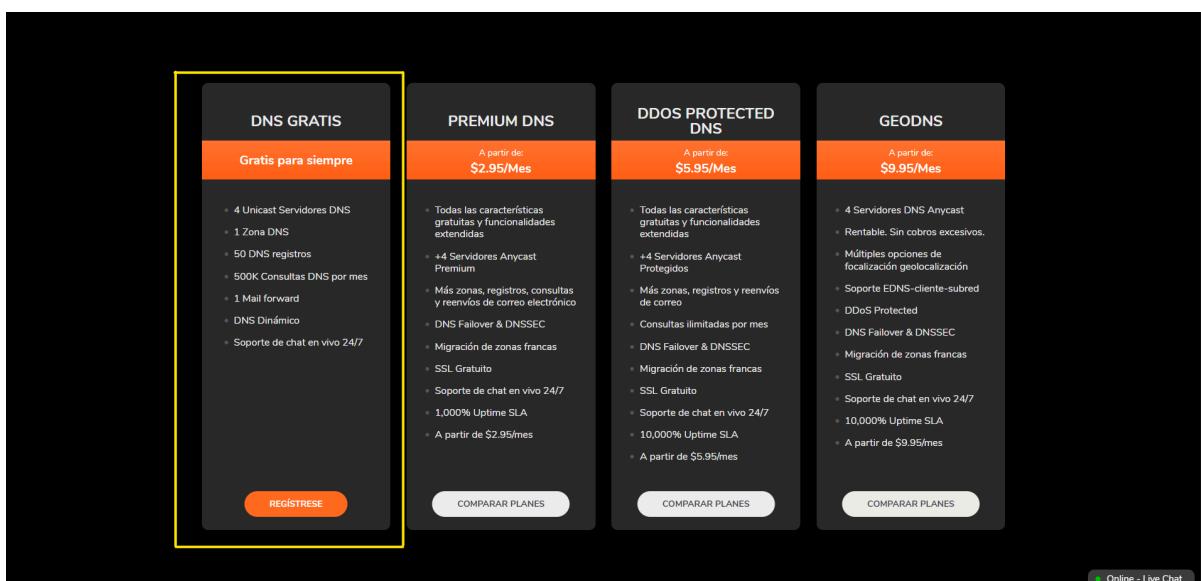


- Se deberían poder agregar notas como se observa en las imágenes anteriores, y si se recarga la página o se accede desde otro dispositivo, se deberían seguir viendo correctamente.

7. Certificado SSL/TLS

Creación del DNS

- Seleccionar un servicio de alojamiento de DNS:
 - En este caso, utilizaremos ClouDNS (<https://www.cloudns.net/index/lang/es/>).
 - Seleccionamos la opción “DNS GRATIS” y nos registramos con la información solicitada.

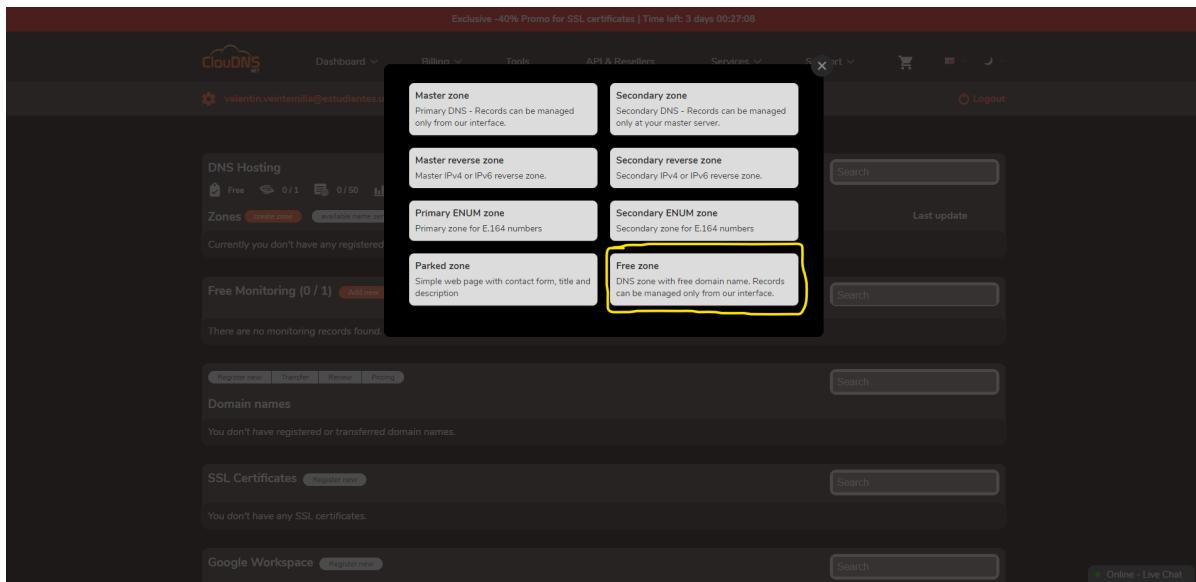


- Revisar y aceptar el correo para activar la cuenta correspondiente.
- Cliqueamos en “create zone” para comenzar.

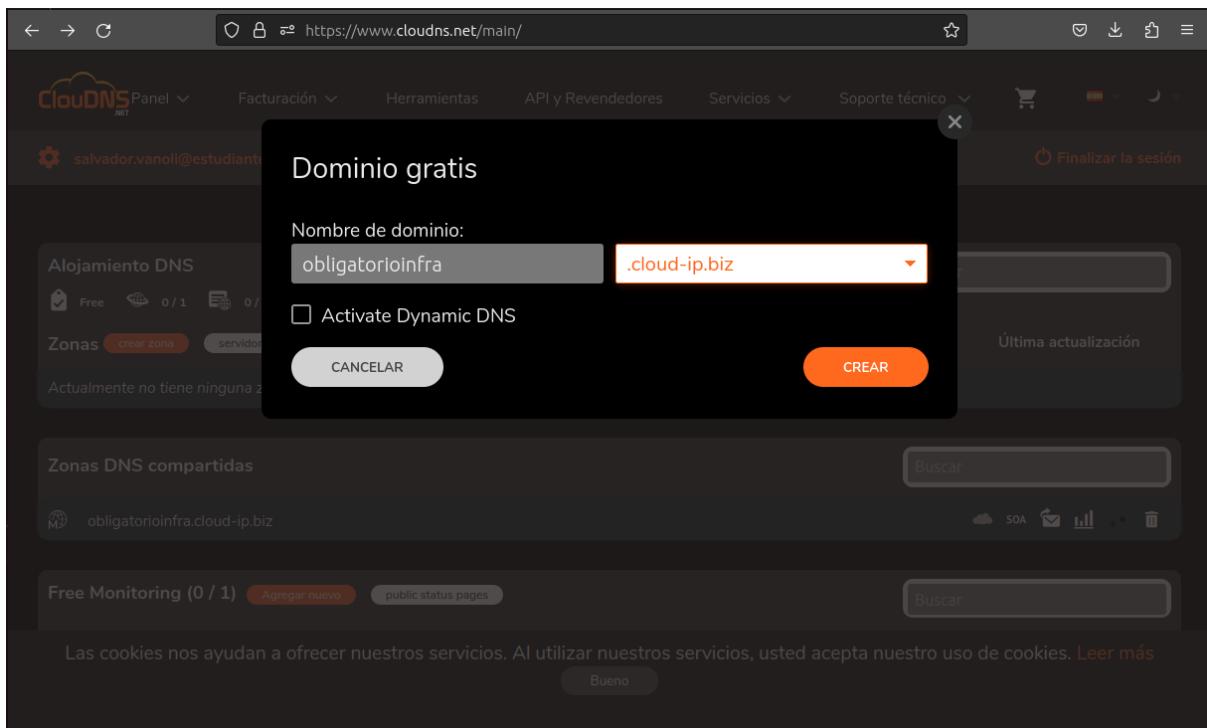
The screenshot shows the ClouDNS dashboard with the following sections:

- DNS Hosting:** Shows 0 registered zones. A red box highlights the "create zone" button.
- Free Monitoring:** Shows 0 monitoring records found.
- Domain names:** Shows 0 registered or transferred domain names.
- SSL Certificates:** Shows 0 SSL certificates.
- Google Workspace:** Shows 0 Google Workspace accounts.

- Luego aparecerá un modal, en el que debemos seleccionar la opción “Free zone”.



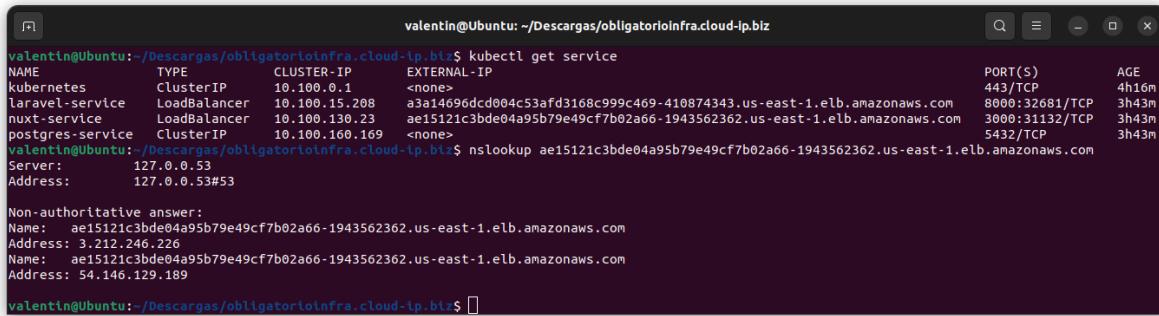
- En este momento, será necesario especificar el dominio de la aplicación (nombre que al ser introducido en la URL, deberá redirigir a la aplicación correspondiente). En el desplegable, debemos elegir la opción “.cloud-ip.biz”.
- Luego, presionamos “CREATE”.



- En la siguiente ventana, presionamos el botón “+ Add new record”.

- Aquí, es importante establecer el tipo “A”, además de especificar la IP que hace referencia al FrontEnd de la aplicación en el apartado de “Points to”.

- Para obtener la IP pública del clúster, se debe conseguir la URL del pod del frontend y ejecutar el comando:
 - nslookup <URL del service del frontend>
 - Recordatorio: para obtener el URL del service del frontend se usa el comando: kubectl get service

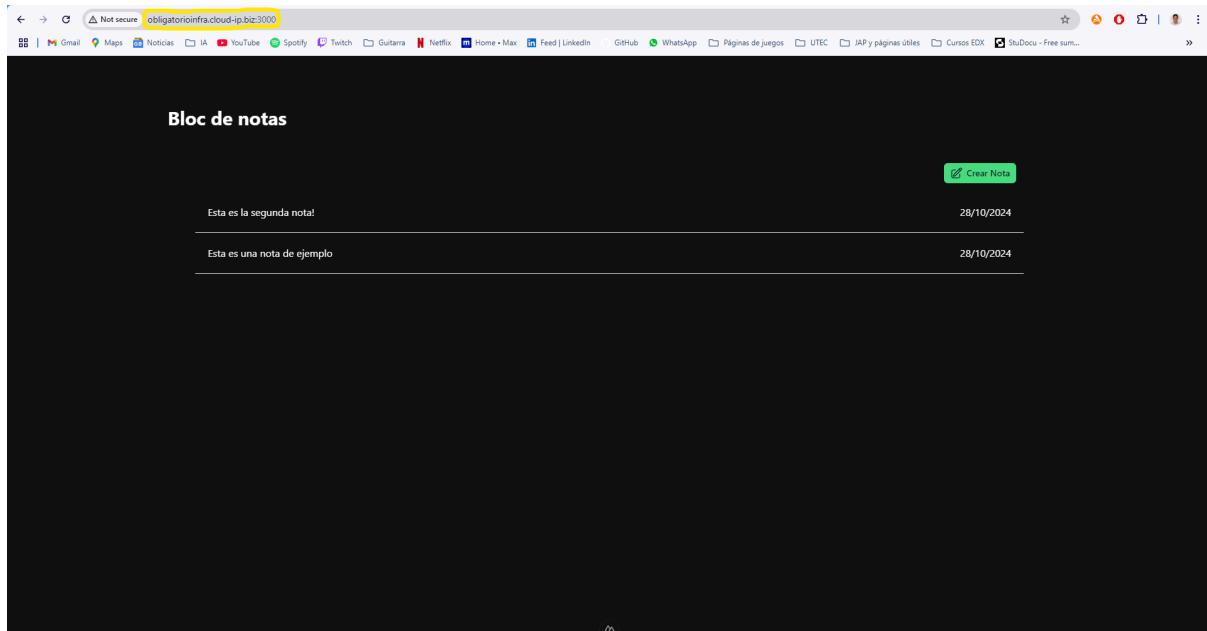


```

valentin@Ubuntu:~/Descargas/obligatorioinfra.cloud-ip.biz$ kubectl get service
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP
kubernetes     ClusterIP  10.100.0.1   <none>
laravel-service LoadBalancer 10.100.15.208  a3a14696dcd004c53af3168c999c469-410874343.us-east-1.elb.amazonaws.com
nuxt-service    LoadBalancer 10.100.130.23  ae15121c3bde04a95b79e49cf7b02a66-1943562362.us-east-1.elb.amazonaws.com
postgres-service ClusterIP  10.100.160.169  <none>
valentin@Ubuntu:~/Descargas/obligatorioinfra.cloud-ip.biz$ nslookup ae15121c3bde04a95b79e49cf7b02a66-1943562362.us-east-1.elb.amazonaws.com
Server:        127.0.0.53
Address:       127.0.0.53#53

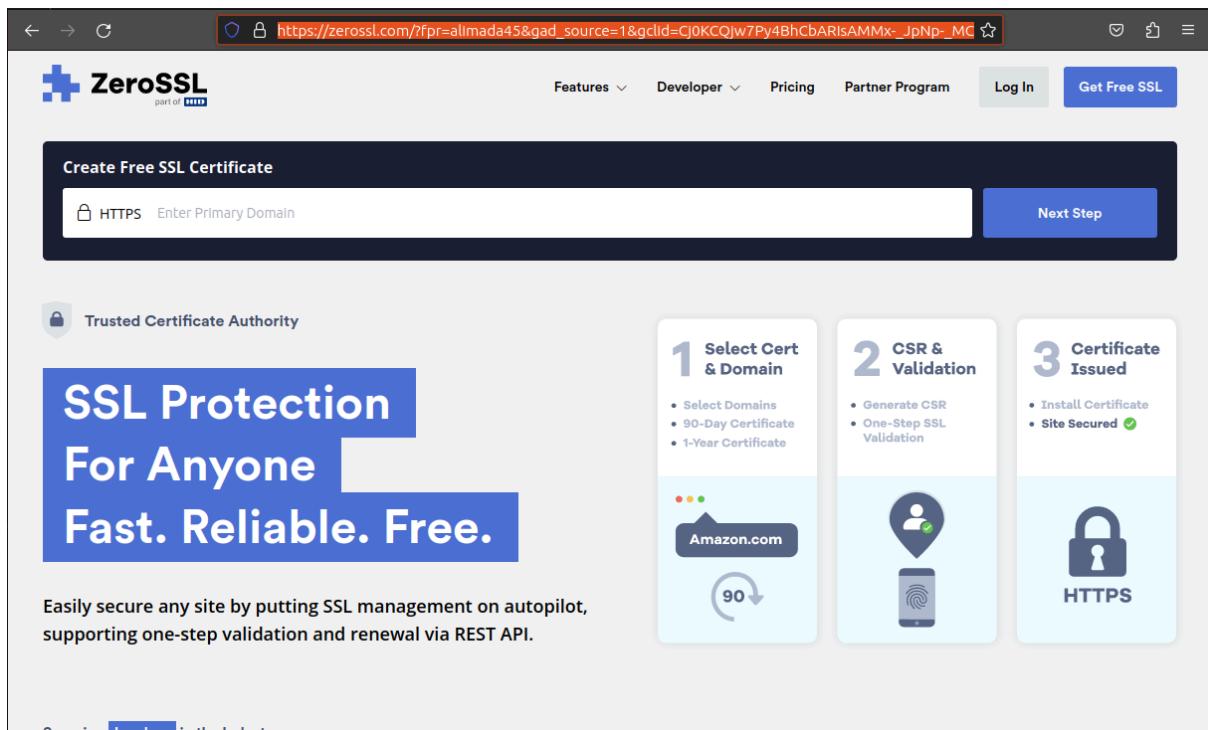
Non-authoritative answer:
Name:  ae15121c3bde04a95b79e49cf7b02a66-1943562362.us-east-1.elb.amazonaws.com
Address: 3.212.246.226
Name:  ae15121c3bde04a95b79e49cf7b02a66-1943562362.us-east-1.elb.amazonaws.com
Address: 54.146.129.189
valentin@Ubuntu:~/Descargas/obligatorioinfra.cloud-ip.biz$ 

```

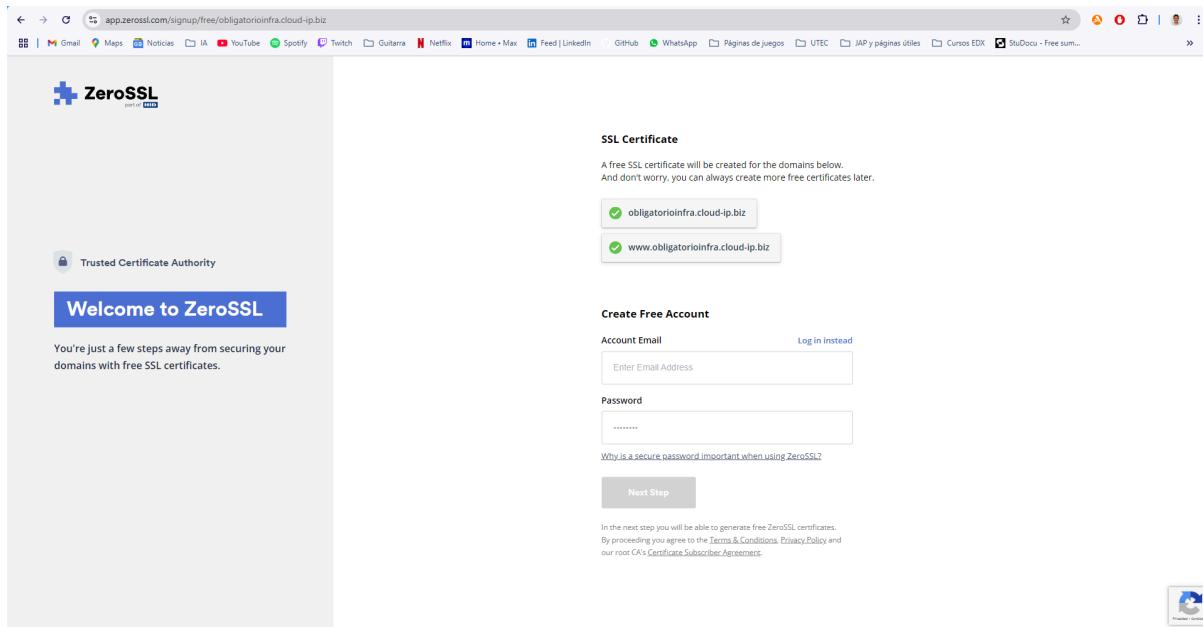


Creación y aplicación del certificado SSL

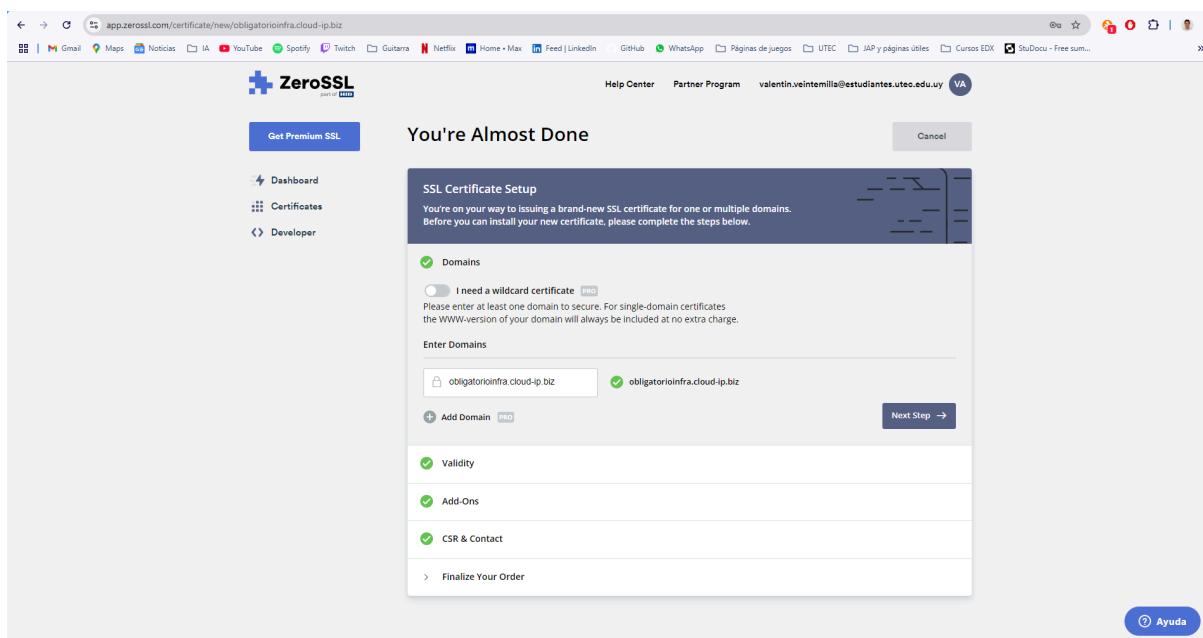
- Accede a ZeroSSL a través del siguiente enlace:
https://zerossl.com/?fpr=alimada45&gad_source=1&gclid=Cj0KCQjw7Py4BhCbARIsAMMx-JpNp-MCkIYjoS9rynxSoye8T35SKvWjk4Us7CUSyzzbXQlhk27gaAiSkEALw_wcB



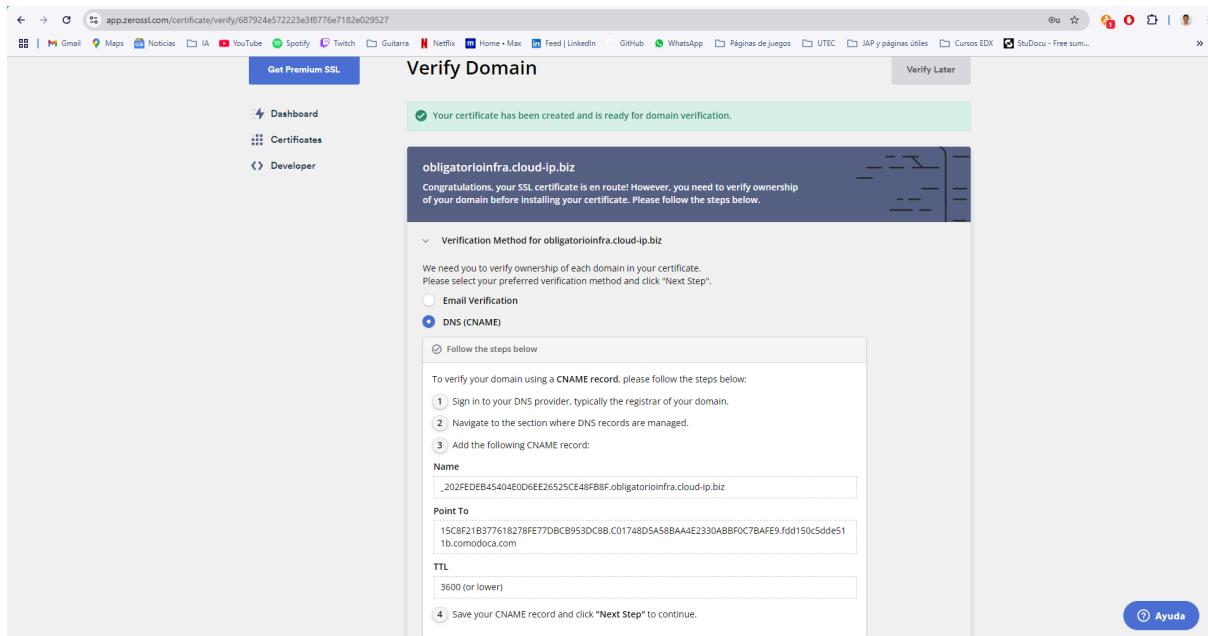
- Pegar la ruta de nuestra página en el input que dice “Enter Primary Domain” y presionar “Next Step”.



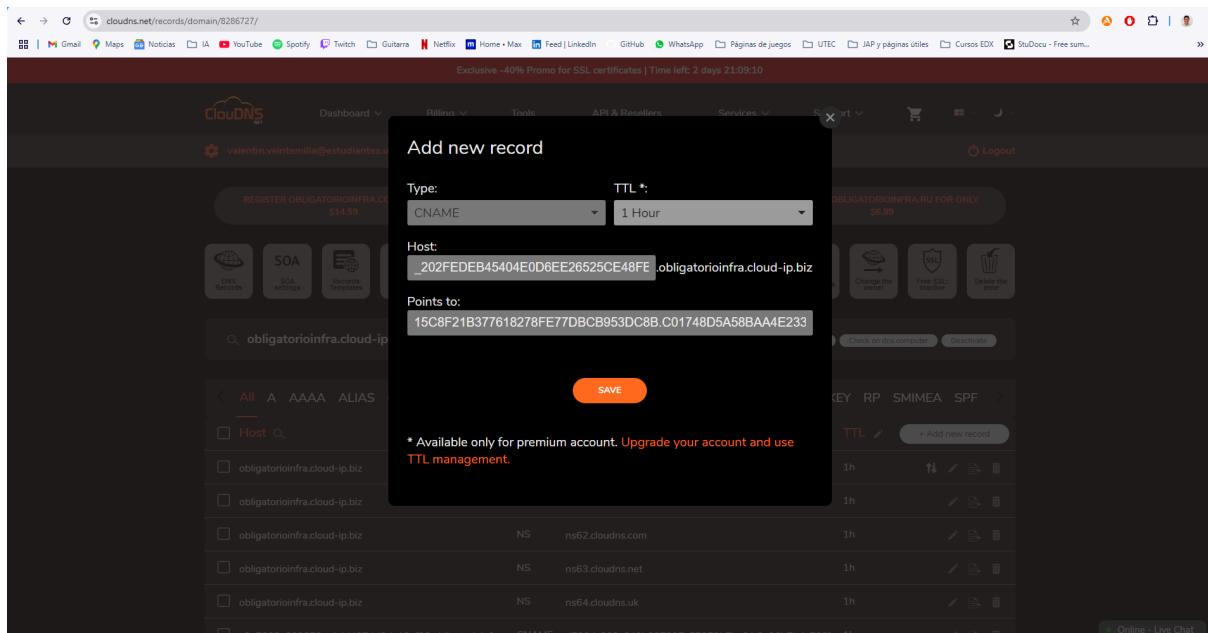
- Para continuar, inicia sesión o crea una cuenta con los datos solicitados.



- De las siguientes secciones, no hace falta modificar ninguna, excepto:
 - Validity: 90-Day Certificate
 - Finalize Your Order: Free



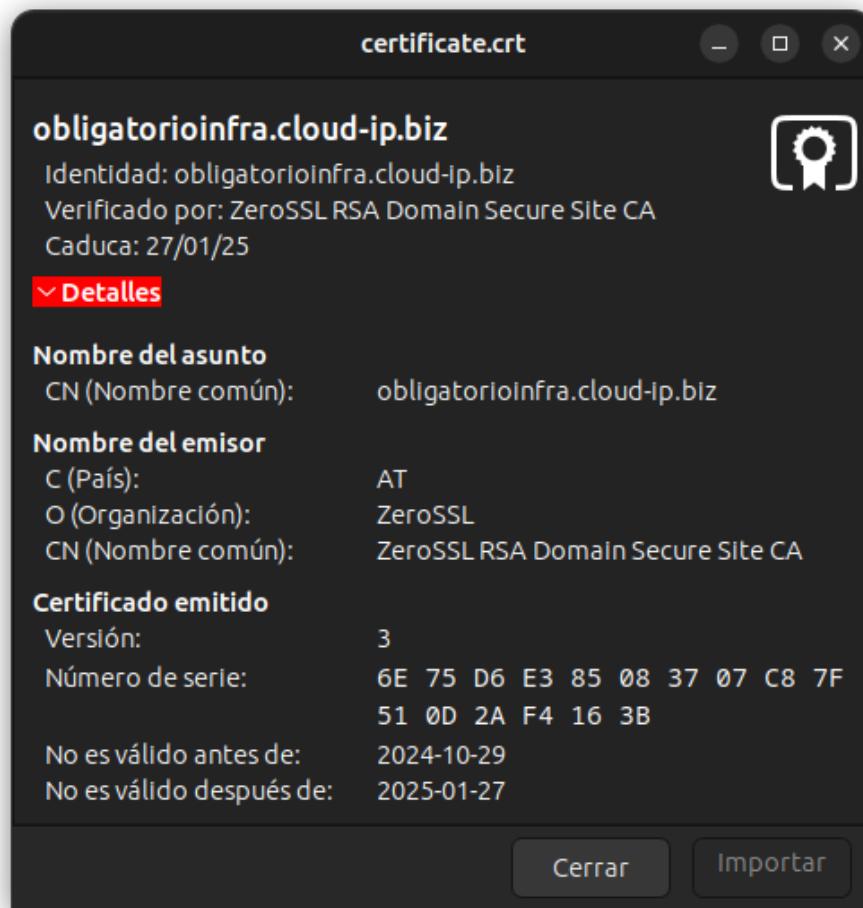
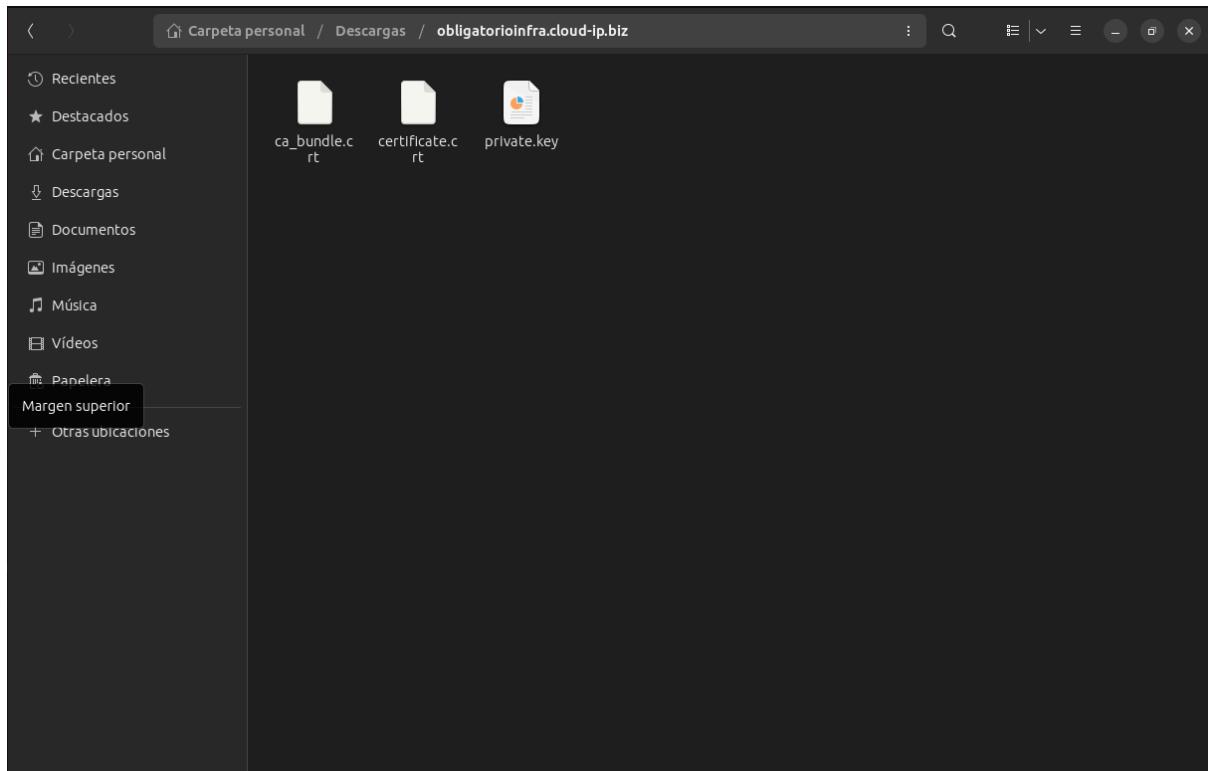
- De las posibles opciones, seleccionar DNS
- Utilizando los datos anteriores, volver a ClouDNS y crear un nuevo registro, esta vez de tipo “CNAME”, y los campos “Host” y “Points to” tomarán los valores de los apartados “Name” (únicamente la parte que está antes del primer punto) y “Point To”.



The screenshot shows a web browser window for the ZeroSSL website at <https://app.zerossl.com/certificate/install/87a9189d9174df19722c70368f630ec6>. The main title is "Install Certificate". On the left, there's a sidebar with "Get Premium SSL", "Dashboard", "Certificates", and "Developer". The main content area says "Our system is currently issuing your certificate. This page will refresh automatically every few seconds." It shows the domain **obligatorioinfra.cloud-ip.biz** and provides instructions: "We've prepared installation instructions for all major server types. To download and install your certificate, please follow the steps below:". Below this are three steps: "Download Certificate", "Install Certificate", and "Installation Complete". A "Finish Later" button is in the top right.

This screenshot shows the same ZeroSSL page after the certificate has been issued. The main title is "Install Certificate". The sidebar and main content area are identical to the previous screenshot, but the message in the main content area has changed to a green success message: "Your certificate has been issued and is ready for installation. To continue, please follow the steps below." The "Download Certificate" step now includes a note: "Your certificate is compatible with any type of web server. Download your certificate right away or make a selection below to get instructions and tutorials specific to your web server." It also shows a dropdown for "Server Type" set to "Default Format" with a "Download Certificate (.zip)" button next to it. A "Next Step →" button is located to the right of the download button. The "Finish Later" button remains in the top right.

- Descargar el certificado:
 - Una vez verificado, descarga el certificado y la clave privada que se generan, estos deberían llamarse cert.crt y private.key respectivamente.



Subir el certificado SSL obtenido a AWS ACM.

- Ingresar a AWS ACM
 - Seleccionar “Importar certificado”
 - Completar los campos
 - Certificate: Aquí debes copiar y pegar el contenido de tu archivo certificate.crt (incluyendo las líneas -----BEGIN CERTIFICATE----- y -----END CERTIFICATE-----).
 - Certificate private key: Copia y pega el contenido del archivo private.key (incluyendo las líneas -----BEGIN RSA PRIVATE KEY----- y -----END RSA PRIVATE KEY-----).
 - Certificate chain: Aquí debes copiar y pegar el contenido del archivo ca_bundle.crt (incluyendo las líneas).

Detalles del certificado

Cuerpo del certificado

```
k42KS0qfnyvNyDz1hOKK/kqOsxZfJWlsdHmJesjD0EmrYt+2TELQc/femhdD
atIMBnC7z3q4f076gpc4lC4f7099zQ6XkOhHd4+uLZ6xqyousxInLLVde/0F
uD05MSGbz2PoxGUk+s+YF5yS+qN/jNtwOCOoque@J0hLeh2wHSj0JMsGaxHle4
r9eIJUctagMBAAGjGKMIChdAfBnNHSMEDGAwBgTlXhoooot2an9utcf7c
tYaGpiJdRgNvH04EFGqfU/dgd7psNaya04SCBfz21+Wz5k2swGyDVR0PAQH/BAQD
AgVghAwGA1UDewEB/
wQCMIAwAHQYDROBBPwvAYKwvBBQUAwEVCCCAGAQUBwMC
MEKGATUJIACRMCAwEwNAYLkwbBAgymQECAk4wJTAjBggBgFBQCBAR8MHowSwY
I
KwvBBQUHMAKGp2h0dHA6Ly62XJv3NLnLyNC5zZWNoaWdv.mnb59a2XJvUTN
M
UINBRG9tYWIuU2VjdXIU2l0ZUNBLrnNydaBgrgBgFBQcwaYVfyaHR0cDovL3pl
cm9zc2wub2hzzCs2ZWNoaWdv.LmNvTCAGMCisGAQzB1nKBAlEqfEgEA7wB1
AMBRwv7UJ.tuouadn2h0kun+kacwtkJtfvAHSw5fcdhUAARRkt7+hoxEAACQnAEYw
```

Clave privada del certificado

```
-----BEGIN RSA PRIVATE KEY-----
MIICowIBAKKCAQEcAqCwugMLNgYUG7PqnTQS5zkVKqjVeTeLPBhGnsIyyDwHD/7r
XU9hGM01c1FowL4EI1PUHeJFR7ExbLrpzds9y2x0dBi+2+gx97b/902wdU
25d/jpID7QonX1w76y3tRHryEg50Mykgd6s2zABwvdYcv3P5KjJGXZ1T1CLLA
6zCYxLjghJq2LemrxCHFx0x02r7fQAUa2240Hz+oKvguQhvezfcOI
lB+4x3PoGelxd0Lb5ly1XXv9Bg0K+TOrm9j0MRCvCrJGBUsplKjzf7cjj
jqmnua49ByGelxd0s5fTBmlx5Xul/xyhRXLQDQAQABai0AGFCQOqlgfzMSxJC
```

- Presionar “Importar certificado”

AWS Certificate Manager (ACM)

Estado del certificado

Identificador	9eb8a3d1-cdcb-4d64-81a0-01c6613c562f	Estado	Emitido
ARN	arn:aws:acm:us-east-1:654654383249:certificate/9eb8a3d1-cdcb-4d64-81a0-01c6613c562f		
Tipo	Importado		

Dominios (1)

Domínio	obligatorioinfra.cloud-ip.biz
---------	-------------------------------

Detalles

En uso	No	Número de serie	16:0ca:91:36:bc:15:0:ffda:7-2:3:21:bc:4b:03:dd	Solicitado a las	octubre 30, 2024, 18:35:10 (UTC-03:00)	Requisitos que se deben cumplir para la renovación	No se cumplen los requisitos
--------	----	-----------------	--	------------------	--	--	------------------------------

Registrar el DNS creado previamente en AWS Route 53

- Ingresar a “Route 53”
 - Clickear “Zonas Hospedadas”

The screenshot shows the AWS Route 53 console. The left sidebar has a tree view with 'Zonas hospedadas' selected. The main area displays a table titled 'Zonas alojadas (0)' with a single row: 'No hay zonas alojadas'. At the bottom is a prominent orange button labeled 'Crear una zona alojada'.

- Clickear “Crear una zona alojada”
 - En el nombre del dominio, poner el obtenido previamente a través de ClouDNS
 - Dejar el resto predeterminado
 - Presionar “Crear una zona alojada”

The screenshot shows the 'Crear una zona alojada' wizard. Step 1: 'Configuración de zona alojada'. Step 2: 'Nombre de dominio' (Domain Name) with the value 'obligatorioinfra.cloud-ip.biz'. Step 3: 'Descripción - opcional' (Optional Description) with the value 'La zona alojada se utiliza para...'. Step 4: 'Tipo' (Type) with 'Zona alojada pública' (Public Hosted Zone) selected. The 'Zona alojada privada' (Private Hosted Zone) option is also shown.

- Ahora, desde la zona alojada, presionar “Crear registro”.
 - Dejar el subdominio vacío.
 - Tipo de registro: “A”
 - Valor: la ip del frontend
 - nslookup <URL del service del frontend>
 - Recordatorio: para obtener el URL del service del frontend se usa el comando: kubectl get service
 - El resto dejarlo predeterminado.

Crear registro Información

Registro de creación rápida

Registro 1

Nombre del registro Información subdomain obligatoriointra2024.cloud-ip.biz Tipo de registro Información A: dirige el tráfico a una dirección IPv4 y a algunos recursos de AWS

Mantenga el espacio en blanco para crear un registro para el dominio raíz.

Alias

Valor Información 34.192.227.169

Introduzca varios valores en líneas separadas.

TTL (segundos) Información 300 1 m 1 h 1 d Política de direccionamiento Información Direccionamiento sencillo

Valores recomendados: de 60 a 172 800 (dos días)

Agregar otro registro

Actualizar los NS de ClouDNS por los de Route 53

- Identificar los registros de Route 53 de tipo NS.

Route 53

El registro de obligatoriointra2024.cloud-ip.biz se ha creado correctamente. Route 53 propaga los cambios a todos los servidores DNS autorizados de Route 53 en 60 segundos. Utilice el botón "Ver estado" para comprobar el estado de propagación.

Detalle de la zona alojada

Registros (3) Información

Tipo	Política...	Difer...	Alias	Valor/Dirigir el tráfico a	TTL (s...)	ID de c...
A	Simple	-	No	34.192.227.169	300	-
NS	Simple	-	No	ns-683.awsdns-21.net. ns-428.awsdns-53.com. ns-1603.awsdns-08.co.uk. ns-1047.awsdns-02.org.	172800	-
SOA	Simple	-	No	ns-683.awsdns-21.net.awsd...	900	-

- Deberemos editar los registros correspondientes en ClouDNS para que coincidan con los mismos.
- Básicamente, presionamos el botón editar en cada uno de los registros, y reemplazamos el valor por uno de los registros de AWS.

Host	Type	Points to	TTL	
obligatorioinfraestructura.cloud-ip.biz	A	34.192.227.169	1h	
obligatorioinfraestructura.cloud-ip.biz	NS	[REDACTED]	1h	
obligatorioinfraestructura.cloud-ip.biz	NS	[REDACTED]	1h	
obligatorioinfraestructura.cloud-ip.biz	NS	[REDACTED]	1h	
obligatorioinfraestructura.cloud-ip.biz	NS	[REDACTED]	1h	

Registrar el certificado en los listeners HTTPS

- Ingresar a EC2, y a Balanceadores de Carga.
- Identificar el LoadBalancer referente al FrontEnd:
 - Seleccionar un LoadBalancer.
 - Ingresar a la sección “Agentes de Escucha”.
 - El LoadBalancer del FrontEnd debe tener el puerto 3000 seteado por defecto.
- Seleccionar la opción “Administrar agentes de escucha”.
- Aquí, se debe realizar lo siguiente:
 - Agregar un nuevo registro, con los siguientes datos:
 - Protocolo de agente de escucha: TCP
 - Puerto: 80
 - Protocolo de instancia 443
 - El resto no aplicable.
 - Referente al registro que estaba inicialmente, modificar el mismo:
 - Protocolo de agente de escucha: HTTPS
 - Puerto: 443
 - Protocolo de instancia: HTTP
 - Certificado SSL/TLS predeterminado: Seleccionar el certificado ingresado recientemente.
 - El resto sin modificar.

https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CLBManageListe 90% ⚡

Catálogo de AMI

Elastic Block Store

Volumenes

Instantáneas

Administrador del ciclo de vida

Red y seguridad

Security Groups

Direcciones IP elásticas

Grupos de ubicación

Pares de claves

Interfaces de red

Equilibrio de carga

Balanceadores de carga

Grupos de destino

Trust Stores Nuevo

Auto Scaling

Grupos de Auto Scaling

Configuración

Buscar [Alt+S]

Norte de Virginia voclabs/user2703209:salvador:vanoli @ 8517-2550-5938

EC2 > Balanceadores de carga > a819bfa7fefbc4f8793e43844415ea71 > Administrar agentes de escucha

Administrador de agentes de escucha

Detalles del equilibrador de carga: a819bfa7fefbc4f8793e43844415ea71

Agentes de escucha (2)

Los agentes de escucha configurados en el balanceador de carga clásico (CLB) definen cómo se enrutan las solicitudes de los clientes y el tráfico de red dentro de la aplicación. Aquí puede agregar nuevos agentes de escucha, modificar los agentes existentes o eliminar los agentes a medida que cambien sus necesidades.

Protocolo de agente de escucha	Puerto	Protocolo de instancia	Puerto de instancia	Política de seguridad	Certificado SSL/TLS predeterminado	Persistencia de las cookies
HTTPS	443	HTTP	641	ELBSecurityPolicy-2016-08	ACM: obligatorioinfraestructura.cloud-ip.biz	Inhabilitado
				Editar	Editar	Eliminar
TCP	80	TCP	443	No aplicable	No aplicable	No aplicable
						Eliminar

Agregar agente de escucha

Puede agregar hasta 98 más.

- Ahora hay que modificar y volver a aplicar el service para que maneje los puertos correctamente.
 - En este caso se están redirigiendo las peticiones web a un puerto seguro (443 - https).

```
apiVersion: v1
kind: Service
metadata:
  name: laravel-service
spec:
  type: LoadBalancer
  ports:
    - port: 80
      targetPort: 8000
      name: http
    - port: 443
      targetPort: 8000
      name: https
  selector:
    app: laravel
```

```
apiVersion: v1
kind: Service
```

```

metadata:
  name: nuxt-service
spec:
  type: LoadBalancer
  ports:
    - port: 80
      targetPort: 3000
      name: http
    - port: 443
      targetPort: 3000
      name: https
  selector:
    app: nuxt
---
apiVersion: v1
kind: Service
metadata:
  name: postgres-service
spec:
  ports:
    - port: 5432
      targetPort: 5432
  selector:
    app: postgres

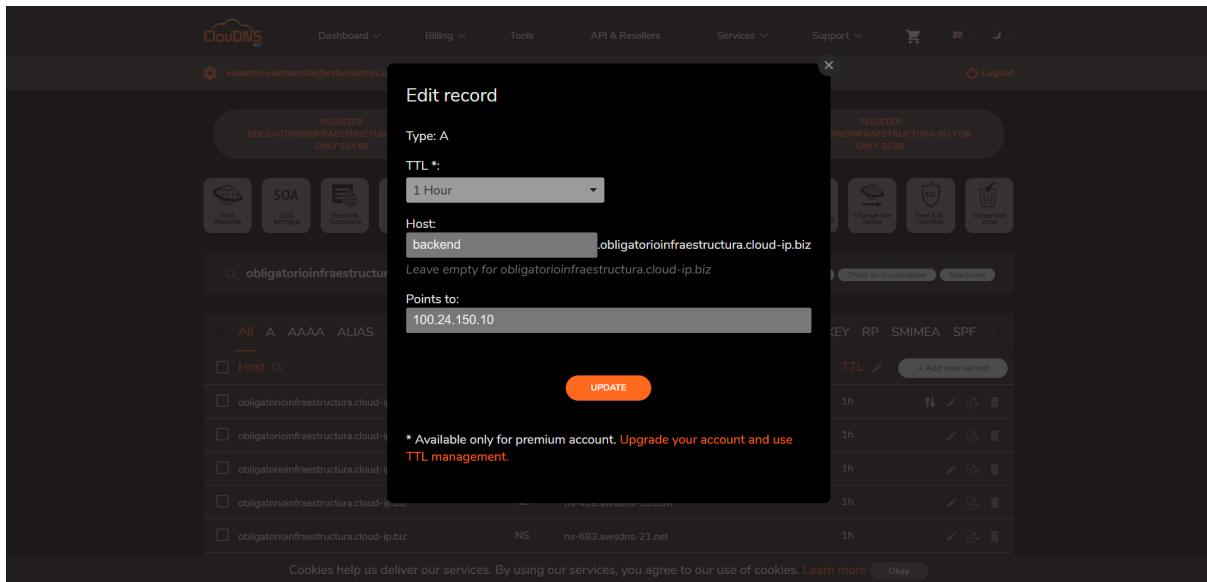
```

- Aplicar el cambio: `kubectl apply -f service.yaml`

Configurar backend para que sea seguro

- Para que la aplicación siga funcionando como corresponde, es necesario que el BackEnd también maneje protocolo HTTPS (ya que una página https no se puede comunicar con una http de manera segura).
- Para esto, lo primero sería modificar los agentes de escucha del LoadBalancer correspondiente al BackEnd.
 - Ingresar a EC2, y a Balanceadores de Carga.
 - Identificar el LoadBalancer referente al Backend:
 - Seleccionar un LoadBalancer.
 - Ingresar a la sección “Agentes de Escucha”.
 - El LoadBalancer del BackEnd debe tener el puerto 8000 seteado por defecto.

- Seleccionar la opción “Administrar agentes de escucha”.
- Aquí, deberían haber dos registros, teniendo que modificar ambos para que sean idénticos a los del FrontEnd, con la excepción de que el registro con puerto 443 (HTTPS) redirige al mismo puerto que estaba inicialmente (HTTPS).
- El registro con el puerto 80, ahora estaría direccionado al puerto 443, ambos manejando TCP.
- Además, es importante asociar un nuevo certificado SSL con el registro HTTPS 443.
 - Para esto es necesario crear un nuevo registro en ClouDNS, de tipo “A”, referente al BackEnd.
 - La IP del BackEnd puede ser obtenida en consola mediante:
 - nslookup <URL del service del BackEnd>
 - Recordatorio: para obtener el URL del service del BackEnd se usa el comando: kubectl get service.
 - Importante: A la hora de crear el registro, es importante diferenciarlo del que creamos inicialmente para el FrontEnd. Para esto, podemos agregar un host que lo diferencie del anterior, como en nuestro caso “backend”.



- A continuación, con el dominio, debemos crear un nuevo certificado SSL, ahora relacionado al BackEnd (siguiendo [los mismos pasos](#) que para el certificado anterior, pero especificando el dominio del BackEnd).
- Ahora, ya con el certificado descargado, es necesario subirlo a AWS ACM para poder asociarlo al LoadBalancer (los pasos [ya fueron descritos](#) con anterioridad).

- Finalmente, en el registro de tipo HTTP 443 del LoadBalancer referido al BackEnd, en la sección “Certificado SSL/TLS predeterminado”, se debe seleccionar el certificado ingresado recientemente (como ya [se hizo previamente](#)).

Protocolo de escucha	Puerto	Protocolo de instancia	Puerto de instancia	Política de seguridad	Certificado SSL/TLS predeterminado	Persistencia de las cookies
HTTPS	443	HTTP	32350	ELBSecurityPolicy-2016-08 Editar	ACM: backend.obiatorioinfraestructura.cloud-ip.biz	Inhabilitado Editar Eliminar
TCP	80	TCP	443	No aplicable	No aplicable	No aplicable Eliminar

[Agregar agente de escucha](#)
Puede agregar hasta 98 más.

- Ya con todos estos cambios realizados, el último paso es modificar una última vez el deployment.yaml cambiando la dirección del backend por la del dominio, ahora seguro.

```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: laravel-deployment
spec:
  replicas: 1
  selector:
    matchLabels:
      app: laravel
  template:
    metadata:
      labels:
        app: laravel
    spec:
      containers:
        - name: laravel-app
          image: <account-id>.dkr.ecr.us-east-1.amazonaws.com/backend:latest
          ports:
            - containerPort: 8000
          env:
            - name: APP_ENV
              value: "local"
            - name: APP_DEBUG
              value: "true"
            - name: DB_CONNECTION

```

```

    value: "sqlite"
  - name: DB_HOST
    value: "postgres-service"
  - name: DB_PORT
    value: "5432"
  - name: DB_DATABASE
    value: "eks_notes"
  - name: DB_USERNAME
    value: "root"
  - name: DB_PASSWORD
    value: "root"
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nuxt-deployment
spec:
  replicas: 1
  selector:
    matchLabels:
      app: nuxt
  template:
    metadata:
      labels:
        app: nuxt
    spec:
      containers:
        - name: nuxt-app
          image: <account-id>.dkr.ecr.us-east-1.amazonaws.com/frontend:latest
          ports:
            - containerPort: 3000
          env:
            - name: NUXT_PUBLIC_API_BASE
              value: "https://<tu-dominio-seguro>/api"
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: postgres-deployment
spec:

```

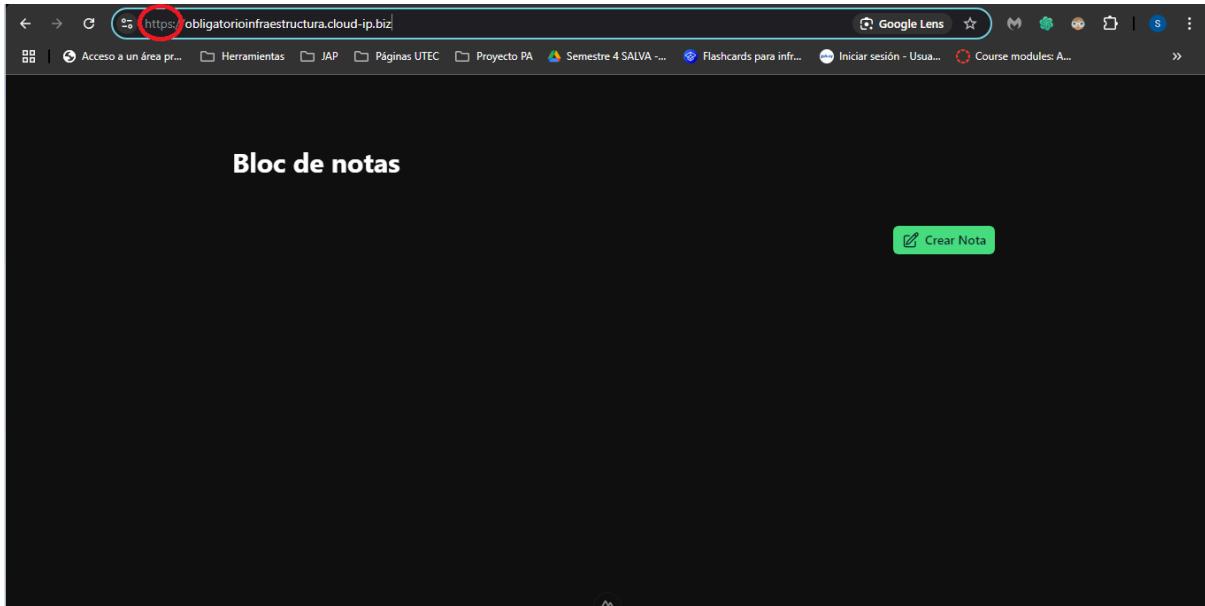
```

replicas: 1
selector:
  matchLabels:
    app: postgres
template:
  metadata:
    labels:
      app: postgres
  spec:
    containers:
      - name: postgres-db
        image: postgres:13
        ports:
          - containerPort: 5432
        env:
          - name: POSTGRES_DB
            value: "eks_notes"
          - name: POSTGRES_USER
            value: "root"
          - name: POSTGRES_PASSWORD
            value: "root"
        volumeMounts:
          - name: postgres-storage
            mountPath: /var/lib/postgresql/data
    volumes:
      - name: postgres-storage
        persistentVolumeClaim:
          claimName: postgres-pvc
---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: postgres-pvc
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi # Debe coincidir con el tamaño del PV
  storageClassName: manual # Debe coincidir con el PV

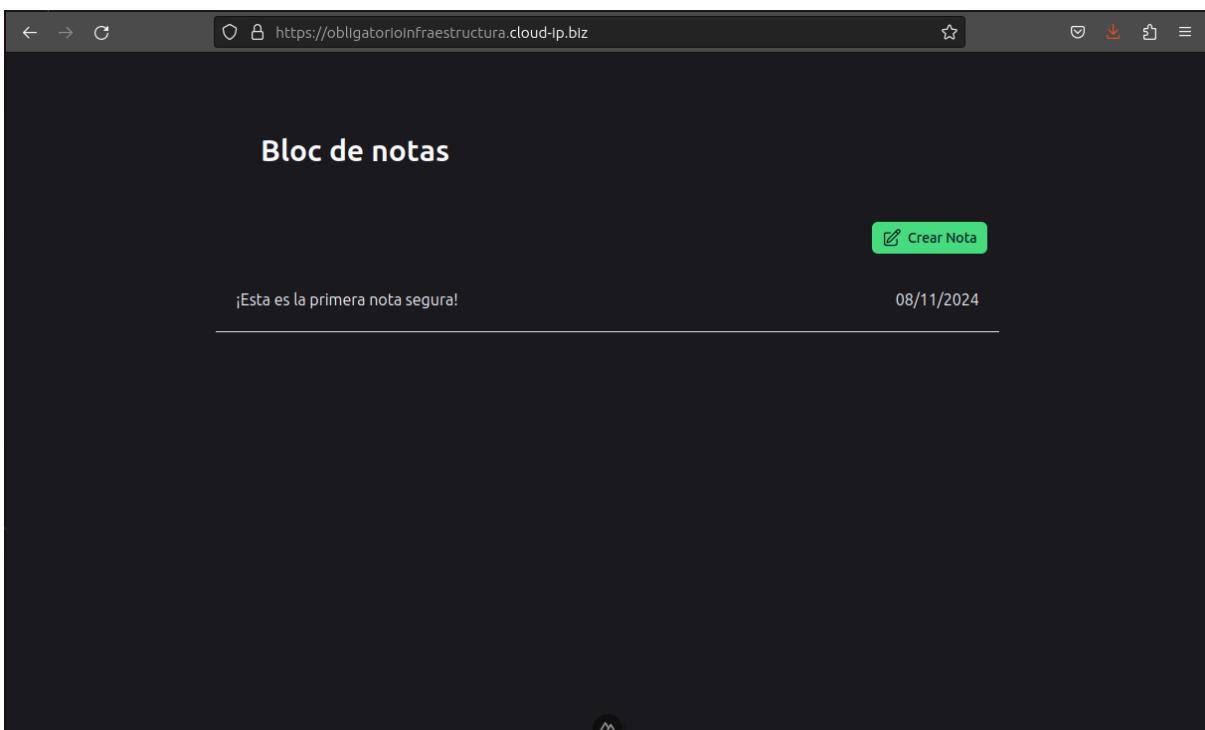
```

Comprobación final

- Si todo quedó correctamente configurado, ahora se debería poder acceder a la página a través de una conexión segura (https).



- Intentar agregar una nota, debería de dejar y guardarse en la bd.



```

{
  "0": {
    "id": 1,
    "data": "¡Esta es la primera nota segura!",
    "created_at": "2024-11-09T01:07:11.000000Z",
    "updated_at": "2024-11-09T01:07:11.000000Z"
  }
}

```

- Ahora, mismamente desde la página web, se pueden observar los metadatos de los certificados. En nuestro caso, el del backend es:

Visor de certificados: backend.obligatorioinfraestructura.cloud-ip.biz

General **Detalles**

Enviado a

Nombre común (CN)	backend.obligatorioinfraestructura.cloud-ip.biz
Organización (O)	<No incluido en el certificado>
Unidad organizativa (OU)	<No incluido en el certificado>

Emitido por

Nombre común (CN)	ZeroSSL RSA Domain Secure Site CA
Organización (O)	ZeroSSL
Unidad organizativa (OU)	<No incluido en el certificado>

Período de validez

Emitido el	viernes, 8 de noviembre de 2024, 21:00:00
Vencimiento el	viernes, 7 de febrero de 2025, 20:59:59

Huellas digitales SHA-256

Certificado	80168d0a09d8e60308372c6155ca7313295f59a6aad8119916c65b2 584541e3b
Clave pública	a4ad25a6394106d485f96db018349538ba7c2ad3242f0cd01646749 4543a09b7

8. Seguridad

Buenas prácticas

- Escanear las imágenes docker antes de subirlas al clúster con Triv para encontrar vulnerabilidades conocidas.
- Manejar credenciales/secretos/claves a través de AWS Secrets Manager.
- Habilitar CloudTrail y IAM para controlar el acceso a los recursos.
- Utilizar HTTPS y conseguir un certificado SSL para permitir el cifrado de información sensible en las peticiones web.
- Realizar análisis de seguridad con herramientas como OWASP ZAP.

Aplicación de buenas prácticas

Escanear las imágenes docker con trivy

- En la consola, ejecutar los siguientes pasos:
 - sudo snap install trivy
 - trivy image <nombre-imagen-front>
 - trivy image <nombre-imagen-back>

Utilizar AWS Secrets Manager

Lo utilizamos al integrar ROUTE 53 en el proyecto, fue necesario para crear el secreto del clúster y utilizarlo en el Ingress.yaml.

CloudTrail y IAM

No podemos utilizar estos servicios libremente debido a las limitaciones de la cuenta, pero consiste en la creación de roles y eventos de monitorización para manejar el acceso a recursos, lo que aumenta significativamente la seguridad.

Uso de HTTPS y certificados SSL

Esto queda incluído en el desarrollo de la consigna, al conseguir el certificado y aplicarlo a través del DNS y Route 53.

Fuzz testing tools

AFL (American Fuzzy Lop)

AFL es una herramienta de Fuzzing muy conocida que funciona guiada por cobertura, generando entradas aleatorias que maximizan la cantidad de rutas de ejecución probadas en un programa. Al ser puesta en marcha, AFL puede realizar mutaciones inteligentes en las entradas, lo que permite detectar fallos de seguridad como desbordamientos de buffer. Es muy eficaz y es capaz de realizar pruebas rápidas en aplicaciones complejas.

ZZUF

ZZUF es una herramienta simple pero efectiva para realizar Fuzzing en programas que manejan archivos, como imágenes, videos o texto. Modifica ligeramente los datos binarios de los archivos de entrada y los introduce al programa objetivo, observando cómo maneja los errores. Es excelente para descubrir vulnerabilidades relacionadas con la manipulación de archivos, como crashes inesperados.

LibFuzzer

LibFuzzer es parte del ecosistema LLVM y se enfoca en pruebas dirigidas a funciones individuales dentro de proyectos C/C++. Inyecta entradas aleatorias directamente en funciones objetivo y monitorea el comportamiento del programa. Al ser muy eficaz para encontrar fallos de memoria, se utiliza en bibliotecas críticas para detectar errores como referencias de punteros nulos o sobreescritura de memoria.

Burp Suite (Intruder Module)

Burp Suite es una herramienta popular para pruebas de seguridad en aplicaciones web, y su módulo Intruder permite realizar Fuzzing enviando grandes cantidades de solicitudes HTTP con entradas malformadas. Este enfoque ayuda a descubrir vulnerabilidades como inyecciones SQL, XSS y fallos de validación de datos en aplicaciones web. Es muy utilizado en auditorías de seguridad web para automatizar pruebas.

Radamsa

Radamsa es una herramienta de Fuzzing de propósito general que produce entradas malformadas al modificar ligeramente los datos de entrada esperados por un programa. Es flexible y puede ser aplicada a muchos tipos de programas y protocolos. Es muy útil para realizar pruebas de robustez y encontrar vulnerabilidades en el manejo de datos, como fallos de validación de entradas.

Peach Fuzzer

Peach Fuzzer es una herramienta avanzada que se utiliza para Fuzzing de protocolos, archivos y sistemas distribuidos. Genera entradas no válidas para probar cómo las aplicaciones manejan datos inesperados, siendo ideal para probar software que interactúa con redes o archivos complejos. Su versatilidad le permite adaptarse a diferentes entornos, haciéndolo útil en pruebas de seguridad de aplicaciones críticas.

SAST (Static Application Security Testing)

SonarQube

SonarQube es una plataforma de código abierto diseñada para inspeccionar continuamente la calidad del código fuente. Permite identificar vulnerabilidades, errores y malas prácticas de programación en una gran variedad de lenguajes. SonarQube se integra fácilmente con sistemas CI/CD, facilitando la detección temprana de problemas en el flujo de desarrollo. Proporciona un análisis profundo mediante reglas de calidad personalizables y tiene un ecosistema de plugins que lo hacen compatible con herramientas de desarrollo populares. Es ideal para equipos que buscan mejorar la calidad y seguridad de sus aplicaciones de manera continua.

Checkmarx CxSAST

Checkmarx CxSAST es una solución de seguridad orientada al análisis de código fuente que permite a los equipos de desarrollo y seguridad identificar, priorizar y remediar vulnerabilidades de manera eficaz. Ofrece una amplia cobertura de lenguajes y marcos de trabajo, y sus algoritmos avanzados permiten clasificar vulnerabilidades para optimizar la remediación. Esta herramienta se integra fácilmente en entornos de CI/CD, lo que ayuda a detectar problemas de seguridad antes de que el código llegue a producción. Es utilizada comúnmente en grandes organizaciones que requieren altos estándares de seguridad.

Veracode Static Analysis

Veracode proporciona un análisis de seguridad robusto basado en la nube que permite a las organizaciones identificar vulnerabilidades en su código de forma rápida. Su enfoque en la detección de errores comunes de seguridad y recomendaciones de mitigación específicas facilita la implementación de buenas prácticas de desarrollo seguro. Con su interfaz en la nube, Veracode se adapta bien a proyectos que operan en infraestructuras de desarrollo moderno y distribuidas. Se integra de forma sencilla en

el flujo de desarrollo, ofreciendo informes detallados y ayudando a los equipos a priorizar las correcciones.

Nmap

Es una herramienta de código abierto utilizada para la exploración y auditoría de seguridad de redes. Su principal función es identificar dispositivos en una red, los servicios que están corriendo en esos dispositivos, y las versiones de los sistemas operativos en uso. Nmap utiliza paquetes IP para determinar qué hosts están disponibles en la red, qué servicios están expuestos y qué puertos están abiertos. Esta herramienta es ampliamente utilizada por administradores de red y expertos en seguridad para realizar auditorías de seguridad y descubrir vulnerabilidades.

Nessus Pro

Es una herramienta de evaluación de vulnerabilidades que permite a las organizaciones identificar y remediar vulnerabilidades en sus sistemas y redes. Desarrollada por Tenable, Nessus ofrece un enfoque integral para la gestión de vulnerabilidades, permitiendo escaneos automatizados y en profundidad que evalúan una amplia variedad de sistemas, incluyendo servidores, dispositivos de red, bases de datos y aplicaciones web.

Nessus Pro incluye características como la evaluación de configuraciones, el escaneo de cumplimiento normativo, y una amplia base de datos de vulnerabilidades que se actualiza continuamente. Los informes generados por Nessus proporcionan una visión clara de las vulnerabilidades encontradas, su severidad y recomendaciones para su remediación. Debido a su facilidad de uso y su enfoque exhaustivo, Nessus Pro es ampliamente adoptado por profesionales de la seguridad y administradores de TI para fortalecer la postura de seguridad de sus organizaciones.

OWASP ZAP (Zed Attack Proxy)

OWASP ZAP, desarrollado por OWASP, es una herramienta que permite realizar Fuzzing en aplicaciones web de forma automatizada. Ofrece un entorno completo para detectar vulnerabilidades comunes como fallos en validación de formularios, parámetros de URL y más. Es ampliamente utilizado en el pentesting (prueba de seguridad que lanza un ciberataque simulado para encontrar vulnerabilidades en un sistema informático) de aplicaciones web debido a su facilidad de uso y capacidad de integración con otros flujos de trabajo.

Pasos para llevar a cabo los tests de seguridad

Para cumplir con la letra, las aplicaciones que vamos a utilizar para realizar las pruebas de seguridad serán:

SonarQube

- Crear un usuario para SonarQube
 - Ejecutar sudo useradd -m -d /opt/sonarqube -c "SonarQube user" sonarqube
- Descargar SonarQube
 - Ejecutar los siguientes comandos:
 - cd /opt
 - sudo wget <https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-9.9.1.69595.zip>
 - sudo unzip sonarqube-9.9.1.69595.zip
 - sudo mv sonarqube-9.9.1.69595 sonarqube
 - sudo chown -R sonarqube:sonarqube /opt/sonarqube
- Configurar SonarQube
 - Crear la base de datos
 - sudo -u postgres psql
 - CREATE DATABASE sonarqube;
 - CREATE USER sonarqube WITH ENCRYPTED PASSWORD 'password123';
 - GRANT ALL PRIVILEGES ON DATABASE sonarqube TO sonarqube;
 - \q
 - Abrir el archivo de configuración: sudo nano /opt/sonarqube/conf/sonar.properties
 - Editar las siguientes líneas con los valores correctos:
 - sonar.jdbc.username=sonarqube (nombre de usuario)
 - sonar.jdbc.password=YOUR_PASSWORD (contraseña)
 - sonar.jdbc.url=jdbc:postgresql://localhost:5432/sonarqube
- Configurar el archivo SonarQube como un Servicio Systemd
 - Ejecutar sudo nano /etc/systemd/system/sonarqube.service
 - Agregar la siguiente configuración al archivo:

```
[Unit]
Description=SonarQube service
After=syslog.target network.target
```

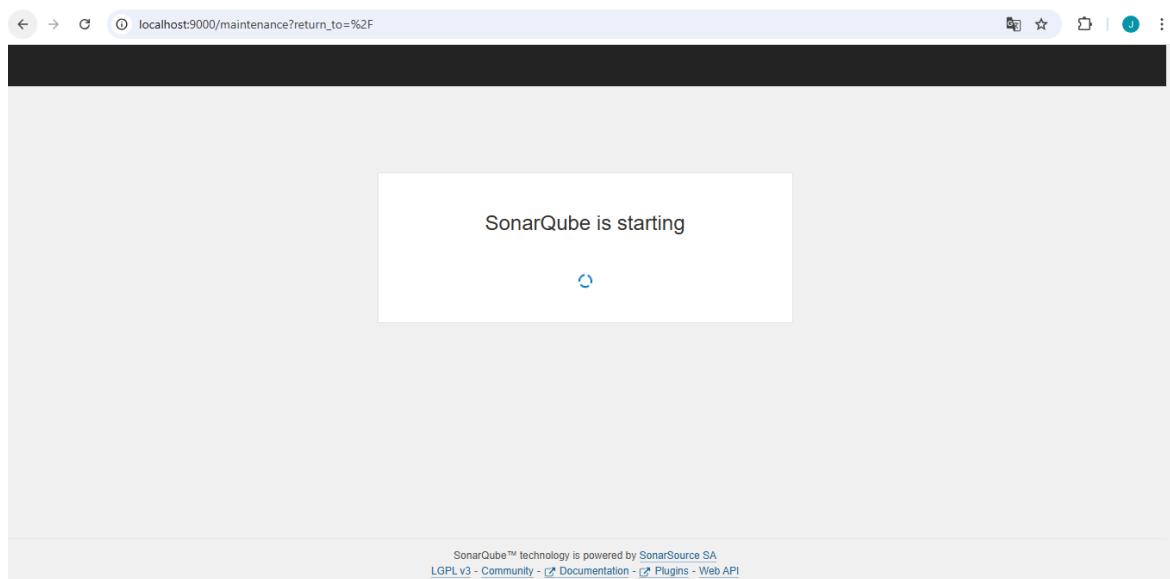
```
[Service]
Type=forking
```

```
ExecStart=/opt/sonarqube/bin/linux-x86-64/sonar.sh start
ExecStop=/opt/sonarqube/bin/linux-x86-64/sonar.sh stop
```

```
User=sonarqube
Group=sonarqube
Restart=on-failure
LimitNOFILE=65536
```

```
[Install]
WantedBy=multi-user.target
```

- Iniciar el servicio SonarQube
 - Ejecutar los siguientes comandos:
 - sudo systemctl daemon-reload
 - sudo systemctl start sonarqube
 - sudo systemctl enable sonarqube
- Acceder a SonarQube en el navegador
 - <http://localhost:9000>



- Cambiar la contraseña de administrador si es solicitado.

The screenshot shows a web browser window with the URL `localhost:9000/account/reset_password`. The main content is a form titled "Update your password". A note at the top says "This account should not use the default password." Below it, instructions say "Enter a new password" and "All fields marked with * are required". There are three input fields: "Old Password *", "New Password *", and "Confirm Password *". An "Update" button is at the bottom. The browser interface includes standard navigation buttons and a tab bar.

- Crear un nuevo proyecto en SonarQube:
- Ir a Projects > Create Project.

The screenshot shows a web browser window with the URL `localhost:9000/projects/create`. The header includes the SonarQube logo and navigation links for Projects, Issues, Rules, Quality Profiles, Quality Gates, and Administration. A search bar is also present. The main content asks "How do you want to create your project?". It provides options for creating projects from various platforms: Azure DevOps, Bitbucket Server, Bitbucket Cloud, GitHub, and GitLab, each with a "Set up global configuration" link. At the bottom, there's a section for manual project creation with a "Manually" link and a "Manually" button.

- Asignar un nombre y clave única al proyecto.

localhost:9000/projects/create?mode=manual

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration Search for projects... A

Create a project

All fields marked with * are required

Project display name *
adminInfraestructura ✓
Up to 255 characters. Some scanners might override the value you provide.

Project key *
adminInfraestructura ✓
The project key is a unique identifier for your project. It may contain up to 400 characters. Allowed characters are alphanumeric, '-' (dash), '_' (underscore), '.' (period) and ':' (colon), with at least one non-digit.

Main branch name *
main
The name of your project's default branch [Learn More](#)

Set Up

localhost:9000/dashboard?id=adminInfraestructura

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration Search for projects... A

adminInfraestructura main

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

How do you want to analyze your repository?

Do you want to integrate with your favorite CI? Choose one of the following tutorials.

With Jenkins **With GitHub Actions** **With Bitbucket Pipelines** **With GitLab CI** **With Azure Pipelines** **Other CI**

Are you just testing or have an advanced use-case? Analyze your project locally.

Locally

- Generar un token de análisis y copiarlo para usarlo en los pasos siguientes.

localhost:9000/dashboard?id=adminInfraestructura&selectedTutorial=local

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration Search for projects... A

adminInfraestructura main

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

Analyze your project

We initialized your project on SonarQube, now it's up to you to launch analyses!

1 Provide a token

Generate a project token

Token name adminInfraestructura123 **Expires in** 30 days **Generate**

Please note that this token will only allow you to analyze the current project. If you want to use the same token to analyze multiple projects, you need to generate a global token in your user account. See the [documentation](#) for more information.

The token is used to identify you when an analysis is performed. If it has been compromised, you can revoke it at any point in time in your [user account](#).

2 Run analysis on your project

Embedded database should be used for evaluation purposes only

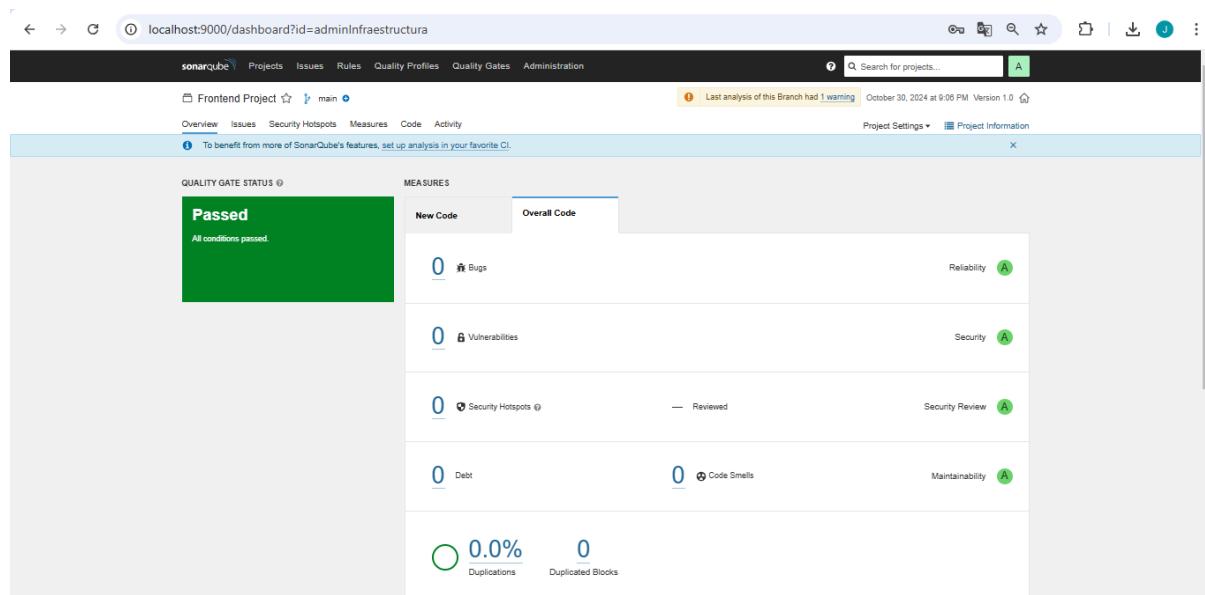
- Configurar SonarScanner
 - Descargar SonarScanner
 - cd /opt
 - sudo wget https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-4.8.0.2856-linux.zip
 - sudo unzip sonar-scanner-4.8.0.2856-linux.zip
 - sudo mv sonar-scanner-4.8.0.2856-linux sonar-scanner
 - sudo chown -R sonarqube:sonarqube /opt/sonar-scanner]
 - Configurar SonarScanner en el PATH
 - echo 'export PATH=\$PATH:/opt/sonar-scanner/bin' | sudo tee -a /etc/profile
 - source /etc/profile
 - Configurar el proyecto para el análisis
 - Crear el archivo sonar-project.properties
 - nano sonar-project.properties
 - Agregar la siguiente configuración al archivo, reemplazando los valores según el proyecto y el token generado:


```
sonar.projectKey=clave-unica-proyecto
sonar.projectName=NombreProyecto
sonar.projectVersion=1.0
sonar.sources=.
sonar.host.url=http://localhost:9000
sonar.login=YOUR_TOKEN
```
 - Ejecutar el análisis del proyecto
 - Ejecutar sonar-scanner

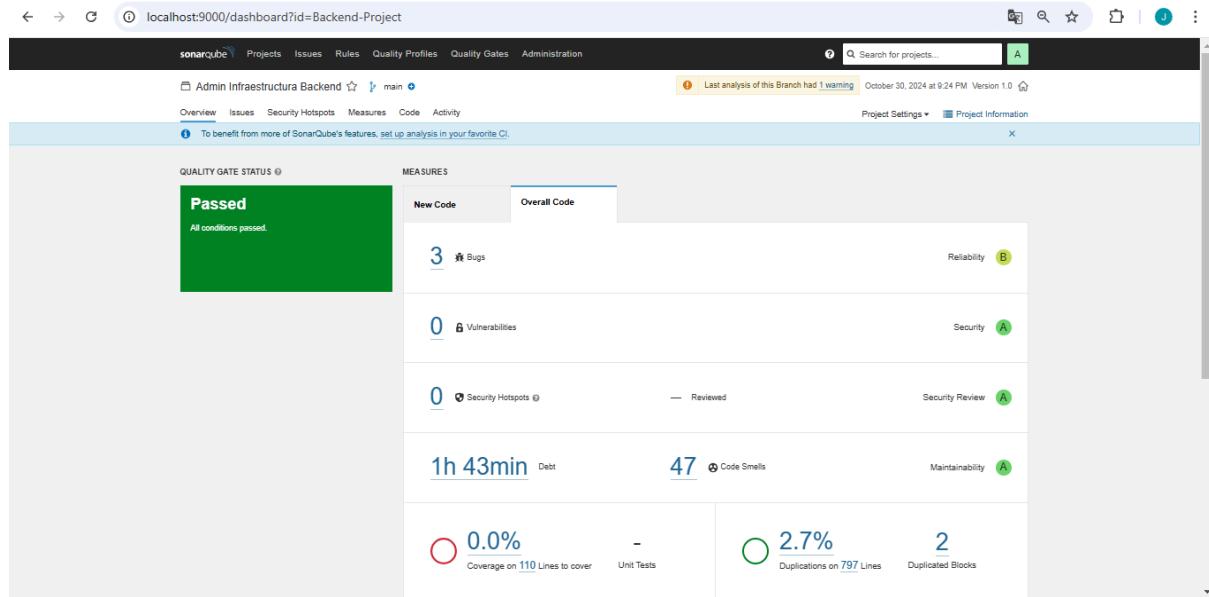
```
joaquin@LAPTOP-FDTEII6P:/opt/EKSApp/backend$ nano sonar-project.properties
joaquin@LAPTOP-FDTEII6P:/opt/EKSApp/backend$ sonar-scanner
INFO: Scanner configuration file: /opt/sonar-scanner/conf/sonar-scanner.properties
INFO: Project root configuration file: /opt/EKSApp/backend/sonar-project.properties
INFO: SonarScanner 4.8.0.2856
INFO: Java 11.0.17 Eclipse Adoptium (64-bit)
INFO: Linux 5.10.16.3-microsoft-standard-WSL2 amd64
INFO: User cache: /home/joaquin/.sonar/cache
INFO: Analyzing on SonarQube server 9.9.1.69595
INFO: Default locale: "en", source code encoding: "UTF-8" (analysis is platform dependent)
INFO: Load global settings
INFO: Load global settings (done) | time=281ms
INFO: Server id: 147B411E-AZLfiykSb4VhdDinYpyN
INFO: User cache: /home/joaquin/.sonar/cache
INFO: Load/download plugins
INFO: Load plugins index
INFO: Load plugins index (done) | time=194ms
INFO: Load/download plugins (done) | time=526ms
INFO: Process project properties
INFO: Process project properties (done) | time=30ms
INFO: Execute project builders
INFO: Execute project builders (done) | time=7ms
```

- Revisar los Resultados en SonarQube
 - Volver a la interfaz web de SonarQube.
 - Navegar al Proyecto.
 - Examinar los resultados de análisis, donde se verán:
 - Bugs: Problemas en el código.
 - Code Smells: Problemas de calidad del código.
 - Vulnerabilidades: Problemas de seguridad.

Resultados del análisis sobre el front-end



Resultados del análisis sobre el backend:



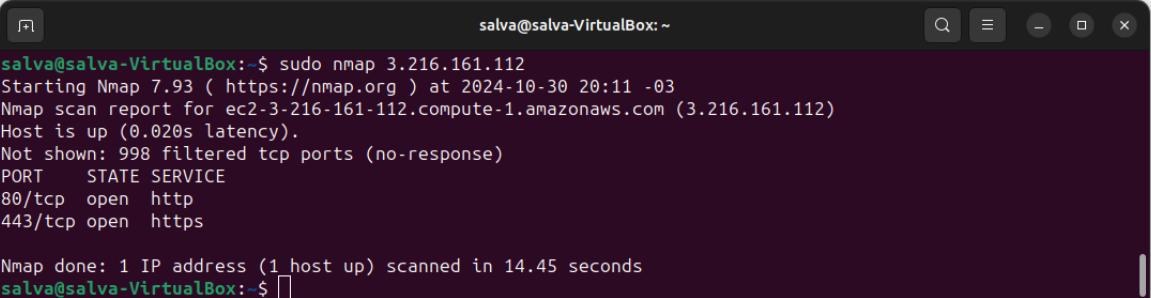
Nmap

- Instalar Nmap
 - sudo apt-get install nmap
- Obtener la IP del service del front.
 - nslookup <EXTERNAL-IP-DEL-FRONT>

```
valentin@Ubuntu:~/Descargas/obligatorioInfra.cloud-ip.biz$ kubectl get services -n ingress-nginx
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP
Ingress-nginx-controller   LoadBalancer  10.100.181.9  ab1c6032e9dcba14827ef10c58be3e3-1806991520.us-east-1.elb.amazonaws.com
  80:31529/TCP,443:31527/TCP  3h9m
ingress-nginx-controller-admin  ClusterIP  <none>
  443/TCP                  3h9m
valentin@Ubuntu:~/Descargas/obligatorioInfra.cloud-ip.biz$ nslookup ab1c6032e9dcba14827ef10c58be3e3-1806991520.us-east-1.elb.amazonaws.com
Server:  127.0.0.53
Address: 127.0.0.53#53

Non-authoritative answer:
Name: ab1c6032e9dcba14827ef10c58be3e3-1806991520.us-east-1.elb.amazonaws.com
Address: 3.216.161.112
valentin@Ubuntu:~/Descargas/obligatorioInfra.cloud-ip.biz$
```

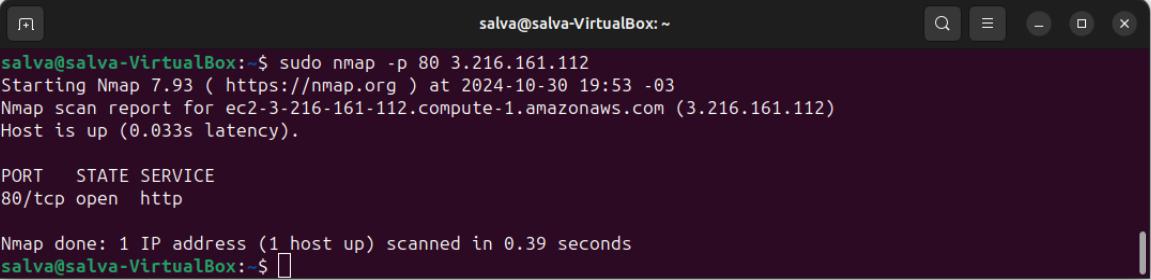
- Escaneo básico de puertos
 - nmap <IP>



```
salva@salva-VirtualBox:~$ sudo nmap 3.216.161.112
Starting Nmap 7.93 ( https://nmap.org ) at 2024-10-30 20:11 -03
Nmap scan report for ec2-3-216-161-112.compute-1.amazonaws.com (3.216.161.112)
Host is up (0.020s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
443/tcp   open  https

Nmap done: 1 IP address (1 host up) scanned in 14.45 seconds
salva@salva-VirtualBox:~$ 
```

- Escaneo de puertos específicos
 - `nmap -p 80,443 <IP>`

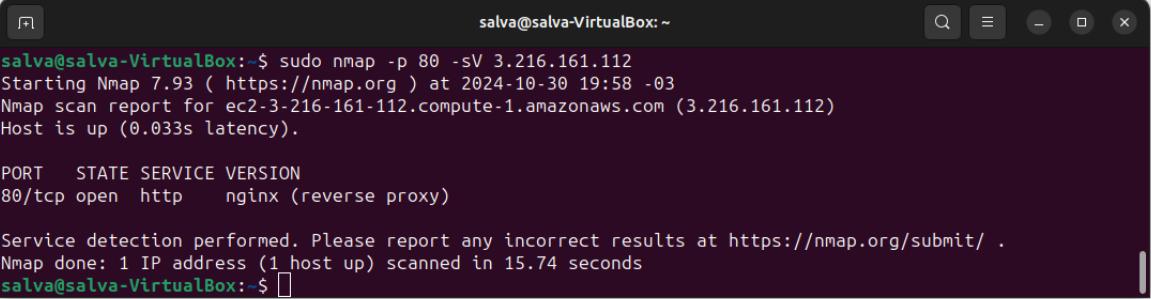


```
salva@salva-VirtualBox:~$ sudo nmap -p 80 3.216.161.112
Starting Nmap 7.93 ( https://nmap.org ) at 2024-10-30 19:53 -03
Nmap scan report for ec2-3-216-161-112.compute-1.amazonaws.com (3.216.161.112)
Host is up (0.033s latency).

PORT      STATE SERVICE
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 0.39 seconds
salva@salva-VirtualBox:~$ 
```

- Escaneo completo (Descubrimientos de Servicios)
 - `nmap -sV <IP>`

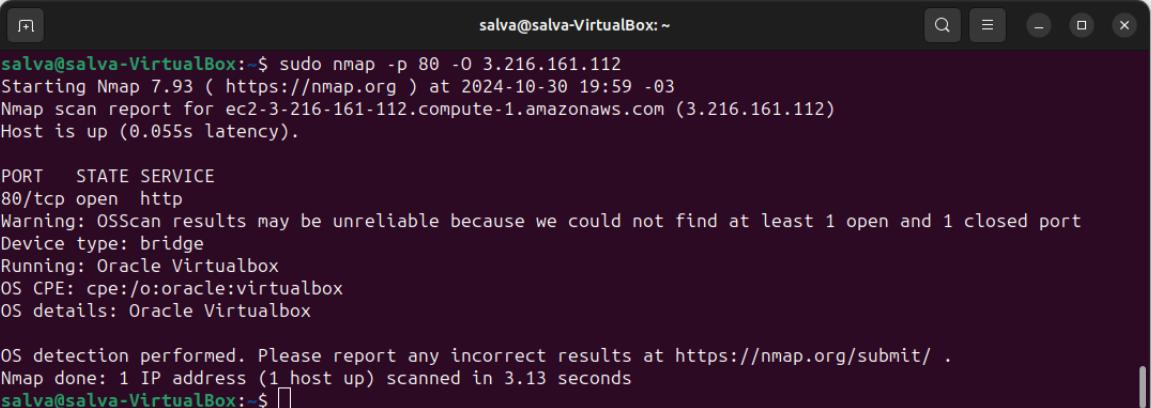


```
salva@salva-VirtualBox:~$ sudo nmap -p 80 -sV 3.216.161.112
Starting Nmap 7.93 ( https://nmap.org ) at 2024-10-30 19:58 -03
Nmap scan report for ec2-3-216-161-112.compute-1.amazonaws.com (3.216.161.112)
Host is up (0.033s latency).

PORT      STATE SERVICE VERSION
80/tcp    open  http      nginx (reverse proxy)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.74 seconds
salva@salva-VirtualBox:~$ 
```

- Escaneo de Sistema Operativo
 - `nmap -O <IP>`

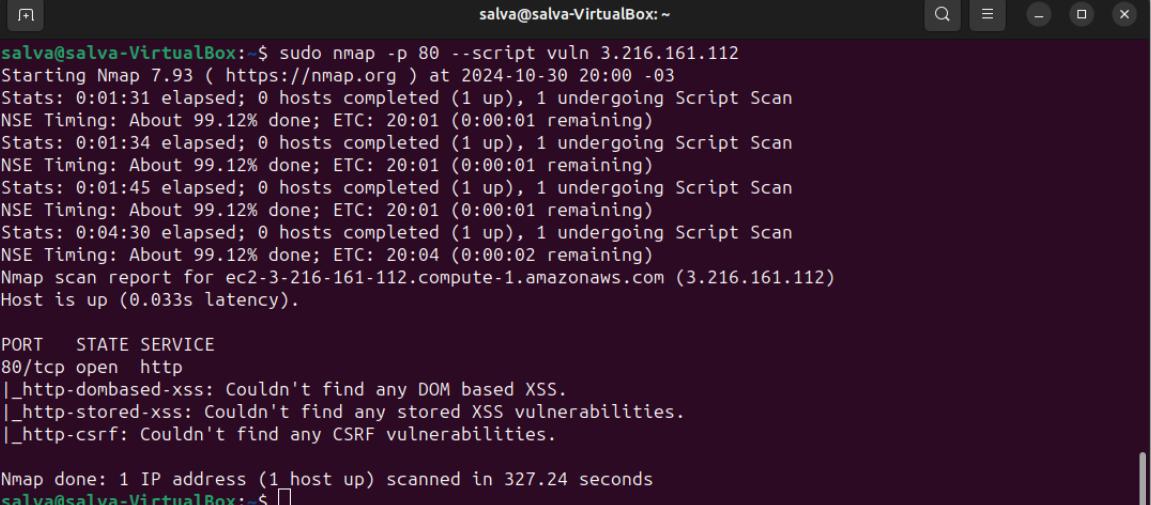


```
salva@salva-VirtualBox:~$ sudo nmap -p 80 -O 3.216.161.112
Starting Nmap 7.93 ( https://nmap.org ) at 2024-10-30 19:59 -03
Nmap scan report for ec2-3-216-161-112.compute-1.amazonaws.com (3.216.161.112)
Host is up (0.055s latency).

PORT      STATE SERVICE
80/tcp    open  http
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: bridge
Running: Oracle Virtualbox
OS CPE: cpe:/o:oracle:virtualbox
OS details: Oracle Virtualbox

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.13 seconds
salva@salva-VirtualBox:~$ 
```

- Escaneo de vulnerabilidades
 - nmap --script vuln <IP>



```
salva@salva-VirtualBox:~$ sudo nmap -p 80 --script vuln 3.216.161.112
Starting Nmap 7.93 ( https://nmap.org ) at 2024-10-30 20:00 -03
Stats: 0:01:31 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.12% done; ETC: 20:01 (0:00:01 remaining)
Stats: 0:01:34 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.12% done; ETC: 20:01 (0:00:01 remaining)
Stats: 0:01:45 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.12% done; ETC: 20:01 (0:00:01 remaining)
Stats: 0:04:30 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.12% done; ETC: 20:04 (0:00:02 remaining)
Nmap scan report for ec2-3-216-161-112.compute-1.amazonaws.com (3.216.161.112)
Host is up (0.033s latency).

PORT      STATE SERVICE
80/tcp    open  http
|_http-dombased-xss: Couldn't find any DOM based XSS.
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-csrf: Couldn't find any CSRF vulnerabilities.

Nmap done: 1 IP address (1 host up) scanned in 327.24 seconds
salva@salva-VirtualBox:~$ 
```

- Guardar resultados en un archivo
 - nmap -oN resultado.txt <IP>

```

salva@salva-VirtualBox:~$ nmap -A 3.216.161.112
Nmap scan report for ec2-3-216-161-112.compute-1.amazonaws.com (3.216.161.112)
Host is up (0.033s latency).

PORT      STATE SERVICE
80/tcp    open  http
|_http-dombased-xss: Couldn't find any DOM based XSS.
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-csrf: Couldn't find any CSRF vulnerabilities.

Nmap done: 1 IP address (1 host up) scanned in 327.24 seconds
salva@salva-VirtualBox:~$ sudo nmap -oN resultado.txt 3.216.161.112
Starting Nmap 7.93 ( https://nmap.org ) at 2024-10-30 20:06 -03
Nmap scan report for ec2-3-216-161-112.compute-1.amazonaws.com (3.216.161.112)
Host is up (0.019s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
443/tcp   open  https

Nmap done: 1 IP address (1 host up) scanned in 8.03 seconds
salva@salva-VirtualBox:~$ 

```

```

GNU nano 7.2
resultado.txt
# Nmap 7.93 scan initiated Wed Oct 30 20:06:26 2024 as: nmap -oN resultado.txt 3.216.161.112
Nmap scan report for ec2-3-216-161-112.compute-1.amazonaws.com (3.216.161.112)
Host is up (0.019s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
443/tcp   open  https

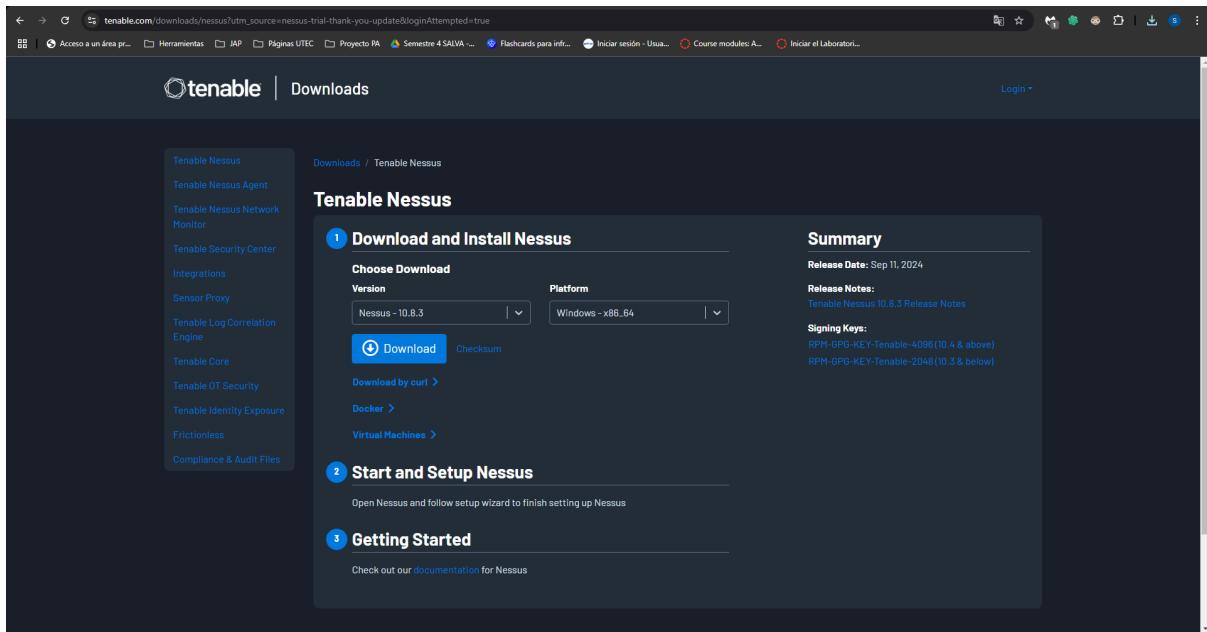
# Nmap done at Wed Oct 30 20:06:34 2024 -- 1 IP address (1 host up) scanned in 8.03 seconds

^G Ayuda      ^O Guardar      ^W Buscar      ^K Cortar      ^T Ejecutar      ^C Ubicación      M-U Deshacer
^X Salir      ^R Leer fich.  ^\ Reemplazar  ^U Pegar       ^J Justificar  ^/ Ir a linea  M-E Rehacer

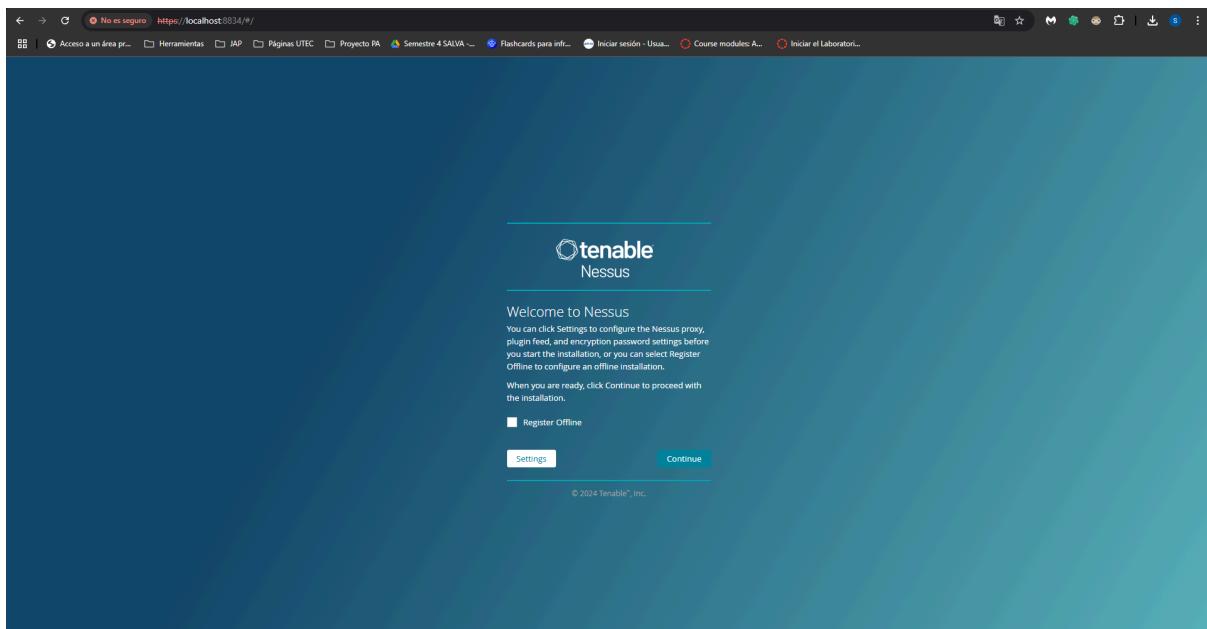
```

Nessus Pro

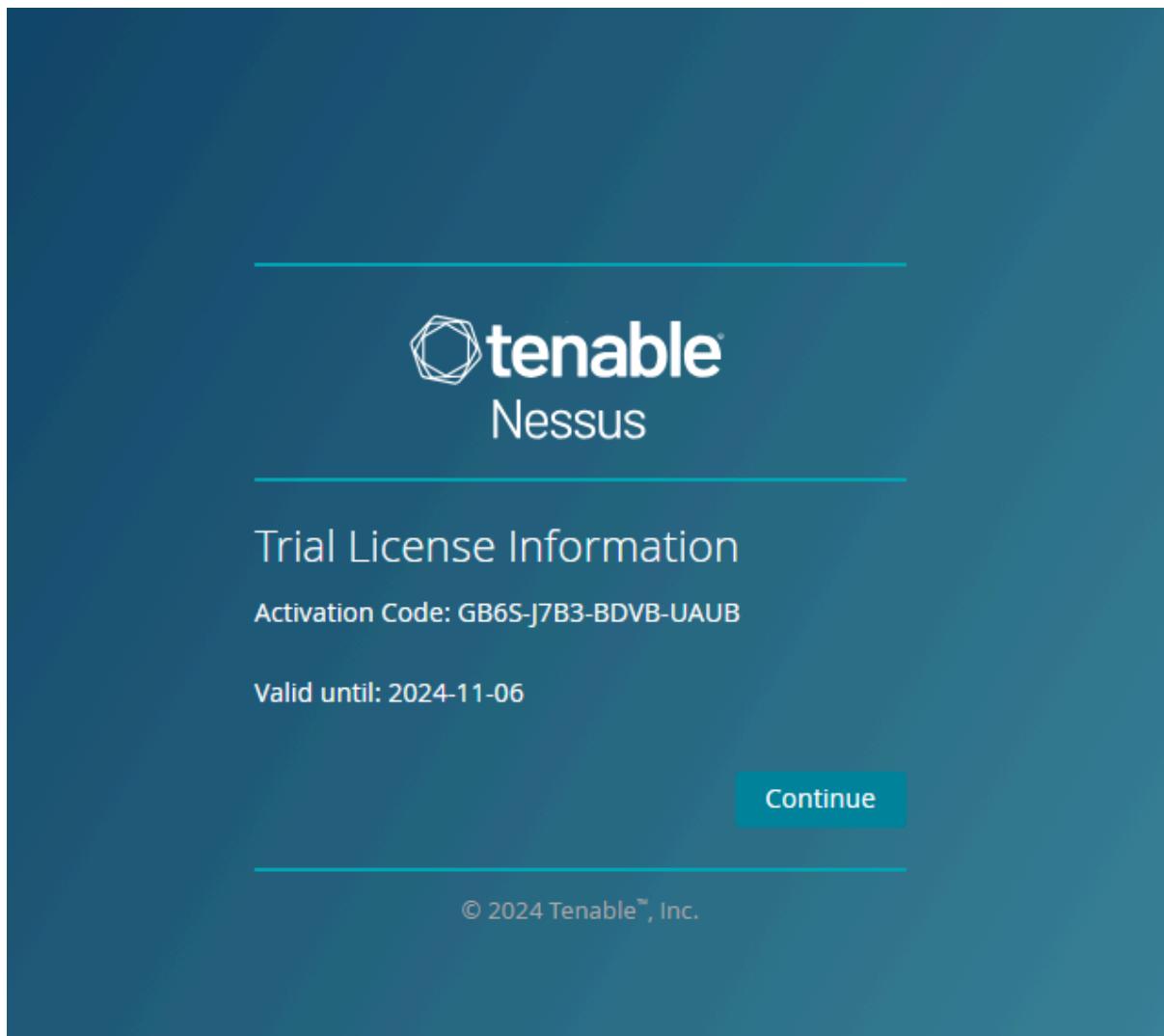
- Instalar y Configurar Nessus Pro
 - Descargar Nessus Pro: Ir al sitio oficial de Tenable y descargar la versión adecuada para tu sistema operativo.
 - Instalar Nessus: Ejecutar el archivo de instalación y seguir las instrucciones del asistente para instalar Nessus Pro en el sistema.



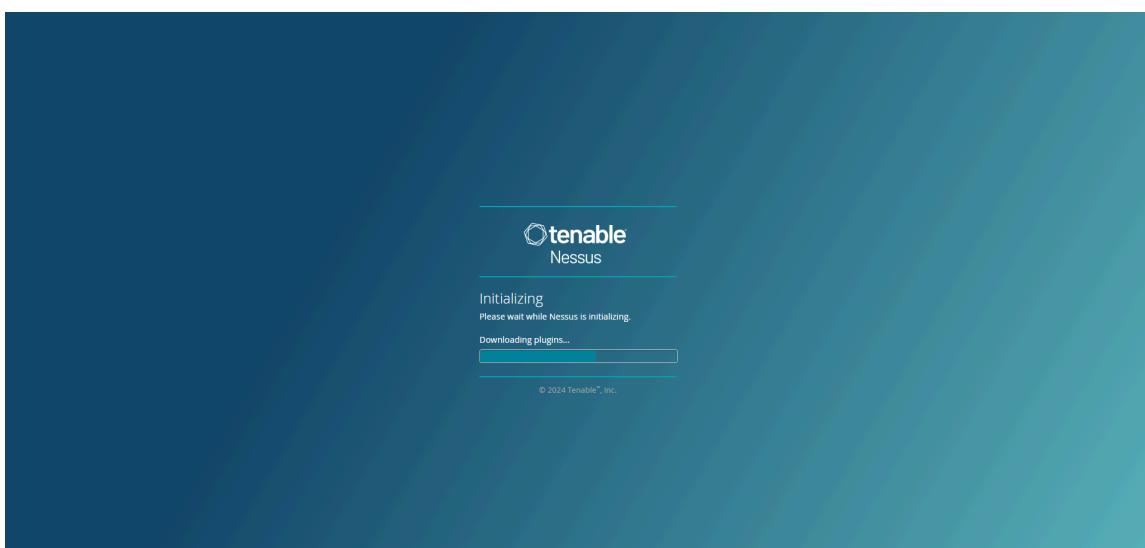
- Configurar un Nuevo Escaneo
 - Iniciar sesión en Nessus: Abrir el navegador e ingresar a la interfaz web de Nessus (generalmente en <https://localhost:8834>).



- Para la licencia, elegir “Nessus Expert” e ingresar el correo electrónico para activar la licencia gratuita de prueba. Luego presionar continuar.



- Ingresar un nombre de usuario y una contraseña para la cuenta de administrador y presionar continuar.
 - Esperar mientras cargan los últimos plugins.



- Crear un nuevo escaneo: En la página de inicio, seleccionar “New Scan” y luego elegir la opción de escaneo “Basic Network Scan”.
 - Configurar el objetivo del escaneo: Ingresar la IP externa pública o el dominio de la aplicación en el campo correspondiente.

The screenshot shows the Tenable Nessus Expert web interface. On the left, there's a sidebar with 'FOLDERS' (My Scans, All Scans, Trash) and 'RESOURCES' (Policies, Plugin Rules, Customized Reports, Terrascan, Web App Scanning). The main area is titled 'My Scans' and shows one scan named 'My Host Discovery Scan'. At the top right, there are buttons for 'Import', 'New Folder', and a prominent blue 'New Scan' button, which is circled in red.

This screenshot shows the 'New Scan / Basic Network Scan' configuration dialog. The left sidebar has 'Scans' selected. The main panel has 'Settings' tab active. Under 'General Settings', the 'Name' field is set to 'Basic Scan Obligatorio'. The 'Targets' field contains the IP addresses '54.146.129.189, 3.216.161.112, obligatorioinfra.cloud-ip.biz'. There are tabs for 'Credentials' and 'Plugins' at the top of the dialog. At the bottom, there are 'Save' and 'Cancel' buttons.

- Presionar guardar e ir a la pantalla principal, ahora estará visible el escaneo creado. Presionar “Launch”.

The screenshot shows the 'My Scans' section of the Otenable Nessus Expert web interface. On the left, there's a sidebar with 'Folders' (My Scans, All Scans, Trash) and 'Resources' (Policies, Plugin Rules, Customized Reports, Terrascan, Web App Scanning). The main area has a search bar ('Search Scans') and a table with one row. The table columns are Name, Scan Type, Schedule, and Last Scanned. The 'Last Scanned' column header is circled in red.

Name	Scan Type	Schedule	Last Scanned
Basic Scan Obligatorio	Vulnerability	On Demand	N/A

- Esperar a que se termine de ejecutar.

This screenshot is identical to the previous one, but the 'Last Scanned' column for the 'Basic Scan Obligatorio' entry now displays the timestamp 'Today at 8:56 PM', indicating the scan has been completed.

Name	Scan Type	Schedule	Last Scanned
Basic Scan Obligatorio	Vulnerability	On Demand	Today at 8:56 PM

- Ahora se puede ver la información extraída durante el análisis al hacer click sobre el escaneo.

Scan Details

- Critical Vulnerabilities: 0
- High Vulnerabilities: 0
- Medium Vulnerabilities: 2
- Low Vulnerabilities: 0

Scan Notes

Search Notes: 1 Notes

Scan Notes: Network Interface Not Supported
127.63.0.1/10180 The network interface '\Device\NPF_{0D8C2DEE-33EB-11EE-9A24-806F6F6E6963}' does not support packet forgery. This prevents Nessus from determining whether some of the target hosts are alive and from performing a full scan.

Severity	CVSS	VPR	EPSS	Name	Family	Count
MIXED	SSL (Multiple Issues)	General	6
INFO	HTTP (Multiple Issues)	Web Servers	3
INFO	TLS (Multiple Issues)	Service detection	2
INFO				Service Detection	Service detection	3
INFO				Nessus SYN scanner	Port scanners	2
INFO				Common Platform Enumeration (CPE)	General	1
INFO				Device Type	General	1
INFO				Host Fully Qualified Domain Name (FQDN) Resolution	General	1
INFO				Nessus Scan Information	Settings	1
INFO				Non-compliant Strict Transport Security (STS)	Service detection	1
INFO				OS Identification	General	1
INFO				SSL / TLS Versions Supported	General	1
INFO				Strict Transport Security (STS) Detection	Service detection	1

Scan Details

- Policy: Basic Network Scan
- Status: Completed
- Severity Base: CVSS v3.0
- Scanner: Local Scanner
- Start: Today at 8:40 PM
- End: Today at 8:56 PM
- Elapsed: 16 minutes

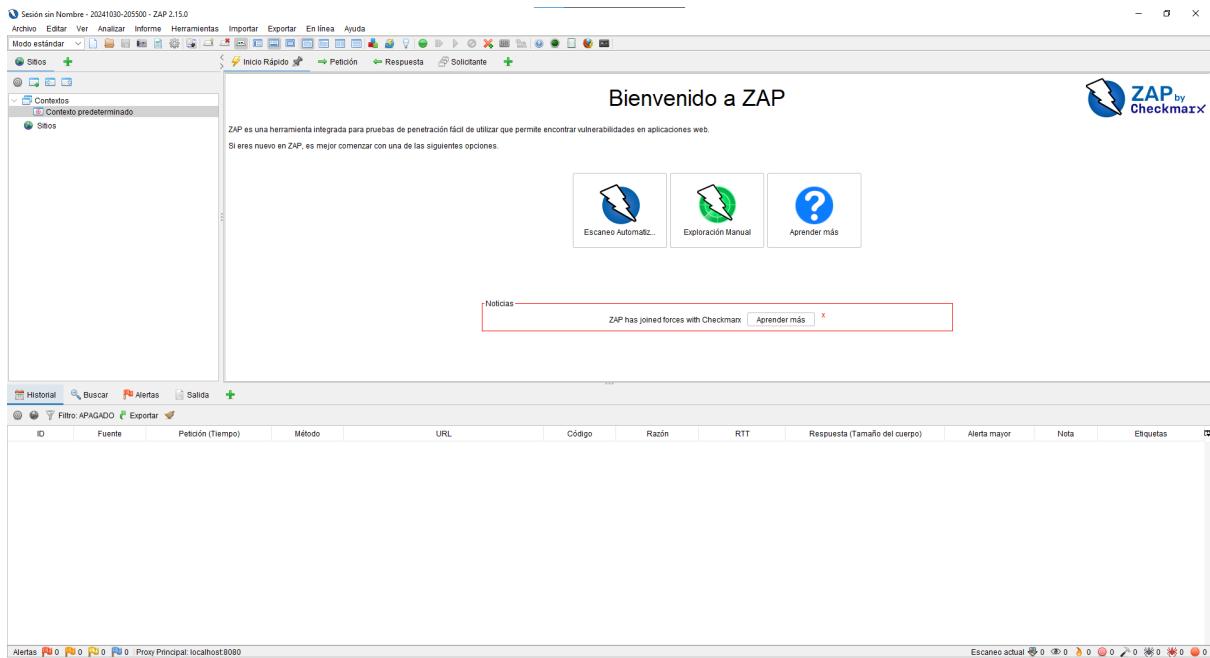
Vulnerabilities

- Opcionalmente se puede exportar el informe para almacenar la información.

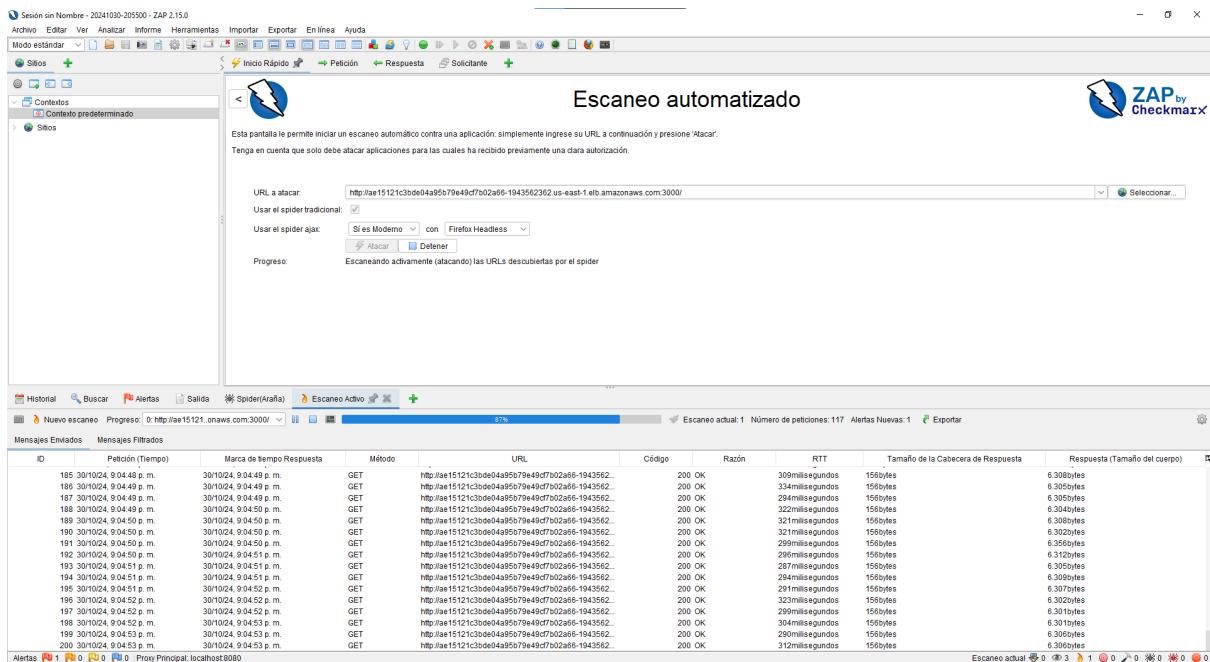
OWASP ZAP

- Descargar OWASP ZAP
 - Ir a la página oficial de OWASP ZAP (<https://www.zaproxy.org/download/>) y descargar la versión adecuada para tu sistema operativo.
 - Instalar el programa, puede requerir tener instalado java (<https://www.oracle.com/java/technologies/downloads/?er=221886>)
 - Ejecutarlo.

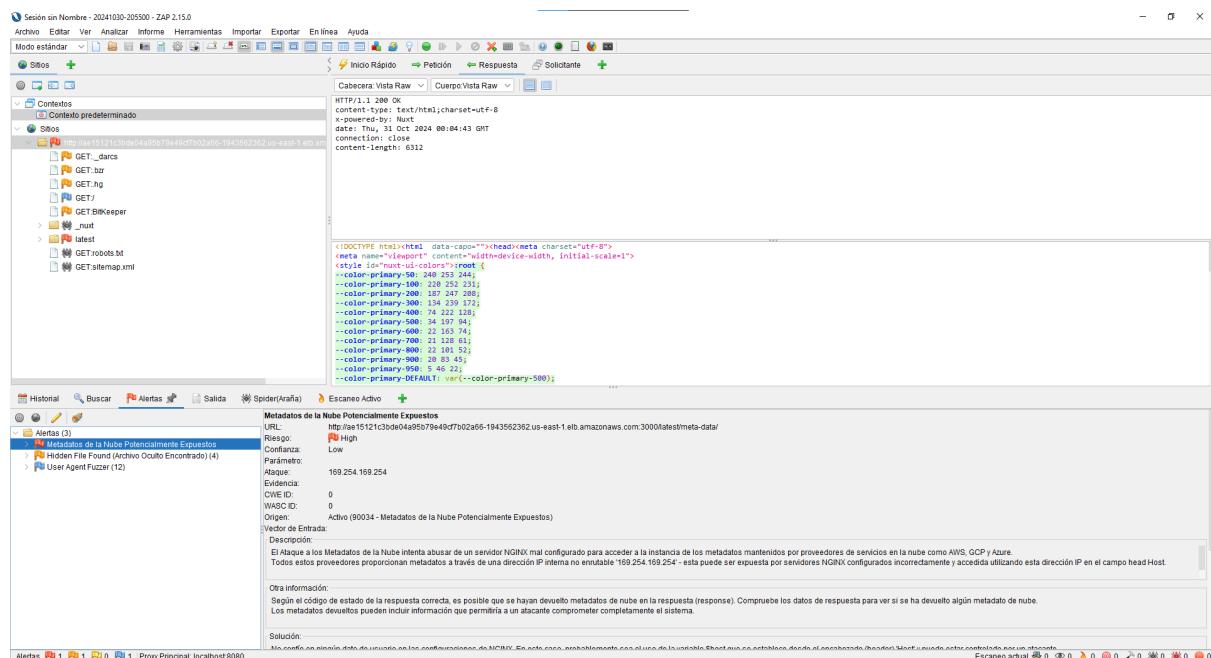
- Elegir guardar la sesión con fecha.



- Elegir “Escaneo automatizado”.
 - Una vez ahí, ingresar la URL del sitio web.
 - Apretar “Atacar”.
 - Esperar a que termine el ataque.

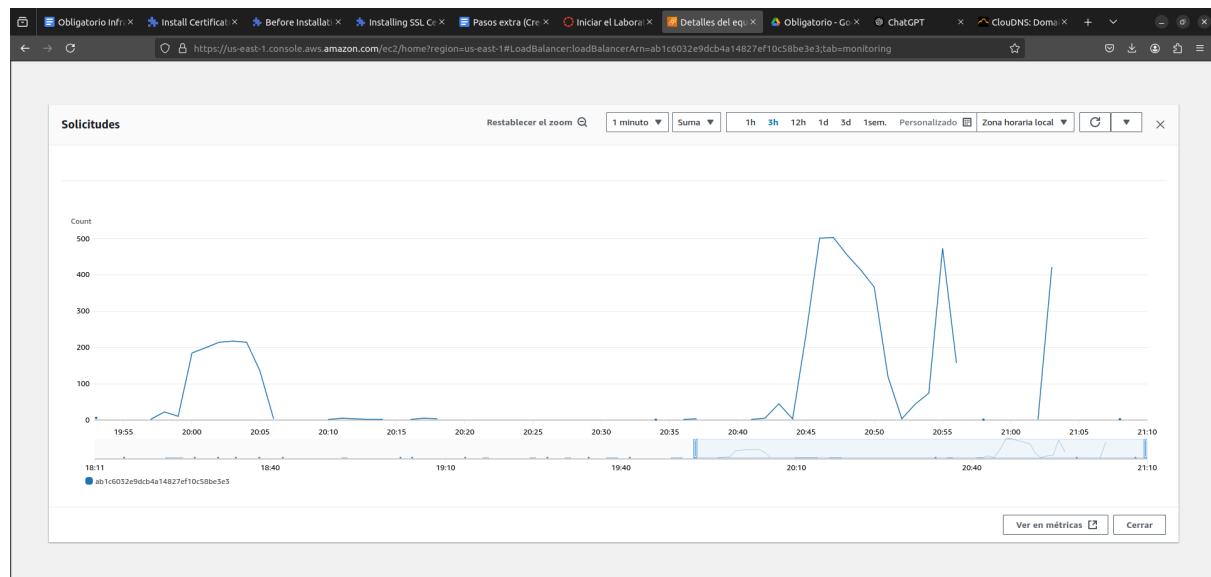


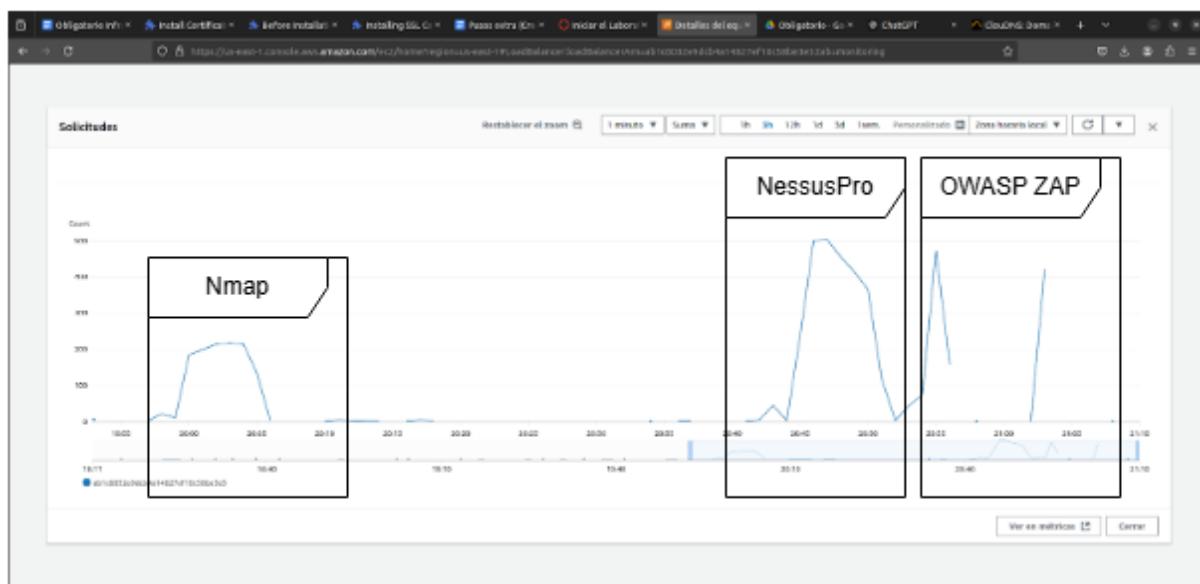
- Una vez terminado el ataque, mostrará alertas y otra información de interés.



Resultados de los tests en la página

Observando las métricas de tráfico de la página web durante los escaneos, este fue el resultado:





Se observan picos claros a la hora de ejecutar los tests de seguridad. Como se puede ver, el mayor pico lo realiza NessusPro, que fue el que por defecto ejecutó más pruebas de vulnerabilidad. A su vez, resulta en el programa más profesional, ya que cuenta con certificaciones reconocidas y tiene el mayor precio de todos los utilizados (4000 dólares por año). Fue también el que devolvió más información y el que tenía más configuraciones (con el programa por defecto, sin plugins extra).

Diagrama de Red de nuestra Aplicación Web

A continuación adjuntamos un link de Google Drive con la imagen del diagrama de red.

https://drive.google.com/file/d/1OQeDXxXqyz8QzbwGBS1o9lkKKn0Pl_3A/view?usp=sharing

Video demostrativo

A continuación adjuntamos un enlace con un vídeo que demuestra la página en funcionamiento y aspectos generales de la configuración.

<https://drive.google.com/file/d/1-7Y04M92CLr--P8GbVy-8Xm9VwjFSWwY/view?usp=sharing>

Conclusión

Logramos levantar una aplicación web utilizando servicios de AWS como EKS, ECR, Route 53, S2, IAM, entre otros. La aplicación web contaba con un frontend, un backend y una base de datos y era accesible desde cualquier dispositivo, permitiendo la creación y visualización de notas.

Pudimos registrar un dominio para la aplicación utilizando DNS y crear un certificado SSL para que la página fuera segura y se permitiera el encriptado de encabezados y cuerpos en las peticiones web.

A su vez, gracias a herramientas de ciberseguridad logramos identificar vulnerabilidades y registrarlas, aprendiendo de primera mano sobre aspectos fundamentales a la hora de desplegar una aplicación en internet.

Por último, creamos una guía detallada de todos los pasos necesarios para llevar a cabo esta tarea, incluyendo detalles, recomendaciones y alternativas; incluyendo en estos pasos imágenes a modo de guía visual.

Fuentes

Chen, X., Kim, M., & Andersen, M. (2022). *A survey on fuzzing for software security: State of the art and challenges*. ACM Computing Surveys, 54(5), 1-42.
<https://doi.org/10.1145/3464373>.

Security Intelligence. (2023). *Fuzz Testing: How it works and why it matters*. Recuperado de <https://securityintelligence.com/news/fuzz-testing-explained/>

Snyk. (s.f.). *Fuzz testing tools & techniques for software security*. Recuperado de <https://snyk.io/learn/fuzzing/>

Palo Alto Networks. (s.f.). *What Is Static Application Security Testing (SAST)?*. Recuperado de <https://www.paloaltonetworks.com/cyberpedia/what-is-sast>

Snyk. (s.f.). *SAST tools & testing: Benefits and limitations*. Recuperado de <https://snyk.io/learn/static-application-security-testing-sast/>

Veracode. (2023). *Static Application Security Testing (SAST) Best Practices*. Recuperado de <https://www.veracode.com/products/sast-static-analysis>

Lyon, G. F. (2009). *Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning*. Insecure.org.

Convery, S. (2023). *Network Security Tools & Techniques: A Guide to Using Nmap for Network Scanning*. Journal of Network Security, 11(4), 128-136.

Nmap Project. (s.f.). *Nmap Network Mapper*. Recuperado de <https://nmap.org/>

Tenable. (s.f.). *Nessus Pro - The Most Comprehensive Vulnerability Scanner on the Market*. Recuperado de <https://www.tenable.com/products/nessus/nessus-professional>

Aldeweesh, A., et al. (2020). *A Comparative Evaluation of Open-Source and Proprietary Vulnerability Scanners*. Journal of Information Security, 10(3), 45-58. <https://doi.org/10.4236/jis.2020.113005>

Cloud Native Computing Foundation. (s.f.). *Kubernetes Security Best Practices*. Recuperado de <https://kubernetes.io/docs/concepts/security/>

Amazon Web Services. (s.f.). *AWS Security Best Practices*. Recuperado de <https://docs.aws.amazon.com/general/latest/gr/aws-security-best-practices.html>

OWASP Foundation. (s.f.). *OWASP Top Ten Project*. Recuperado de <https://owasp.org/www-project-top-ten/>

AWS. (s.f.). *Best Practices for Securing Amazon EKS*. Recuperado de <https://aws.amazon.com/blogs/containers/best-practices-for-securing-amazon-eks/>