



Proyecto

Objetivos

- Conocer e implementar el protocolo de red https
- Configurar conexiones seguras entre servidores a través de ssh.
- Conocer qué es un certificado SSL y su relación con https.
- Crear e instalar un certificado SSL.
- Conocer e implementar nombres de dominio.
- Conocer el uso de los registros CNAME y A.
- Aplicar los conocimientos adquiridos anteriormente sobre redes en la nube.
- Utilizar balanceo de carga.
- Crear ACLs para controlar el tráfico en las subredes.
- Configuración de comunicación a través del protocolo SSH.
- Configurar Security Group para controlar el tráfico en las instancias.
- Creación de redes públicas y privadas en AWS.
- Uso de tablas de ruteo para garantizar el tráfico dentro de una VPC.

Descripción del problema

La Escuela de Ciencias y Sistemas de la Universidad de San Carlos de Guatemala se encuentra muy satisfecha con el demo que usted presentó para el nuevo sistema, por lo que ha decidido que usted está más que capacitado para desarrollar la primera fase del proyecto.

El proyecto luego de un análisis ha sufrido algunos cambios, como el hecho de que ahora ya no se desea utilizar un solo servidor en la nube, si no que se desea contar con una infraestructura conformada por varios servidores.

Se desea mantener el módulo de inserción de reportes de los estudiantes, por lo que deberá implementar dicha funcionalidad en el nuevo sistema.

Referencia del módulo en el enunciado de la práctica: [Enunciado Practica 4](#)

Módulo de Asistencia

Adicionalmente se desea agregar un nuevo módulo, el cual permita pasar asistencia a los estudiantes en los eventos organizados por la escuela.

Dicho módulo de asistencia debe recolectar los siguientes datos:

- Carnet
- Nombre Estudiante
- Nombre del evento
- Id de evento(es un valor numérico entero)
- Cargar Imagen con captura de asistencia(Se recomienda usar AWS S3 para almacenar las imágenes).

Adicionalmente a esos datos, se debe registrar la fecha y hora en que se ingresó la asistencia, así como el servidor que atendió la petición de inserción de información.

Se debe poder realizar las siguientes acciones en dicho módulo:

- Agregar asistencia.
- Listar asistencia al evento(por id de evento).
- Listar eventos a los que asistió un alumno (por carnet).

Arquitectura

Inicialmente el sistema contará con dos componentes principales:

- Backend.
- Frontend.

Backend

Se le solicita desarrollar una API con los endpoints necesarios para proveer los servicios del módulo de asistencia y el módulo de reportes.

Dicha API será consumida por un cliente web, el cual debe consumir la API a través de https.

De lo anterior, se le presenta el siguiente diagrama:

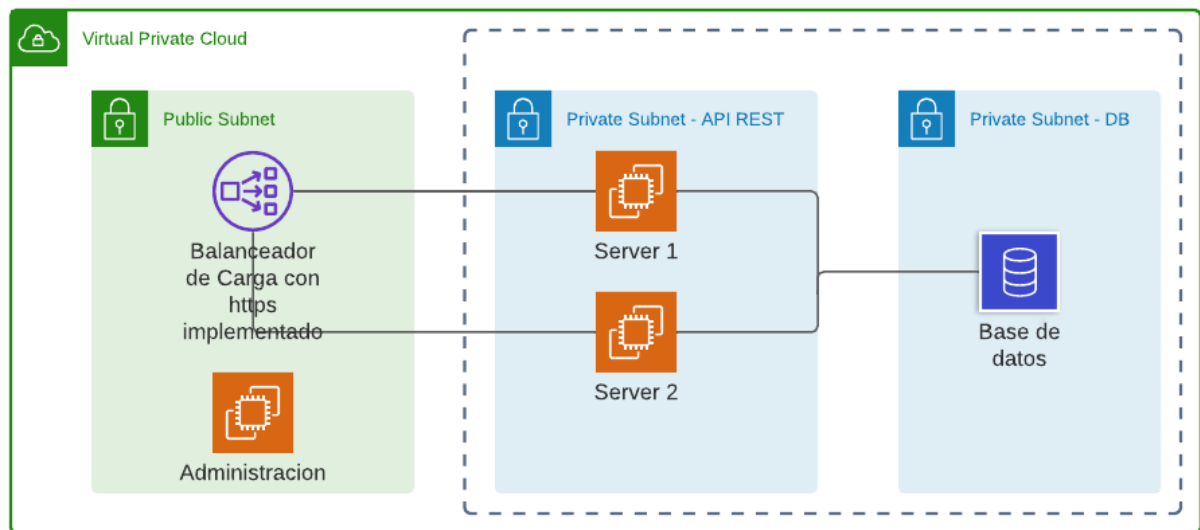


Diagrama de API

Se debe crear una VPC de nombre ProyectoFinalR2.

Dicha VPC contendrá las siguientes subredes:

- 1 subred pública, de nombre R2Publica, que contendrá un balanceador de carga y una EC2 que servirá para administrar las instancias en subredes privadas.
- 2 subredes privadas:
 - R2Privada1: esta subred contendrá las dos instancias del api.
 - R2Privada2: que contendrá la base de datos.

Configuración de conexiones ssh entre EC2 administración y EC2 en subredes privadas

Se requiere que realice las configuraciones necesarias para permitir la conexión a través de ssh entre la instancia "Administracion" en la subred pública y las instancias de servidores y base de datos en las subredes privadas.

Por lo tanto se le solicita la creación de un usuario en la instancia “Administración” con el nombre **adm_r2p1_g#grupo**, donde #grupo es el número de su grupo, con contraseña “redes2_pro2@2021”.

Debe realizar las configuraciones necesarias para permitir la conexión a través de ssh con este usuario.

Balanceo de carga

Se debe configurar un balanceo de carga que reparta el tráfico entre el Server 1 y el Server 2. Es importante aclarar que la comunicación entre el cliente web y la API debe ser por medio de https, por lo que debe asegurarse de implementar dicho protocolo.

Nota: Se deja a su elección la forma en que implementara el balanceo de carga, pudiendo elegir entre el servicio de balanceo de carga de AWS o instalar algún software como NGINX en una instancia EC2.

HTTPS

Para la implementación de https se le recomienda utilizar un certificado gratuito. Puede elegir entre utilizar un certificado de AWS CM, o un certificado de Let's Encrypt.

Seguridad

Se le solicita que proponga y cree las políticas de seguridad para el tráfico a través de la creación de ACLs y Security Groups. Este paso es de vital importancia para agregar capas extra de seguridad dentro de la VPC.

Debe documentar y justificar las reglas creadas en la documentación (documentar reglas de las ACL utilizadas y puertos abiertos de las instancias EC2 o LoadBalancer).

Nombres de Dominio

También se le solicita crear nombres de dominio, que permitan acceder a la API.

Se le solicita un nombre de dominio principal y un nombre de dominio secundario que funcionará como un alias para el dominio principal.

- api.<dominio1> (principal)
- servicios.<dominio2> (secundario)

La API debe poder ser accedida desde cualquiera de los dos dominios definidos.

Debe agregar a la documentación la información relacionada a los registros utilizados para llevar a cabo lo solicitado.

También se debe configurar las siguientes urls que permitan acceder al frontend:

- portal.<dominio1>
- portal.<dominio2>

Debe configurar los registros DNS de tipo A y CNAME necesarios para cumplir con lo requerido.

Nota: Puede utilizar cualquier proveedor de nombres de dominio gratuito para adquirir los dominios que utiliza, o puede utilizar dominios que ya tenga, pero debe respetar el nombre de los subdominios propuestos.

Cliente con interfaz de usuario

Inicialmente, se le solicita un Cliente sencillo que permita consumir la API. Las herramientas a utilizar para el desarrollo del cliente quedan a discreción del estudiante. **El sitio web debe ser alojado en github pages, y debe funcionar sobre https.**

Restricciones

- El proyecto se realizará en grupos de máximo 4 integrantes.
- Todos los integrantes del grupo deben de tener conocimiento del desarrollo de la práctica.
- Se debe utilizar AWS como proveedor de servicios en la nube.
- Para la calificación se debe presentar el proyecto en una computadora de los integrantes del grupo.
- Se debe de crear un repositorio de GitHub donde se irá actualizando el desarrollo de la práctica, el cual debe de contener como mínimo 2 commits por semana por parte de cada uno de los integrantes del grupo.
- Durante la calificación se preguntará información relevante de la práctica para comprobar la autoría del mismo.
- El manual técnico debe ser un pdf con el nombre **Proyecto_Manual_#grupo.pdf**

Penalizaciones

- Falta de seguimiento de desarrollo continuo por medio de Github tendrá una penalización del 10%.
- Falta de seguimiento de instrucciones conforme al método de entrega (nombre del repositorio) tendrá una penalización del 5%.
- Falta de puntualidad conforme a la entrega tendrá una penalización de la siguiente manera:
 - a. 1-10 minutos – 10%.
 - b. 11-59 minutos – 30%.
 - c. Pasados 60 minutos tendrá una nota de 0 y no se calificará.

Observaciones

- La entrega se realizará por medio de **Github**, el código se debe guardar en un directorio llamado Proyecto1 dentro del repositorio creado anteriormente

- Además de tener a su auxiliar como colaborador del repositorio para tener un control y orden de las personas que entreguen deberán de colocar el Link de su repositorio en la Tarea que cada auxiliar asignará en su plataforma correspondiente.
- Fecha y hora de entrega: **Martes 2 de Noviembre, antes de las 23:59 horas.**
- **Las copias serán penalizadas con una nota de 0 y castigadas según lo indique el reglamento.**