**Título del Video**: Implementación de un Sitio Web Seguro en Google Cloud Platform con Apache2 y Certificados Digitales

**Introducción**

¡Hola a todos! En este video, voy a explicar paso a paso cómo implementé un sitio web seguro utilizando una máquina virtual en Google Cloud Platform, Apache2 como servidor web y certificados digitales para habilitar HTTPS. También mostraré cómo configuré el firewall y cómo utilicé NoMachine para conectarme a la máquina virtual a través del puerto 4000. Además, explicaré el funcionamiento de los certificados digitales y su importancia en la seguridad web.

**Parte 1: Creación y Configuración de la Máquina Virtual en Google Cloud Platform**

1. **Creación de la Instancia en GCP**

Comencé creando una cuenta en Google Cloud Platform y aproveché el crédito gratuito que ofrecen. En la consola de GCP, fui a "Compute Engine" y seleccioné "VM Instances" para crear una nueva instancia.

* + **Configuración de la Instancia**:
    - **Nombre**: my-server
    - **Región**: elegí una región cercana para mejorar la latencia.
    - **Tipo de Máquina**: seleccioné e2-micro, que es suficiente para este proyecto.
    - **Sistema Operativo**: opté por **Ubuntu 22.04 LTS**.

1. **Configuración del Firewall en GCP**

Para permitir el acceso necesario a la instancia, configuré las reglas de firewall:

* + **Puertos Permitidos**:
    - **Puerto 80**: para tráfico HTTP.
    - **Puerto 443**: para tráfico HTTPS.
    - **Puerto 4000**: para conectarme a través de NoMachine.
  + **Creación de Reglas de Firewall**:
    - En "VPC Network" > "Firewall rules", creé una regla llamada allow-ports que permite tráfico entrante en los puertos 80, 443 y 4000 desde cualquier IP (0.0.0.0/0).

**Parte 2: Conexión a la Máquina Virtual usando NoMachine**

1. **Instalación de NoMachine en la Máquina Virtual**
   * **Descarga e Instalación**:

wget https://download.nomachine.com/download/7.10/Linux/nomachine\_7.10.1\_1\_amd64.deb

sudo dpkg -i nomachine\_7.10.1\_1\_amd64.deb

* + NoMachine es una herramienta que permite acceso remoto a escritorios Linux. Elegí utilizarlo en el puerto 4000 para mayor seguridad y evitar conflictos con puertos comunes.

1. **Configuración de NoMachine**
   * **Cambiar el Puerto por Defecto a 4000**:
     + Edité el archivo de configuración de NoMachine:

sudo nano /usr/NX/etc/server.cfg

* + - Modifiqué la línea:

NXPort = 4000

* + **Reinicié el Servicio de NoMachine**:

sudo /etc/NX/nxserver --restart

1. **Conexión desde mi Equipo Local**
   * **Instalación de NoMachine en mi Computadora**: Descargué e instalé NoMachine desde su sitio oficial.
   * **Establecer la Conexión**:
     + Ingresé la dirección IP externa de la instancia y el puerto 4000.
     + Utilicé las credenciales de usuario lab14 que creé previamente en la máquina virtual.

**Parte 3: Instalación y Configuración de Apache2**

1. **Instalación de Apache2**
   * Actualicé los paquetes y luego instalé Apache2:

sudo apt update

sudo apt install apache2 -y

1. **Configuración del Sitio Web**
   * **Creación del Directorio del Sitio**:

sudo mkdir -p /var/www/redes2lab14.online/public\_html

* + **Asignación de Permisos**:

sudo chown -R $USER:$USER /var/www/redes2lab14.online/public\_html

sudo chmod -R 755 /var/www

* + **Creación de la Página Principal**:

nano /var/www/redes2lab14.online/public\_html/index.html

* + - **Contenido del index.html**:

<!DOCTYPE html>

<html>

<head>

<title>Práctica de Certificados Digitales R2-2024</title>

</head>

<body>

<h1>Nombre Completo: [Mi Nombre]</h1>

<h2>Número de Carné: [Mi Número de Carné]</h2>

<h3>Carrera: [Mi Carrera]</h3>

<p>Práctica de Certificados Digitales R2-2024</p>

</body>

</html>

1. **Configuración del Virtual Host en Apache2**
   * **Crear el Archivo de Configuración**:

sudo nano /etc/apache2/sites-available/redes2lab14.online.conf

* + - **Contenido del Archivo**:

<VirtualHost \*:80>

ServerAdmin webmaster@localhost

ServerName redes2lab14.online

ServerAlias www.redes2lab14.online

DocumentRoot /var/www/redes2lab14.online/public\_html

ErrorLog ${APACHE\_LOG\_DIR}/error.log

CustomLog ${APACHE\_LOG\_DIR}/access.log combined

</VirtualHost>

* + **Habilitar el Sitio y Deshabilitar el Predeterminado**:

sudo a2dissite 000-default.conf

sudo a2ensite redes2lab14.online.conf

* + **Reiniciar Apache2**:

sudo systemctl restart apache2

**Parte 4: Compra y Configuración del Dominio en GoDaddy**

1. **Compra del Dominio**
   * Compré el dominio redes2lab14.online en GoDaddy, aprovechando las ofertas disponibles para dominios de bajo costo.
2. **Configuración de los Registros DNS**
   * **Registro A (@)**:
     + **Tipo**: A
     + **Nombre**: @
     + **Valor**: [Dirección IP externa de mi instancia en GCP]
   * **Registro CNAME (www)**:
     + **Tipo**: CNAME
     + **Nombre**: www
     + **Valor**: redes2lab14.online
   * Esto asegura que tanto redes2lab14.online como www.redes2lab14.online apunten a mi servidor.

**Paso 4: Investigar y utilizar Let's Encrypt**

**4.1. ¿Qué es Let's Encrypt?**

* **Let's Encrypt** es una autoridad certificadora que proporciona certificados SSL/TLS gratuitos para habilitar HTTPS en sitios web de forma segura y automatizada.

**4.2. ¿Para qué sirve?**

* Permite cifrar la comunicación entre el servidor y los clientes, mejorando la seguridad y la confianza en tu sitio web.

**4.3. ¿Cómo se utiliza?**

* Usando herramientas como **Certbot** para automatizar la obtención y renovación de certificados.

**Parte 5: Instalación y Configuración de Certbot para SSL**

1. **Instalación de Certbot y el Plugin de Apache**

sudo apt install certbot python3-certbot-apache -y

1. **Obtención y Configuración del Certificado SSL**
   * Ejecuté Certbot:

sudo certbot --apache

* + **Proceso Interactivo**:
    - Ingresé mi correo electrónico.
    - Acepté los términos de servicio.
    - Seleccioné mis dominios (redes2lab14.online y www.redes2lab14.online).
    - Opté por redirigir todo el tráfico HTTP a HTTPS.

1. **Verificación del Acceso Seguro**
   * Visité https://redes2lab14.online y verifiqué que el sitio se cargaba de forma segura, con el candado en la barra de direcciones.
2. **Configuración de Renovación Automática**
   * Probé la renovación automática:

sudo certbot renew --dry-run

* + Confirmé que la renovación automática está configurada correctamente.

**Parte 6: Funcionamiento de los Certificados Digitales**

1. **¿Qué es un Certificado Digital?**
   * Un certificado digital es un archivo que vincula una clave criptográfica con la identidad de una organización o individuo. Permite establecer conexiones seguras y cifradas entre un servidor y un cliente.
2. **Funcionamiento de SSL/TLS**
   * SSL/TLS es un protocolo que utiliza certificados digitales para cifrar la comunicación entre el navegador del usuario y el servidor web.
   * Garantiza:
     + **Confidencialidad**: La información transmitida no puede ser leída por terceros.
     + **Integridad**: Los datos no han sido alterados durante la transmisión.
     + **Autenticidad**: Verifica que el servidor es quien dice ser.
3. **Importancia de HTTPS**
   * Proporciona seguridad y confianza a los usuarios.
   * Mejora el posicionamiento en motores de búsqueda.
   * Es esencial para proteger información sensible.

**Parte 7: Configuración de Firewall en GCP y en la Instancia**

1. **Firewall en Google Cloud Platform**
   * Apliqué reglas de firewall para asegurar mi instancia:
     + **Permití solo los puertos necesarios**: 80 (HTTP), 443 (HTTPS) y 4000 (NoMachine).
     + **Restringí otros puertos** para mejorar la seguridad.
2. **Firewall en la Instancia (UFW)**
   * Instalé y configuré UFW (Uncomplicated Firewall) en Ubuntu:

sudo apt install ufw -y

sudo ufw allow 'Apache Full'

sudo ufw allow 4000/tcp

sudo ufw enable

* + Esto ayuda a proteger la instancia de accesos no autorizados.

**Parte 8: Demostración Final**

* **Acceso al Sitio Web Seguro**
  + Mostré cómo acceder al sitio web utilizando HTTPS y verificando que el certificado es válido.
* **Verificación del Certificado**
  + En el navegador, hice clic en el candado y mostré detalles del certificado emitido por Let's Encrypt.
* **Redirección Automática**
  + Probé accediendo a http://redes2lab14.online y comprobé que redirige automáticamente a https://redes2lab14.online.

**Conclusión**

En este proyecto, aprendí a desplegar un sitio web seguro en la nube utilizando Google Cloud Platform, Apache2 y certificados digitales de Let's Encrypt. Implementé medidas de seguridad adicionales, como la configuración de firewalls y el uso de conexiones seguras para acceder a la máquina virtual.

La seguridad en la web es fundamental en el mundo actual, y comprender cómo funcionan los certificados digitales y cómo implementarlos es una habilidad valiosa. Este laboratorio me permitió aplicar conocimientos teóricos en una práctica real, fortaleciendo mi comprensión de la infraestructura de redes y la seguridad informática.