Laboratorio Seguridad

Daniel Felipe Gomez Suarez

September 2020

1 Introduccion

Este laboratorio es el complemento de las actividades pasadas donde hemos trabajado aplicaciones web. en este programa evidenciamos el intercambio de certificados de seguridad garantizando con la autenticación una seguridad más completa. con los certificados garantizamos que se puedan intercambiar los navegadores.

Spark nos provee una opción de restringir la navegación dentro de los paths. dentro de una etiqueta BEFORE la cual nos permite compartirla cual mediante una función lambda validar los usuarios.

1.1 Certificados SSL

Figure 1: SSL

SSL (Secure Sockets Layer o capa de conexión segura) es un estándar de seguridad global que permite la transferencia de datos cifrados entre un navegador y un servidor web. Es utilizado por millones1 de empresas e individuos en línea a fin de disminuir el riesgo de robo y manipulación de información confidencial (como números de tarjetas de crédito, nombres de usuario, contraseñas, correos electrónicos, etc.) por parte de hackers y ladrones de identidades. Básicamente, la capa SSL permite que dos partes tengan una "conversación" privada

1.1.1 TrustStore

Un truststore se utiliza para negociar una conexión segura entre un cliente y un servidor. Por ejemplo, tenemos una aplicación cliente que tiene hacer una petición https a un web service y obtener una serie de datos, para que esto funcione es necesario establecer una relación de confianza entre el cliente y el servidor

Aqui es donde hacemos la confiabilidad de un servicio, para ello cada servicio web que usamos tiene sus correspondientes KEYSTORE las cuales son las que se guardan y posteriormente nos dejaran generaran el certificado.

1.2 Keystore

Un keystore se suele usar por ejemplo en un servicio web, este para que puedas enviar una petición puede requerir un certificado con clave privada. Entonces en tu aplicación debes tener un keystore con el certificado que requiera el web service.

2 Diseño

Este modelo es el propuesto para el trabajo, donde todos los servicios están desplegados en AWS en un contenedor EC2 como se mencionó anteriormente, para esto lo primero que tenemos que hacer es tener la instancia del EC2 habilitada y los puertos que se requieren para el despliegue, es decir los puertos mapeados en el Docker, estos puertos serán los usados para acceder a cada uno de los microservicios.

En este modelo contamos con dos contenedores donde se desplegaron los dos servicios de forma independiente. esto con el fin de modelar o simular dos servicios aparte, cada cual con sus certificados y llaves de seguridad.

Figure 2: Contenedores

En la Estructura del modelo podemos ver que un servicio web en el cual se está manejando la autenticidad mediante un login, y adicionalmente con el protocolo HTTPS Como se observa a continuación

Figure 3: Login con HTTPS

Para validar el certificado se observa como el navegador reconoce los certificados de forma correcta con los detalles especificados en su creación.

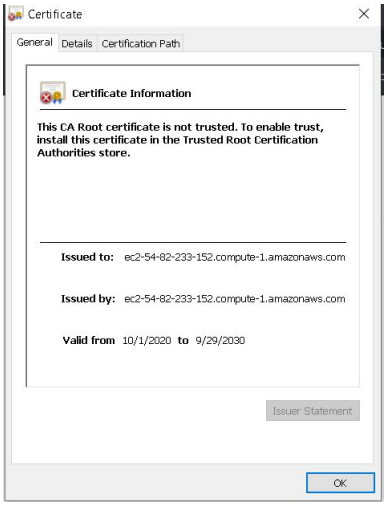

Figure 4: Ejemplo de certificado

3 Modelo

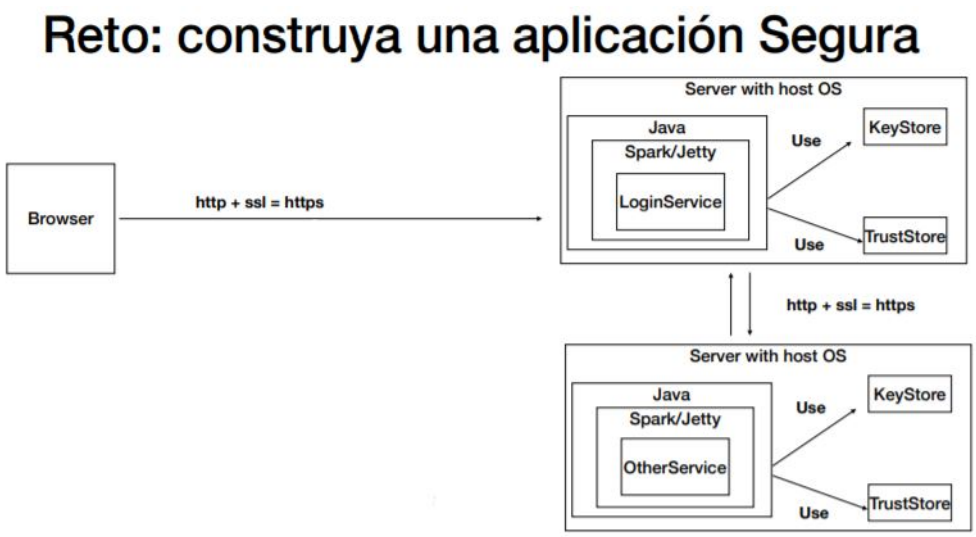

Figure 5: Author : Daniel Benavides

- Main aplicación (p 4500): Este servicio es el que interactúa directamente con el browser, la primera comunicación con el usuario. allí se aloja el login y las peticiones al segundo servicio, el cual es una página sencilla donde podemos ver la respuesta de este microservicio, se hizo mediante una solitud web a la dirección https://ec2-34-226-211-147.compute-1.amazonaws.com:46000/app la cual aloja el resultado.

  El Main App tiene una un objeto el cual interpreta las respuestas, en este caso a nuestro microservicio y las muestra, es decir recibe y guarda en una cadena, para posteriormente mostrar el resultado.

- Service App (p 4600): Este servicio es una aplicación sencilla donde se retorna una página web al acceder al path /aplicación Donde el usuario evidenciara un cambio de servicio. este intercambio se hace por medio de protocolo HTTPS el cual era el objetivo principal y parte de los requerimientos dados. este servicio es el que podemos ver en la parte inferior, el cual interactúa con el MAIN APP

4 Conclusiones

Durante el desarrollo del laboratorio nos enfrentamos a dos tipos de retos, los cuales principalmente consisten en la implementación de seguridad del framework Spark y segundo el manejo y la creación de los certificados. El manejo en cuanto De los certificados SSL. Pero estos retos valen el esfuerzo para un garantizar una seguridad completa de un servicio web, ya que da confiabilidad al usuario o al menos una sensación de seguridad a la hora de usar la aplicación.

El primer reto se realizó finalmente con la etiqueta BEFORE de spark la cual nos permite limitar el acceso a paths determinados, en el caso de la autenticación debemos hacer que el usuario se autentique con sus credenciales, en términos prácticos, la clave del usuario debe estar cifrada para un mayor porcentaje de seguridad. Para este Laboratorio la clave este sin cifrar. Una vez autenticado, se le permite al usuario avanzar al paths que desee, en este caso a la conexión con el servicio que se encuentra en el otro contenedor de EC2 de AWS.

En cuanto a los certificados, debemos ser claros a la hora de crearlos, ya que estos también tienen sus propias claves como los archivos KEYSTORE y TRUSTSTORE, se debe manejar de forma correcta, ya que en caso de que las claves sean erróneas, no se podrá usar el protocolo HTTPS y dejaría de ser una navegación segura en términos generales. para el manejo en JAVA debemos ser claros en las rutas de donde están almacenados y no limitar el acceso a este recurso. para validar si el certificado se emite de forma correcta el navegador nos dará la opción de ver el detalle de estos.

Podemos concluir que hacer una aplicación segura puede tomar su tiempo, pero actualmente es necesario, SPARK Y SPRING son herramientas que nos ayudan a manejar la autenticidad a su manera. Como desarrolladores debemos garantizar al usuario una experiencia de manera Segura. Hoy en día los ciber ataques son más que comunes, por ellos debemos hacer de nuestros servicios , servicios confiables.

References

[1] definicion SSL
https://www.verisign.com/es-LA/website-presence/online/ssl-certificates/index.xhtml 2020.

[2] definicion KeyStore

http://minisconlatex.blogspot.com/2014/02/contenidos-de-este-blog.html
2020.

[3] QUE ESTrustStore
https://aprenderdiaadia.wordpress.com/2011/07/23/keystore-truststore/text=Un0truststore20se20utiliza20parael20cliente20y20el20servidor

2020.

[4] Especificaciones y Modelo
http://campusvirtual.escuelaing.edu.co/moodle/pluginfile.php/230587/modresource/content/0/TallerAllSe
2020.