Taller 3: Certificados Digitales con OpenSSL y Apache 2

Para el desarrollo de este taller debe crear un informe en formato `.doc` o formato `.odt` en el cual debe colocar las respuestas a todas las preguntas **resaltadas en negritas** que encuentre en las siguientes Secciones. Al finalizar el taller envíe su respectivo informe al profesor. El informe debe incluir su nombre y número de cédula al principio.

1. Creación y Uso de Certificados Auto-firmados

1.1. Creación de una autoridad de certificación

El primer paso para poder generar un certificado auto-firmado es la creación de un par de claves para la "autoridad de certificación". Esto se realiza con el comando "`openssl genrsa -des3 -out ca.key 4096`". Con este comando se genera una clave privada de 4096 bits la cual será cifrada con el algoritmo Triple-DES y guardada en el archivo `ca.key`.

Luego se debe generar el certificado de la autoridad. Esto se realiza con el comando "`openssl req -new -x509 -days 7300 -key ca.key -out ca.crt`". Llene la información que se solicita a su gusto. Para verificar el contenido de un certificado se utiliza el comando "`openssl x509 -noout -text -in ca.crt | less`". **Verifique que el certificado es auto-firmado. ¿Que campos del certificado puede utilizar para verficar esta información?**.

1.2. Creación de un certificado de servidor

Para generar un certificado de servidor, lo primero que se tiene que hacer es generar una solicitud de certificación o CSR en inglés. Pero antes es necesario generar un par de claves RSA para el servidor. **Utilizando el mismo comando de la sección anterior, genere una clave para el servidor que llamará "`server.key`". Copie el contenido de la clave generada en su informe**. Luego genere una solicitud de certificación con el comando "`openssl req -new -sha256 -key server.key -out server.csr`". Es de suma importancia que NO deje vacío los campos "`Common Name`" ni "`Challenge Password`" que se solicitan al generar el CSR. **Copie el contenido de la solicitud de certificación en su informe**. Comparta su solicitud de certificación con algún compañero, quien deberá compartirle su respectiva solicitud de certificación.

Luego debemos crear unos archivos que se utilizaran como la "base de datos" de la autoridad de certificación. Primero cree un directorio dentro del directorio de trabajo que deberá llamar "`demoCA`" con el siguiente comando "`mkdir -p demoCA/newcerts`". Luego cree un archivo dentro del directorio recién creado con el comando "`touch demoCA/index.txt`". Finalmente cree un archivo adicional con el comando "`echo 1000 >demoCA/serial`". El archivo "`index.txt`" contendrá un listado de los certificados generados por esta autoridad de certificación. El archivo "`serial`" contendrá el número de serie siguiente a utilizar al firmar el próximo certificado. La ubicación y nombre de esta "base de datos" se puede configurar en el archivo "`/usr/lib/ssl/openssl.cnf`" en sistemas operativos derivados de Debian.

Genere un certificado para su compañero con el siguiente comando "`openssl ca -days 365 -notext -md sha256 -in compañero.csr -out certificado_compañero.crt -keyfile ca.key -cert ca.crt`". **Visualice el certificado generado y verifique que corresponde con los datos suministrados por su compañero en su respectiva solicitud de certificación**. Envíe el certificado generado a su compañero y reciba su correspondiente certificado generado. **Visualice el certificado que le generó su compañero, verifique que es correcto y copie su contenido en su informe**.

1.3. Instalación del certificado de servidor en Apache2

Instale el servidor Apache 2 con el comando "`apt install apache2`". El docente le suministrará la clave del usuario `root` para que pueda realizar esta y las siguientes actividades.

Habilite el módulo de SLL de Apache 2 con el comando "`a2enmod ssl`" como usuario `root`. Luego edite el archivo "`/etc/apache2/sites-available/default-ssl.conf`" para aplicar los siguientes cambios:

1. Busque la linea que contiene el texto "`SSLCertificateFile`" y cambie la ruta allí indicada por la ruta absoluta donde se encuentra su certificado de servidor que le generó su compañero. La linea debe quedar de la siguiente forma: "`SSLCertificateFile /ruta/al/certificado.crt`".
2. Busque la linea que contiene el texto "`SSLCertificateKeyFile` y cambie la ruta allí indicada por la ruta absoluta donde se encuentra su archivo de clave del servidor. La linea debe quedar de la siguiente forma: "`SSLCertificateKeyFile /ruta/al/certificado.key`".

Luego recargue la configuración del servidor utilizando el comando "`systemctl reload apache2`". **Verifique que el servidor se recargó correctamente con el comando "`systemctl status apache2`"**. Conéctese al servidor de su compañero usando la dirección "`https://190.169.74.XXX`", sustituyendo las XXX por el octeto correspondiente de la dirección IP de su compañero. **Verifique que la conección se hace cifrada con el certificado firmado por usted**.

2. Post-Taller

Investigue las siguientes cuestiones:

1. ¿Como se puede hacer para agregar el certificado de su autoridad de certificación al navegador y/o sistema operativo para no recibir la advertencia de certificado no reconocido al conectarse a su servidor?
2. Los certificados X.509 pueden ser almacenados en una gran variedad de archivos contenedores. Investigue las diferencias de los siguientes archivos de certificados (los siguientes nombres se refieren a la extensión del archivo):
   - PEM.
   - DER.
   - PFX.
   - P12.
   - CER.
   - KEY.

Lea el artículo "*Here's Why Your Static Website Needs HTTPS*" de Troy Hunt, disponible en la subsección "enlaces" de la sección de descargas de la página de la materia y haga un resumen de a lo sumo 1 página de su contenido, donde debe contestar la pregunta **¿Por que es recomendable instalar un certificado de servidor para un servicio estático que nunca recibirá entrada de datos de parte de los usuarios?**.