

## Práctica: Servidores de certificados en Apache

Ahora vamos a usar el protocolo HTTPS basado en SSL (security Socket Layer) para establecer un canal seguro en la interacción entre el navegador del cliente y el servidor web. Este protocolo se usa para proteger información confidencial como datos personales en banca electrónica, tiendas online, etc.

En la barra del navegador aparece un candado y el protocolo `https://`. El puerto que usa el método SSL es el 443 para diferenciarlo del que usa por defecto `http`.

El objetivo es que cuando accedamos a nuestro host virtual la información vaya cifrada. Nuestro objetivo es que tecleando `https://www.MiVirtualHost.com` (donde pongo MiVirtualHost debe aparecer el nombre del Host Virtual que creaste) se visualice el fichero `index.html` y el servidor muestre que es un sitio seguro.

Aquí os doy algunas pistas para configurarlo.

Habilitamos el módulo SSL de apache:

```
a2enmod ssl
```

Creamos un directorio para almacenar los certificados:

```
mkdir /etc/apache2/MiVirtualHost/ssl
```

Vamos a la carpeta anterior y creamos los certificados. Primero un certificado clave de SSL de 2048 bits de longitud. Nos pedirá una clave para usarla en el futuro, para generarla escribimos:

```
openssl genrsa -des3 -out key 2048
```

Se crea el CSR (Certificate Signing Request) a partir de la clave privada creada. Se solicita la contraseña creada antes.

```
openssl req -new -key key -out MiVirtualHost.csr
```

Se genera el certificado para nuestra web, nos pedirá de nuevo la clave:



```
openssl x509 -req -sha256 -days 365 -in MiVirtualHost.csr  
-signkeykey -out MiVirtualHost.crt
```

Ahora vamos a configurar el host virtual con los siguientes parámetros:

```
<VirtualHost 10.0.2.15:443>  
ServerAdmin admin@MiVirtualHost.com  
DocumentRoot /var/www/MiVirtualHost  
ServerName www.MiVirtualHost.com  
ServerAlias MiVirtualHost.es  
SSLEngine on  
SSLCertificateFile  
/etc/apache2/MiVirtualHost/ssl/MiVirtualHost.com.crt  
SSLCertificateKeyFile /etc/apache2/MiVirtualHost/ssl/key  
ErrorFile ${APACHE_LOG_DIR} MiVirtualHost-access-ssl.log  
LogFormat "%h %l %u %t\" %r\" %>s" combined  
</VirtualHost 10.0.2.15:443>
```

Ahora debemos reiniciar el servicio Apache:

```
service apache2 restart
```

Comprueba que funciona en el navegador, debe aparecer https:// y en el candado se muestra el certificado.

**Entrega un documento en pdf con capturas de pantalla en el que se pueda comprobar la autoría del trabajo.**