apache, ssl, certificados, https

Implementación de Cetificados SSL en Apache

Generar certificados autofirmados

Por ejemplo para generar unos certificados autofirmados para nextcloud

Si no tenemos el módulo ssl lo instalamos

```
yum install mod_ssl
```

Creamos un directorio específico para almacenar la clave privada

```
sudo mkdir /etc/ssl/private
sudo chmod 700 /etc/ssl/private/
```

```
sudo openssl req -x509 -days 365 -nodes -newkey rsa:2048 -keyout
/etc/ssl/private/nextcloud.key -out /etc/ssl/certs/nextcloud.crt
```

contestamos a las preguntas del asistente y se generaran nuestros certificados privados.

Hay que tener en cuenta que cuando nos pregunte el **Common Name** poner el nombre de dominio asociado a nuestro servidor o la ip en caso de no tener dominio

Para tener una negociación de certificados fuerte **Perfect Forward Secrecy** necesitamos crear un grupo DH (Diffie-Hellman group) y añadirlo a nuestro certificado. Para ello ejecutamos

```
sudo openssl dhparam -out /etc/ssl/certs/dhparam.pem 2048
```

y procedemos a añadirlo a nuestro certificado autofirmado

```
cat /etc/ssl/certs/dhparam.pem | sudo tee -a /etc/ssl/certs/nextcloud.crt
```

Configurar Apache para SSL

Editamos el fichero de configuración SSL de nuestro servidor

Tendrás que editar/crear el fichero de configuración **/etc/httpd/conf.d/ssl.conf** o bien editar el fichero de configuración de tu Virtual Host.

```
##Fichero ejemplo de
https://www.sugeek.co/instalar-certificado-ssl-en-centos-7/
Listen 443
```

```
NameVirtualHost *:443
SSLPassPhraseDialog  builtin
SSLSessionCacheTimeout  300
SSLRandomSeed startup file:/dev/urandom  256
SSLRandomSeed connect builtin
SSLCryptoDevice builtin
SSLStrictSNIVHostCheck off

<VirtualHost *:443>
DocumentRoot /var/www/html/misitio
ServerName www.misitio.com
ServerAlias misitio.com
SSLEngine on
SSLProtocol all -SSLv2 -SSLv3
SSLCipherSuite ALL:!ADH:!EXPORT:!SSLv2:RC4+RSA:+HIGH:+MEDIUM:+LOW
SSLCertificateFile /etc/letsencrypt/live/www.misitio.com/cert.pem
SSLCertificateKeyFile /etc/letsencrypt/live/www.misitio.com/privkey.pem
ErrorLog logs/ssl_error_log
TransferLog logs/ssl_access_log
LogLevel warn
<Files ~ “\.(cgi|shtml|phtml|php3?)$”>
SSLOptions +StdEnvVars
</Files>
SetEnvIf User-Agent “.*MSIE.*” \
nokeepalive ssl-unclean-shutdown \
downgrade-1.0 force-response-1.0
CustomLog logs/ssl_request_log \
“%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \”%r\” %b”
</VirtualHost>
```

El comando “apache2ctl -t” te puede ayudar para detectar posibles errores de sintaxis en la configuración del archivo del virtualhost

Reiniciamos Apache para que se apliquen los cambios.

```
systemctl restart httpd
```

Redireccionar el tráfico http a https

Podrías redirigir todo el tráfico de http (normal) a https (seguro) mediante una configuración en el virtual host para el puerto 80. Para ello creamos un fichero de configuración

```
sudo vi /etc/httpd/conf.d/non-ssl.conf
```

Añadimos las siguientes línas al fichero que hemos creado para redireccionar el tráfico

```
<VirtualHost *:80>
```

```
        ServerName www.example.com
        Redirect "/" "https://www.example.com/"
</VirtualHost>
```

Referencias

- https://www.digitalocean.com/community/tutorials/how-to-create-an-ssl-certificate-on-apache-for-centos-7
- http://terminus.ignaciocano.com/k/2011/06/14/configurar-apache-para-servir-conexiones-seguras/
- http://terminus.ignaciocano.com/k/2012/05/10/forzar-el-uso-de-sslhttps-de-un-directorio-en-apache2-mediante-htaccess-y-mod_rewrite/