

# Configuración de un certificado SSL en la Release1

## Introducción

El protocolo SSL (Secure Socket Layer) está destinado a encriptar de manera segura los datos intercambiados entre dos máquinas. En su servidor dedicado, puede ser útil utilizar SSL con el fin de proteger datos sensibles.

Encontrará más información sobre el funcionamiento de SSL en esta página.

## Procedimiento

**Atención** : Este procedimiento está basado en la Release 1 de OVH.

Esta guía le ayudará a crear su propio certificado SSL con sus datos de manera gratuita y posteriormente instalar este nuevo certificado en los servicios que desee de su servidor.

Igualmente si ha comprado un certificado de pago y dispone de una pareja de 2 ficheros (KEY y CRT) válida, podrá utilizar esta guía para instalar los dos ficheros de su certificado en los servicios deseados.

Nota : Un certificado SSL autogenerado y gratuito, puede provocar una alerta de autenticación. Esto no reduce en nada la seguridad aportada por este certificado en la transmisión de datos.

## Paso 1. Activar la opción SSL en Apache

La activación de SSL en Apache se hace desde SSH. Para ello deberá conectarse a través de SSH a su servidor (usuario root).

## Verificación

El servidor simple (HTTP) funciona sobre el puerto 80. El puerto en modo SSL (HTTPS) funciona sobre el puerto 443. Verificamos primero si este puerto ya está escuchándose por Apache :

```
[root@crashtest root]# netstat -tanpu | grep ":443"  
[root@crashtest root]#
```

Si no hay resultado (como es el caso), el servidor aún no ha sido activado para el SSL. Hace falta descomentar la opción de SSL.

En caso de que exista resultado, podemos pasar al segundo paso.

## Activación

Introducimos el comando en SSH :

```
# pico /etc/sysconfig/apache
```

Para descomentar, retiramos el caracter # delante de la opción en cuestión :

```
# Uncomment to active SSL  
OPTIONS="-DSSL"
```

Guardamos el fichero con ctrl+x y con 'Y' o bien 'S', luego pulsamos enter.

A continuación reiniciamos Apache para aplicar los cambios.

```
# /etc/init.d/httpd restart  
Parada de httpd : [ OK ]  
Inicio de httpd : [ OK ]
```

Ahora si verificamos el puerto 443, aparecerá Apache escuchando en este puerto :

```
[root@crashtest root]# netstat -tanpu | grep ":443"  
tcp 0 0 0.0.0.0:443 0.0.0.0:* LISTEN 3291/httpd
```

## Seguridad

**Atención** : Este apartado afecta a las Releases anteriores a 1.14.

Existe un fallo de seguridad muy importante en las versiones de OpenSSL inferiores a 0.9.6k. Hace falta obligatoriamente aplicar los parches de seguridad de OVH antes de utilizar SSL sobre las máquinas con versiones de la Release antiguas.

Para verificar la versión de su SSH, introduzca :

```
[root@crashtest root]# rpm -qa | grep ssl
```

Debería obtener una salida del tipo :

```
openssl-devel-0.9.6k-1  
openssl-0.9.6k-1  
openssl-perl-0.9.6k-1
```

o superior.

Si tiene una versión inferior a la que aparece arriba, (por ejemplo openssl-0.9.6i), debe actualizarse y aplicar los parches de seguridad antes de poder utilizar SSL.

Si es igual o superior, puede pasar al paso siguiente.

Para aplicar los parches de seguridad, siga la guía : [ReleasePatchSeguridad](#).

## Paso 2. Instalación del certificado

Ahora que el servidor tiene el soporte SSL activado y accesible, nos hace falta un certificado SSL.

Puede usar un certificado autofirmado gratuito para ello, aunque algunos navegadores dan un mensaje de aviso en este tipo de certificados. Los certificados de pago no dan este aviso, pero eso no significa que la seguridad de las transmisiones de datos sea menor.

Si necesita un certificado de pago y tiene el nombre de dominio registrado con OVH, puede comprar un certificado de OVH. Si tiene el nombre de dominio registrado en otro registrador, puede contactar con una empresa de certificados o bien contratar un certificado en su registrador actual.

En esta guía le explicamos cómo crear un certificado gratuito autofirmado.

Para contratar un certificado de pago con OVH, consulte la guía : [PedidoSsl](#)

### **Creación de un certificado autofirmado**

Diríjase a la carpeta donde se almacenan las keys de SSL en Apache:

```
[root@crashtest root]# cd /usr/local/apache/conf/ssl.key
```

A continuación cree una key con el comando siguiente :

```
[root@crashtest ssl.key]# openssl genrsa 1024 > mi_dominio.com.key  
[root@crashtest ssl.key]# chmod -c 400 mi_dominio.com.key
```

## OVH

Reemplazando mi\_dominio.com por el dominio sobre el que desea instalar el certificado

Ahora vamos a rellenar la solicitud de certificado y responder a las preguntas :

```
[root@crashtest ssl.key]# openssl req -new -key mi_dominio.com.key > mi_dominio.com.csr
Using configuration from /usr/share/ssl/openssl.cnf
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
Country Name (2 letter code) [AU]: ES
State or Province Name (full name) [Some-State]: Madrid
Locality Name (eg, city) []:Madrid
Organization Name (eg, company) [Internet Widgits Pty Ltd] : OVH HISPANO
Organizational Unit Name (eg, section) []: Soporte
Common Name (eg, YOUR name) []: www.mi_dominio.com
Email Address []: mi_correo@mi_dominio.com__
```

Please enter the following 'extra' attributes  
to be sent with your certificate request  
A challenge password []: **Certificado**  
An optional company name []: **OVH**

Luego se autofirma el certificado :

```
[root@crashtest ssl.key]# openssl x509 -req -days 365 -in
mi_dominio.com.csr -signkey mi_dominio.com.key -out mi_dominio.com.crt
Signature ok
subject=/C=ES/ST=MADRID/L=MADRID/O=OVH
HISPANO/OU=Soporte/CN=www.mi_dominio.com/Email=mi_correo@mi_dominio.com
Getting Private key
[root@crashtest ssl.key]#
```

Luego basta con desplazar los ficheros a los directorios correctos :

```
[root@crashtest root]# mv
/usr/local/apache/conf/ssl.key/mi_dominio.com.crt
/usr/local/apache/conf/ssl.crt/
```

```
[root@crashtest root]# mv
/usr/local/apache/conf/ssl.key/mi_dominio.com.csr
/usr/local/apache/conf/ssl.csr/
```

y nuestro certificado está ya listo.

### Paso 3. Configuración de Apache

Modificamos en SSH nuestro fichero httpd.conf :

```
[root@crashtest root]# pico /httpd.conf
```

y buscamos esta primera sección para verificar que el soporte SSL está activo :

```
##
## SSL Support
##
## When we also provide SSL we have to listen to the
## standard HTTP port (see above) and to the HTTPS port
##
<IfDefine SSL>
Listen 80
Listen 443
</IfDefine>
```

En la segunda sección, a la altura de "NameVirtualHost" añadimos la IP con el puerto **443**:

```
#NameVirtualHost 12.34.56.78:80
#NameVirtualHost 12.34.56.78
NameVirtualHost 213.186.37.141:80
NameVirtualHost 213.186.37.141:443
```

Luego buscamos la tercera sección y modificamos el Virtual Host por defecto para añadir nuestro nuevo certificado

**Atención :** en la parte siguiente hemos cortado las líneas en comentario para que el texto no sea demasiado largo. Por favor, no borre estas líneas en su fichero, sólo modifique las que aparecen a continuación.

```

<IfDefine SSL>
<VirtualHost _default_:443>
#General setup for the virtual host
DocumentRoot "/home/mi_login/www"
ServerName crashtest.mi_dominio.com
ServerAdmin mi_correo@mi_dominio.com
ErrorLog logs/error_ssl_log
TransferLog logs/access_ssl_log
#SSL Engine Switch:
#Enable/Disable SSL for this virtual host.
SSLEngine on
#Server Certificate:
SSLCertificateFile /usr/local/apache/conf/ssl.crt/mi_dominio.com.crt
#Server Private Key:
SSLCertificateKeyFile /usr/local/apache/conf/ssl.key/mi_dominio.com.key

```

Ya tenemos indicado el emplazamiento del certificado. Ahora, hace falta añadir el Virtual Host para el sitio.

El virtual host se debe situar entre <If Define SSL> y </If Define>.

```

<VirtualHost 213.186.37.141:443>
DocumentRoot "/home/mi_login/www"
ServerName mi_dominio.com
ServerAdmin mi_correo@mi_dominio.com
ErrorLog logs/error_ssl_log
TransferLog logs/access_ssl_log
SSLEngine on
SSLCipherSuite ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP:+eNULL
SSLCertificateFile /usr/local/apache/conf/ssl.crt/mi_dominio.com.crt
SSLCertificateKeyFile /usr/local/apache/conf/ssl.key/mi_dominio.com.key
<Files ~ "(cgi|shtml|phtml|php3?)$" >
  SSLOptions +StdEnvVars
</Files>
<Directory "/usr/local/apache/cgi-bin">
  SSLOptions +StdEnvVars
</Directory>
</VirtualHost>
</IfDefine>

```

A continuación, guardamos la configuración y reiniciamos Apache :

```

[root@crashtest www]# /etc/init.d/httpd restart
Arrêt de httpd : [ OK ]

```

```
Démarrage de httpd : [ OK ]
[root@crashtest www]#
```

#### Paso 4. Añadir un segundo dominio con certificado SSL

Puede añadir un dominio adicional al certificado, añadiendo simplemente un Virtual Host más.

Añadimos para ello otra sección Virtual Host después del que acabamos de crear, la sección `</VirtualHost>` y antes del `</IFDefine>` :

```
<VirtualHost 213.186.37.141:443>
DocumentRoot "/home/mi_login2/www"
ServerName mi_dominio2.org
ServerAdmin mi_correo@mi_dominio.com
ErrorLog logs/error_ssl_log
TransferLog logs/access_ssl_log
SSLEngine on
SSLCipherSuite ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP:+eNULL
SSLCertificateFile /usr/local/apache/conf/ssl.crt/mi_dominio.com.crt
SSLCertificateKeyFile /usr/local/apache/conf/ssl.key/mi_dominio.com.key
<Files ~ "(.cgi|shtml|phtml|php3?)$" >
  SSLOptions +StdEnvVars
</Files>
<Directory "/usr/local/apache/cgi-bin">
  SSLOptions +StdEnvVars
</Directory>
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>
```

Al utilizar el mismo certificado para un dominio distinto, es posible que dé un mensaje de advertencia por el CN (Common Name) del certificado. Puede jugar con los Wildcard para evitar el error.

#### Conclusión

Si ha efectuado bien las manipulaciones, puede a partir de ahora acceder a sus sitios web a través de HTTPS. No olvide que una alerta se mostrará si utiliza un certificado que se ha generado usted mismo, y que para todos los dominios del servidor, sólo se utilizará un único certificado.

Como en el ejemplo mostrado, todos los dominios añadidos en la sección "VirtualHost" responderán con el certificado generado al principio de la guía. Si necesita poner en marcha varios sitios web bajo HTTPS con certificados distintos, cuenta con la posibilidad de utilizar varias IPs, una por cada dominio.

Más información

: ServidorCertificados :: Configuración de un certificado SSL en la Release2

: PleskCertificadosSSL :: Configuración de un certificado SSL en Plesk

: SslRelease2 :: Configuración de un certificado SSL de OVH en la Release2

: SslPlesk :: Configuración de un certificado SSL de OVH en Plesk