

# Certificados Digitales / TLS y SSL

## Descripción

Debido a las carencias de seguridad del protocolo HTTP se implementa una capa adicional para el transporte de información que resuelve los aspectos de seguridad carentes en los servidores web.

Esta capa viene definida por el estándar TLS o Transport Layer Protocol que gracias a las librerías OpenSSL permite encriptar la información a través de conexiones seguras SSL o Secure Sockets Layer.

SSL es un estándar definido por Netscape Communication Corporation y que actualmente implementan todos los servidores seguros. El estándar SSL no solamente proporciona la encriptación y confidencialidad de los datos sino que también proporciona la autenticación de cliente y servidor y garantiza la integridad de los datos a través del protocolo TCP/IP. Estas son las características deseables para que podamos considerar una conexión segura.

*Aunque SSL proporciona autenticación y privacidad de la información entre extremos sobre Internet mediante el uso de criptografía con normalidad sólo el servidor es autenticado mientras que el cliente se mantiene sin autenticar. -fuera??*

Como la mayoría de servidores (entre ellos apache) ya implementan estos estándares a través de las herramientas de OpenSSL sólo necesitamos de un Certificado válido para poder establecer conexiones tipo HTTPS.

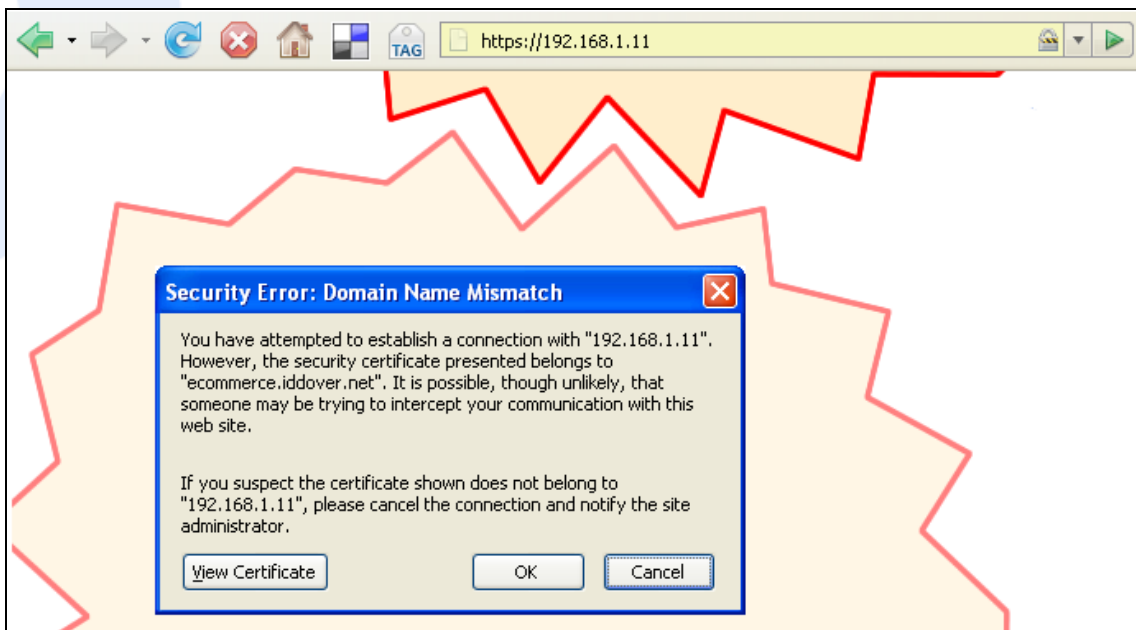
El Certificado Digital es un documento digital mediante el cual un tercero confiable EC (Entidad Certificadora) garantiza la vinculación entre la identidad de un sujeto o entidad y su clave pública.

Instalando el Certificado Digital el servidor apache proporcionará automáticamente la información sobre la web al navegador de cualquier cliente si este está verificado y confirmado por una EC (Entidad Certificadora) o tercero confiable ésta confirmará que el certificado es válido y garantizará la veracidad de los datos. Además toda comunicación se realizará a través del

canal seguro que nos proporcione SSL garantizando la confidencialidad, autenticidad y privacidad de los datos.

## Actuaciones

Existen tres condiciones para que el certificado sea válido y por lo tanto aceptado por la mayoría de navegadores. De no cumplirse estas tres condiciones el navegador alertará de la invalidez del certificado y en muchos casos recomendará al usuario abandonar la conexión.



- El nombre común CN (Common Name) debe coincidir con la dirección URL que el usuario teclea en su navegador
- El certificado debe estar firmado por una EC (Entidad Certificadora) en la que el navegador o navegador del cliente haya depositado su confianza
- Las fechas de validez del certificado deben corresponderse con las actuales

A continuación se detallan los pasos a seguir para la instalación de un certificado SSL confiable.

## Previo

Antes de iniciar el proceso para adquirir e instalar nuestro certificado digital, debemos comprobar si el *Módulo SSL* de Apache está instalado y si no es así realizar la instalación a través de yum:

```
rpm -q mod_ssl || yum install mod_ssl
```

## Organización en disco.

La Distribución de Redhat FC6 dispone de directorios específicos para colocar estos archivos tan sensibles éstos tienen asignados los permisos adecuados para garantizar la seguridad y el acceso solamente a usuarios verificados.

Toda la estructura de certificados será generada sobre `/etc/pki/tls/`. Sobre éste se despliegan los directorios siguientes

certs	Certificados
misc	CA Bundle y Otros
private	Clave Privada y CSR
pub	Certificados para el servidor web

## Generar el CSR

El CSR (Certificate Signing Request) es el archivo que se debe suministrar a la EC (Entidad Certificadora) para que ésta pueda expedir el certificado. Este contiene toda la información con la información relativa a la empresa, además de la clave publica de la PKI.

Para generar un par de claves (clave publica y privada) y el CSR para un servidor web a través de la aplicación openssl se usará el comando genérico siguiente:

```
openssl req -new -nodes -keyout nombredelaclave.key -out  
nombredelcsr.csr
```

Para generar el CSR y las claves para `ecommerce.iddover.net` sería:

```
openssl req -new -nodes -keyout private/ecommerce.iddover.net.key  
-out private/ecommerce.iddover.net.csr
```

Este comando generará dos archivos. El archivo `private/ecommerce.iddover.net.key` contiene la clave privada que debe ser guardada en secreto.

**Nota:** Es importante garantizar la persistencia de la clave privada, en el caso

de extraviarse el certificado resultaría inválido y deberíamos solicitar otro.

A Continuación se deben indicar los detalles referentes a la Empresa o Entidad que solicita el certificado. La información introducida en este punto aparecerá en el apartado *Subject* del certificado final emitido.

```
Country Name (2 letter code) [AU]: ES
State or Province Name (full name) [Some-State]: Barcelona
Locality Name (eg, city) []: Vilafanca del Penedes
Organization Name (eg, company) [Internet Widgits Pty Ltd]: FAVSHARE
Organizational Unit Name (eg, section) []: IT
Common Name (eg, YOUR name) []: ecommerce.iddover.net
Email Address []: ssl@iddover.net
```

```
A challenge password []:
An optional company name []:
```

En el campo CN (Common Name) se indicará el nombre del servidor y dominio (*fully qualified domain name*) al que se emitirá el certificado y estará alojada la web. Es importante tener en cuenta que si indicamos en este campo el valor: *iddover.net*, el certificado no será válido para el subdominio [www.iddover.net](http://www.iddover.net) o cualquier otro, por lo que debe ser seleccionado con sumo cuidado.

Los campos Email Address, challenge password y optional company name pueden dejarse en blanco.

Si indicamos un valor para el campo *challenge password* cada vez que se tenga que acceder a los valores del certificado (como por ejemplo en un reinicio del Apache) se deberá introducir la contraseña de forma manual, por lo que se recomienda dejar en blanco.

**Nota:** Para verificar los datos de un CSR ya generado se puede usar el comando: `openssl req -text -noout -verify -in nombredelcsr.csr` que presentará por pantalla toda la información que el CSR contiene

### **Enviar el CSR a la Entidad Certificadora**

Una vez se haya generado el archivo que contiene el CSR se debe enviar el contenido de tal a la Entidad Certificadora (normalmente a través de un formulario)

## Instalación del Certificado en Apache/ModSSL

La Entidad Certificadora tiene la obligación de verificar que la información del CSR es cierta y posteriormente emitirá el certificado definitivo que se enviará normalmente a través de correo electrónico.

El certificado definitivo (visto mediante un editor de texto) debe tener un aspecto similar a el siguiente:

```
-----BEGIN CERTIFICATE-----
MIIEEdjCCA16gAwIBAgIRALMw0xXFLYmFkuXTPOr0W9MwDQYJKoZIhvcNAQEF
BQAw
WDELMAkGA1UEBhMCVVMxHjAcBgNVBAoTFVJlZ2lzdGVyRmx5LmNvbSwgaW5
jLjEp
MCCGA1UEAxMgUmVzZWxsZXJGbhkgQ2VydG1maWNhdGUgU2VydmljZXMwHhc
NMDYx
( . . . )
mD6qAxL92/qjJERI0bTRn2BJYNSRGGND1aAf8kh7LuJiONnooIHfdlrv
-----END CERTIFICATE-----
```

Este archivo debe ser ubicado con un nombre descriptivo en el directorio correspondiente dentro del servidor web : /etc/pki/tls/certs/

## Instalación del Certificado Intermedio

Algunas entidades certificadoras, para evitar estar bombardeadas con peticiones de verificación para sus certificados prefieren distribuir el *Certificado Intermedio* que no es más que el certificado de ésta, para que cada servidor que haya sido certificado disponga de la información necesaria para el cliente que establezca la conexión SSL.

Situamos este archivo dentro de la estructura de directorios, concretamente en el directorio del servidor siguiente: /etc/pki/tls/misc/

## Configuraciones

Una vez estén generados y almacenados correctamente los certificados, se añadirán las directrices necesarias para que el servicio web Apache pueda manejar conexiones cifradas. Para ello generamos un archivo específico en /etc/httpd/conf.d/ssl.conf

Indicamos al servidor Apache que no se limite al puerto 80 (http) sino que también escuche el puerto correspondiente a las conexiones web cifradas (https) que se corresponde con el número 443.

```
Listen 443
```

Normalmente, configuraremos un *servidor virtual* en el que activamos el soporte ssl mediante la siguiente línea:

```
SSLEngine on
```

Indicamos los *protocolos cifrados* permitidos para la comunicación entre cliente y servidor. Desactivamos SSLv2 por débil haber sido explotado recientemente.

```
SSLProtocol all -SSLv2
```

Indicamos el *tipo de cifrado* permitido, se permite el *low*, de 128 Bits por mantener la compatibilidad con Internet Explorer 6

```
SSLCipherSuite ALL:!ADH:!EXPORT:!SSLv2:RC4+RSA:+HIGH:+MEDIUM:+LOW
```

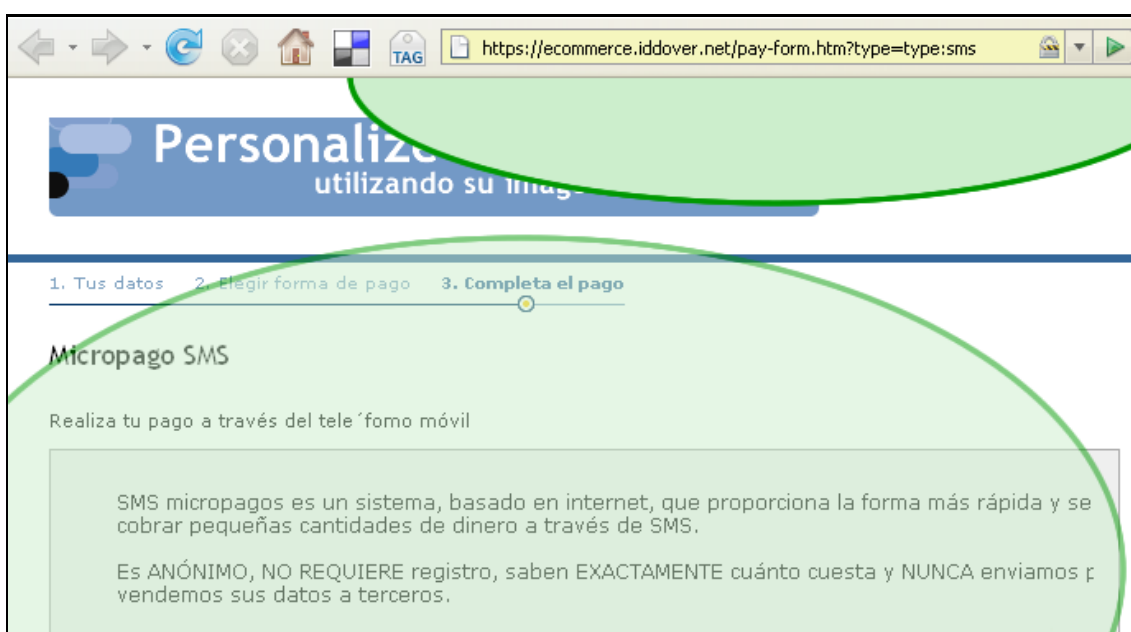
Se indican las rutas del certificado y la clave privada

```
SSLCertificateFile /etc/pki/tls/certs/ecommerce.iddover.net.crt  
SSLCertificateKeyFile /etc/pki/tls/private/ecommerce.iddover.net.key
```

Añadimos la ruta del Certificado Intermedio.

```
SSLCACertificateFile /etc/pki/tls/misc/ecommerce.iddover.net.ca-bundle
```

Si hemos instalado el certificado satisfactoriamente, la próxima vez que accedemos al servidor de forma segura, aparecerá el candadito cerrado al lado de la Dirección Web, sin haber aparecido ninguna alerta de seguridad.





## Referencias

Para ampliar información consulta las siguientes referencias:

[http://httpd.apache.org/docs/2.2/mod/mod\\_ssl.html](http://httpd.apache.org/docs/2.2/mod/mod_ssl.html)

[http://shib.kuleuven.be/docs/ssl\\_commands.shtml](http://shib.kuleuven.be/docs/ssl_commands.shtml)

<http://es.wikipedia.org/wiki/Ssl>

[http://es.wikipedia.org/wiki/Certificado\\_digital](http://es.wikipedia.org/wiki/Certificado_digital)

<http://www.ietf.org/rfc/rfc2246.txt>

## ¿Que es favshare.com?

Favshare es un espacio para almacenar compartir fotografías, cuyo acceso lo puedes realizar a través de FTP y en el que te damos 20 GB de disco gratis para que uses a tu antojo.