

TEMA 2: Administración de credenciales de acceso a sistemas

Práctica 2.6: Uso de certificados en servicios web

Realizaremos una instalación de un servidor apache en nuestro sistema Ubuntu y configuraremos dos dominios con acceso https mediante la creación de sus certificados con la CA instalada anteriormente.

Primero procedemos a crear los dos certificados para los dominios siguientes:

dominio1.com
dominio2.com

Revisar en la práctica anterior para realizar estas operaciones el punto “Firmar certificados de servidor y cliente”.

```
openssl genrsa -out intermedia/private/dominio1.key.pem 2048
```

```
openssl genrsa -out intermedia/private/dominio2.key.pem 2048
```

```
openssl req -config intermedia/openssl.cnf -key intermedia/private/dominio1.key.pem -new -sha256 -out intermedia/csr/dominio1.csr.pem
```

```
openssl req -config intermedia/openssl.cnf -key intermedia/private/dominio2.key.pem -new -sha256 -out intermedia/csr/dominio2.csr.pem
```

```
openssl ca -config intermedia/openssl.cnf -extensions server_cert -days 365 -notext -md sha256 -in intermedia/csr/dominio1.csr.pem -out intermedia/certs/dominio1.cert.pem
```

```
openssl ca -config intermedia/openssl.cnf -extensions server_cert -days 365 -notext -md sha256 -in intermedia/csr/dominio2.csr.pem -out intermedia/certs/dominio2.cert.pem
```

Cuando hemos obtenido los certificados, copiamos las claves privadas en /etc/ssl/private y los certificados en /etc/ssl/certs.

Realizamos también la copia de la cadena de certificados intermedios en /etc/ssl/certs.

```
root@ubuntu:~/ca/intermedia/certs# ls
ca-cadena.cert.pem dominio1.cert.pem dominio2.cert.pem intermedia.cert.pem midominio.cert.pem
root@ubuntu:~/ca/intermedia/certs# cp dominio* /etc/ssl/certs/
root@ubuntu:~/ca/intermedia/certs#
```

```
root@ubuntu:~/ca/intermedia/private# ls
dominio1.key.pem dominio2.key.pem intermedia.key.pem midominio.key.pem
root@ubuntu:~/ca/intermedia/private# cp dominio* /etc/ssl/private/
root@ubuntu:~/ca/intermedia/private#
```

```
root@ubuntu:~/ca/intermedia/certs# ls
ca-cadena.cert.pem dominio1.cert.pem dominio2.cert.pem intermedia.cert.pem midominio.cert.pem
root@ubuntu:~/ca/intermedia/certs# cp ca-cadena.cert.pem /etc/ssl/certs/
root@ubuntu:~/ca/intermedia/certs#
```

Ahora instalamos apache en el servidor

```
root@ubuntu:/root/# apt-get install apache2
```

Y configuramos la visibilidad de los dos dominios con sus certificados mediante el uso de virtual hosts de apache creando el archivo /etc/apache2/sites-available/dominios-ssl.conf

```

GNU nano 4.8
<IfModule mod_ssl.c>
    <VirtualHost _default_:443>
        ServerAdmin webmaster@localhost
        ServerName dominiol.com
        ServerAlias www.dominiol.com
        DocumentRoot /var/www/html/dominiol

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        # SSL Engine Switch:
        # Enable/Disable SSL for this virtual host.
        SSLEngine on

        SSLCertificateFile      /etc/ssl/certs/dominiol.cert.pem
        SSLCertificateKeyFile   /etc/ssl/private/dominiol.key.pem
        SSLCertificateChainFile /etc/ssl/certs/ca-cadena.cert.pem

        <FilesMatch "\.(cgi|shtml|phtml|php)$">
            SSLOptions +StdEnvVars
        </FilesMatch>
        <Directory /usr/lib/cgi-bin>
            SSLOptions +StdEnvVars
        </Directory>

    </VirtualHost>
    <VirtualHost _default_:443>
        ServerAdmin webmaster@localhost
        ServerName dominio2.com
        ServerAlias www.dominio2.com
        DocumentRoot /var/www/html/dominio2

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        # SSL Engine Switch:
        # Enable/Disable SSL for this virtual host.
        SSLEngine on

        SSLCertificateFile      /etc/ssl/certs/dominio2.cert.pem
        SSLCertificateKeyFile   /etc/ssl/private/dominio2.key.pem
        SSLCertificateChainFile /etc/ssl/certs/ca-cadena.cert.pem

        <FilesMatch "\.(cgi|shtml|phtml|php)$">
            SSLOptions +StdEnvVars
        </FilesMatch>
        <Directory /usr/lib/cgi-bin>
            SSLOptions +StdEnvVars
        </Directory>

    </VirtualHost>
</IfModule>

```

Activamos SSL en apache
 root@ubuntu:/root/# a2enmod ssl
 Realizamos una redirección de http a https

```

<VirtualHost *:80>
    ServerAdmin webmaster@localhost
    ServerName dominio1.com
    ServerAlias www.dominio1.com
    Redirect Permanent / https://dominio1.com
    DocumentRoot /var/www/html/dominio1

    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

</VirtualHost>

<VirtualHost *:80>
    ServerAdmin webmaster@localhost
    ServerName dominio2.com
    ServerAlias www.dominio2.com
    Redirect Permanent / https://dominio2.com
    DocumentRoot /var/www/html/dominio2

    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

```

Creamos los directorios dominio1 y dominio2 en /var/www/html

Copiamos el archivo index.html en cada uno de los directorios para poder tener al menos una página para mostrar.

Asignamos el propietario www-data a los directorios

```

root@ubuntu:/var/www/html# ls
index.html
root@ubuntu:/var/www/html# mkdir dominio1
root@ubuntu:/var/www/html# mkdir dominio2
root@ubuntu:/var/www/html# chown www-data:www-data dominio*
root@ubuntu:/var/www/html# ls -lh
total 20K
drwxr-xr-x 2 www-data www-data 4.0K Jan 17 11:15 dominio1
drwxr-xr-x 2 www-data www-data 4.0K Jan 17 11:15 dominio2
-rw-r--r-- 1 root      root      11K Jan 17 10:46 index.html
root@ubuntu:/var/www/html# cp index.html dominio1/
root@ubuntu:/var/www/html# cp index.html dominio2

```

Activamos nuestro fichero de configuración dominios-ssl.conf en apache

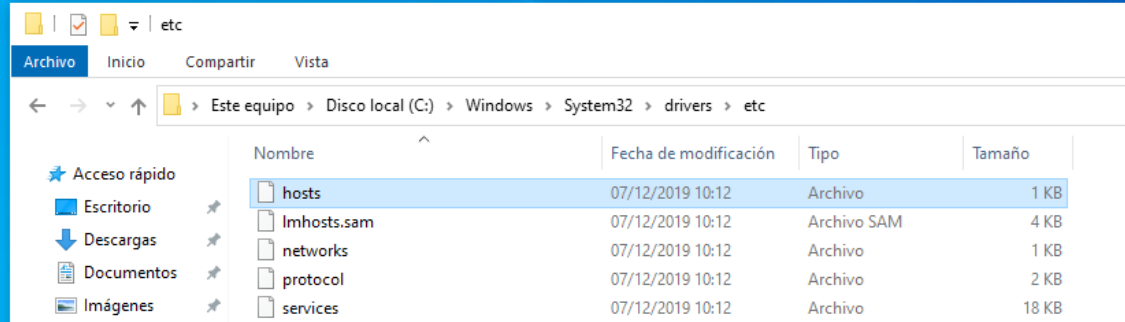
a2ensite dominios-ssl.conf

Probamos la configuración de apache antes de reiniciarlo

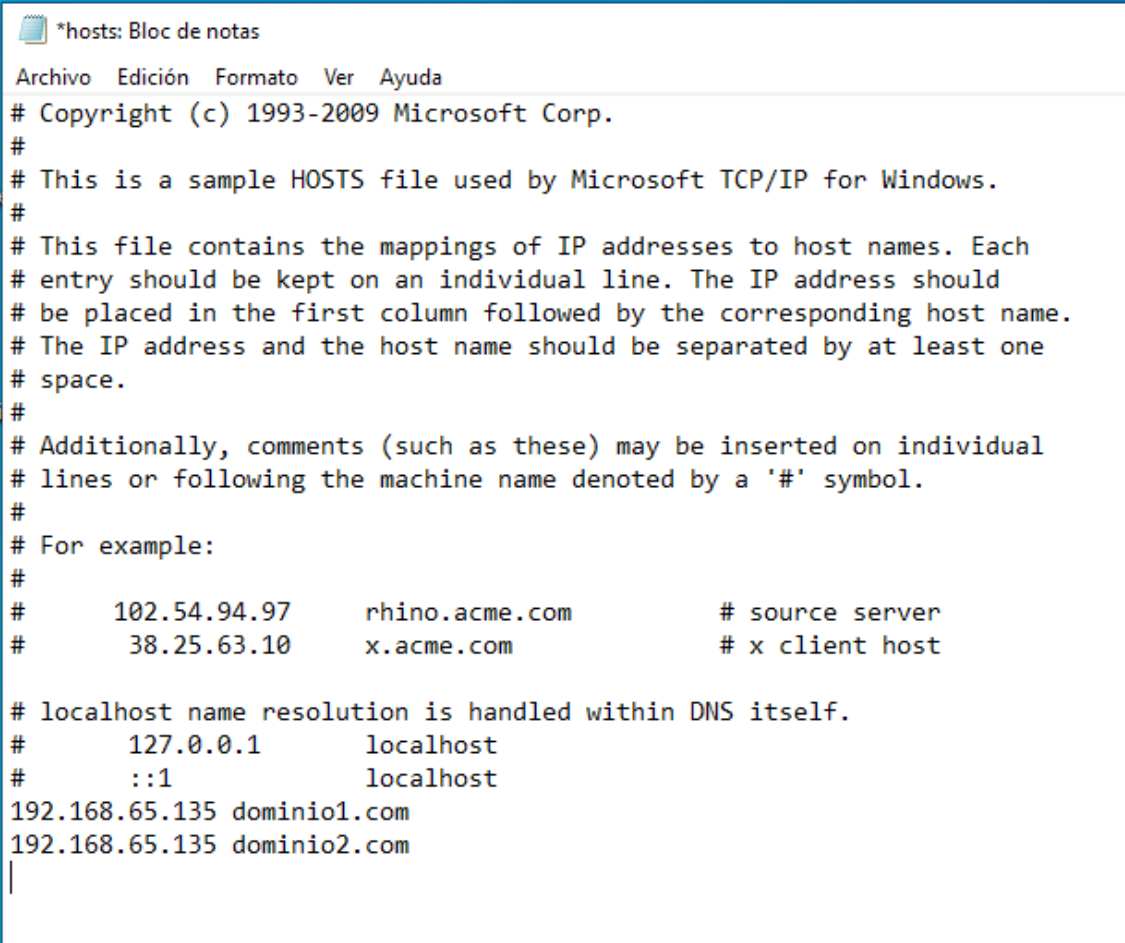
```
#apache2ctl configtest  
Y reiniciamos el servicio  
# service apache2 restart
```

Solamente quedaría probar los certificados, para ello nos vamos a nuestra máquina con Windows 10.

Primero introducimos en nuestro archivo host los nombres de los dominios con sus direcciones ip para que resuelva correctamente.



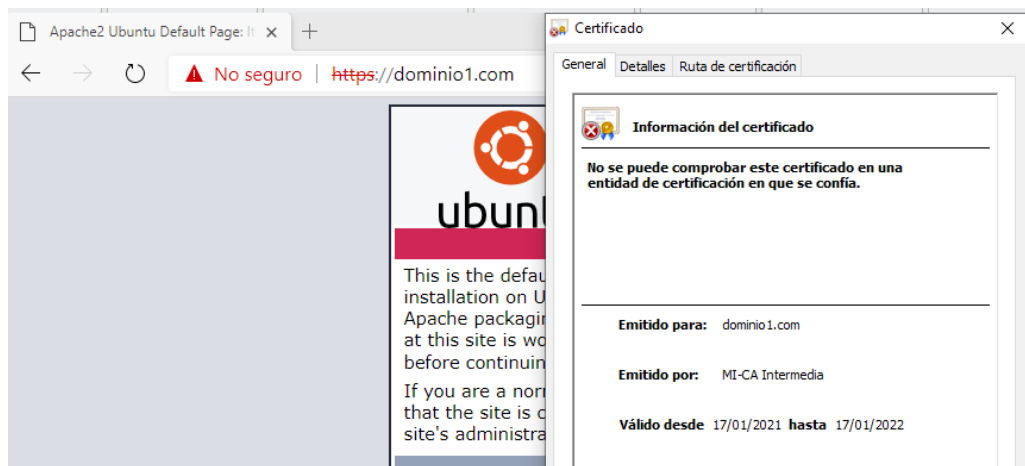
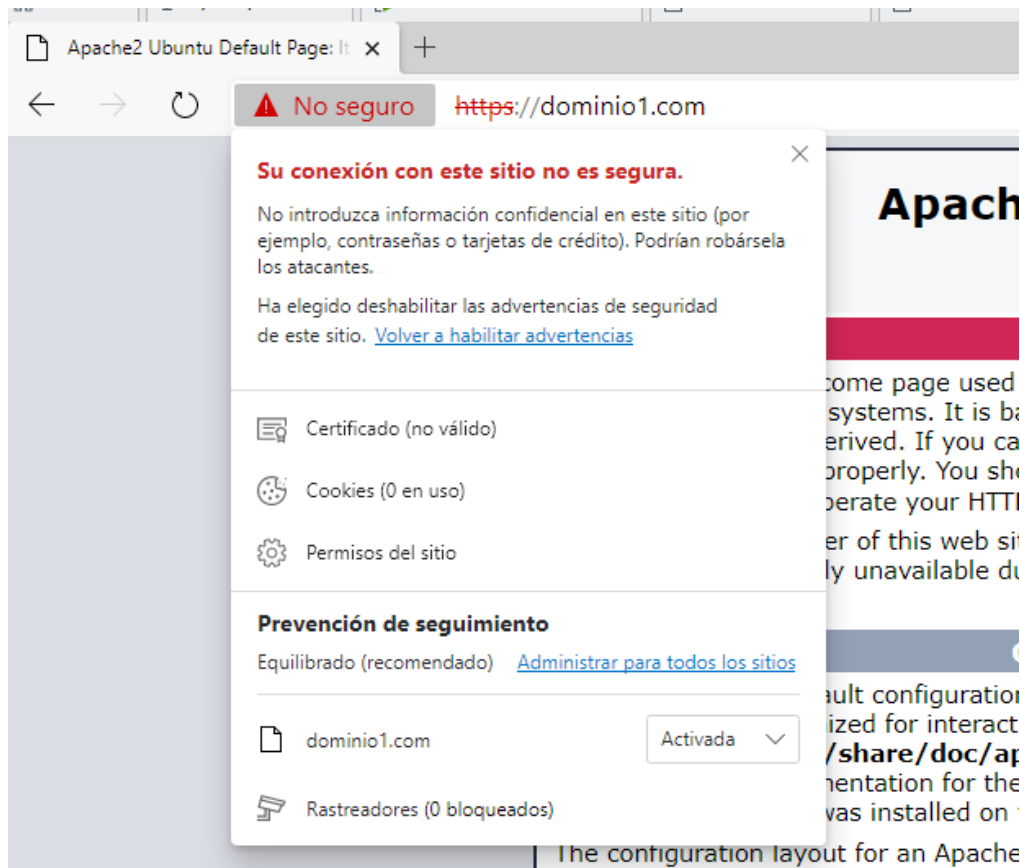
Nombre	Fecha de modificación	Tipo	Tamaño
hosts	07/12/2019 10:12	Archivo	1 KB
lmhosts.sam	07/12/2019 10:12	Archivo SAM	4 KB
networks	07/12/2019 10:12	Archivo	1 KB
protocol	07/12/2019 10:12	Archivo	2 KB
services	07/12/2019 10:12	Archivo	18 KB

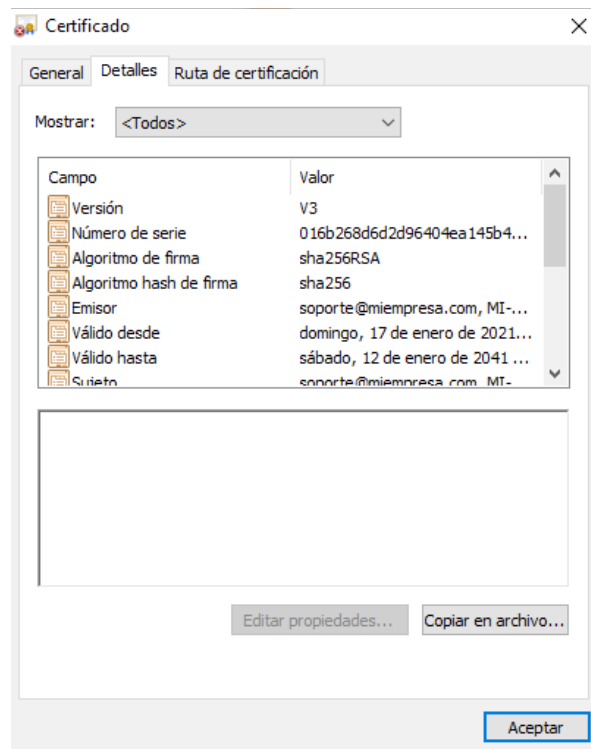


```
*hosts: Bloc de notas  
Archivo Edición Formato Ver Ayuda  
# Copyright (c) 1993-2009 Microsoft Corp.  
#  
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.  
#  
# This file contains the mappings of IP addresses to host names. Each  
# entry should be kept on an individual line. The IP address should  
# be placed in the first column followed by the corresponding host name.  
# The IP address and the host name should be separated by at least one  
# space.  
#  
# Additionally, comments (such as these) may be inserted on individual  
# lines or following the machine name denoted by a '#' symbol.  
#  
# For example:  
#  
#      102.54.94.97      rhino.acme.com          # source server  
#      38.25.63.10      x.acme.com              # x client host  
  
# localhost name resolution is handled within DNS itself.  
#      127.0.0.1        localhost  
#      ::1              localhost  
192.168.65.135 dominio1.com  
192.168.65.135 dominio2.com  
|
```

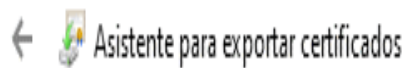
Seguidamente abrimos un explorador y visitamos uno de los dominios.

Vemos que nos aparece con el certificado, pero como página no segura ya que no tenemos la confianza sobre la CA que ha generado los certificados.



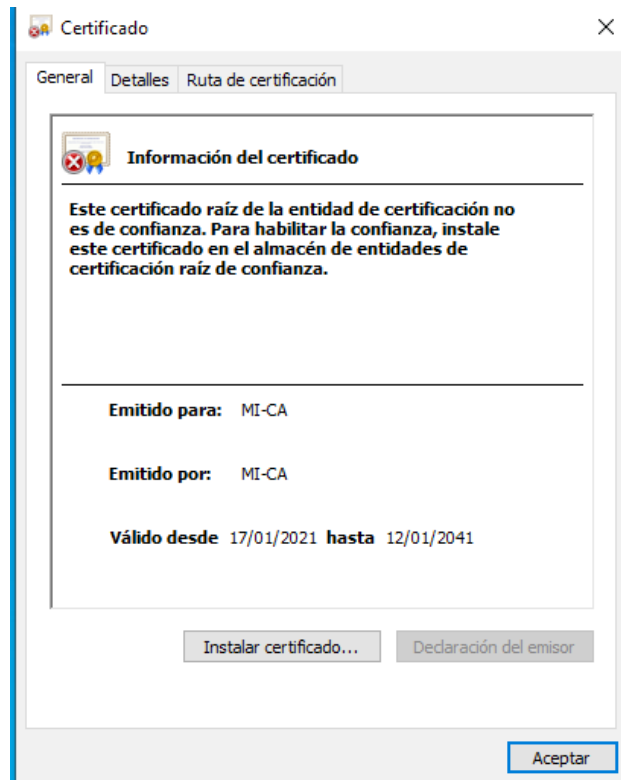


Podemos solucionar esto en algunos exploradores incluyendo los certificados de la cadena de certificación en nuestro almacén de certificados.




Archivo que se va a exportar

X



Seleccionamos como ubicación del almacén “Equipo local” para que esté disponible para todos los usuarios del equipo

←  Asistente para importar certificados

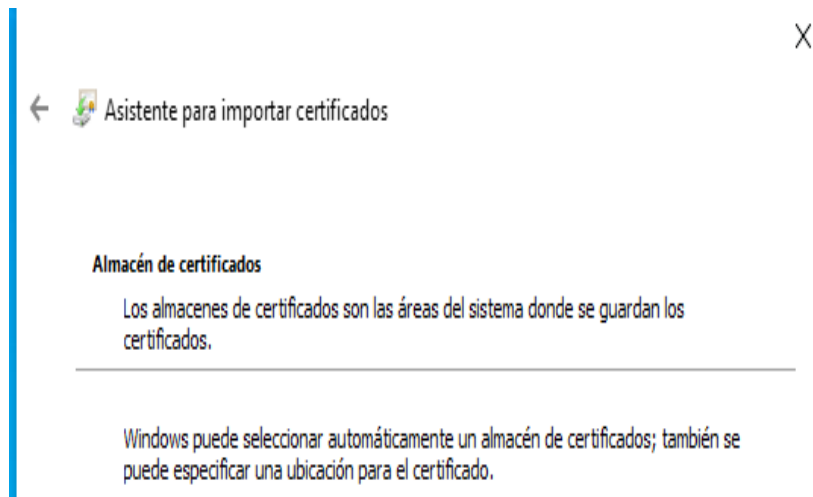
Este es el Asistente para importar certificados

Este asistente lo ayuda a copiar certificados, listas de certificados de confianza y listas de revocación de certificados desde su disco a un almacén de certificados.

Un certificado, que lo emite una entidad de certificación, es una confirmación de su identidad y contiene información que se usa para proteger datos o para establecer conexiones de red seguras. Un almacén de certificados es el área del sistema donde se guardan los certificados.

Ubicación del almacén

Y colocamos el certificado en el almacén “Entidades de certificación raíz de confianza”.



Ahora si comprobamos la página con internet explorer veremos que nos aparece con el candado sin ningún error.