
Práctica 5.19: Servidor virtual HTTPS por defecto en *Linux*

Realiza la siguiente configuración en el servidor *Apache* instalado en **ServidorLinuxXX**.

- Habilita el **servidor virtual por defecto**.
- Deshabilita los **servidores virtuales** creados en las prácticas anteriores.
- Habilita el modulo *mod_ssl*.
- Habilita el servidor virtual *ssl* por defecto.

Prueba la configuración.

1. Inicia una sesión en **ServidorLinuxXX** con un usuario con privilegios de administración.
2. Habilita el servidor virtual por defecto de *Apache*.

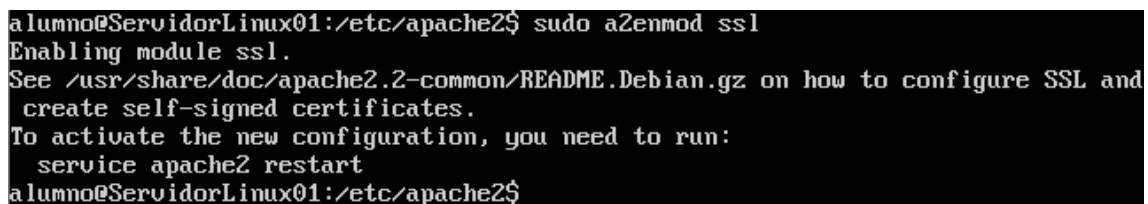
```
sudo a2ensite default
```

3. Verifica que dentro del directorio **/etc/apache2/sites-enabled** se ha creado el enlace **000-default**.
4. Deshabilita los servidores virtuales creados en prácticas anteriores.

```
sudo a2dissite software
sudo a2dissite hardware
```

5. Reinicia el servidor para que los cambios tengan efecto.
6. Habilita el módulo *modssl* que permite usar *https*, Figura 1.

```
sudo a2enmod ssl
```



```
alumno@ServidorLinux01:/etc/apache2$ sudo a2enmod ssl
Enabling module ssl.
See /usr/share/doc/apache2.2-common/README.Debian.gz on how to configure SSL and
create self-signed certificates.
To activate the new configuration, you need to run:
  service apache2 restart
alumno@ServidorLinux01:/etc/apache2$
```

Figura 1: Habilitar el modulo modssl

7. Reinicia el servidor para que los cambios tengan efecto.
8. Consulta el fichero **/etc/apache2/port.conf** y observa que si habilita el modulo *ssl* el servidor escuchará en el puerto 443, Figura 2.

```
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default
# This is also true if you have upgraded from before 2.2.9-3 (i.e. from
# Debian etch). See /usr/share/doc/apache2.2-common/NEWS.Debian.gz and
# README.Debian.gz

NameVirtualHost *:80
Listen 80

<IfModule mod_ssl.c>
    # If you add NameVirtualHost *:443 here, you will also have to change
    # the VirtualHost statement in /etc/apache2/sites-available/default-ssl
    # to <VirtualHost *:443>
    # Server Name Indication for SSL named virtual hosts is currently not
    # supported by MSIE on Windows XP.
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>
```

Figura 2: Fichero `/etc/apache2/port.conf`

9. Verifica que el servidor escucha en los puertos 80/TCP y 443/TCP.

```
netstat -ltn
```

10. Accede al directorio `/etc/apache2/sites-availables` y observa que existe un fichero denominado **default-ssl** que contiene la configuración por defecto de un servidor HTTPS.
11. Habilita el servidor virtual ssl defecto (default-ssl) de *Apache*.

```
sudo a2ensite default-ssl
```

12. Reinicia el servidor para que los cambios tengan efecto.
13. Consulta el fichero `/etc/apache2/sites-availables/default-ssl` y observa su configuración. Fíjate en las directivas que habilitan SSL y que definen la ruta del certificado digital que usará el servidor, Figuras 3 y 4.

El servidor utiliza por defecto un certificado digital autofirmado que se ha creado al instalar *Apache*. Un certificado autofirmado no está firmado por una autoridad de certificación (tercera parte de confianza) y por tanto, no existen mecanismos automáticos que garanticen su autenticidad. Por eso los navegadores nos pedirán confirmación cuando el servidor se lo envíe.

```

<IfModule mod_ssl.c>
<VirtualHost _default_:443>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /var/www/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>

```

Figura 3: Fichero /etc/apache2/sites-available/default-ssl

```

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

# A self-signed (snakeoil) certificate can be created by installing
# the ssl-cert package. See
# /usr/share/doc/apache2.2-common/README.Debian.gz for more info.
# If both key and certificate are stored in the same file, only the
# SSLCertificateFile directive is needed.
SSLCertificateFile /etc/ssl/certs/ssl-cert-snakeoil.pem
SSLCertificateKeyFile /etc/ssl/private/ssl-cert-snakeoil.key

# Server Certificate Chain:
# Point SSLCertificateChainFile at a file containing the
# concatenation of PEM encoded CA certificates which form the
# certificate chain for the server certificate. Alternatively
# the referenced file can be the same as SSLCertificateFile
# when the CA certificates are directly appended to the server
# certificate for convinience.
#SSLCertificateChainFile /etc/apache2/ssl.crt/server-ca.crt

```

Figura 4: Fichero /etc/apache2/sites-available/default-ssl

14. Desde **DesarrolloW7XX** abre el navegador y establece una conexión a `http://192.168.1.X7`, Figura 5.

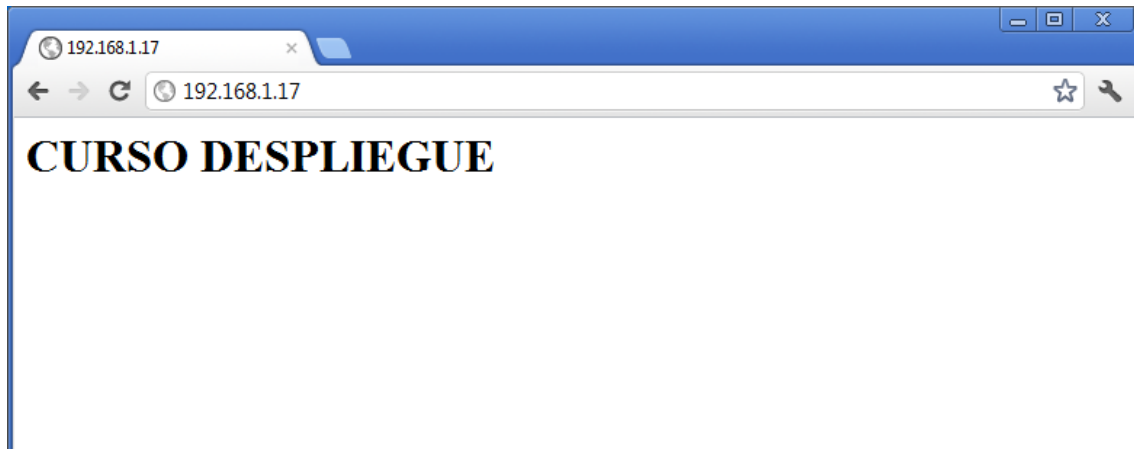


Figura 5: Conexión http

15. Desde **DesarrolloW7XX** abre el navegador y establece una conexión a `https:\\192.168.1.X7`, Figuras 6 y 7.

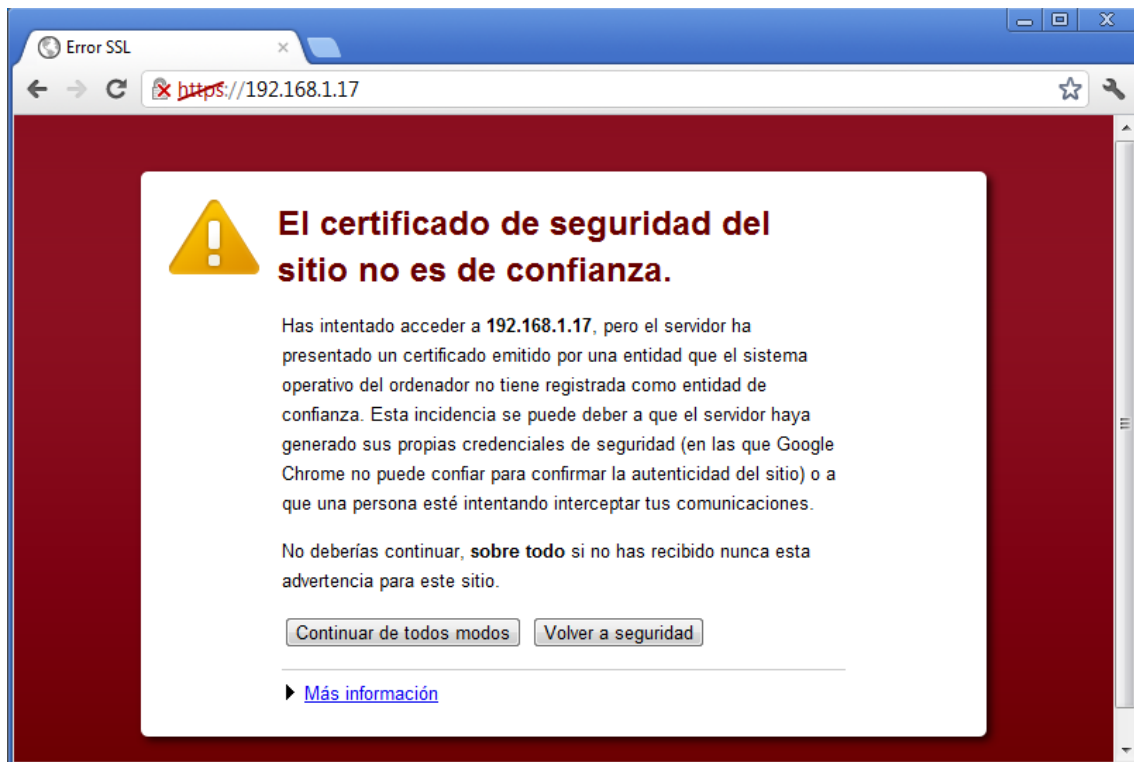


Figura 6: Conexión https

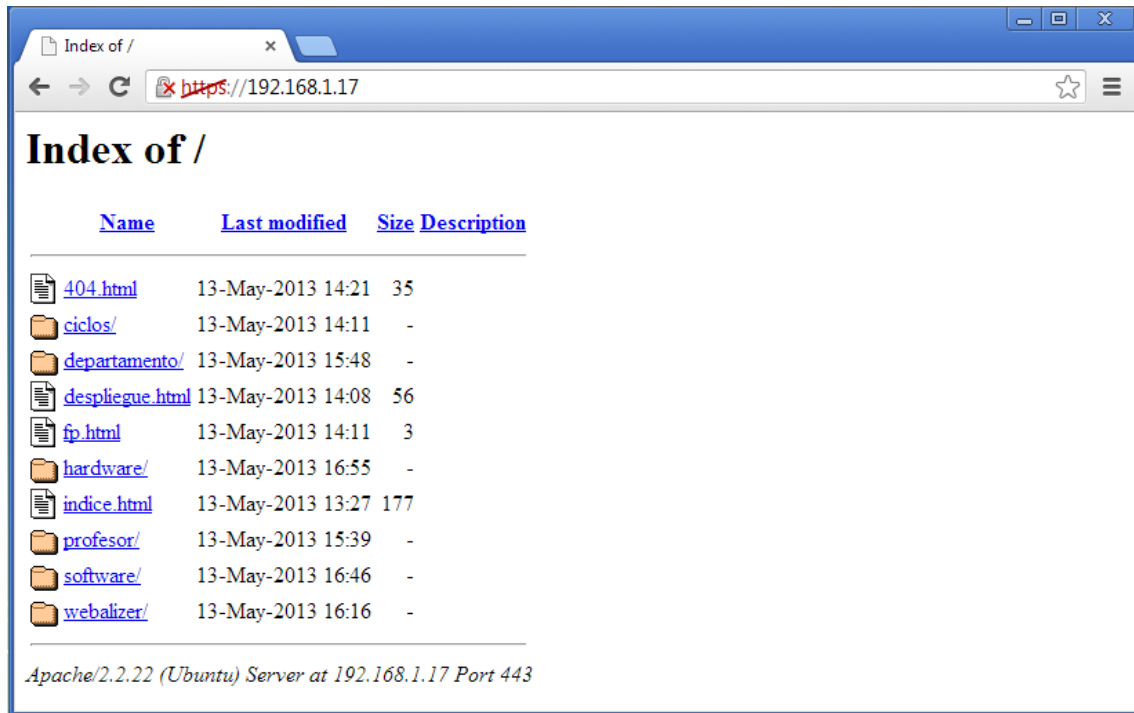


Figura 7: Conexión https

◇