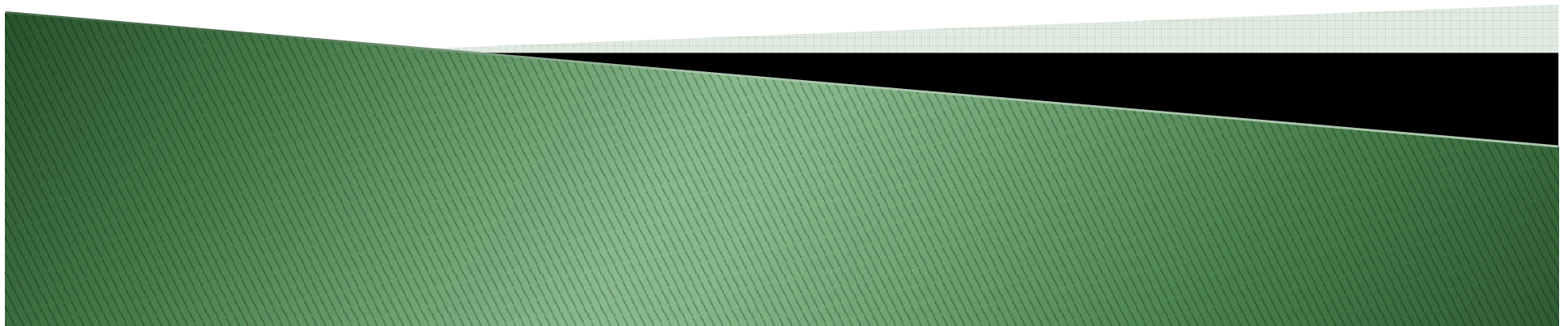


Unidad 4

Administración de *Apache* – 3

Despliegue de aplicaciones web



Índice

- ▶ Módulo mod_ssl.
- ▶ Configuración
 - Certificados.
 - Directivas.
- ▶ Bibliografía.

Módulo mod_ssl

- ▶ Es posible configurar *Apache* para que sirva contenidos seguros usando el protocolo HTTPS.
- ▶ Para ello hay que configurar y habilitar el módulo mod_ssl.



Módulo mod_ssl

- ▶ Webs

- <http://www.modssl.org/>
- http://httpd.apache.org/docs/2.2/mod/mod_ssl.html

- ▶ Utiliza las herramientas proporcionadas por el proyecto OpenSSL.

Configuración

Certificados

► 1) Generar clave privada

```
alumno@ServidorLinux01:~$ openssl genrsa -out seguro.key 2048
Generating RSA private key, 2048 bit long modulus
.....+++
.....+++
e is 65537 (0x10001)
alumno@ServidorLinux01:~$
```

Configuración

Certificados

► 2) Generar una solicitud de certificado (CSR)

```
alumno@ServidorLinux01:~$ openssl req -new -key seguro.key -out seguro.csr
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]:daw01
Organizational Unit Name (eg, section) []:daw01
Common Name (e.g. server FQDN or YOUR name) []:seguro.daw01.net
Email Address []:admin@daw01.net

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
alumno@ServidorLinux01:~$ _
```

Configuración

Certificados

- ▶ 3) Generar un certificado (1)
 - A) Enviar la solicitud de certificado a una autoridad de certificación .
 - Se envía, habitualmente, a través de un formulario web a la empresa certificadora
 - Validan si se ha pagado (si es necesario).
 - Validan la solicitud.
 - Generan el Certificado (.crt)
 - Remiten el certificado e instrucciones.

Configuración

Certificados

- ▶ 3) Generar un certificado (2)
 - B) Crear un certificado autofirmado

```
alumno@ServidorLinux01:~$ openssl x509 -req -days 365 -in seguro.csr -signkey se
guero.key -out seguro.crt
Signature ok
subject=/C=ES/ST=Madrid/L=Madrid/O=daw01/OU=daw01/CN=seguro.daw01.net/emailAddre
ss=admin@daw01.net
Getting Private key
alumno@ServidorLinux01:~$ _
```


Configuración

Directivas

- ▶ SSLEngine
- ▶ SSLCertificateFile
- ▶ SSLCertificateKeyFile
- ▶ ...
- ▶ **Web**
 - <http://httpd.apache.org/docs/2.2/ssl/>

Práctica

► Práctica 4.19

- Servidor virtual HTTPS por defecto en *Linux*.

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

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

    # 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
```

Práctica

► Práctica 4.20

- Servidor virtual HTTPS en *Linux*.

```
alumno@ServidorLinux01:~$ openssl req -new -key seguro.key -out seguro.csr
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's called a 'certificate request'
There are quite a few fields but you can leave some blank
For some fields there will be a '*' character that means 'any character'
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:
State or Province Name (full name) []:
Locality Name (eg, city) []:Madr
Organization Name (eg, company) []:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:admin@daw01.net

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
alumno@ServidorLinux01:~$
```

```
<IfModule mod_ssl.c>
<VirtualHost *:443>
    ServerName seguro.daw01.net
    DocumentRoot /var/www/seguro
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /var/www/seguro>
        DirectoryIndex index.html
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>
    ErrorLog ${APACHE_LOG_DIR}/seguro.error.log
    LogLevel warn
    CustomLog ${APACHE_LOG_DIR}/seguro.access.log combined

    SSLEngine on
    SSLCertificateFile      /etc/ssl/certs/seguro.crt
    SSLCertificateKeyFile   /etc/ssl/private/seguro.key
</VirtualHost>
</IfModule>
```

Práctica

► Práctica 4.21

◦ Servidor virtual HTTPS por defecto en *Windows*

```
#LoadModule mime_magic_module modules/mod_mime_magic.so
LoadModule negotiation_module modules/mod_negotiation.so
#LoadModule proxy_module modules/mod_proxy.so
#LoadModule proxy_ajp_module modules/mod_proxy_ajp.so
#LoadModule proxy_balancer_module modules/mod_proxy_balancer.so
#LoadModule proxy_connect_module modules/mod_proxy_connect.so
#LoadModule proxy_ftp_module modules/mod_proxy_ftp.so
#LoadModule proxy_http_module modules/mod_proxy_http.so
#LoadModule proxy_scgi_module modules/mod_proxy_scgi.so
#LoadModule reqtimeout_module modules/mod_reqtimeout.so
#LoadModule rewrite_module modules/mod_rewrite.so
LoadModule setenvif_module modules/mod_setenvif.so
#LoadModule spelling_module modules/mod_spelling.so
LoadModule ssl_module modules/mod_ssl.so
#LoadModule status_module modules/mod_status.so
#LoadModule substitute_module modules/mod_substitute.so
#LoadModule unique_id_module modules/mod_unique_id.so
LoadModule userdir_module modules/mod_userdir.so
#LoadModule usertrack_module modules/mod_usertrack.so
#LoadModule version_module modules/mod_version.so

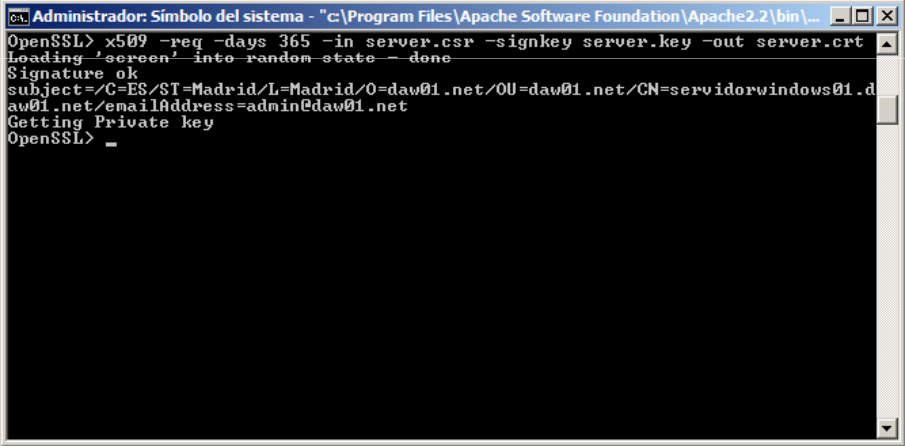
#Include conf/extra/httpd-vhosts.conf

# Local access to the Apache HTTP Server Manual
#Include conf/extra/httpd-manual.conf

# Distributed authoring and versioning (WebDAV)
#Include conf/extra/httpd-dav.conf

# Various default settings
#Include conf/extra/httpd-default.conf

# Secure (SSL/TLS) connections
Include conf/extra/httpd-ssl.conf
#
# Note: The following must must be present to support
#       starting without SSL on platforms with no /dev/random equivalent
#       but a statically compiled-in mod_ssl.
#
<IfModule ssl_module>
SSLRandomSeed startup builtin
```



```
Administrador: Símbolo del sistema - "c:\Program Files\Apache Software Foundation\Apache2.2\bin\..."
OpenSSL> x509 -req -days 365 -in server.csr -signkey server.key -out server.crt
Loading 'screen' into random state - done
Signature ok
subject=/C=ES/ST=Madrid/L=Madrid/O=daw01.net/OU=daw01.net/CN=servidorwindows01.daw01.net/emailAddress=admin@daw01.net
Getting Private key
OpenSSL> _
```

Bibliografía

- ▶ Servicios de Red e Internet. Álvaro García Sánchez, Luis Enamorado Sarmiento, Javier Sanz Rodríguez. Editorial Garceta.
- ▶ <http://httpd.apache.org>