

Primero creamos una llave privada RSA de 2048 bits

```
debian@debian: ~  
File Edit View Search Terminal Help  
debian@debian:~$ openssl genrsa -out /etc/ssl/private/myserver.key 2048
```

Despues creamos un CSR certificate signing request

```
[sudo] password for debian:  
debian@debian:~$ openssl req -new -key /etc/ssl/private/myserver.key -out /etc/ssl/certs/myserver.csr
```

Aqui ingresamos informacion sobre la empresa

```
debian@debian:~$ sudo openssl req -new -key /etc/ssl/private/myserver.key -out /etc/ssl/certs/myserver.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]:MX  
State or Province Name (full name) [Some-State]:Chihuahua  
Locality Name (eg, city) []:Delicias  
Organization Name (eg, company) [Internet Widgits Pty Ltd]:MiEmpresa  
Organizational Unit Name (eg, section) []:IT  
Common Name (e.g. server FQDN or YOUR name) []:mi-dominio.com  
Email Address []:admin@mi-dominio.com
```

Aqui firmamos el CSR con nuestra llave privada para validar el certificado por 265 dias.

```
debian@debian:~$ sudo openssl x509 -req -days 365 -in /etc/ssl/certs/myserver.csr -signkey /etc/ssl/private/myserver.key -out /etc/ssl/certs/myserver.crt  
Certificate request self-signature ok  
subject=C = MX, ST = Chihuahua, L = Delicias, O = MiEmpresa, OU = IT, CN = mi-dominio.com, emailAddress = admin@mi-dominio.com
```

Aqui deditamos el archivo de configuracion

```

GNU nano 7.2 /etc/apache2/sites-available/default-ssl.conf *
# practice often causes hanging connections with brain-dead browsers>
# this only for browsers where you know that their SSL implementation
# works correctly.
# Notice: Most problems of broken clients are also related to the HTTP
# keep-alive facility, so you usually additionally want to disable
# keep-alive for those clients, too. Use variable "nokeepalive" for t>
# Similarly, one has to force some clients to use HTTP/1.0 to workar>
# their broken HTTP/1.1 implementation. Use variables "downgrade-1.0">
# "force-response-1.0" for this.
BrowserMatch "MSIE [2-6]" \
    nokeepalive ssl-unclean-shutdown \
    downgrade-1.0 force-response-1.0

BrowserMatch "MSIE [17-9]" ssl-unclean-shutdown

</VirtualHost>

```

```

#SSLOptions +FakeBasicAuth +ExportCertData +StrictRequire
<FilesMatch "\.(?:cgi|shtml|phtml|php)$">
    SSLOptions +StdEnvVars
</FilesMatch>
<Directory /usr/lib/cgi-bin>
    SSLOptions +StdEnvVars
</Directory>

```

```

GNU nano 7.2 /etc/apache2/sites-available/default-ssl.conf *

# SSL Engine Switch:
# Enable/Disable SSL for this virtual host.
SSLEngine on
SSLCertificateFile /etc/ssl/certs/myserver.crt
SSLCertificateKeyFile /etc/ssl/private/myserver.key

# A self-signed (snakeoil) certificate can be created by installing
# the ssl-cert package. See
# /usr/share/doc/apache2/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

```

```
File Edit View Search Terminal Help
GNU nano 7.2 /etc/apache2/sites-available/default-ssl.conf *
<VirtualHost *:443>
    ServerAdmin admin@mi-dominio.com
    ServerName mi-dominio.com

    DocumentRoot /var/www/html

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

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

Aqui habilitamos SSL y el modulo SSL

```
debian@debian:~$ sudo a2enmod ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create s
elf-signed certificates.
To activate the new configuration, you need to run:
    systemctl restart apache2
debian@debian:~$ sudo a2ensite default-ssl
Enabling site default-ssl.
To activate the new configuration, you need to run:
    systemctl reload apache2
```



Warning: Potential Security Risk Ahead

Firefox detected a potential security threat and did not continue to **mi-dominio.com**. If you visit this site, attackers could try to steal information like your passwords, emails, or credit card details.

[Learn more...](#)

Go Back (Recommended)

Advanced...



debian

Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/  
|-- apache2.conf  
|   |-- ports.conf
```

El reporte jSON menciona que la llave se encuentra en el archivo `/etc/ssl/private/myserver.key`. Abajo confirmo que si lo esta.

```

debian@debian: ~
File Edit View Search Terminal Help
GNU nano 7.2 /etc/ssl/private/myserver.key
-----BEGIN PRIVATE KEY-----
MIIEvQIBADANBgkqhkiG9w0BAQEFAASCBCwggSjAgEAAoIBAQDLcsoUavIaDVtK
WovtxxDrVTQSQ62cKTHGsx3i93x0P0IJYahnRWaNRU5gA/JtvN7MHMK49Xj8KB4A
U07FfkfY+gZwXuK95+ssESvozv9aS3A+pZKJXFxaZYMWysBdGla08VrHZ50i8JAF
sCC9fLH9MMA90igH8WpQpBwvy8SONUzo7pQUF4SuzjT86J2Enq1WfvJExH9sAVn/
089719ERINDJH/UjRICyprroi8PMLo0t9p3WgIUiNuP7tujk/73qmyX3EMMURzlr
Hiplb/qIUmvHAL/QsIClsCCHxJsIZmdv6lyv4TlmJNwGfYwBztx7fCvznhE2EMil
WtREfUIhAgMBAAECggEAG1qSL/+XEIimMw+Mi8o9jJ6WkzQVRg9F2YTgqlEDIzAU
JWXDeisbGDhuYrsNH9MKk5/0UGD0xNZx/jHjgJsZds1bD0W6r7Tcaj7ey/z02UOW
U135n01TP3YycxtMlNgD0dDF11TnjDiS7KjbJg77cWLaHhXA+xxq4m/G0GjppwLK
Qhnych0rrFZRMekIQmqVh5U30S/CjCQMjuif19cLiP3a5/Upglc3chBJum3av/TS
pyBvCLticGwxB7vJ1dUKijvpVzhTvufkLP04MYIanQ5UGGtpmgD4NzqY3C74erzu
+aAm2Ih63G9rJhUxAzbXFGxZkGPqgxB2BLXTrqBN7wKBgQDwiXI7H6x1catJXvDg
BCLnWXY4Gby2Vj14/DvzzBxYAcRnsqyXwmaTiTf1Sh/ifX7w9z3RJC83mSU0gvi
xPc9323qZjg8xyoqmV5EOXM5oFCKMD3TT1bAmh1/w7xZZdMlXgcG+uFPZS/GgZ0q
twmCHWbbV2/zVjXk+G2Iqhr1VwKBgQDYhvJHpch1HNve13iv0cc1/0dQGUAgeqnI
-----END PRIVATE KEY-----

```