

Configuración de HTTPS en Apache2

Creamos un dominio en duckdns



Configuramos las variables de entorno

```
export DOMAIN="jaimesimarро.duckdns.org"
export WEBROOT="/var/www/miweb"
```

Instalamos apache

```
sudo apt update
sudo apt install -y apache2
```

Creamos el directorio web y una pagina basica

```
jaime@server:~$ sudo mkdir -p $WEBROOT
jaime@server:~$ echo "<h1>HTTPS con Let's Encrypt (DNS-01) - Jaime Simarro</h1>" | sudo tee $WEBROOT/index.html
<h1>HTTPS con Let's Encrypt (DNS-01) - Jaime Simarro</h1>
jaime@server:~$ cat $WEBROOT/index.html
<h1>HTTPS con Let's Encrypt (DNS-01) - Jaime Simarro</h1>
jaime@server:~$ |
```

Creamos el virtualhost HTTP de Apache

```
sudo nano /etc/apache2/sites-available/miweb.conf
```

A screenshot of a terminal window titled "Windows PowerShell". It contains the configuration for an Apache2 virtual host. The configuration includes setting the server name to "jaimesimarро.duckdns.org" and the document root to "/var/www/miweb". It also defines a directory for "/var/www/miweb" with specific options like "-Indexes +FollowSymLinks" and "AllowOverride All". Error and custom logs are specified as well. The configuration ends with a closing tag for the virtual host.

Activamos el sitio web con apache

```
sudo a2ensite miweb.conf  
sudo systemctl reload apache2
```

Instalamos Certbot

```
sudo apt install -y certbot  
certbot --version
```

Emitimos un certificado DNS

```
sudo certbot -d jaimesimarro.duckdns.org --manual --preferred-challenges dns certonly
```

(Haciendolo con los pasos de la practica no me funcionaba y he realizadodo otros que llegan al mismo punto)

Instalamos los plugins de DuckDNS

```
sudo apt install -y python3-pip  
sudo pip3 install certbot-dns-duckdns --break-system-packages
```

Creamos el archivo con el token y le damos permisos

```
sudo mkdir -p /etc/letsencrypt  
echo "dns_duckdns_token=393dc506-d8bb-4f2a-9829-611d94bf99f7" | sudo tee  
/etc/letsencrypt/duckdns.ini  
sudo chmod 600 /etc/letsencrypt/duckdns.ini
```

Solicitamos el certificado de manera automatica

```
sudo certbot certonly --authenticator dns-duckdns --dns-duckdns-credentials  
/etc/letsencrypt/duckdns.ini -d jaimesimarro.duckdns.org
```

```
jaime@server:~$ sudo certbot certonly --authenticator dns-duckdns --dns-duckdns-credentials /etc/letsencrypt/duckdns.ini --dns-duckdns-propagation-seconds 6  
0 -d jaimesimarro.duckdns.org  
Saving debug log to '/var/log/letsencrypt/letsencrypt.log'  
Requesting a certificate for jaimesimarro.duckdns.org  
Waiting 60 seconds for DNS changes to propagate  
  
Successfully received certificate.  
Certificate is saved at: /etc/letsencrypt/live/jaimesimarro.duckdns.org/fullchain.pem  
Key is saved at: /etc/letsencrypt/live/jaimesimarro.duckdns.org/privkey.pem  
This certificate expires on 2026-02-11.  
These files will be updated when the certificate renews.  
Certbot has set up a scheduled task to automatically renew this certificate in the background.  
- - - - -  
If you like Certbot, please consider supporting our work by:  
* Donating to ISRG / Let's Encrypt: https://letsencrypt.org/donate  
* Donating to EFF: https://eff.org/donate-le  
- - - - -  
jaime@server:~$
```

Habilitamos el modulo SSL

```
sudo a2enmod ssl
```

Creamos el virtualhost de HTTPS

```
GNU nano 7.2                                         /etc/apache2/sites-enabled/miweb-ssl.conf
<VirtualHost *:443>
    ServerName jaimesimarro.duckdns.org
    DocumentRoot /var/www/miweb

    SSLEngine on
    SSLCertificateFile /etc/letsencrypt/live/jaimesimarro.duckdns.org/fullchain.pem
    SSLCertificateKeyFile /etc/letsencrypt/live/jaimesimarro.duckdns.org/privkey.pem

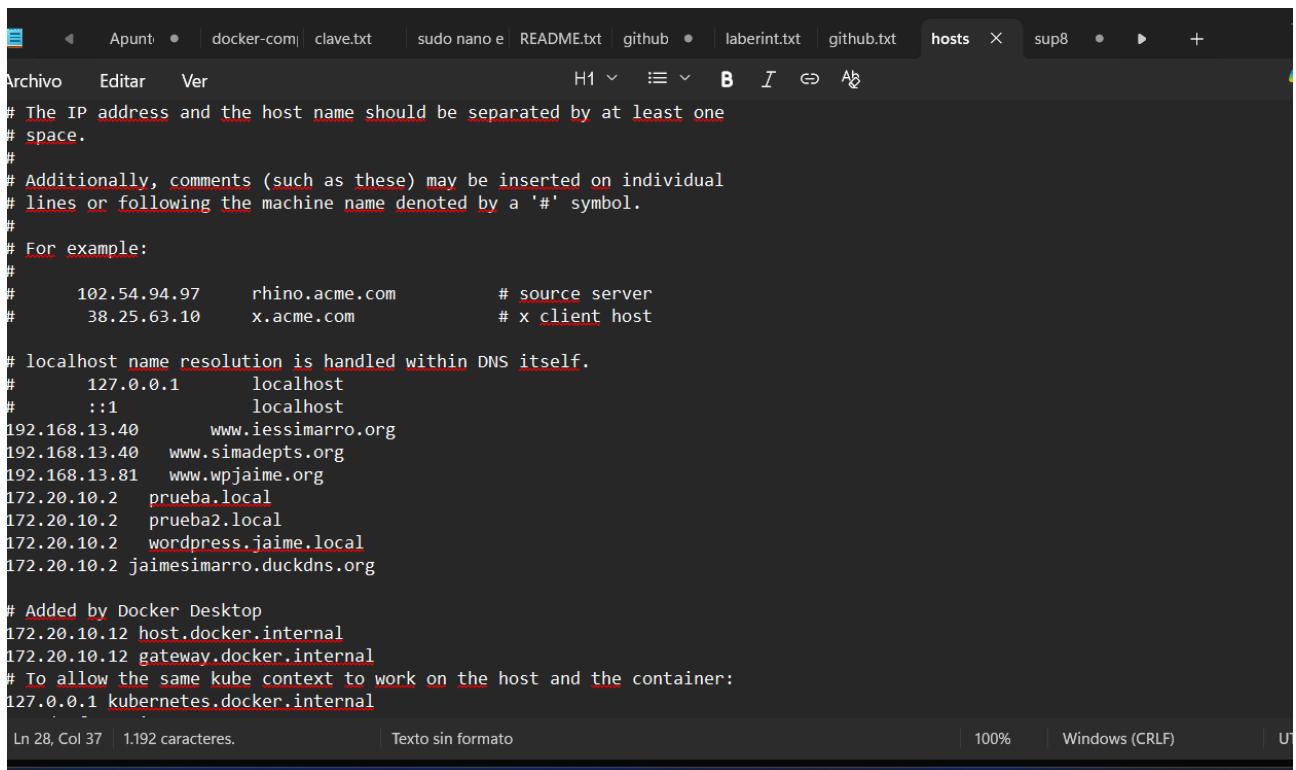
    <Directory /var/www/miweb>
        Options -Indexes +FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/miweb_ssl_error.log
    CustomLog ${APACHE_LOG_DIR}/miweb_ssl_access.log combined
</VirtualHost>
```

Activamos el sitio y reiniciamos

```
sudo apache2ctl configtest
sudo systemctl reload apache2
```

Añadimos el archivo al /etc/hosts

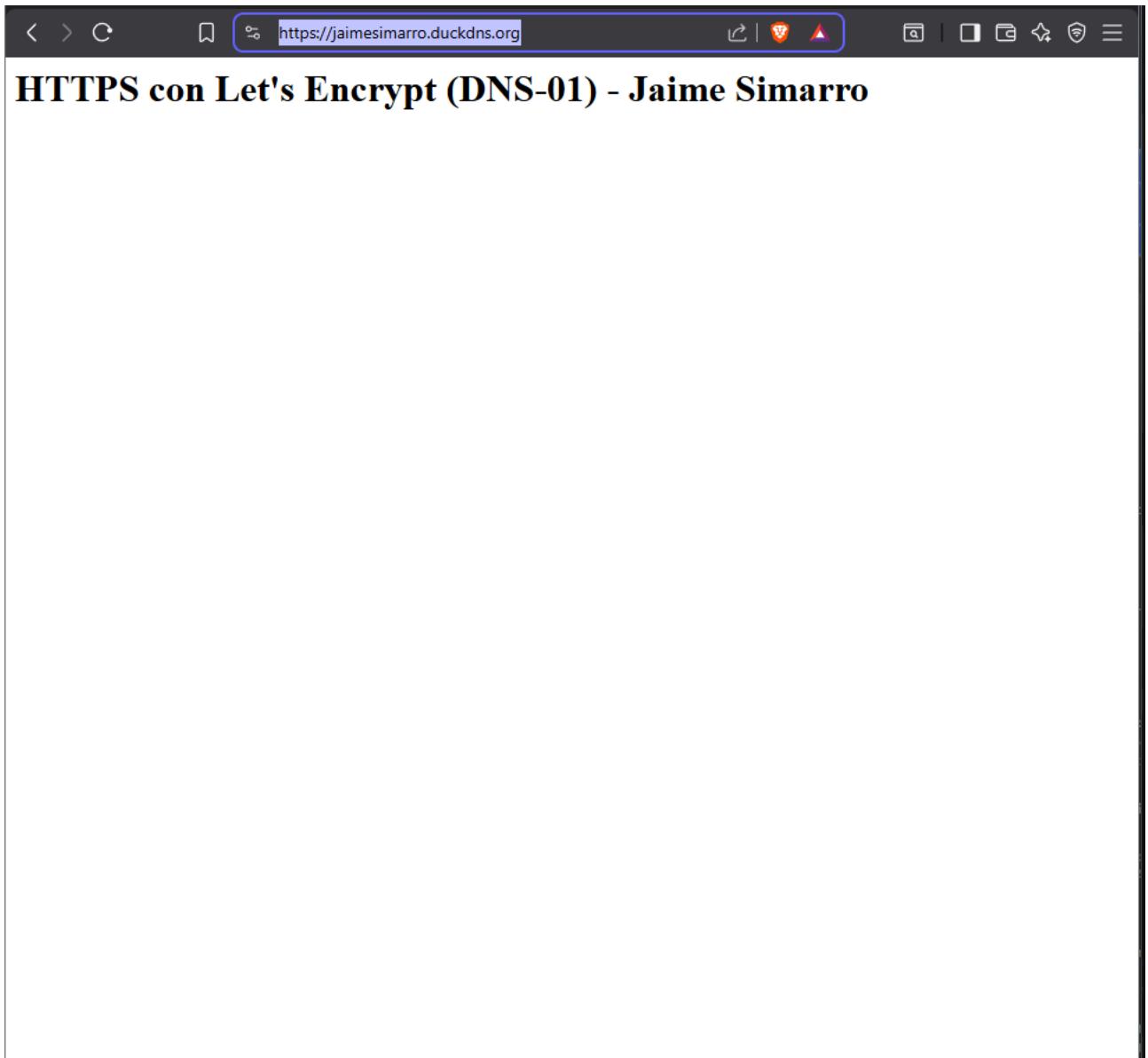


```
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#       102.54.94.97      rhino.acme.com      # source server
#       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
192.168.13.40      www.iessimarro.org
192.168.13.40      www.simadepts.org
192.168.13.81      www.wpjaime.org
172.20.10.2        prueba.local
172.20.10.2        prueba2.local
172.20.10.2        wordpress.jaime.local
172.20.10.2        jaimesimarro.duckdns.org

# Added by Docker Desktop
172.20.10.12 host.docker.internal
172.20.10.12 gateway.docker.internal
# To allow the same kube context to work on the host and the container:
127.0.0.1 kubernetes.docker.internal
```

Y realizamos la prueba



```
jaime@server:~$ sudo certbot certificates
Saving debug log to /var/log/letsencrypt/letsencrypt.log

-----
Found the following certs:
  Certificate Name: jaimesimarro.duckdns.org
    Serial Number: 6c1fd2549979dc65274c0ff34512d3d6e00
    Key Type: ECDSA
    Domains: jaimesimarro.duckdns.org
    Expiry Date: 2026-02-11 06:54:07+00:00 (VALID: 89 days)
    Certificate Path: /etc/letsencrypt/live/jaimesimarro.duckdns.org/fullchain.pem
    Private Key Path: /etc/letsencrypt/live/jaimesimarro.duckdns.org/privkey.pem
-----
jaime@server:~$ |
```