

Configuración de SSL

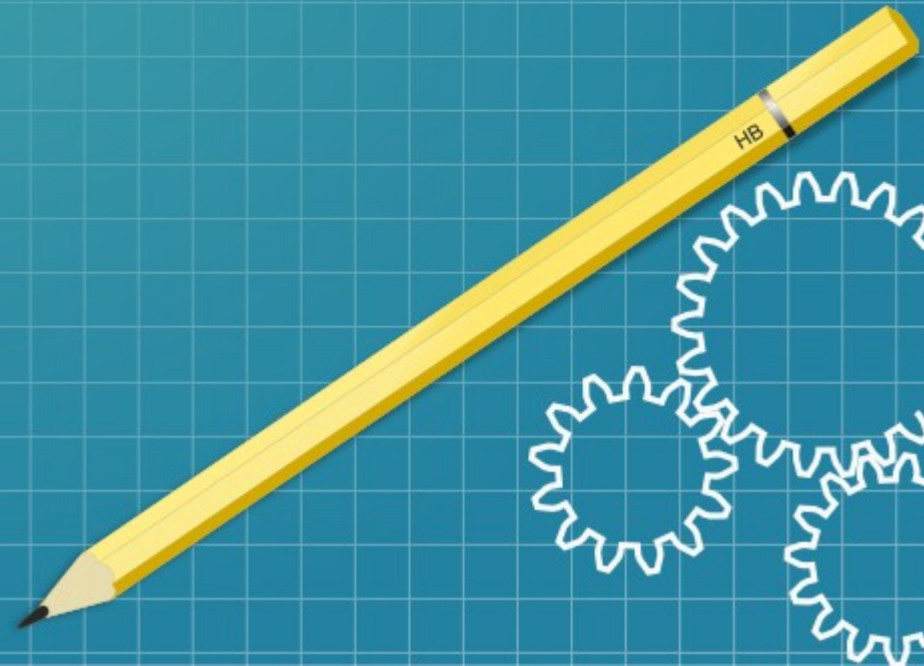
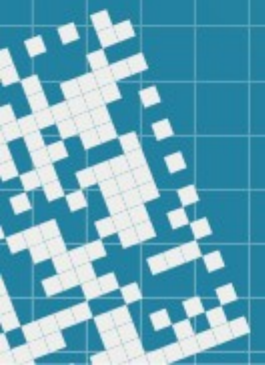


¿Qué es el SSL?



- Secure Sockets Layer, o el nivel de conectores seguros, es un protocolo de cifrado que mantiene segura una conexión a internet, así cómo también protege cualquier información confidencial que se envía entre dos sistemas e impide que los delincuentes lean y modifiquen cualquier dato que se transfiera.
- Actualmente tiene una versión mejorada que se llama TLS (Transport Layer Security) que la hace más segura porque usa sistema de cifrados mucho más resistentes a los ataques.

Procedimiento

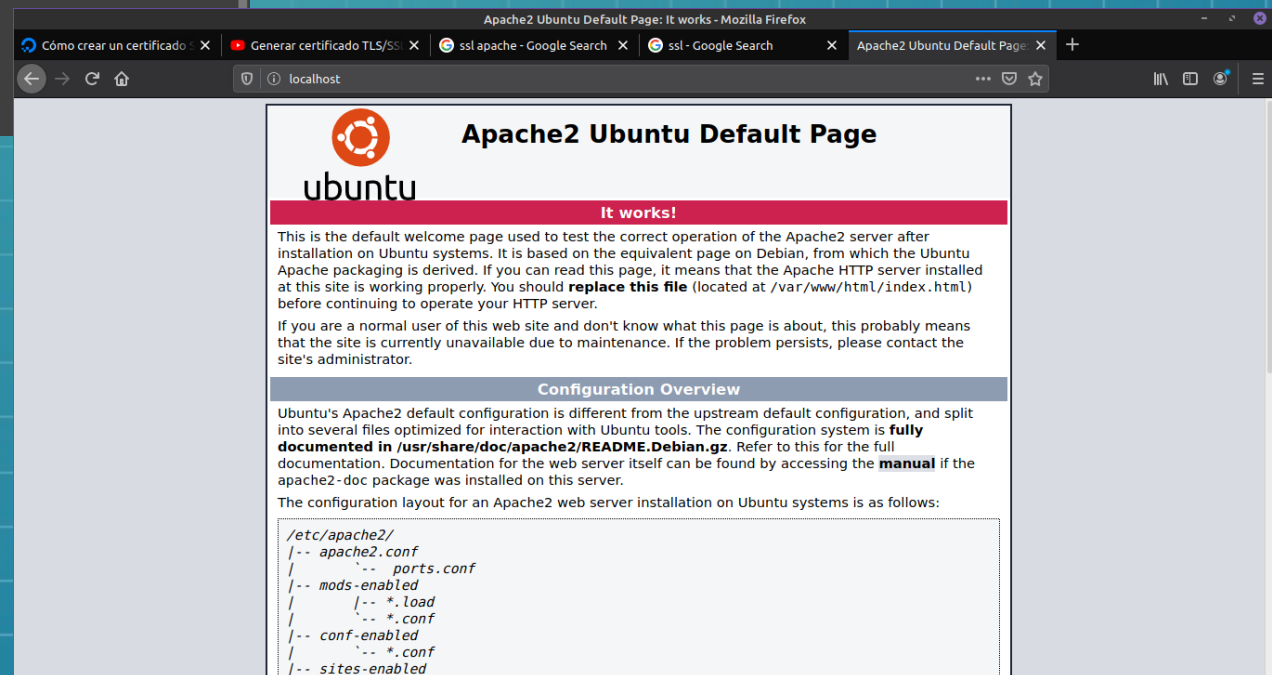


1. Instalar y verificar el funcionamiento del Apache

```
cristian@cristian-VirtualBoxPC: ~  
File Edit View Search Terminal Help  
cristian@cristian-VirtualBoxPC:~$ service apache2 status  
● apache2.service - The Apache HTTP Server  
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor prese  
   Active: active (running) since Sun 2020-11-22 03:07:40 CET; 2h 14min ago  
     Docs: https://httpd.apache.org/docs/2.4/  
   Process: 710 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUC  
   Main PID: 814 (apache2)  
     Tasks: 55 (limit: 1686)  
    Memory: 3.5M  
   CGroup: /system.slice/apache2.service  
           └─814 /usr/sbin/apache2 -k start  
             └─815 /usr/sbin/apache2 -k start  
               └─816 /usr/sbin/apache2 -k start  
  
Nov 22 03:07:31 cristian-VirtualBoxPC systemd[1]: Starting The Apache HTTP Serv  
Nov 22 03:07:40 cristian-VirtualBoxPC apachectl[733]: AH00558: apache2: Could n  
Nov 22 03:07:40 cristian-VirtualBoxPC systemd[1]: Started The Apache HTTP Serve  
lines 1-16/16 (END)
```

Instalar: `sudo apt install apache2`

Verificar: `services apache2 status`



2. Crear el fichero SSL autofirmado

```
cristian@cristian-VirtualBoxPC:~$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/apache-selfsigned.key -out /etc/ssl/certs/apache-selfsigned.crt
[sudo] password for cristian:
Generating a RSA private key
.....+++++
.....+++++
writing new private key to '/etc/ssl/private/apache-selfsigned.key'
-----
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.
-----
```

`sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/apache-selfsigned.key -out /etc/ssl/certs/apache-selfsigned.crt`

openssl: es la herramienta básica para configurar las llaves virtuales y los certificados creados.

req: subcomando que especifica que deseamos usar la administración de la solicitud de firma de certificados (CSR-Certificate Signing Request) X.509 (que es un estándar de infraestructura de claves públicas al que se adecúan SSL y TLS para la administración de claves y certificados).

-x509: modifica aún más el subcomando anterior al indicar a la utilidad que deseamos crear un certificado autofirmado en el lugar de generar una solicitud de firma de certificados, como normalmente sucede.

-nodes: indica a OpenSSL que omita la opción para proteger nuestro certificado con una frase de contraseña. Es necesario para que Apache pueda leer el archivo en cada reinicio sin que el usuario intervenga, ya que la frase solo podrá ser usada por el servidor.

-days 365: esta opción sirve para especificar el tiempo de validez del certificado (en este caso un año).

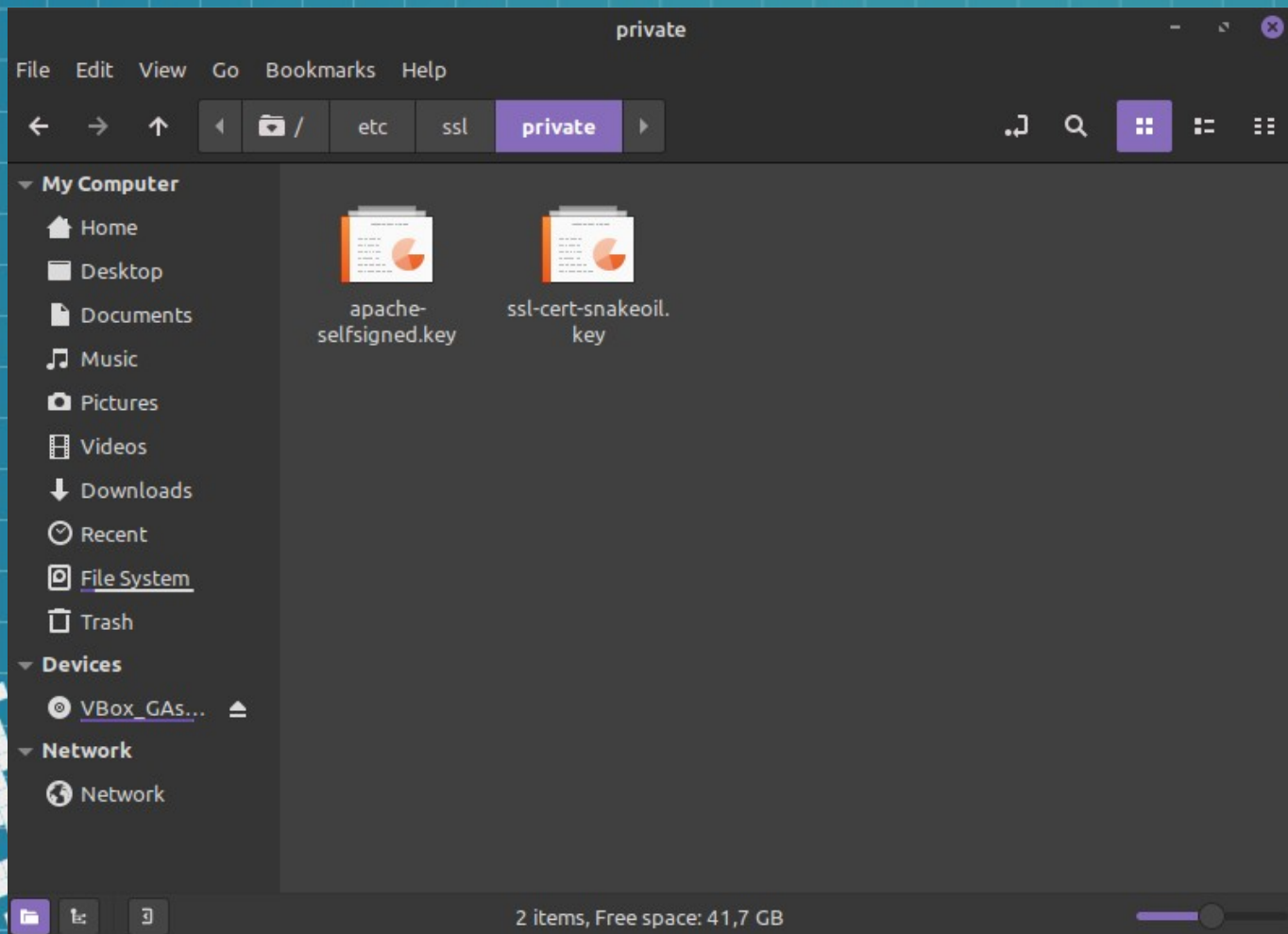
-newkey rsa:2048: especifica que deseamos crear un nuevo certificado y una nueva clave al mismo tiempo. Se puede crear previamente a hacer la creación del certificado, pero en este caso se especifica que se cree en conjunto con el comando `rsa:2048`, que indica crear una clave RSA* de 2048 bits de extensión.

-keyout: esta línea indica a OpenSSL donde colocar el archivo de la clave privada generado que se está creando.

-out: indica a OpenSSL dónde colocar el certificado que se ha creado.

Se rellena un pequeño formulario con los datos de la empresa. Lo más importante es la IP del servidor, que tiene que ser idéntica a la del apache.

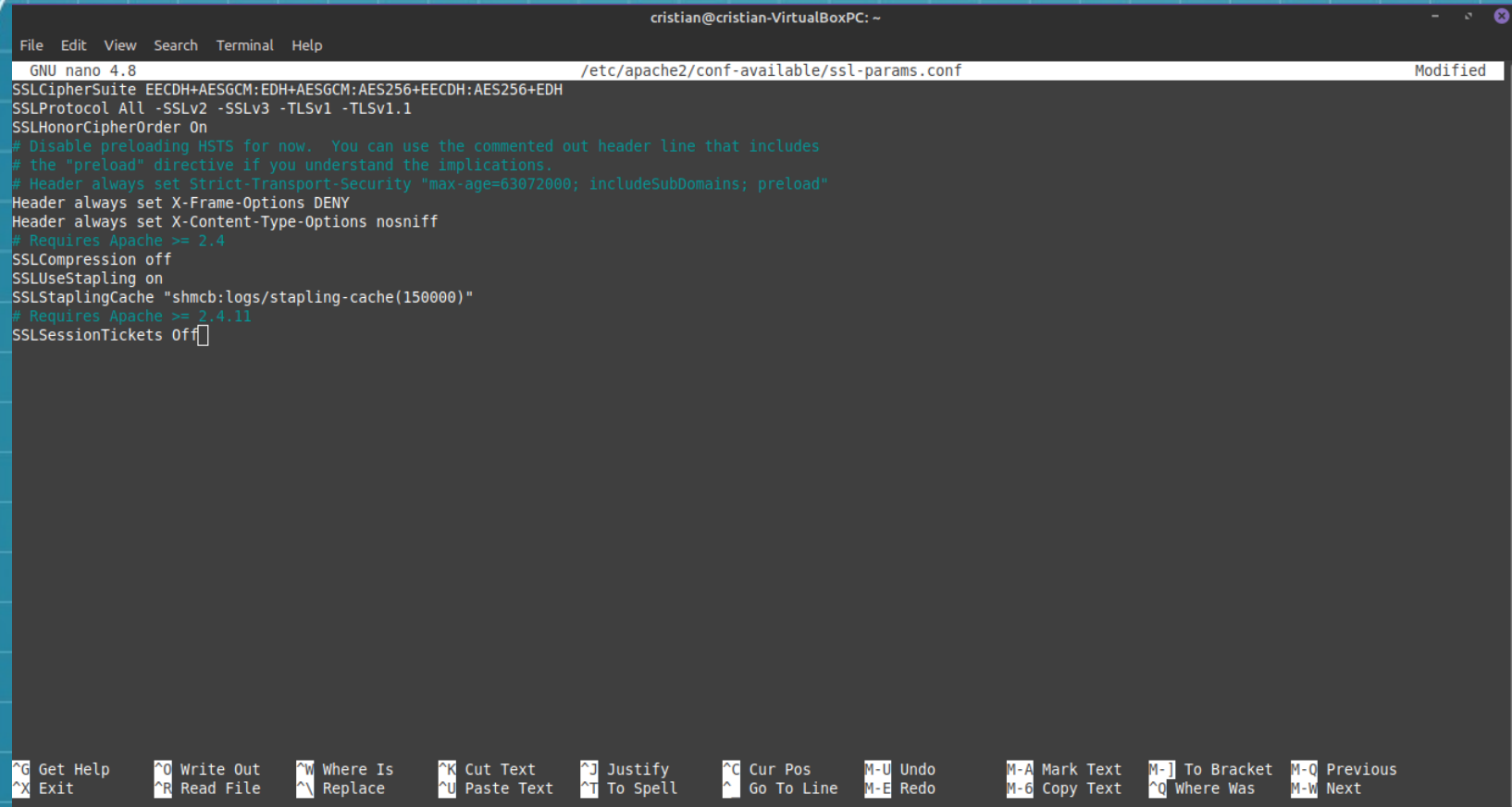
```
Country Name (2 letter code) [AU]:ES  
State or Province Name (full name) [Some-State]:MURCIA  
Locality Name (eg, city) []:YECLA  
Organization Name (eg, company) [Internet Widgits Pty Ltd]:AEROVUELO CO. S.A.  
Organizational Unit Name (eg, section) []:Base de Operacion  
Common Name (e.g. server FQDN or YOUR name) []:10.0.2.15  
Email Address []:info@aerovuelo.com.es
```



Se confirma que se guarde la llave y el certificado recién creado.

3. Crear un fragmento de configuración de Apache con ajustes de cifrado seguro.

```
sudo nano /etc/apache2/conf-available/ssl-params.conf
```

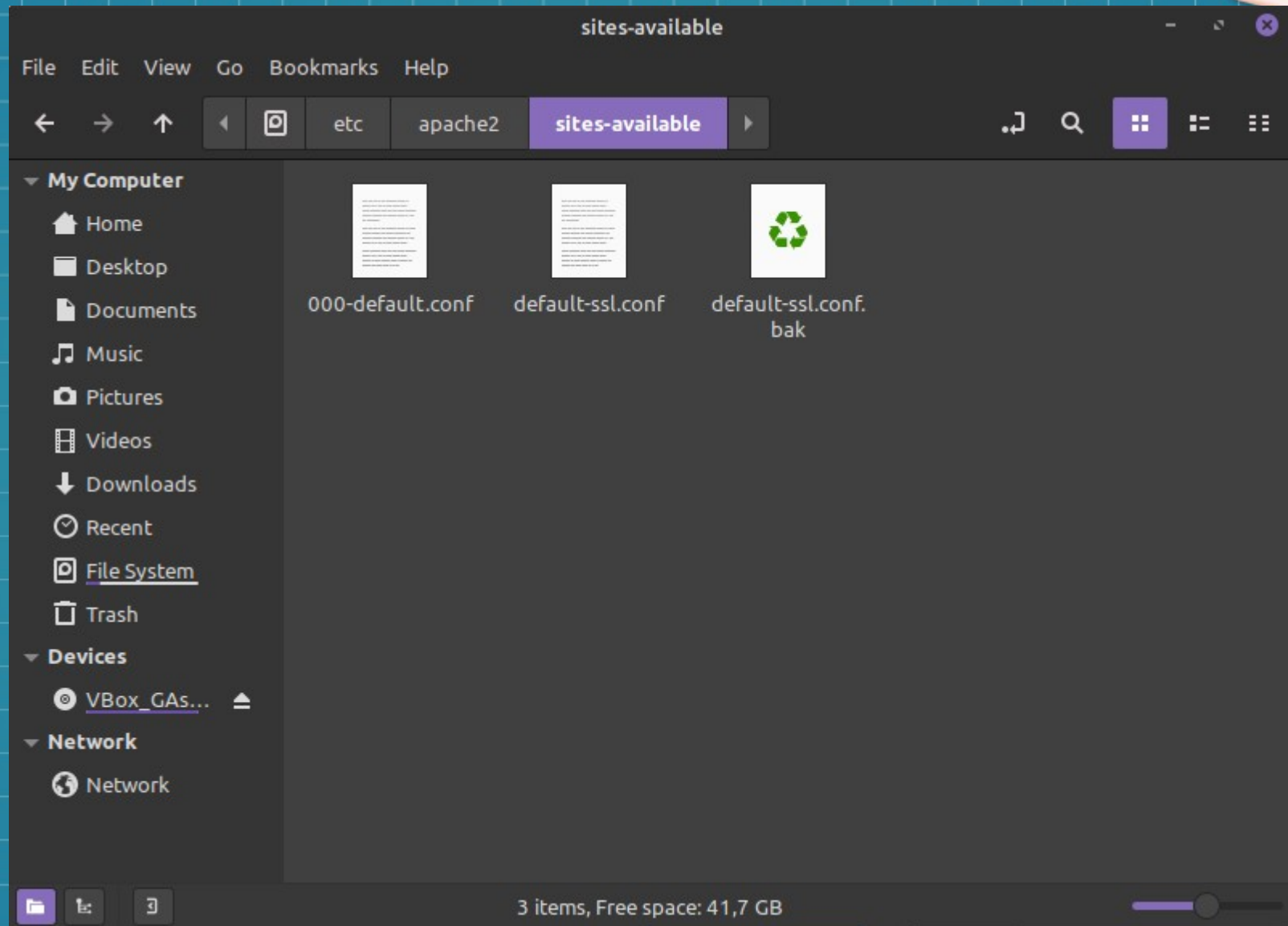


```
cristian@cristian-VirtualBoxPC: ~  
File Edit View Search Terminal Help  
GNU nano 4.8 /etc/apache2/conf-available/ssl-params.conf Modified  
SSLCipherSuite ECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH  
SSLProtocol All -SSLv2 -SSLv3 -TLSv1 -TLSv1.1  
SSLHonorCipherOrder On  
# Disable preloading HSTS for now. You can use the commented out header line that includes  
# the "preload" directive if you understand the implications.  
# Header always set Strict-Transport-Security "max-age=63072000; includeSubDomains; preload"  
Header always set X-Frame-Options DENY  
Header always set X-Content-Type-Options nosniff  
# Requires Apache >= 2.4  
SSLCompression off  
SSLUseStapling on  
SSLStaplingCache "shmcb:logs/stapling-cache(150000)"  
# Requires Apache >= 2.4.11  
SSLSessionTickets Off
```

^G Get Help ^O Write Out ^W Where Is ^X Cut Text ^J Justify ^C Cur Pos ^M-U Undo ^M-A Mark Text ^M-J To Bracket ^M-Q Previous
^X Exit ^R Read File ^N Replace ^U Paste Text ^T To Spell ^G Go To Line ^M-E Redo ^M-G Copy Text ^Q Where Was ^M-W Next

Con esto, se configurará Apache con un conjunto de cifrado SSL seguro y se habilitarán algunas características avanzadas que ayudarán a mantener protegido nuestro servidor. Los parámetros que se configurarán pueden utilizarse a través de cualquier hosting virtual que habilite SSL.


```
sudo cp /etc/apache2/sites-available/default-ssl.conf /etc/apache2/sites-available/default-ssl.conf.bak
```



Se hace una copia del archivo por defecto del host virtual antes de cambiarlo.


```
GNU nano 4.8 /etc/apache2/sites-
<IfModule mod_ssl.c>
  <VirtualHost _default :443>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www/html

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

    SSLEngine on

    SSLCertificateFile      /etc/ssl/certs/ssl-cert-snakeoil.pem
    SSLCertificateKeyFile   /etc/ssl/private/ssl-cert-snakeoil.key

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

  </VirtualHost>
</IfModule>
```

Archivo modificado
para que el archivo
de configuración por
defecto apunte a la
clave y al certificado
ya creado.

Archivo por defecto antes
de modificar.

```
GNU nano 4.8 /etc/apache2/sites-available/default-ssl.conf Modifi
<IfModule mod_ssl.c>
  <VirtualHost _default :443>
    ServerAdmin info@aerovuelo.com.es
    ServerName 10.0.2.15

    DocumentRoot /var/www/html

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

    SSLEngine on

    SSLCertificateFile      /etc/ssl/certs/apache-selfsigned.crt
    SSLCertificateKeyFile   /etc/ssl/private/apache-selfsigned.key

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

  </VirtualHost>
</IfModule>
```

4. Configurar el Firewall

Activar el Firewall:

```
cristian@cristian-VirtualBoxPC:~$ sudo ufw enable
Firewall is active and enabled on system startup
```

Verificar los perfiles para usar:

```
cristian@cristian-VirtualBoxPC:~$ sudo ufw applist
Available applications:
Apache
Apache Full
Apache Secure
CUPS
cristian@cristian-VirtualBoxPC:~$
```

Se agrega la regla del Apache full que habilita el uso del puerto 443 y el puerto 80:

```
cristian@cristian-VirtualBoxPC:~$ sudo ufw allow 'Apache Full'
Rule added
Rule added (v6)
```

Se confirma el uso de las nuevas reglas:

```
cristian@cristian-VirtualBoxPC:~$ sudo ufw status
Status: active

To Action From
--
Apache Full ALLOW Anywhere
Apache Full (v6) ALLOW Anywhere (v6)
```


5. Habilitar los cambios y el SSL en el servidor Apache

```
cristian@cristian-VirtualBoxPC:~$ 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 self
-signed certificates.
To activate the new configuration, you need to run:
    systemctl restart apache2
```

Se habilita
directamente el ssl.

```
cristian@cristian-VirtualBoxPC:~$ sudo a2enmod headers
Enabling module headers.
To activate the new configuration, you need to run:
    systemctl restart apache2
```

Se habilitan las cabeceras.

Los parámetros de cifrado
creados se aplican al servidor.

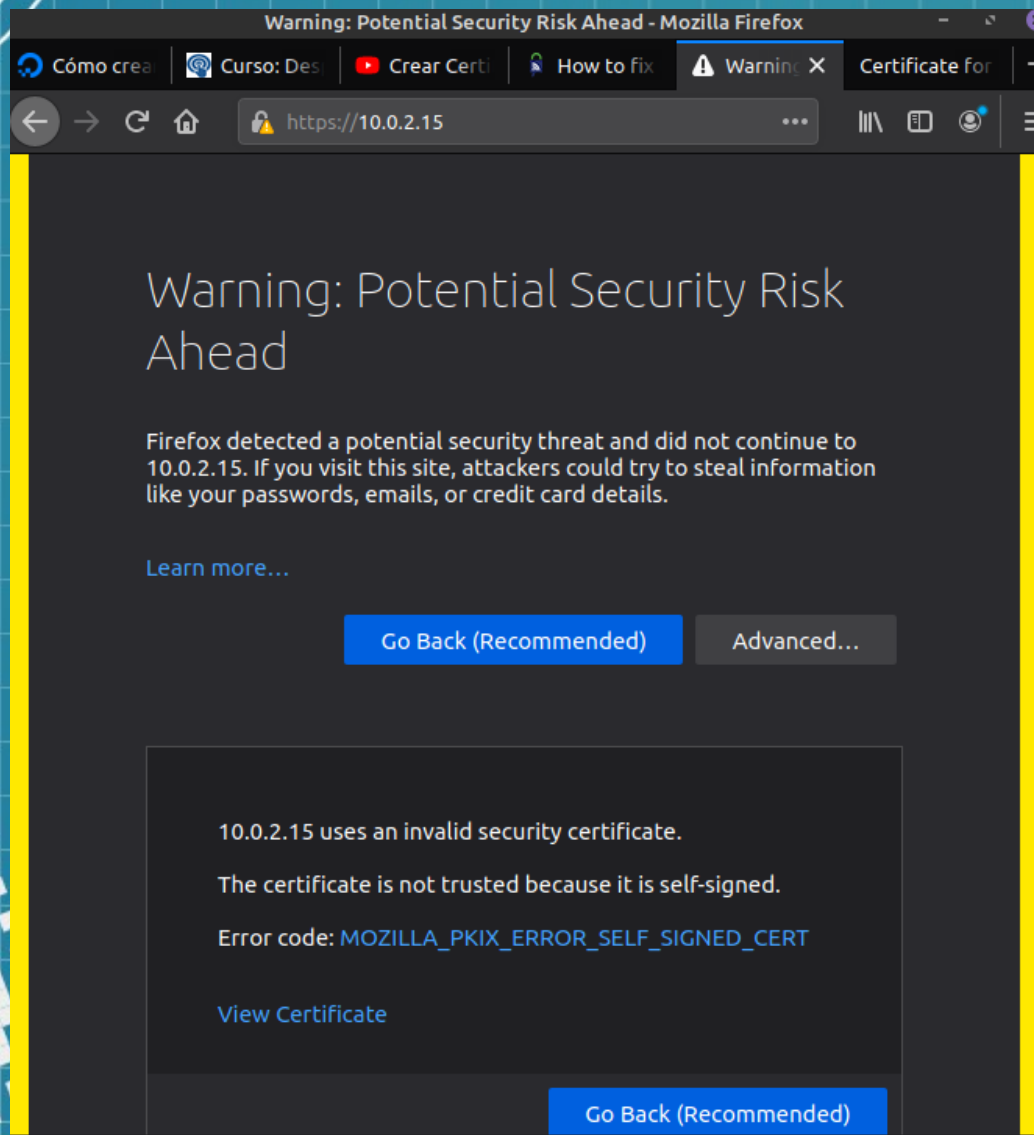
```
cristian@cristian-VirtualBoxPC:~$ sudo a2enconf ssl-params
Enabling conf ssl-params.
To activate the new configuration, you need to run:
    systemctl reload apache2
```

Se aplica el archivo por
defecto que carga el SSL en el
servidor.

```
cristian@cristian-VirtualBoxPC:~$ sudo a2ensite default-ssl.conf
Enabling site default-ssl.
To activate the new configuration, you need to run:
    systemctl reload apache2
```

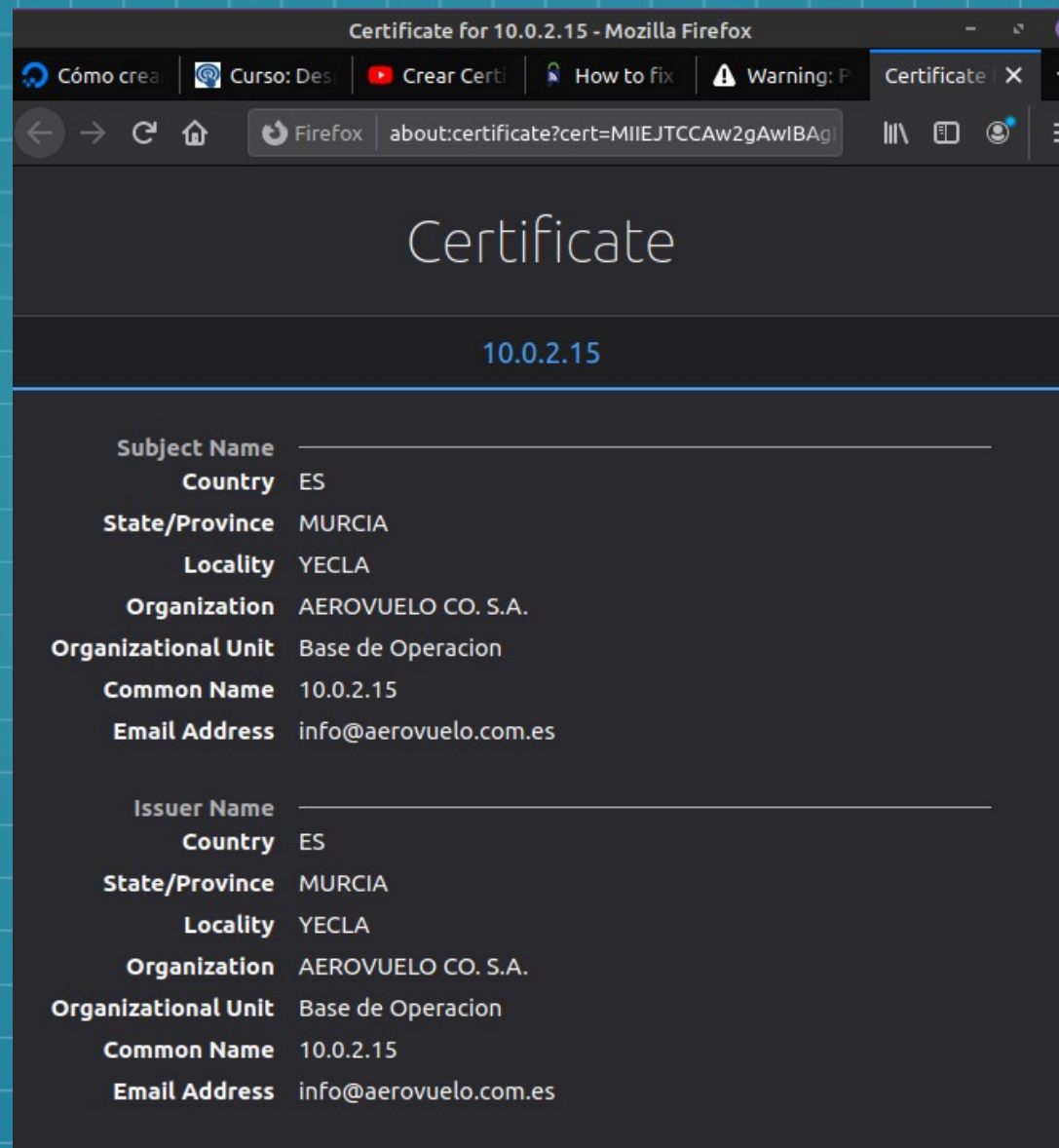

Se confirma
que la sintaxis
introducida sea
correcta.

```
cristian@cristian-VirtualBoxPC:~$ sudo apache2ctl configtest
AH00558: apache2: Could not reliably determine the server's fully qualified domain
name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this mes
sage
Syntax OK
```



Se evidencia que el
servidor está cargando
con el ssl, pero muestra
este resultado porque es
un certificado
autofirmado, certificado
no valido para los
navegadores web ya que
estos son firmados por
autoridades competentes
(de paga).

Se confirma el certificado creado visitando los ajustes avanzados





Conclusiones

1. La seguridad en la web es muy importante, ya que en el tráfico de datos, hay muchos que son sensibles (desde nombres completos, direcciones, hasta cuentas bancarias). Es importante que cualquier página que visitemos nos garantice una conexión segura.
2. Los tipos de cifrado han avanzado y se han desarrollado muy bien en los últimos años, cosa que favorece la seguridad en la red. Este cifrado al cubrir tanto el emisor como el receptor, hace casi imposible que intrusos puedan ver los datos que se envían y se reciben en una página web.
3. El uso de esta herramienta, principalmente, está orientada para todo e-commerce o página que necesite administrar y recibir información por parte del cliente, incluso con todo lo referido a transacciones bancarias.