Problem Statement : **Generation of self signed SSL certificate.**

Theory: SSL (Secure Sockets Layer) is a standard security technology for establishing an encrypted link between a server and a client—typically a web server (website) and a browser, or a mail server and a mail client (e.g., *Outlook*).
SSL allows sensitive information such as credit card numbers, social security numbers, and login credentials to be transmitted securely. Normally, data sent between browsers and web servers is sent in plain text—leaving you vulnerable to eavesdropping. If an attacker is able to intercept all data being sent between a browser and a web server, they can see and use that information.
This link ensures that all data passed between the web server and browsers remain private and integral.

Why SSL exists?

1. Encryption- Hiding what is sent from one computer to another computer.
2. Identification- Making sure the computer you are speaking to is the one you trust.

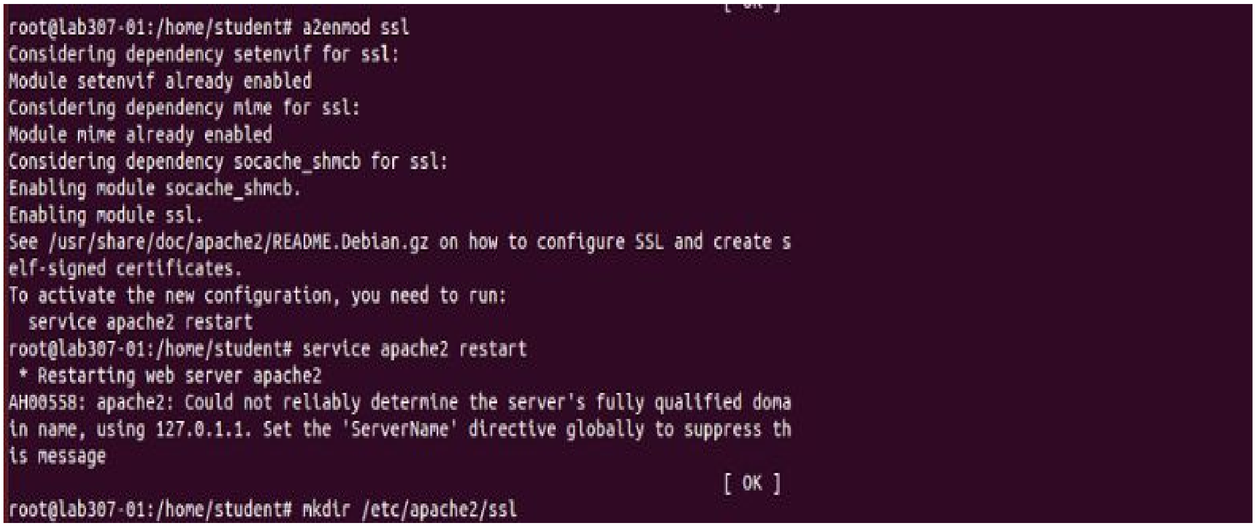
Why do I need SSL on my site?

1. The Internet has new global business opportunities for enterprises conducting online commerce.
2. However, that growth has also attracted fraudsters and cyber criminals who are ready to exploit any opportunity to steal consumer bank account numbers and card details.
3. Unless the connection between a client (e.g. internet browser) and a web server is encrypted, then any moderately skilled hacker can easily intercept and read the traffic.

How can I tell when my site uses SSL?

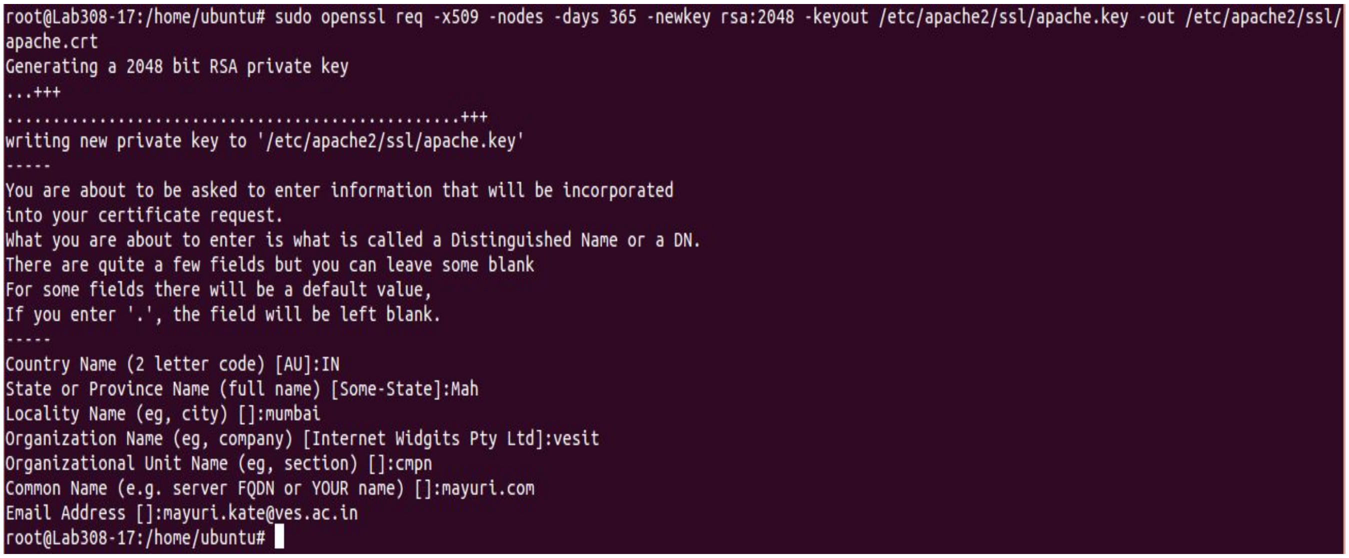
1. When a digital certificate is installed on a web page, users will see a padlock icon in the browser address bar. When an Extended Validation Certificates is installed on a web site, the address bar will turn green during secure sessions.
2. Users on sites with SSL Certificates will also see https:// in the address bar

Step 1: Activation of SSL Module:-



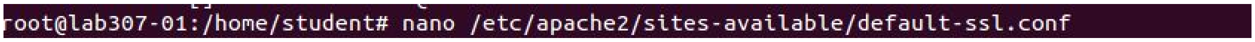
```
root@lab307-01:/home/student# a2enmod ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create s
elf-signed certificates.
To activate the new configuration, you need to run:
  service apache2 restart
root@lab307-01:/home/student# service apache2 restart
 * Restarting web server apache2
AH00558: apache2: Could not reliably determine the server's fully qualified doma
in name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress th
is message
                                                                         [ OK ]
root@lab307-01:/home/student# mkdir /etc/apache2/ssl
```

Step 2: Creation of Self-Signed SSL certificate

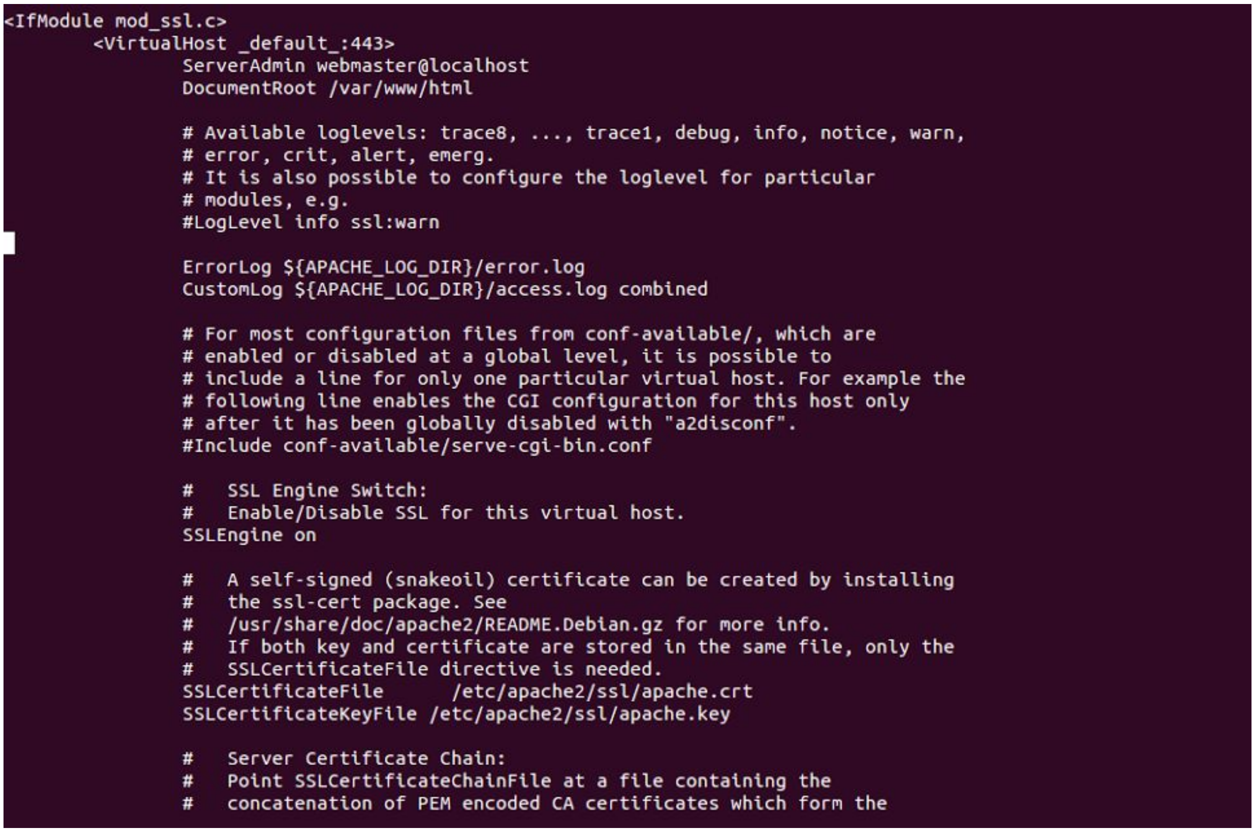


```
root@Lab308-17:/home/ubuntu# sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/apache2/ssl/apache.key -out /etc/apache2/ssl/
apache.crt
Generating a 2048 bit RSA private key
...+++
..............................................+++
writing new private key to '/etc/apache2/ssl/apache.key'
-----
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.
-----
Country Name (2 letter code) [AU]:IN
State or Province Name (full name) [Some-State]:Mah
Locality Name (eg, city) []:mumbai
Organization Name (eg, company) [Internet Widgits Pty Ltd]:vesit
Organizational Unit Name (eg, section) []:cmpn
Common Name (e.g. server FQDN or YOUR name) []:mayuri.com
Email Address []:mayuri.kate@ves.ac.in
root@Lab308-17:/home/ubuntu# 
```

Step 3: Setting up the certificate

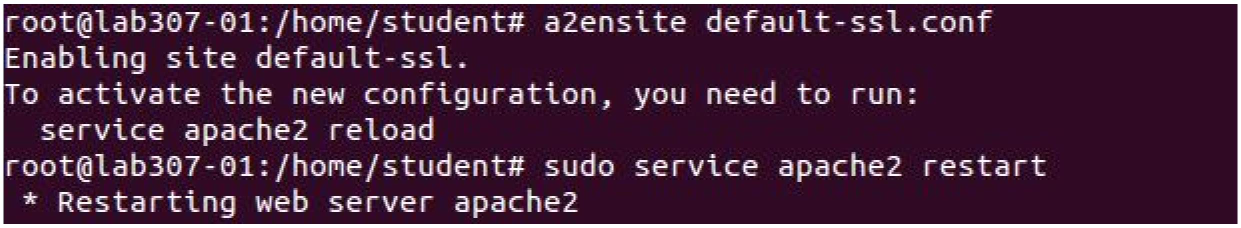


```
root@lab307-01:/home/student# nano /etc/apache2/sites-available/default-ssl.conf
```



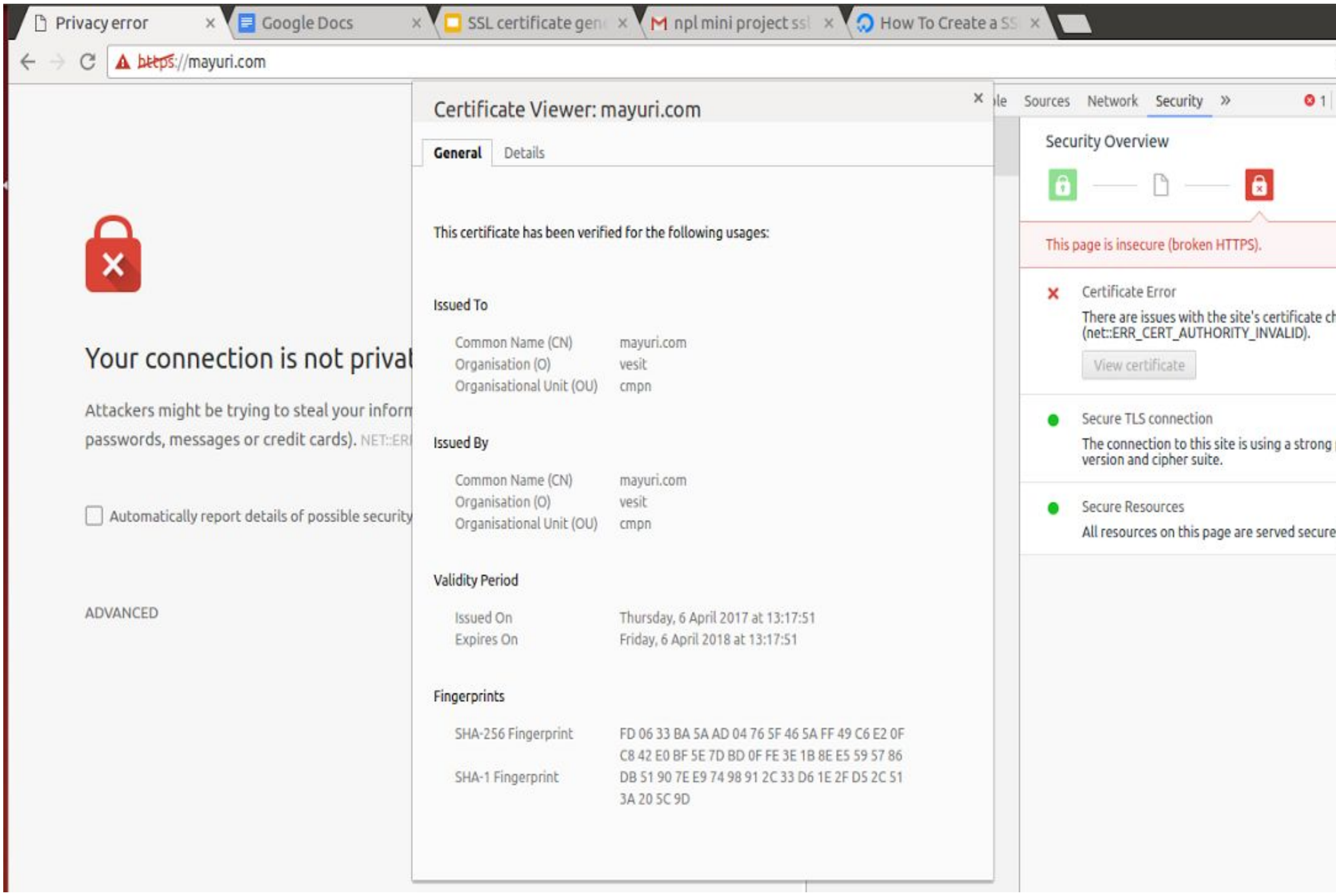
```
<IfModule mod_ssl.c>
        <VirtualHost _default_:443>
                ServerAdmin webmaster@localhost
                DocumentRoot /var/www/html

                # 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

                # For most configuration files from conf-available/, which are
                # enabled or disabled at a global level, it is possible to
                # include a line for only one particular virtual host. For example the
                # following line enables the CGI configuration for this host only
                # after it has been globally disabled with "a2disconf".
                #Include conf-available/serve-cgi-bin.conf

                #   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/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/apache2/ssl/apache.crt
                SSLCertificateKeyFile /etc/apache2/ssl/apache.key

                #   Server Certificate Chain:
                #   Point SSLCertificateChainFile at a file containing the
                #   concatenation of PEM encoded CA certificates which form the
```

Step 4: Activation of new Virtual Host



```
root@lab307-01:/home/student# a2ensite default-ssl.conf
Enabling site default-ssl.
To activate the new configuration, you need to run:
  service apache2 reload
root@lab307-01:/home/student# sudo service apache2 restart
 * Restarting web server apache2
```

Output:



Certificate Viewer: mayuri.com

General Details

This certificate has been verified for the following usages:

Issued To

Common Name (CN) mayuri.com
Organisation (O) vesit
Organisational Unit (OU) cmpn

Issued By

Common Name (CN) mayuri.com
Organisation (O) vesit
Organisational Unit (OU) cmpn

Validity Period

Issued On Thursday, 6 April 2017 at 13:17:51
Expires On Friday, 6 April 2018 at 13:17:51

Fingerprints

SHA-256 Fingerprint FD 06 33 BA 5A AD 04 76 5F 46 5A FF 49 C6 E2 0F
C8 42 E0 BF 5E 7D BD 0F FE 3E 1B 8E E5 59 57 86
SHA-1 Fingerprint DB 51 90 7E E9 74 98 91 2C 33 D6 1E 2F D5 2C 51
3A 20 5C 9D

Your connection is not privat

Attackers might be trying to steal your inform
passwords, messages or credit cards). NET::ERI

□ Automatically report details of possible security

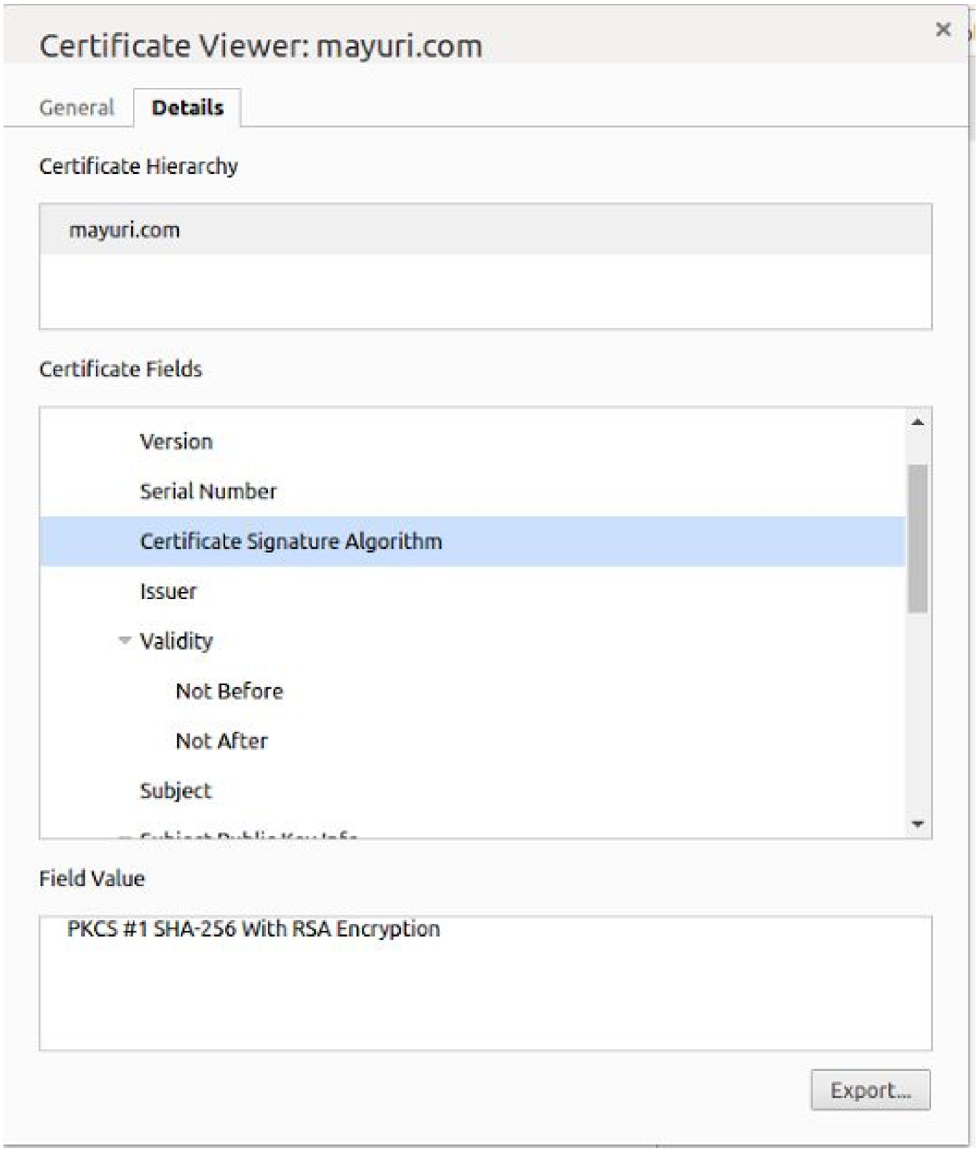
Security Overview

This page is insecure (broken HTTPS).

Certificate Error
There are issues with the site's certificate ch
(net::ERR_CERT_AUTHORITY_INVALID).

Secure TLS connection
The connection to this site is using a strong p
version and cipher suite.

Secure Resources
All resources on this page are served secure



Certificate Viewer: mayuri.com

Certificate Hierarchy

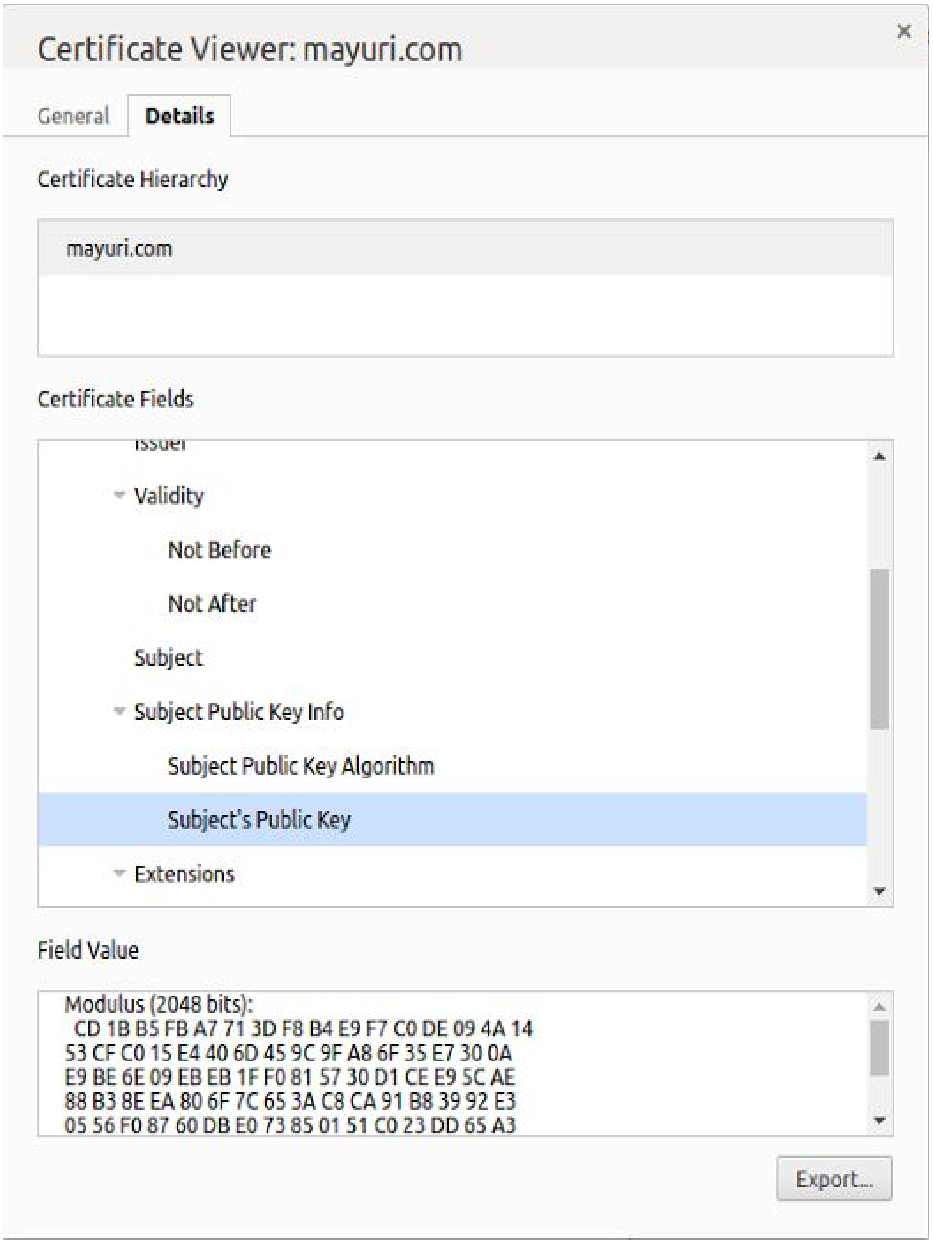
mayuri.com

Certificate Fields

```
Version
Serial Number
Certificate Signature Algorithm
Issuer
Validity
    Not Before
    Not After
Subject
Subject Public Key Info
```

Field Value

PKCS #1 SHA-256 With RSA Encryption



```
Modulus (2048 bits):
  CD 1B B5 FB A7 71 3D F8 B4 E9 F7 C0 DE 09 4A 14
53 CF C0 15 E4 40 6D 45 9C 9F A8 6F 35 E7 30 0A
E9 BE 6E 09 EB EB 1F F0 81 57 30 D1 CE E9 5C AE
88 B3 8E EA 80 6F 7C 65 3A C8 CA 91 B8 39 92 E3
05 56 F0 87 60 DB E0 73 85 01 51 C0 23 DD 65 A3
```