Server World   Privacy Policy   Support/Contact us   33399 / 114452916

Apache2 : Configure SSL/TLS 2022/05/10

Configure SSL/TLS setting to use secure encrypt HTTPS connection.

[1] Get SSL Certificate, refer to here.

[2] Enable SSL/TLS settings.

```
root@www:~#
vi /etc/apache2/sites-available/default-ssl.conf
# line 3 : change admin email

ServerAdmin
webmaster@srv.world
# line 32,33 : change to the certs gotten in section [1]


SSLCertificateFile        /etc/letsencrypt/live/www.srv.world/cert.pem
SSLCertificateKeyFile     /etc/letsencrypt/live/www.srv.world/privkey.pem


# line 42 : uncomment and change to the chain-file gotten in section [1]


SSLCertificateChainFile /etc/letsencrypt/live/www.srv.world/chain.pem


root@www:~#
a2ensite default-ssl


Enabling site default-ssl.
To activate the new configuration, you need to run:
  systemctl reload apache2


root@www:~#
a2enmod ssl


Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
```
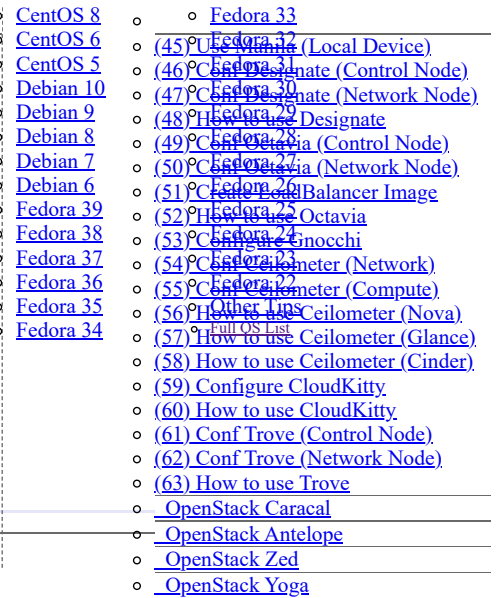
```
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-signed certificates.
To activate the new configuration, you need to run:
  systemctl restart apache2
```

root@www:~#
systemctl restart apache2

[3] If you'd like to set HTTP connection to redirect to HTTPS (Always on SSL/TLS), Set RewriteRule to each Host settings.
For example, if you set Virtual Hostings like the link here, Add RewriteRule like follows. Or It's possible to set RewriteRule in [.htaccess] not in [httpd.conf].

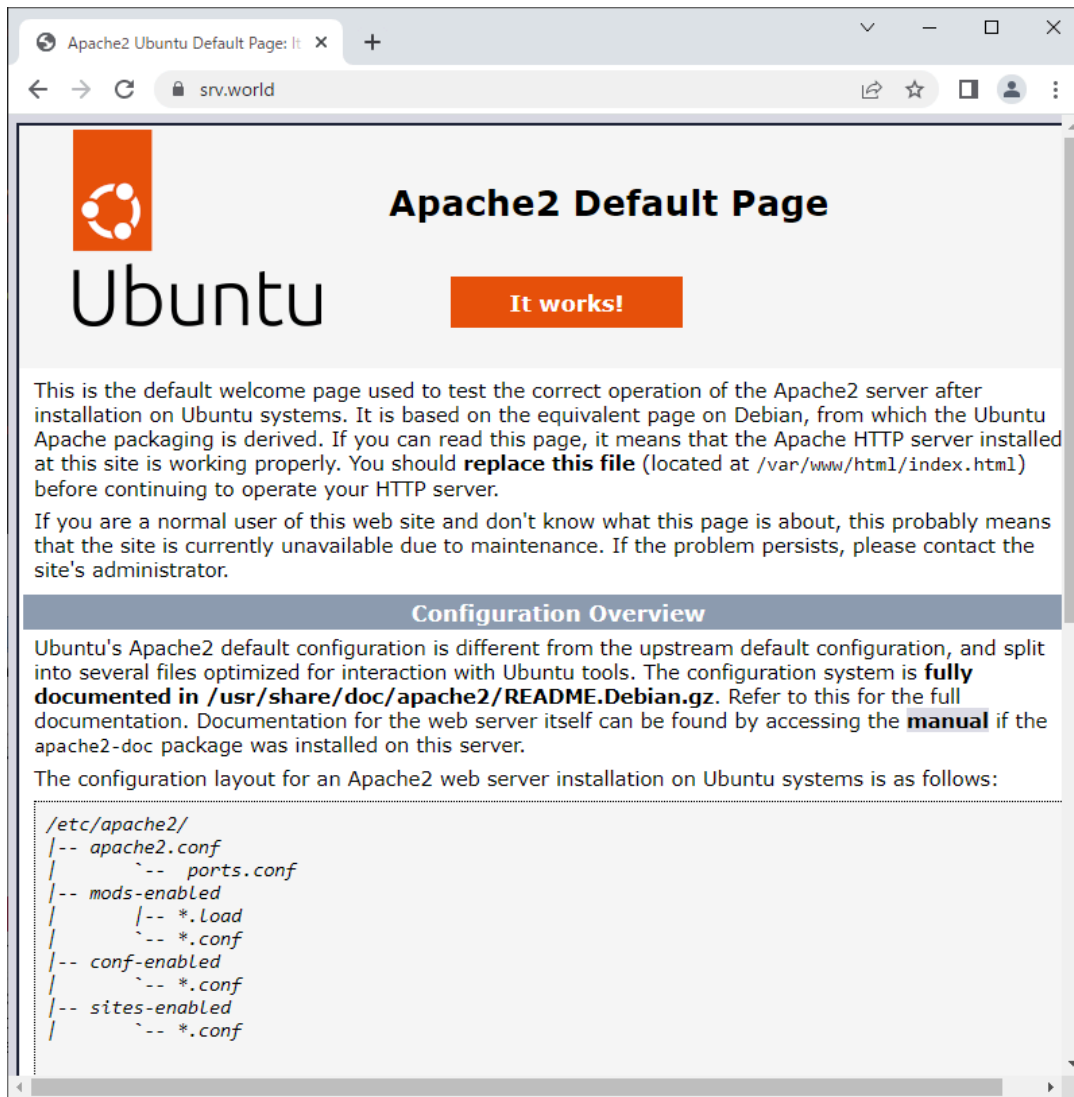root@www:~#
vi /etc/apache2/sites-available/virtual.host.conf

```
<VirtualHost *:80>
    DocumentRoot /var/www/virtual.host
    ServerName www.virtual.host
    ServerAdmin webmaster@virtual.host
    ErrorLog /var/log/apache2/virtual.host.error.log
    CustomLog /var/log/apache2/virtual.host.access.log combined
    RewriteEngine On
    RewriteCond %{HTTPS} off
    RewriteRule ^(.*)$ https://%{HTTP_HOST}%{REQUEST_URI} [R=301,L]
</VirtualHost>
```

root@www:~#
a2enmod rewrite

```
Enabling module rewrite.
To activate the new configuration, you need to run:
  systemctl restart apache2
```

root@www:~#
systemctl restart apache2

[4] Verify to access to the test page from any client computer with Web browser via HTTPS.

```
| Apache2 Ubuntu Default Page: It  X | +
|  ←  →  C    🔒 srv.world |
```

Apache2 Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. 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

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu 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 Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|       `--   ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
|       `-- *.conf
|-- sites-enabled
|       `-- *.conf
```

M a t c h e d   C o n t e n t

Ubuntu 22.04 : Apache2

- (01) Install Apache2
- (02) Configure Virtual Hostings
- (03) Configure SSL/TLS
- (04) Enable Userdir
- (05) Use CGI Scripts
- (06) Use PHP Scripts
- (07) PHP + PHP-FPM
- (08) Basic Authentication
- (09) Configure WebDAV Folder
- (10) Basic Authentication + PAM
- (11) Kerberos Authentication
- (12) Configure mod_md
- (13) Configure mod_proxy
- (14) Configure mod_security
- (15) Configure mod_ratelimit
- [Next Page]

Sponsored Link