Apache

en.wikibooks.org
On the 28th of April 2012 the contents of the English as well as German Wikibooks and Wikipedia projects were licensed under Creative Commons Attribution-ShareAlike 3.0 Unported license. A URI to this license is given in the list of figures on page 39. If this document is a derived work from the contents of one of these projects and the content was still licensed by the project under this license at the time of derivation this document has to be licensed under the same, a similar or a compatible license, as stated in section 4b of the license. The list of contributors is included in chapter Contributors on page 37. The licenses GPL, LGPL and GFDL are included in chapter Licenses on page 43, since this book and/or parts of it may or may not be licensed under one or more of these licenses, and thus require inclusion of these licenses. The licenses of the figures are given in the list of figures on page 39. This PDF was generated by the \LaTeX typesetting software. The \LaTeX source code is included as an attachment (source.7z.txt) in this PDF file. To extract the source from the PDF file, you can use the pdfdetach tool including in the poppler suite, or the http://www.pdflabs.com/tools/pdftk-the-pdf-toolkit/ utility. Some PDF viewers may also let you save the attachment to a file. After extracting it from the PDF file you have to rename it to source.7z. To uncompress the resulting archive we recommend the use of http://www.7-zip.org/. The \LaTeX source itself was generated by a program written by Dirk Hünniger, which is freely available under an open source license from http://de.wikibooks.org/wiki/Benutzer:Dirk_Huenniger/wb2pdf.
# Contents

1 Introduction ................................................. 3
   1.1 History and name ........................................ 4
   1.2 License .................................................. 5
   1.3 References ............................................... 5

2 Architecture ............................................... 7
   2.1 Features ............................................... 7

3 Installation ............................................... 9
   3.1 Windows ................................................ 9
   3.2 Unix-like ............................................. 17

4 Configuration ............................................. 19
   4.1 Performance ............................................ 20

5 Virtual hosting ........................................... 21
   5.1 Principle .............................................. 21
   5.2 Configuration ......................................... 21
   5.3 References ............................................ 21

6 SSL ........................................................... 23
   6.1 Required Packages ..................................... 23
   6.2 Configuration Steps ................................... 23
   6.3 Example ................................................ 24

7 .htaccess ................................................... 27
   7.1 Principle .............................................. 27
   7.2 Installation ........................................... 27
   7.3 References ............................................ 27

8 Security ..................................................... 29
   8.1 Protection by provenance ................................. 29
   8.2 Protection by password .................................. 30
   8.3 References ............................................ 31

9 CGI ............................................................. 33
   9.1 CGI scripts ............................................. 33
   9.2 References ............................................ 36

10 Contributors ............................................... 37

List of Figures .............................................. 39

III
## 11 Licenses

11.1 GNU GENERAL PUBLIC LICENSE ............................................. 43  
11.2 GNU Free Documentation License ........................................ 44  
11.3 GNU Lesser General Public License ..................................... 45  

1 Introduction

Apache is primarily used to serve both static content and dynamic Web pages on the World Wide Web. Many web applications are designed expecting the environment and features that Apache provides.

Apache is redistributed as part of various proprietary software packages including the Oracle Database and the IBM WebSphere application server. Mac OS X integrates Apache as its built-in web server and as support for its WebObjects application server. It is also supported in some way by Borland in the Kylix and Delphi development tools. Apache is included with Novell NetWare 6.5, where it is the default web server. Apache is included with many Linux distributions.

Apache is used for many other tasks where content needs to be made available in a secure and reliable way. One example is sharing files from a personal computer over the Internet. A user who has Apache installed on their desktop can put arbitrary files in Apache's document root which can then be shared.

Programmers developing web applications often use a locally installed version of Apache in order to preview and test code as it is being developed.
Microsoft Internet Information Services\(^1\) (IIS) is the main competitor to Apache, followed by Sun Microsystems' Sun Java System Web Server and a host of other applications such as Zeus Web Server or Nginx.

1.1 History and name

The first version of the Apache web server software was created by Robert McCool, who was heavily involved with the National Center for Supercomputing Applications web server, known simply as NCSA HTTPd. When McCool left NCSA in mid-1994, the development of httpd stalled, leaving a variety of patches for improvements circulating through e-mails. These patches were provided by a number of other developers besides McCool, and they thus helped to form the original "Apache Group".

There have been two explanations of the project's name. According to the Apache Foundation, the name was chosen out of respect for the Native American tribe of Apache\(^2\) (Indé), well-known for their endurance and their skills in warfare. However, the original FAQ on the Apache Server project's website, from 1996 to 2001, claimed that "The result after combining [the NCSA httpd patches] was a patchy server. The first explanation was supported at an Apache Conference and in an interview in 2000 by Brian Behlendorf, who said that the name connoted "Take no prisoners. Be kind of aggressive and kick some ass". Behlendorf then contradicted this in a 2007 interview, stating that "The Apache server isn't named in honor of Geronimo's tribe” but that so many revisions were sent in that "the group called it 'a patchy Web server'”. Both explanations are probably appropriate.

Version 2 of the Apache server was a substantial re-write of much of the Apache 1.x code, with a strong focus on further modularization and the development of a portability layer, the Apache Portable Runtime\(^3\). The Apache 2.x core has several major enhancements over Apache 1.x. These include UNIX threading, better support for non-Unix platforms (such as Microsoft Windows), a new Apache API, and IPv6\(^4\) support. The first alpha release of Apache 2 was in March 2000, with the first general availability release on April 6, 2002.

Version 2.2 introduced a more flexible authorization API. It also features improved cache modules and proxy modules.

---

\(^1\) https://en.wikipedia.org/wiki/Internet_Information_Services
\(^2\) https://en.wikipedia.org/wiki/Apache
\(^3\) https://en.wikibooks.org/w/index.php?title=Apache_Portable_Runtime&action=edit&redlink=1
1.2 License

The software license under which software from the Apache Foundation is distributed is a distinctive part of the Apache HTTP Server's history and presence in the open source software community. The Apache License allows for the distribution of both open and closed source derivatives of the source code.

The Free Software Foundation does not consider the Apache License to be compatible with version 2 of the GPL in that software licensed under the Apache License cannot be integrated with software that is distributed under the GPL:

This is a free software license but it is incompatible with the GPL. The Apache Software License is incompatible with the GPL because it has a specific requirement that is not in the GPL: it has certain patent termination cases that the GPL does not require. We don't think those patent termination cases are inherently a bad idea, but nonetheless they are incompatible with the GNU GPL.[1]

However, version 3 of the GPL includes a provision (Section 7e) which allows it to be compatible with licenses that have patent retaliation clauses, including the Apache License.

The name Apache is a registered trademark and may only be used with the trademark holder's express permission.

1.3 References


[^1]: [http://www.gnu.org/philosophy/license-list.html](http://www.gnu.org/philosophy/license-list.html)
2 Architecture

2.1 Features

Apache supports a variety of features, many implemented as compiled modules which extend the core functionality. These can range from server-side programming language support to authentication schemes. Some common language interfaces support Perl\(^1\), Python\(^2\), Tcl\(^3\), and PHP\(^4\). Popular authentication modules include mod access\(^5\), mod auth\(^6\), mod digest\(^7\), and mod auth digest\(^8\), the successor to mod_digest. A sample of other features include SSL\(^9\) and TLS\(^10\) support (mod ssl\(^11\)), a proxy module (mod proxy\(^12\)), a URL rewriter (implemented under mod rewrite\(^13\)), custom log files (mod log config\(^14\)), and filtering support (mod include\(^15\) and mod ext filter\(^16\)).

Popular compression methods on Apache include the external extension module, mod gzip\(^17\), implemented to help with reduction of the size of web pages served over HTTP. ModSecurity is an open source intrusion detection and prevention engine for web applications. Apache logs can be analyzed through a web browser using free scripts such as AWStats/W3Perl or Visitors.

\(^{3}\)https://en.wikibooks.org/w/index.php?title=Tcl&action=edit&redlink=1
\(^{4}\)https://en.wikibooks.org/wiki/PHP
\(^{15}\)https://en.wikibooks.org/w/index.php?title=User:Dirk_Hünniger/mod_include&action=edit&redlink=1
Virtual hosting\textsuperscript{18} allows one Apache installation to serve many different actual websites. For example, one machine with one Apache installation could simultaneously serve www.example.com, www.test.com, test47.test-server.test.com, etc.

Apache features configurable error messages, DBMS-based authentication databases, and content negotiation. It is also supported by several graphical user interfaces.
3 Installation

3.1 Windows

The program can be installed alone from \(^1\).

However the best is generally to get it with an all-in-one solution like WAMP\(^2\) or EasyPHP\(^3\), because it includes the installation or PHP\(^4\) and MySQL\(^5\) at the same time.

Apart from that, a portable all-in-one is XAMPP\(^6\).

![Installation Wizard](http://www.apache.org/dyn/closer.cgi)

Figure 2

---

1 http://www.apache.org/dyn/closer.cgi
2 https://en.wikipedia.org/wiki/WAMP
3 https://en.wikipedia.org/wiki/EasyPHP
4 https://en.wikibooks.org/wiki/PHP
5 https://en.wikibooks.org/wiki/MySQL
6 https://en.wikipedia.org/wiki/XAMPP
Figure 3
Figure 4

Apache HTTP Server

What is it?
The Apache HTTP Server is a powerful and flexible HTTP/1.1 compliant web server. Originally designed as a replacement for the NCSA HTTP Server, it has grown to be the most popular web server on the Internet. As a project of the Apache Software Foundation, the developers aim to collaboratively develop and maintain a robust, commercial-grade, standards-based server with freely available source code.

The Latest Version
Details of the latest version can be found on the Apache HTTP server project page under:

http://httpd.apache.org/
Figure 5
Setup Type

Choose the setup type that best suits your needs.

- **Typical**
  Typical program features will be installed. (Headers and Libraries for compiling modules will not be installed.)

- **Custom**
  Choose which program features you want installed and where they will be installed. Recommended for advanced users.

Figure 6
Installation

Figure 7
Figure 8
Installation

Figure 9

[Image of an installation wizard for Apache HTTP Server 2.2 showing the destination folder set to C:\Apache\]
Remark (on a PC):

By default Apache is set to automatically launch at startup which is convenient on a server. However on a PC, it may be better to create a script to start and stop it manually. Example with EasyPHP:

```bash
net start ews-dbserver
net start ews-httpserver
net start ews-dashboard
pause
net stop ews-dashboard
net stop ews-httpserver
net stop ews-dbserver
```

### 3.2 Unix-like

#### 3.2.1 Debian

```bash
apt-get install apache2
```
3.2.2 Gentoo
emerge apache

3.2.3 Configuration
The service should be restarted manually after the most part of the configuration modifications:

vim /etc/httpd/conf/httpd.conf
#or
vim /etc/apache2/apache2.conf
/etc/init.d/apache2 restart
4 Configuration

Configuring Apache WebServer

GUI Tools:

There are some GUI tools to configure Apache web server but they mostly do the basic or intermediate levels of configurations. For advanced configuration, modifying the configuration file is the way to go. Some GUI tools are: Comanche, TkApache, LinuxConf, WebMin, ApacheConf, user_manage. Among them, Webmin is browser based and may be a better choice than the others.

Basic Apache Configuration Information is as below:

Apache has three configuration files:

httpd.conf
access.conf
srm.conf

httpd.conf is the main file, you can forget about the others.

A basic configuration file may look like the following:

```
ServerName www.justetc.net  # host name ... serving website URL
listen 192.168.0.1:80  # listen to the network interface 192.168.0.1 and port 80
listen 192.168.0.1:400  # listen to the network interface 192.168.0.1 and port 400
User nobody  # connect to the apache server as the user nobody # default
Group nobody
ServerAdmin admin@justetc.net # email of the administrator
ServerRoot /usr/locale/apache # apache's main working folder, apache keeps essential files here
ErrorLog logs/error_log # to keep track of errors
TransferLog logs/access_log # check the access, may make website slower
DocumentRoot /home/www/justetc
```

You can create your own user and set that to access webpages

```
groupadd -g httpd
useradd -u 999 -g httpd -s /bin/false -c 'Web Server'
```

then use

```
User httpd
Group httpd
```
4.1 Performance

Although the main design goal of Apache is not to be the "fastest" web server, Apache does have performance comparable to other "high-performance" web servers. Instead of implementing a single architecture, Apache provides a variety of MultiProcessing Modules (MPMs) which allow Apache to run in a process-based, hybrid (process and thread) or event-hybrid mode, to better match the demands of each particular infrastructure. This implies that the choice of correct MPM and the correct configuration is important. Where compromises in performance need to be made, the design of Apache is to reduce latency and increase throughput, relative to simply handling more requests, thus ensuring consistent and reliable processing of requests within reasonable time-frames.
5 Virtual hosting

5.1 Principle

Virtual hosting in Apache allows a single instance of the Apache software to host more than one distinct web site. Since only one instance of Apache can use port 80 on a server at once, without virtual hosting each new web site you hosted would need a new server.

Sites hosted under virtual hosting appear to end users as normal sites in every way; the only difference is that fewer server machines are needed to host them. Virtual hosting is commonly used by high-volume low-price web hosting companies, who give each of their customers a virtual host, with potentially hundreds of customers sharing the same server.

5.2 Configuration

To add a virtual host, edit apache2.conf (or httpd.conf). Example:[1]

```
<VirtualHost MyIP:80>
  ServerAdmin admin@site1.com
  DocumentRoot /home/site1/public_html
  ServerName site1.com
  ServerAlias www.site1.com
</VirtualHost>

<VirtualHost MyIP:80>
  ServerAdmin admin@site2.com
  DocumentRoot /home/site2/public_html
  ServerName site2.com
  ServerAlias www.site2.com
  AccessLog /home/site2/access.log
  ErrorLog /home/site2/error.log
  <Directory /home/site2/public_html>
    AllowOverride All
  </Directory>
</VirtualHost>
```

5.3 References

1. [1]

---

6 SSL

Install and Configure Apache2 with PHP5 and SSL Support

6.1 Required Packages

apache2
openssl ssl-cert
libapache2-mod-php5
php5-cli
php5-common
php5-cgi

6.2 Configuration Steps

6.2.1 Step 1: generate certificate

For generating certificate Use the following command to generate certificates

```
sudo openssl req $@ -new -x509 -days 365 -nodes -out /etc/apache2/apache.pem -
keyout /etc/apache2/apache.pem
```

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]:West Bengal
Locality Name (eg, city) []:Kolkata
Organization Name (eg, company) [Internet Widgits Pty Ltd]:MAT3 Impex Pvt. Ltd.
Organizational Unit Name (eg, section) []:Crypto-Devel
Common Name (eg, YOUR name) []:Promathesh Mandal
Email Address []:promathesh@mat3impex.com

This will complete the certificate now you need to make sure you have the correct permissions for .pem file if not use the following command to set the correct permissions

```
sudo chmod 600 /etc/apache2/apache.pem
```

Note: For generating a certificate signing request give the following command

```
sudo openssl req -new -key apache.pem -out chikpea.csr
```
6.2.2 Step 2: listen the port

By default the server will listen for incoming HTTP requests on port 80 - and not SSL connections on port 443. So you need to enable SSL support by entering the following entry to the file `/etc/apache2/ports.conf` save and exit the file.

\['\textbf{Listen 443}\]

6.2.3 Step 3: enable SSL support

If you want to enable SSL support for your apache web server you need to use the following command

\['\textbf{sudo a2enmod ssl}\]

Module ssl installed; run `/etc/init.d/apache2 force-reload` to enable. Now you need to restart the apache2 server using the following command

\['\textbf{sudo /etc/init.d/apache2 restart}\]

6.2.4 Step 4: configuring SSL Certificate to Virtual Hosts in Apache2

First you need to edit the `/etc/apache2/sites-available/default` file change

\['\textbf{NameVirtualHost \*}\]

to

\['\textbf{NameVirtualHost \*:80} \\
\textbf{NameVirtualHost \*:443}\]

Now you need to configure Virtual hosts using port 80.

6.3 Example

\['\textbf{ServerAdmin webmaster@localhost} \\
\textbf{.} \\
\textbf{.} \\
\textbf{.}\]

configure Virtual hosts using port 443 the main difference is you need to use the following two lines for each SSL hosts.

\['\textbf{SSLEngine on} \\
\textbf{SSLCertificateFile /etc/apache2/apache.pem}\]

Example

\['\textbf{ServerAdmin webmaster@localhost} \\
\textbf{.} \\
\textbf{.}\]
Example

```
SSLEngine on
SSLCertificateFile /etc/apache2/apache.pem
```

Now you need to restart your apache web server using the following command

```
sudo /etc/init.d/apache2 reload
```

Sample Files: sample for “ports.conf” file

```
Listen 80
Listen 443
```

Sample for “default” file

```
NameVirtualHost *:80
NameVirtualHost *:443
<VirtualHost *:80>
  DocumentRoot /var/www/
  <Directory />
    Options FollowSymLinks
    AllowOverride None
  </Directory>
  <Directory /var/www/>
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from all
    # This directive allows us to have apache2’s default start page
    # in /apache2-default/, but still have / go to the right place
    #RedirectMatch ^/$ /apache2-default/
  </Directory>
  ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
  <Directory /usr/lib/cgi-bin>
    AllowOverride None
    Options ExecCGI -MultiViews +SymLinksIfOwnerMatch
    Order allow,deny
    Allow from all
  </Directory>
  ErrorLog /var/log/apache2/error.log
  # Possible values include: debug, info, notice, warn, error, crit,
  # alert, emerg.
  LogLevel warn
  CustomLog /var/log/apache2/access.log combined
  ServerSignature On
  Alias /doc/ "/usr/share/doc/"
  <Directory "/usr/share/doc/">
    Options Indexes MultiViews FollowSymLinks
    AllowOverride None
    Order deny,allow
    Deny from all
    Allow from 127.0.0.0/255.0.0.0 ::1/128
  </Directory>
</VirtualHost>
<VirtualHost *:443>
  ServerAdmin webmaster@localhost
  SSLEngine on
  SSLCertificateFile /etc/apache2/apache.pem
  DocumentRoot /var/www/
  <Directory />
    Options FollowSymLinks
    AllowOverride None
  </Directory>
</VirtualHost>
```
<Directory /var/www/>
  Options Indexes FollowSymLinks MultiViews
  AllowOverride None
  Order allow,deny
  allow from all
  # This directive allows us to have apache2's default start page
  # in /apache2-default/, but still have / go to the right place
  #RedirectMatch ^/$ /apache2-default/
</Directory>
ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
<Directory "/usr/lib/cgi-bin”>
  AllowOverride None
  Options ExecCGI -MultiViews +SymLinksIfOwnerMatch
  Order allow,deny
  Allow from all
</Directory>
ErrorLog /var/log/apache2/error.log
# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
LogLevel warn
CustomLog /var/log/apache2/access.log combined
ServerSignature On
Alias /doc/ "/usr/share/doc/"
<Directory "/usr/share/doc/”>
  Options Indexes MultiViews FollowSymLinks
  AllowOverride None
  Order deny,allow
  Deny from all
  Allow from 127.0.0.0/255.0.0.0 ::1/128
</Directory>
</VirtualHost>
7 .htaccess

7.1 Principle

In order to protect a directory in particular (and its subdirectories), it suffices to place a file called .htaccess inside. Apache will instantly apply its rules after, only in this tree structure. The syntax is the same as the general vhost rules (eg: URL rewriting or protection), unless it will only affect the .htaccess directory (so no Directory clause).

Attention: the Windows explorer doesn't allow to name some files beginning by a dot, but a text editor is able to save as .htaccess.

7.2 Installation

To authorize the .htaccess in the site .conf, use AllowOverride:

AllowOverride All

To forbid them:

AllowOverride None

7.3 References

2. [http://httpd.apache.org/docs/2.2/mod/core.html#allowoverride](http://httpd.apache.org/docs/2.2/mod/core.html#allowoverride)
8 Security

Numerous robots use to try to crack some databases (for instance via PhpMyAdmin\(^1\) or WordPress\(^2\)). To be protected from them, we can play on several criteria.

For example, to forbid to visualize a directory files which hasn't got any index (e.g.: .html, .php), add the code: `Options -Indexes`.

8.1 Protection by provenance

Authorize only two IP to read the directory:

8.1.1 whitelist with Require\(^{[1]}\)

```
<Directory /usr/share/phpmyadmin/>
  <IfModule mod_authz_core.c>
    <RequireAny>
      Require all denied
      Require ip 127.0.0.1
      Require ip 127.0.0.2
    </RequireAny>
  </IfModule>
</Directory>
```

8.1.2 whitelist with allow (obsolete in Apache 2.4)

```
<Directory /usr/share/phpmyadmin/>
  <IfModule mod_access_compat.c>
    deny from all
    allow from 127.0.0.1
    allow from 127.0.0.2
  </IfModule>
</Directory>
```

If the authorization ranges have some addresses in common with the prohibited ranges, it's better to specify their precedence (the lines order in the .htaccess file doesn't change anything):

**order allow, deny**

begin by the authorizations and then start the interdictions, by risking to ban what was previously allowed.

**order deny, allow**

---

1 https://en.wikipedia.org/wiki/PhpMyAdmin
the contrary is less restrictive.

8.1.3 blacklist avec Require

```html
<Directory /usr/share/phpmyadmin/
  <IfModule mod_authz_core.c>
    Require all granted
    Require not ip 127.0.0.1
  </IfModule>
</Directory>
```

8.1.4 blacklist avec deny (obsolete in Apache 2.4)

```html
<Directory /usr/share/phpmyadmin/
  <IfModule mod_access_compat.c>
    order allow,deny
    allow from all
    deny from 127.0.0.1
  </IfModule>
</Directory>
```

8.2 Protection by password

8.2.1 Authentication configuration

It's imperative to allow the authentication parameters modifications in the Apache settings. The directive AllowOverride of a parent directory must contain the option `AuthConfig`.

The directives to place in the `.htaccess` are:

`AuthType basic`

authentication type communally adopted but poorly secured.

`AuthName "My message"`

the text as an invite in the dialog box.

`AuthUserFile /etc/apache2/my_passwd`

the passwords file path.

`Require valid-user`

specifies that a valid account is needed to accede to the folder.

We can also use `Require user toto sasa` to authorize only the two accounts toto & sasa.

The authentication type `basic` uses not crypted passwords.

Some other more secured types exist, like `digest`, which is recommended to combine with HTTPS.

The first request is addressed to the protected directory and provokes the displaying of the dialog box, from which the user should identify (with login and password):

> [3](https://en.wikibooks.org/w/index.php?title=User:Dirk_H%C3%BCnniger/SSL&action=edit&redlink=1)
• If the password is invalid, the dialog will be displayed again.
• If it's valid, the navigator can record it, and never ask it again until the next relaunching.

8.2.2 Passwords file

The following command creates a passwords file called with one user toto:

```
htpasswd -c /home/user/www/.htpasswd toto
```

To add or modify a user:

```
htpasswd /home/user/www/.htpasswd sasa
```

Then, tell to .htaccess the .htpasswd path with:

```
AuthName "Protected page"
AuthType Basic
AuthUserFile "/home/user/www/.htpasswd"
Require valid-user
```

8.3 References

1. 4
2. 5
3. 6

---

4 https://httpd.apache.org/docs/2.4/en/howto/access.html
5 https://httpd.apache.org/docs/2.4/en/upgrading.html
6 http://httpd.apache.org/docs/2.2/mod/core.html#allowoverride
9 CGI

9.1 CGI scripts

The CGI (Common Gateway Interface) is a norm permitting Apache to execute some programs, which can be written in any programming language (Bash, C, Java, Perl, PHP, Python...), from the moment it's executable and it respects certain in/out constraints.

9.1.1 Configure the CGI scripts access

To make Apache interpret the scripts, it's necessary to do a minimum of settings in the site configuration.

**ScriptAlias**

The directive (from `httpd.conf`):

```
ScriptAlias /cgi-bin/ ''/scripts path''
```

precise the folder name where Apache is authorized to execute the CGI scripts.[1]

Unix example:

```
ScriptAlias /cgi-bin/ /var/www/cgi-bin
```

Windows example, use the URL format (no backslash):

```
ScriptAlias /cgi-bin/ "C:/wamp/bin/apache/apache2.2.27/cgi-bin/
```

Actually the path `/cgi-bin/` doesn't really exist, it's redirected to the scripts path, set by the directive, and it allows to write some URL like `http://server/cgi-bin/my_script`.

**ExecCGI**

The following clause activates the option `ExecCGI` in `/var/www/cgi-bin`, which authorize Apache to execute some scripts on the server:

---

For example, if a script is called `essai.cgi` into `/home/httpd/cgi-bin`:

```
<Directory /home/httpd/cgi-bin>
  Options ExecCGI
</Directory>
```

Then, call the URL: `http://serveur/cgi-bin/essai.cgi`

**AddHandler**

This clause permits to choose the files extensions which will be authorized, eg:

```
AddHandler cgi-script .cgi .exe .pl .py .vbs
```

**Recapitulation**

Full example on Windows, in the Apache configuration:

```
ScriptAlias /cgi-bin/ "E:/www/cgi-bin/*
<Directory "E:/www/cgi-bin/*/"
  Options FollowSymLinks Indexes
  AllowOverride All
  Order deny,allow
  Allow from all
  Require all granted
</Directory>
```

In `E:/www/cgi-bin/.htaccess`:

```
AddHandler cgi-script .cgi .exe .pl .py .vbs
```

**9.1.2 Write a CGI program**

The main constraint concerns the program outputs. If a CGI script generates some data on its standard output, he must display an HTTP header before, allowing to identify them.

**Bash**

```
#!/bin/bash

# Header
echo "Content-type: text/html"

# Header end
echo ""

echo "<html><body>Hello World!</body></html>"
```
This script generates an HTML page.

**Perl**

```perl
#!/c:/perl/perl/bin/perl.exe -w
use CGI;
my $query = new CGI;
my $Name = $query->param('Name');
print $query->header();
print "Hello World!"
```

**Python**

```python
#!/C:\Program Files (x86)\Python\python.exe
# -*- coding: UTF-8 -*-
print "Content-Type: text/plain; charset=utf-8"
print
print "Hello World!"
```

**VBS**

For Windows.[2]

```vbs
'!c:/windows/system32/cscript //nologo
Wscript.Echo "Content-type: text/html" & vbLF & vbLF
WScript.Echo "Hello World!"
Wscript.Quit 0
```

### 9.1.3 Known errors

- **Error 500** *Server error!: replace a **Deny** from all by a **Allow** from all.

  or

  # setsebool -P httpd_enable_cgi 1
  # chcon -R -t httpd_sys_script_exec_t cgi-bin/your_script.cgi

- **Error 403** *Forbidden access*: impossible to list this folder, so call directly its files.

- If the file source code is appearing in the navigator: the `.htaccess` is not properly set.

- **couldn’t create child process**: replace the path after shebang\(^8\). For example:

  ```bash
  #!/usr/bin/perl par #!/c:/perl/perl/bin/perl.exe -w.
  #!/usr/bin/env python par #!/C:\Program Files (x86)\Python\python.exe.
  ```

- **End of script output before headers**: missing header (eg: move the importation before print "Content-Type: text/plain; charset=utf-8"). But it can also be the symptom of a compilation error in the script language.

- **malformed header from script**: *Bad header*: the header is not adapted (eg: replace `# print "Content-Type: text/plain; charset=utf-8"` by `print "Content-type: text/html\n\n" if there is a print "<html>" after).

Otherwise consult the Apache logs...

---

\(^8\) [https://en.wiktionary.org/wiki/shebang](https://en.wiktionary.org/wiki/shebang)
9.2 References


• This page was last edited on 26 June 2024, at 11:44.
• Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy.
## 10 Contributors

<table>
<thead>
<tr>
<th>Edits</th>
<th>User</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Adrignola¹</td>
</tr>
<tr>
<td>1</td>
<td>Cic²</td>
</tr>
<tr>
<td>16</td>
<td>DannyS712³</td>
</tr>
<tr>
<td>1</td>
<td>Dirk Hünniger⁴</td>
</tr>
<tr>
<td>2</td>
<td>Fishpi⁵</td>
</tr>
<tr>
<td>58</td>
<td>JackPotte⁶</td>
</tr>
<tr>
<td>1</td>
<td>KonstantinaG07⁷</td>
</tr>
<tr>
<td>4</td>
<td>MarcGarver⁸</td>
</tr>
<tr>
<td>10</td>
<td>Mike.lifeguard⁹</td>
</tr>
<tr>
<td>1</td>
<td>Minorax¹⁰</td>
</tr>
<tr>
<td>5</td>
<td>PokestarFan¹¹</td>
</tr>
<tr>
<td>1</td>
<td>Ridernyc¹²</td>
</tr>
<tr>
<td>1</td>
<td>Thenub314¹³</td>
</tr>
</tbody>
</table>

List of Figures

• cc-by-sa-4.0: Creative Commons Attribution ShareAlike 4.0 License. https://creativecommons.org/licenses/by-sa/4.0/deed.en
• cc-by-sa-3.0: Creative Commons Attribution ShareAlike 3.0 License. http://creativecommons.org/licenses/by-sa/3.0/
• cc-by-sa-2.5: Creative Commons Attribution ShareAlike 2.5 License. http://creativecommons.org/licenses/by-sa/2.5/
• cc-by-sa-2.0: Creative Commons Attribution ShareAlike 2.0 License. http://creativecommons.org/licenses/by-sa/2.0/
• cc-by-sa-1.0: Creative Commons Attribution ShareAlike 1.0 License. http://creativecommons.org/licenses/by-sa/1.0/
• cc-by-1.0: Creative Commons Attribution 1.0 License. https://creativecommons.org/licenses/by/1.0/deed.en
• cc-by-2.0: Creative Commons Attribution 2.0 License. http://creativecommons.org/licenses/by/2.0/
• cc-by-2.5: Creative Commons Attribution 2.5 License. http://creativecommons.org/licenses/by/2.5/deed.en
• cc-by-3.0: Creative Commons Attribution 3.0 License. http://creativecommons.org/licenses/by/3.0/deed.en
• cc-by-4.0: Creative Commons Attribution 4.0 License. https://creativecommons.org/licenses/by/4.0/deed.de
• GPL: GNU General Public License. http://www.gnu.org/licenses/gpl-2.0.txt
• LGPL: GNU Lesser General Public License. http://www.gnu.org/licenses/lgpl.html
• PD: This image is in the public domain.
• ATTR: The copyright holder of this file allows anyone to use it for any purpose, provided that the copyright holder is properly attributed. Redistribution, derivative work, commercial use, and all other use is permitted.
• EURO: This is the common (reverse) face of a euro coin. The copyright on the design of the common face of the euro coins belongs to the European Commission. Authorised
is reproduction in a format without relief (drawings, paintings, films) provided they are not detrimental to the image of the euro.

- CFR: Copyright free use.

Copies of the GPL, the LGPL as well as a GFDL are included in chapter Licenses\textsuperscript{14}. Please note that images in the public domain do not require attribution. You may click on the image numbers in the following table to open the webpage of the images in your webbrowser.

\textsuperscript{14} Chapter 11 on page 43
<table>
<thead>
<tr>
<th>Figure</th>
<th>Author</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>arichnad\textsuperscript{15}, arichnad\textsuperscript{16}</td>
<td>CC-BY-SA-3.0</td>
</tr>
<tr>
<td>2</td>
<td>Shirley de Oliveira</td>
<td>CC-BY-SA-3.0</td>
</tr>
<tr>
<td>3</td>
<td>Shirley</td>
<td>CC-BY-SA-3.0</td>
</tr>
<tr>
<td>4</td>
<td>Shirley</td>
<td>CC-BY-SA-3.0</td>
</tr>
<tr>
<td>5</td>
<td>Shirley</td>
<td>CC-BY-SA-3.0</td>
</tr>
<tr>
<td>6</td>
<td>Shirley</td>
<td>CC-BY-SA-3.0</td>
</tr>
<tr>
<td>7</td>
<td>Shirley</td>
<td>CC-BY-SA-3.0</td>
</tr>
<tr>
<td>8</td>
<td>Shirley</td>
<td>CC-BY-SA-3.0</td>
</tr>
<tr>
<td>9</td>
<td>Shirley</td>
<td>CC-BY-SA-3.0</td>
</tr>
<tr>
<td>10</td>
<td>Shirley</td>
<td>CC-BY-SA-3.0</td>
</tr>
</tbody>
</table>

\textsuperscript{15} [http://commons.wikimedia.org/wiki/User:Arichnad]
\textsuperscript{16} [https:///wiki/User:Arichnad]
11 Licenses

11.1 GNU GENERAL PUBLIC LICENSE

Version 2, June 1991
Copyright © 1989, 1991 Free Software Foundation, Inc.
<https://www.gnu.org/licenses/gpl.html>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

The GNU General Public License is a free, copy-once license that distributes source code verbatim and fixes the source files for the user's benefit, without trying to impose systematic control. It keeps all users legally able to run the programs as they are, whatever they want to do with them.

What the License Guarantees.

The "corresponding source code" for a work is the code for implementing the work. The "executable form" of a work is the code that runs it.

The "corresponding source code" for a work in object code form shall be the source code for implementing the work without non殘差, that is the source code for implementing the work without modules added after the cessation.

A "copy" of a work means a verbatim copy of a portion or the whole of a work, whether uniformly treated or not.

A "derivative work" means a work that is transformed by an additional copy of the corresponding source code and that is itself not a defunct license.

A "tertiary component" means a component designated by license agreement as a tertiary component of the work.

A "patent claim" means an item of patent protection in which a patent applicant (or a patent holder that is applicable to a party's product) specifies conditions under which royalty-free use of that claim will be allowed receive a corresponding source code for the corresponding functions in accordance with section 4.

The Corresponding Source for a work for which no corresponding source code exists means either the corresponding object code or the object code together with all the corresponding宗卷 or enables use of the work with that module without making changes to it. (If a module could make use of that work, but applied to a different portion of it, those portions of the work that make use of the module are treated as a whole and counted as a "work on the same subject" as the original work.)

The "corresponding source code" for a work in object code form shall be the source code for implementing the work with the same manifestation of those changes. The Corresponding Source for a work in object code form shall be a complete, machine-readable copy of the corresponding source code for the object code, and must include all the files listed in section 3, except that the source code for a work that uses a third-party library (or a modified version of such a library) need not be included in the corresponding source code.

The Corresponding Source for a work in object code form shall be a complete, machine-readable copy of the corresponding source code for the object code, and must include all the files listed in section 3, except that the source code for a work that uses a third-party library (or a modified version of such a library) need not be included in the corresponding source code.

The Corresponding Source for a work in object code form shall be a complete, machine-readable copy of the corresponding source code for the object code, and must include all the files listed in section 3, except that the source code for a work that uses a third-party library (or a modified version of such a library) need not be included in the corresponding source code.

The Corresponding Source for a work in object code form shall be a complete, machine-readable copy of the corresponding source code for the object code, and must include all the files listed in section 3, except that the source code for a work that uses a third-party library (or a modified version of such a library) need not be included in the corresponding source code.

If you convey only part of an inhomogeneous work, you may convey part of the work under this License, provided that the part that you convey under this License contains no part of the work that you did not convey under this License. This 11.1 GNU GENERAL PUBLIC LICENSE

Copyright © 1989, 1991 Free Software Foundation, Inc.
<https://www.gnu.org/licenses/gpl.html>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

The GNU General Public License is a free, copy-once license that distributes source code verbatim and fixes the source files for the user's benefit, without trying to impose systematic control. It keeps all users legally able to run the programs as they are, whatever they want to do with them.

What the License Guarantees.

The "corresponding source code" for a work is the code for implementing the work. The "executable form" of a work is the code that runs it.

The "corresponding source code" for a work in object code form shall be the source code for implementing the work without non 残差, that is the source code for implementing the work without modules added after the cessation.

A "copy" of a work means a verbatim copy of a portion or the whole of a work, whether uniformly treated or not.

A "derivative work" means a work that is transformed by an additional copy of the corresponding source code and that is itself not a defunct license.

A "tertiary component" means a component designated by license agreement as a tertiary component of the work.

A "patent claim" means an item of patent protection in which a patent applicant (or a patent holder that is applicable to a party's product) specifies conditions under which royalty-free use of that claim will be allowed receive a corresponding source code for the corresponding functions in accordance with section 4.

The Corresponding Source for a work for which no corresponding source code exists means either the corresponding object code or the object code together with all the corresponding 宗卷 or enables use of the work with that module without making changes to it. (If a module could make use of that work, but applied to a different portion of it, those portions of the work that make use of the module are treated as a whole and counted as a "work on the same subject" as the original work.)

The "corresponding source code" for a work in object code form shall be the source code for implementing the work with the same manifestation of those changes. The Corresponding Source for a work in object code form shall be a complete, machine-readable copy of the corresponding source code for the object code, and must include all the files listed in section 3, except that the source code for a work that uses a third-party library (or a modified version of such a library) need not be included in the corresponding source code.

The Corresponding Source for a work in object code form shall be a complete, machine-readable copy of the corresponding source code for the object code, and must include all the files listed in section 3, except that the source code for a work that uses a third-party library (or a modified version of such a library) need not be included in the corresponding source code.

The Corresponding Source for a work in object code form shall be a complete, machine-readable copy of the corresponding source code for the object code, and must include all the files listed in section 3, except that the source code for a work that uses a third-party library (or a modified version of such a library) need not be included in the corresponding source code.

The Corresponding Source for a work in object code form shall be a complete, machine-readable copy of the corresponding source code for the object code, and must include all the files listed in section 3, except that the source code for a work that uses a third-party library (or a modified version of such a library) need not be included in the corresponding source code.

If you convey only part of an inhomogeneous work, you may convey part of the work under this License, provided that the part that you convey under this License contains no part of the work that you did not convey under this License. This
The "Corresponding Application Code" for a Combined Work means
the object code and/or source code for the Application, including any
data and utility programs needed for reproducing the Combined Work
from the Application, but excluding the System Libraries of the Com-
bined Work. 1. Exception to Section 3 of the GNU GPL. 2. Conveying
Identical Versions, Except with Corresponding Source. If you convey
a Combined Work under the terms of this License, and the Library is
incorporated into the Combined Work, then the Corresponding Source
for the Combined Work shall be the output from compiling or linkage
the object code and/or source code of the Library. 3. Object Code
Incorporating Material from Library Header Files. The object code
form of an Application may incorporate material from a header file
that is part of the Library. You may convey such object code under ter-
ms of your choice, provided that, if the object code is made available
under the terms of Section 4 of the GNU GPL, then the Library is
made available under the terms of this License. 4. Combined Works.

The "Corresponding Application Code" for a Combined Work means
the object code and/or source code for the Application, including any
data and utility programs needed for reproducing the Combined Work
from the Application, but excluding the System Libraries of the Com-
bined Work. 1. Exception to Section 3 of the GNU GPL. 2. Conveying
Identical Versions, Except with Corresponding Source. If you convey
a Combined Work under the terms of this License, and the Library is
incorporated into the Combined Work, then the Corresponding Source
for the Combined Work shall be the output from compiling or linkage
the object code and/or source code of the Library. 3. Object Code
Incorporating Material from Library Header Files. The object code
form of an Application may incorporate material from a header file
that is part of the Library. You may convey such object code under ter-
ms of your choice, provided that, if the object code is made available
under the terms of Section 4 of the GNU GPL, then the Library is
made available under the terms of this License. 4. Combined Works.

You may convey a Combined Work under terms of your choice, that
are based on the Application, and not on the Linked Version.

corresponding to the License, supplemented by the additional permis-
sions listed below. 1. Exception to Section 3 of the GNU GPL. 2. Conveying
Identical Versions, Except with Corresponding Source. If you convey
a Combined Work under the terms of this License, and the Library is
incorporated into the Combined Work, then the Corresponding Source
for the Combined Work shall be the output from compiling or linkage
the object code and/or source code of the Library. 3. Object Code
Incorporating Material from Library Header Files. The object code
form of an Application may incorporate material from a header file
that is part of the Library. You may convey such object code under ter-
ms of your choice, provided that, if the object code is made available
under the terms of Section 4 of the GNU GPL, then the Library is
made available under the terms of this License. 4. Combined Works.

You may convey a Combined Work under terms of your choice, that
are based on the Application, and not on the Linked Version.

You may convey a Combined Work under terms of your choice, that
are based on the Application, and not on the Linked Version.

You may convey a Combined Work under terms of your choice, that
takes together, effectively do not restrict modification of the portions
of the Library contained in the Combined Work and reverse engineer-
ging for debugging such modifications, if you also do the following:

a) Give prominent notice with each copy of the Combined Work
that the Library is used in it and that the Library and its use are
covered by this License. * b) Accompany the Combined Work with a
copy of the GNU GPL and this license document. * c) For a Com-
bined Work that displays copyright notices during execution, include
the copyright notice for the Library among these notices, as well as a
reference directing the user to the copies of the GNU GPL and this
license document. * d) Do one of the following: o 0) Convey the
Minimal Corresponding Source under the terms of this License, and
the Corresponding Application Code in a form suitable for, and un-
der terms that permit, the user to recalculate or retranslate the Library
with a modified version of the Linked Version to produce a modified
Combined Work, in the manner specified by section 6 of the GNU
GPL for conveying Corresponding Source. * i) Use a suitable closed
library mechanism for linking with the Library. A suitable mechanism
is one that (a) uses at run time a copy of the Library already present
on the user’s computer, and (b) will operate properly with a
modified version of the Library that is interface-compatible with the
Linked Version. * j) Provide Installation Information, but only if you
choose to provide INSTALL files and other documentation of the
same quality of detail and or thoroughness as those normally used
with a modified version of the Linked Version. * k) Provide an un-
politicized notice and a means of obtaining full source code of the
Library. * l) Provide a method of modifying the Library, such as by
providing an installer or source code to the Library. * m) Provide
the Library in uncombined or object code form, if supplied under a
license that does not require linking in combination with other library
modules.

You may place library facilities that are works based on the Library
side by side in a single library together with other library facilities
that are not Applications and are not covered by this License; and convey
such a combined library under terms of your choice, if you do both of
the following:

The Library is used in it and that the Library and its use are
covered by this License. * b) Accompany the Combined Work with a
copy of the GNU GPL and this license document. * c) For a Com-
bined Work that displays copyright notices during execution, include
the copyright notice for the Library among these notices, as well as a
reference directing the user to the copies of the GNU GPL and this
license document. * d) Do one of the following: 0) Convey the
Minimal Corresponding Source under the terms of this License, and
the Corresponding Application Code in a form suitable for, and un-
der terms that permit, the user to recalculate or retranslate the Library
with a modified version of the Linked Version to produce a modified
Combined Work, in the manner specified by section 6 of the GNU
GPL for conveying Corresponding Source. * i) Use a suitable closed
library mechanism for linking with the Library. A suitable mechanism
is one that (a) uses at run time a copy of the Library already present
on the user’s computer, and (b) will operate properly with a
modified version of the Library that is interface-compatible with the
Linked Version. * j) Provide Installation Information, but only if you
choose to provide INSTALL files and other documentation of the
same quality of detail and or thoroughness as those normally used
with a modified version of the Linked Version. * k) Provide an un-
politicized notice and a means of obtaining full source code of the
Library. * l) Provide a method of modifying the Library, such as by
providing an installer or source code to the Library. * m) Provide
the Library in uncombined or object code form, if supplied under a
license that does not require linking in combination with other library
modules.

The Free Software Foundation may publish revised and/or new ver-
sions of the GNU Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version, but
may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
as you received it specifies that a proxy can decide, with a version
number of the GNU Lesser General Public License "or any later version" applies to it, you
may choose any version of the GNU Lesser General Public License ever
published by the Free Software Foundation. 6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new ver-
sions of the GNU Lesser General Public License and/or new license
commands approved by the Free Software Foundation. If the Library as you received it does not specify a ver-
sion number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever
published by the Free Software Foundation.

The Library as you received it states that it is governed by this
License. If the Library as you received it does not state it is governed
by this License, this version of the GNU Lesser General Public License shall
apply, that proxy’s public statement of acceptance of any ver-
sion is permanent authorization for you to choose that version for the
Library.