

OFFICIAL REPOSITORY

[httpd \(/r/ /httpd/\)](#) 

Last pushed: 8 days ago

[Repo Info \(/ /httpd/\)](#)

Short Description

The Apache HTTP Server Project

Full Description

Supported tags and respective Dockerfile links

- [2.4.37](#) , [2.4](#) , [2](#) , [latest](#) ([2.4/Dockerfile](#)) (<https://github.com/docker-library/httpd/blob/a426e299988c48f4937dfb4a9a67ac479ca011e1/2.4/Dockerfile>)
- [2.4.37-alpine](#) , [2.4-alpine](#) , [2-alpine](#) , [alpine](#) ([2.4/alpine/Dockerfile](#)) (<https://github.com/docker-library/httpd/blob/a426e299988c48f4937dfb4a9a67ac479ca011e1/2.4/alpine/Dockerfile>)

Quick reference

- **Where to get help:**
[the Docker Community Forums](#) (<https://forums.docker.com/>), [the Docker Community Slack](#) (<https://blog.docker.com/2016/11/introducing-docker-community-directory-docker-community-slack/>), or [Stack Overflow](#) (<https://stackoverflow.com/search?tab=newest&q=docker>)
- **Where to file issues:**
<https://github.com/docker-library/httpd/issues> (<https://github.com/docker-library/httpd/issues>)
- **Maintained by:**
[the Docker Community](https://github.com/docker-library/httpd) (<https://github.com/docker-library/httpd>)

- **Supported architectures:** (more info (<https://github.com/docker-library/official-images#architectures-other-than-amd64>))
[amd64](https://hub.docker.com/r/amd64/httpd/) (<https://hub.docker.com/r/amd64/httpd/>), [arm32v5](https://hub.docker.com/r/arm32v5/httpd/) (<https://hub.docker.com/r/arm32v5/httpd/>), [arm32v6](https://hub.docker.com/r/arm32v6/httpd/) (<https://hub.docker.com/r/arm32v6/httpd/>), [arm32v7](https://hub.docker.com/r/arm32v7/httpd/) (<https://hub.docker.com/r/arm32v7/httpd/>), [arm64v8](https://hub.docker.com/r/arm64v8/httpd/) (<https://hub.docker.com/r/arm64v8/httpd/>), [i386](https://hub.docker.com/r/i386/httpd/) (<https://hub.docker.com/r/i386/httpd/>), [ppc64le](https://hub.docker.com/r/ppc64le/httpd/) (<https://hub.docker.com/r/ppc64le/httpd/>), [s390x](https://hub.docker.com/r/s390x/httpd/) (<https://hub.docker.com/r/s390x/httpd/>)
- **Published image artifact details:**
[repo-info](https://github.com/docker-library/repo-info/blob/master/repos/httpd/) [repo's](https://github.com/docker-library/repo-info/blob/master/repos/httpd/) [repos/httpd/](https://github.com/docker-library/repo-info/blob/master/repos/httpd/) [directory](https://github.com/docker-library/repo-info/blob/master/repos/httpd/) (https://github.com/docker-library/repo-info/blob/master/repos/httpd) ([history](https://github.com/docker-library/repo-info/commits/master/repos/httpd/) (https://github.com/docker-library/repo-info/commits/master/repos/httpd))
 (image metadata, transfer size, etc)
- **Image updates:**
[official-images](https://github.com/docker-library/official-images/pulls?q=label%3Alibrary%2Fhttpd) PRs with label [library/httpd](https://github.com/docker-library/official-images/pulls?q=label%3Alibrary%2Fhttpd) (<https://github.com/docker-library/official-images/pulls?q=label%3Alibrary%2Fhttpd>)
[official-images](https://github.com/docker-library/official-images/blob/master/library/httpd) [repo's](https://github.com/docker-library/official-images/blob/master/library/httpd) [library/httpd](https://github.com/docker-library/official-images/blob/master/library/httpd) [file](https://github.com/docker-library/official-images/blob/master/library/httpd) (<https://github.com/docker-library/official-images/blob/master/library/httpd>) ([history](https://github.com/docker-library/official-images/commits/master/library/httpd) (<https://github.com/docker-library/official-images/commits/master/library/httpd>))
- **Source of this description:**
[docs](https://github.com/docker-library/docs/tree/master/httpd) [repo's](https://github.com/docker-library/docs/tree/master/httpd) [httpd/](https://github.com/docker-library/docs/tree/master/httpd) [directory](https://github.com/docker-library/docs/tree/master/httpd) (<https://github.com/docker-library/docs/tree/master/httpd>)
 ([history](https://github.com/docker-library/docs/commits/master/httpd) (<https://github.com/docker-library/docs/commits/master/httpd>))
- **Supported Docker versions:**
[the latest release](https://github.com/docker/docker-ce/releases/latest) (<https://github.com/docker/docker-ce/releases/latest>) (down to 1.6 on a best-effort basis)

What is httpd?

The Apache HTTP Server, colloquially called Apache, is a Web server application notable for playing a key role in the initial growth of the World Wide Web. Originally based on the NCSA HTTPd server, development of Apache began in early 1995 after work on the NCSA code stalled. Apache quickly overtook NCSA HTTPd as the dominant HTTP server, and has remained the most popular HTTP server in use since April 1996.

[wikipedia.org/wiki/Apache_HTTP_Server](http://en.wikipedia.org/wiki/Apache_HTTP_Server)
 (http://en.wikipedia.org/wiki/Apache_HTTP_Server)



How to use this image.

This image only contains Apache httpd with the defaults from upstream. There is no PHP installed, but it should not be hard to extend. On the other hand, if you just want PHP with Apache httpd see the [PHP image \(https://hub.docker.com/_/php/\)](https://hub.docker.com/_/php/) and look at the `-apache` tags. If you want to run a simple HTML server, add a simple Dockerfile to your project where `public-html/` is the directory containing all your HTML.

Create a Dockerfile in your project

```
FROM httpd:2.4
COPY ./public-html/ /usr/local/apache2/htdocs/
```

Then, run the commands to build and run the Docker image:

```
$ docker build -t my-apache2 .
$ docker run -dit --name my-running-app -p 8080:80 my-apache2
```

Visit <http://localhost:8080> (<http://localhost:8080>) and you will see It works!

Without a Dockerfile

If you don't want to include a `Dockerfile` in your project, it is sufficient to do the following:

```
$ docker run -dit --name my-apache-app -p 8080:80 -v "$PWD":/usr/local/apache2/htdocs/ my-apache2
```

Configuration

To customize the configuration of the httpd server, just `COPY` your custom configuration in as `/usr/local/apache2/conf/httpd.conf`.

```
FROM httpd:2.4
COPY ./my-httpd.conf /usr/local/apache2/conf/httpd.conf
```

SSL/HTTPS

If you want to run your web traffic over SSL, the simplest setup is to `COPY` or mount (`-v`) your `server.crt` and `server.key` into `/usr/local/apache2/conf/` and then customize the `/usr/local/apache2/conf/httpd.conf` by removing the comment symbol from the following lines:

```
...
#LoadModule socache_shmcb_module modules/mod_socache_shmcb.so
...
#LoadModule ssl_module modules/mod_ssl.so
...
#Include conf/extra/httpd-ssl.conf
...
```

The `conf/extra/httpd-ssl.conf` configuration file will use the certificate files previously added and tell the daemon to also listen on port 443. Be sure to also add something like `-p 443:443` to your `docker run` to forward the https port.

This could be accomplished with a `sed` line similar to the following:

```
RUN sed -i \
    -e 's/^#\ (Include .*httpd-ssl.conf\)/\1/' \
    -e 's/^#\ (LoadModule .*mod_ssl.so\)/\1/' \
    -e 's/^#\ (LoadModule .*mod_socache_shmcb.so\)/\1/' \
    conf/httpd.conf
```

The previous steps should work well for development, but we recommend customizing your conf files for production, see httpd.apache.org (https://httpd.apache.org/docs/2.4/ssl/ssl_faq.html) for more information about SSL setup.

Image Variants

The `httpd` images come in many flavors, each designed for a specific use case.

`httpd:<version>`

This is the defacto image. If you are unsure about what your needs are, you probably want to use this one. It is designed to be used both as a throw away container (mount your source code and start the container to start your app), as well as the base to build other images off of.

`httpd:<version>-alpine`

This image is based on the popular [Alpine Linux project](http://alpinelinux.org) (<http://alpinelinux.org>), available in the [alpine official image](https://hub.docker.com/_/alpine) (https://hub.docker.com/_/alpine). Alpine Linux is much smaller than most distribution base images (~5MB), and thus leads to much slimmer images in general.

This variant is highly recommended when final image size being as small as possible is desired. The main caveat to note is that it does use [musl libc](http://www.musl-libc.org) (<http://www.musl-libc.org>) instead of [glibc and friends](http://www.etalabs.net/compare_libcs.html) (http://www.etalabs.net/compare_libcs.html), so certain software might run into issues depending on the depth of their libc requirements. However, most software doesn't have an issue with this, so this variant is usually a very safe choice.

See [this Hacker News comment thread \(https://news.ycombinator.com/item?id=10782897\)](https://news.ycombinator.com/item?id=10782897) for more discussion of the issues that might arise and some pro/con comparisons of using Alpine-based images.

To minimize image size, it's uncommon for additional related tools (such as `git` or `bash`) to be included in Alpine-based images. Using this image as a base, add the things you need in your own Dockerfile (see the [alpine image description \(https://hub.docker.com/_/alpine/\)](https://hub.docker.com/_/alpine/) for examples of how to install packages if you are unfamiliar).

License

View [license information \(https://www.apache.org/licenses/\)](https://www.apache.org/licenses/) for the software contained in this image.

As with all Docker images, these likely also contain other software which may be under other licenses (such as Bash, etc from the base distribution, along with any direct or indirect dependencies of the primary software being contained).

Some additional license information which was able to be auto-detected might be found in the [repo-info repository's httpd/ directory \(https://github.com/docker-library/repo-info/tree/master/repos/httpd\)](https://github.com/docker-library/repo-info/tree/master/repos/httpd).

As for any pre-built image usage, it is the image user's responsibility to ensure that any use of this image complies with any relevant licenses for all software contained within.

Docker Pull Command



```
docker pull httpd
```