

 無料電子ブック

学習

apache

Free unaffiliated eBook created from  
**Stack Overflow contributors.**

#apache

.....	1
<b>1: apache</b> .....	<b>2</b>
.....	2
.....	2
Apachehttpd.....	2
Examples.....	2
.....	2
Ubuntu.....	2
Windows.....	2
CentOS.....	2
macOS.....	3
[Ubuntu>Hello World.....	3
.....	3
HTML.....	3
.....	4
.....	4
<b>2: Apache Flume</b> .....	<b>5</b>
.....	5
Examples.....	5
/.....	5
<b>3: Apache</b> .....	<b>6</b>
.....	6
Examples.....	6
.....	6
PHP.....	7
WAMP.....	7
1IP23Apache2.4.....	9
HTTPS.....	10
<b>4: Apache.htaccess</b> .....	<b>11</b>
Examples.....	11
.....	11

HTTPS.....	11
CORS.....	12
.....	13
301 Htaccess.....	13
.....	<b>14</b>

---

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [apache](#)

It is an unofficial and free apache ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official apache.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to [info@zzzprojects.com](mailto:info@zzzprojects.com)

# 1: apacheをいめる

このセクションでは、Apacheのとがなぜそれをしたいのかをしします。

また、Apacheのきなテーマについてもし、トピックにリンクしてください。Apacheのドキュメントはしくなっているなので、それらのトピックのバージョンをするがあります。

## バージョン

### さまざまなApacheのhttpdリリース

バージョン		リリース
1.3	1.3.42	199866
2.0	2.0.65	2002-04-06
2.2	2.2.32	2005-12-01
2.4	2.4.25	2012-02-21

## Examples

インストールまたはセットアップ

Apacheのまたはインストールにするな。

### Ubuntuのインストール

```
sudo apt-get install apache2
```

### Windowsのインストール

WAMPスタックをチェックしてください。WAMPは、Windows、Apache、MySQL、PhpMyAdminのです。

### CentOSのインストール

Apache 2.2にはCentOS6がしていますが、2.4にはCentOS7がしています。

```
yum -y install httpd
```

## macOSのインストール

macOSにはApacheがプリインストールされていますが、ApacheはHomebrewでインストールできます

すでにApacheがみまれているは、まずシャットダウンして、ロードスクリプトをするがあります

。

```
$ sudo apachectl stop
$ sudo launchctl unload -w /System/Library/LaunchDaemons/org.apache.httpd.plist 2>/dev/null
$ brew install httpd24 --with-privileged-ports --with-http2
```

## [Ubuntu]なHello Worldの

ここでは、Hello WorldのHTMLページをするバックエンドのについてします。

## のインストール

このステップのためにをしてください

- `sudo apt-get install apache2`

## HTMLの

Apacheファイルは `/var/www/html/` ます。すぐにそこにきましょう。にルートディレクトリについていることをし、`cd`、に `cd /var/www/html/` ます。

この`html`ディレクトリには、あなたのウェブサイトのすべてのファイルがされます。なHello Worldファイルをすばやくできます。

おにりのテキストエディタをして、のようにします。

```
<!DOCTYPE html>
<html>
<head>
  <title>Hello World!</title>
</head>
<body>
  <h1>Hello World!</h1>
</body>
```

```
</html>
```

このファイルをのディレクトリに `index.html` としてしてください

## あなたのウェブページをれる

したばかりのページにアクセスするには、したブラウザで `localhost` にします。それでもがしないは、 `127.0.0.1` してください。「Hello World」がされます。 `h1` として。あなたはわった

サーバーがしていることをします。

ブラウザがサーバーにできないというメッセージがされたは、まずサーバーがしていることをします。

```
$ ps -aef | grep httpd
```

Apacheがしているは、いくつかの `httpd` プロセスがされます。

オンラインで `apache` をいめるをむ <https://riptutorial.com/ja/apache/topic/964/apacheをいめる>

---

## 2: Apache Flume

き

Apache Flumeは、ログファイル、イベントetc...などのストリーミングデータをして、さまざまなソースからデータストアにするツール/サービス/データリミメカニズムです。

Flumeは、一般的なツールです。に、さまざまなWebサーバーからストリーミングデータログデータをHDFSにコピーするようにされています。

### Examples

#### ストリーミング/ログデータ

に、されるデータのほとんどは、アプリケーションサーバー、ソーシャルネットワーキングサイト、クラウドサーバー、エンタープライズサーバーなどのさまざまなデータソースによってされます。このデータは、ログファイルとイベントのでされます。

ログファイル - に、ログファイルは、オペレーティングシステムであるイベント/アクションをするファイルです。たとえば、Webサーバーは、サーバーにしてわれたすべてのをログファイルにします。

そのようなログデータをするには、

アプリケーションのパフォーマンスをし、さまざまなソフトウェアやハードウェアのをします。よりいビジネスインサイトをきすことができます。データをHDFSシステムにするには、putコマンドをすることです。putコマンドのいをてみましょう。

オンラインでApache Flumeをむ <https://riptutorial.com/ja/apache/topic/9630/apache-flume>

## 3: Apacheでホストをする

ApacheのVirtualHostのなエントリーポイントは、[Apache Virtual Hostのドキュメント](#)です。そこから、ホストのにするなドキュメント、VirtualHostおよびするディレクティブについてのリファレンスドキュメントがあります。

### Examples

#### ベースのホスト

Apacheのベースのホスティングは、[Apache Webサイト](#)でのようにされています。

ベースのホスティングでは、サーバーはクライアントにしてHTTPヘッダーのとしてホストをします。このをすると、くのなるホストがじIPアドレスをできます。

したがって、このでのWebサイトを1のサーバーでホストすることができます。ubuntuでは、ファイルは/etc/apache2/sites-availableにあります。そのディレクトリには、000-default.confがあります。これがデフォルトです。のがするまで、すべてのリクエストはこのファイルにられます。

ホストをするには、ここで**example.com**がされますが、**domain.com**できえるがあります。デフォルトのファイルをコピーする

```
cp 000-default.conf example.com.conf
```

ファイルには、のディレクティブをめることができます。

```
<VirtualHost *:80>
    ServerAdmin admin@example.com
    ServerName example.com
    ServerAlias www.example.com

    DocumentRoot /var/www/example.com/html

    ErrorLog /var/log/apache/logs/error.log
    # Possible values include: debug, info, notice, warn, error, crit,
    # alert, emerg.
    LogLevel warn

    CustomLog /var/log/apache/logs/access.log combined
</VirtualHost>
```

- のは、ポート80デフォルトのhttpポートのすべてののがするがあることをします。\*のわりにIPアドレスをつこともできます。これはサーバーのIPです。
- ServerAdminは、httpエラーメッセージをするためにされるWebサイトののです。
- ServerNameはウェブサイトのドメインです。
- ServerAliasはウェブサイトのセカンダリです。はwww.domain.comになります

- DocumentRoot は、Web サイトをするにみまれるルートフォルダです。
- エラー ErrorLog は、エラーが ErrorLog れるファイルです
- LogLevel 。 ログにされるエラーのレベルです
- CustomLog はアクセスがられるファイルです

example.com のわりに、Web サイトのドメインと Web サイトファイルのなディレクトリをします。

ファイルをし、の Apache コマンドをしてサイトをにします。

```
sudo a2ensite example.com.conf
```

## Apache のリロード

```
sudo service apache2 reload
```

チェックするがあるいくつかの

- ドメインの DNS がしい IP にされていることをしてくださいプロポジットにがかかることがあります
- ファイアウォールでポート 80 がいていることをする
- ファイルのアクセスがサーバーファイルでしくされていることをします。は www-data www-data で、ディレクトリのアクセスは 750、ファイルのアクセスは 640 です。

あなたのホストがしているはずでしファイルしをと /var/www ののディレクトリをして、しサーバーのの Web サイトでこれをりすことができます。

## PHP ホスト

これは、とデバッグのためにホストサイトで PHP エラーログをするのです。

- PHP モジュールがインストールされています。
- はではありません。

```
<VirtualHost *:80>
  ServerName example.com
  DocumentRoot /var/www/domains/example.com/html
  ErrorLog /var/www/domains/example.com/apache.error.log
  CustomLog /var/www/domains/example.com/apache.access.log common
  php_flag log_errors on
  php_flag display_errors on
  php_value error_reporting 2147483647
  php_value error_log /var/www/domains/example.com/php.error.log
</VirtualHost>
```

ホストはです。なぜなら、display\_errors がになっており、ではそれをとしないからです。

## WAMP のホスト

あなたが**Windows 7 PC**でしているとします

ステップ**1** GOTO -> C:\Windows\System32\drivers\etc "hosts"というのファイルをつけたら、じにコピーしてりけてください。そこにホストのコピーファイルがされます。

このファイルでいくつかのをえるがありますが、メモやメモ++などのエディタでしようとする、ファイルをすることはできません。

もうじファイルをコピーしてデスクトップにりけると、このファイルをにできます。

1つまたはのエントリがつかります127.0.0.1 localhostそのファイルにあります。そののののをします127.0.0.1 myproject1.localこのようにして、"localhost / myproject1"のわりにするしいサブドメイン "myproject1.local"をしました。

ステップ**2**これで、しくされたドメインにアクセスするためのルートパスをするときです。

GOTO C:\wamp\bin\apache\Your-Apache-Version\conf\extra "httpd-vhosts"というのファイルがあります。エディターで、のをりけてください。

```
<VirtualHost *:80>
  ServerAdmin webmaster@dummy.example.com
  DocumentRoot "c:/wamp/www/myproject1/"
  ServerName myproject1.local
  ErrorLog "logs/myproject1.local-error.log"
  CustomLog "logs/myproject1.local.log" common
</VirtualHost>
```

これで、ほぼ「c:/wamp/www/myproject1/」にあるプロジェクトにアクセスできます。

ステップ**3** GOTO C:\wamp\bin\apache\your-Apache-version\conf

"httpd.conf"というのファイルをし、それをコピーしてのためにじにりけます。エディタでファイルをし、"ホスト"というをつけます。"Include conf/extra/httpd-vhosts.conf"というがあります。コメントがあればコメントをし、wamp-serverのサービスをしてください。

あなたのウェブブラウザにき、myproject1.localとくと、すぐのプロジェクトをることができます。

localhostをURLとしてしてローカルホストがしないというにすることがあります。しないでください...このコードを"httpd-vhosts"ファイルにりけてください。

```
<VirtualHost *:80>
  ServerAdmin webmaster@dummy.example.com
  DocumentRoot "c:/wamp/www"
  ServerName localhost
  ErrorLog "logs/localhost-error.log"
  CustomLog "logs/localhost.log" common
</VirtualHost>
```

WAMPのすべてのサービスをする、はです。

ありがとう**Chintan Gor**

## 1IPベースのホスト2じポートをつのホスト3マクロをってホストをするApache2.4

### 1IPベースのホスト

```
<VirtualHost 192.168.13.37>
  ServerName example.com
  DocumentRoot /var/www/domains/example.com/html
  ErrorLog /var/log/example.com/error.log
  CustomLog /var/log/example.com/access.log common
</VirtualHost>

<VirtualHost 192.168.47.11>
  ServerName otherurl.com
  DocumentRoot /srv/www/htdocs/otherurl.com/html
  ErrorLog /var/log/otherurl.com/error.log
  CustomLog /var/log/otherurl.com/access.log common
</VirtualHost>
```

あなたのえられたIPにポートをするだけです。ポートはホストがばれるというにはです。

### 2じポートをつのホスト

NameVirtualHostはもうではないので、じポートでのvhostをくことができます。

```
<VirtualHost *:80>
  DocumentRoot /srv/www/htdocs/otherurl.com/html
  ErrorLog /var/log/otherurl.com/error.log
  CustomLog /var/log/otherurl.com/access.log common
</VirtualHost>

<VirtualHost *:80>
  ServerName example.com
  ServerAlias ex1.com ex2.com
  DocumentRoot /var/www/domains/example.com/html
  ErrorLog /var/log/example.com/error.log
  CustomLog /var/log/example.com/access.log common
</VirtualHost>
```

ここではですIPはですが、がポート80でされたは、したがされます。あなたはex1.comにしましたかURLotherurl.com、example3.comなどをびすと、のURLがされます。あなたがむならば、このホストを「フォールバック」としてうことができます。

### 3マクロをってホストをするApache2.4

```
<Macro VHost $port $host>
  <VirtualHost *:$port>
    Servername $host
    DocumentRoot /srv/www/htdocs/$host
    ErrorLog /var/log/$host/error.log
  </VirtualHost>
</Macro>
```

```
Use VHost 80 example.com
Use VHost 443 secure_example.com
```

2つのホストをします。1つはポート80、もう1つは443で、それにじてをします。

ホストをして**HTTPS**をする

リダイレクトをして、ユーザーにセキュリティされたURLへのをします。

```
<VirtualHost *:80>
  ServerName example.com
  SSLProxyEngine on
  Redirect permanent / https://secure_example.com/
</VirtualHost>
```

りのはすべてがリダイレクトされるので、sslホストポート443にくことができます。

```
<VirtualHost _default_:443>
  ServerName secure_example.com
  ServerAdmin webmaster@example.com
  DocumentRoot /var/www/domains/secure_example.com/html
  ErrorLog /var/log/secure_example.com/error.log
  CustomLog /var/log/secure_example.com/access.log common
  SSLEngine On
  ...
</VirtualHost>
```

オンラインでApacheでホストをするをむ <https://riptutorial.com/ja/apache/topic/4856/apacheでホストをする>

## 4: Apacheの.htaccessファイル

### Examples

リライトエンジン

ApacheのRewriteEngineモジュールは、されるさまざまにじて、URLとパスをにきえます。

```
<IfModule mod_rewrite.c>
RewriteEngine On
RewriteBase /
RewriteRule ^index\.php$ - [L]
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule . /index.php [END]
</IfModule>
```

のは、PHPファイルをきえてもはやそのをしないので、index.phpはindex.htmlにられるとののドメインとしてされます。のルールはWordPressにされています。

Apache httpd 2.2.16では、このブロックをFallbackResourceディレクティブをして1にきえることができます。

```
FallbackResource /index.php
```

### HTTPS

.htaccessをして、HTTPサイトをHTTPSにリダイレクトすることができます。

ドメインのコードをするはありません。

```
RewriteEngine On
RewriteCond %{HTTPS} =off
RewriteRule ^ https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]
```

のコードでは、`%{HTTP_HOST}`があなたのドメインをしているとできるとしてしています。

リダイレクトがのドメインであることをするがあるは、`%{HTTP_HOST}`をあなたのドメインにきえてください。

のコードはこれをいます

1. RewriteEngineをにします。
2. のがHTTPSをしていないはします。
3. `https://%{HTTP_HOST}%{REQUEST_URI}`へのHTTP 301リダイレクトをい  
`https://%{HTTP_HOST}%{REQUEST_URI}`
  - `%{HTTP_HOST}`はブラウザからされたホストであり、

- `%(REQUEST_URI)` は、ブラウザからされたURIですドメインのろのすべて。

WebアプリケーションはHTTPSをできるがあり、ホストのApacheにはなサイトをす  
るがあります。

.htaccessファイルでこれらののリクエストごとのをうよりも、httpホストでにRedirectをうがはる  
かにです。こののについては、 <http://wiki.apache.org/httpd/RedirectSSL>をしてください。

## CORSをにする

Apacheで**Cross-Origin Resource Sharing CORS** をにするには、するHTTPヘッダーをなくと  
も1つするがありますデフォルトのはCORSをブロックすることです。のでは、このHTTPヘッダ  
ーを.htaccessにするですが、`site your-site.conf` ファイルまたはApacheファイルですることもでき  
ます。がどのようになっていても、Apacheのブロック `<VirtualHost>`、`<Directory>`、`<Location>`  
、`<Files>`するHTTPヘッダーをできます。

レスポンスですることができるCORSにするいくつかのHTTPヘッダーがあります。

```
Access-Control-Allow-Origin  
  
Access-Control-Allow-Credentials  
Access-Control-Allow-Methods  
Access-Control-Max-Age  
Access-Control-Allow-Headers  
Access-Control-Expose-Headers
```

のうちのいくつかは、"プリフライト"にです。のHTTPクライアントすなわちのブラウザは、  
されたのにをし、サーバーでのをうをているかどうかをします。プリフライトリクエストのにつ  
いては、 [https://en.wikipedia.org/wiki/Cross-origin\\_resource\\_sharing](https://en.wikipedia.org/wiki/Cross-origin_resource_sharing)をしてください。

なメインのHTTPヘッダーはAccess-Control-Allow-Originあり、これをします。しかし、じがすべ  
てのにされますすものをするだけでよい。

のでは、SSLでされたクライアントのドメインFQDNをにするために、`<Directory>` configブロッ  
クになHTTPヘッダーをしています。

```
<Directory /path/to/your/site/>  
    Header set Access-Control-Allow-Origin "https://my.CLIENT.domain"  
</Directory>
```

これをサーバーにした、 <https://my.client.domain>からサーバーにをできるようになり、するがあり  
ます。

くのが、 `Access-Control-Allow-Origin: "*"`しています `Access-Control-Allow-Origin: "*"`はワイルド  
カードであり、すべてのドメインからのをけるがあります。これは、あなたがらかのAPIま  
たはファイルのリポジトリをしているをき、はです。また、HTTPヘッダのコンテキストにして  
ください。APIのHTTPリクエストはしたいが、イメージの「ホットリンク」などはしたくないか  
もしれない。あなたのApache configフローののにこのヘッダをして、のでのみすることができ

。たとえば、`Options +FollowSymLinks`、`Options +Indexes`、`RewriteEngine On`、`RewriteCond`、`RewriteRule`、`Header set`、`Access-Control-Allow-Origin`、`index.php`、`[L]`、`</Directory>`。たとえば、`Options +FollowSymLinks`、`Options +Indexes`、`RewriteEngine On`、`RewriteCond`、`RewriteRule`、`Header set`、`Access-Control-Allow-Origin`、`index.php`、`[L]`、`</Directory>`。たとえば、`Options +FollowSymLinks`、`Options +Indexes`、`RewriteEngine On`、`RewriteCond`、`RewriteRule`、`Header set`、`Access-Control-Allow-Origin`、`index.php`、`[L]`、`</Directory>`。

```
<Directory /path/to/your/site/>
  Options +FollowSymLinks
  Options +Indexes
  RewriteEngine On

  #Make sure it's not a specific file or directory that they're trying to reach
  RewriteCond %{SCRIPT_FILENAME} !-f
  RewriteCond %{SCRIPT_FILENAME} !-d
  Header set Access-Control-Allow-Origin "*"
  RewriteRule ^(.*)$ index.php/$1 [L]
</Directory>
```

[mod\\_headers](#)をインストールしてにするがあります `a2enmod headers`

### 301 Htaccessによるリダイレクト

HTTPレスポンスステータスコード301 Moved Permanentlyは、URLリダイレクトにされます。つまり、レスポンスをしたURLをしているのリンクまたはレコードをするがあります。しいURLは、にまれている[Location]フィールドにするがあります。301リダイレクトは、ユーザーをHTTPからHTTPSにアップグレードするためのベストプラクティスとみなされます。PHP-APACHEのhtaccessファイルにこのコードをいてください

```
Redirect 301 /oldpage/ /newpage/
```

に、htaccessファイルをして、ドメインにSSLがされたwwwにリダイレクトするをします。

```
RewriteEngine On
RewriteCond %{HTTPS} off
RewriteCond %{HTTP_HOST} ^www\.(.*)$ [NC]
RewriteRule ^(.*)$ http://%1/$1 [R=301,L]

RewriteCond %{HTTPS} on
RewriteCond %{HTTP_HOST} ^www\.(.*)$ [NC]
RewriteRule ^(.*)$ https://%1/$1 [R=301,L]

RewriteEngine On
RewriteCond %{SERVER_PORT} 80
RewriteRule ^(.*)$ https://example.com/$1 [R,L]
```

オンラインでApacheのhtaccessファイルをむ <https://riptutorial.com/ja/apache/topic/2089/apacheのhtaccessファイル>

## クレジット

S. No		Contributors
1	apacheをいめる	<a href="#">Community</a> , <a href="#">fab</a> , <a href="#">Flamewires</a> , <a href="#">hjpotter92</a> , <a href="#">James</a> , <a href="#">Katie</a> , <a href="#">Kuhan</a> , <a href="#">Nicholas Qiao</a> , <a href="#">Rich Bowen</a>
2	Apache Flume	<a href="#">Vinod Kumar</a>
3	Apacheでホストをす る	<a href="#">Chintan Gor</a> , <a href="#">Clutch</a> , <a href="#">fab</a> , <a href="#">Harikrishnan</a> , <a href="#">Hello Fishy</a> , <a href="#">Katie</a> , <a href="#">Olaf Dietsche</a>
4	Apacheの.htaccess ファイル	<a href="#">Chintan Gor</a> , <a href="#">Deltik</a> , <a href="#">ezra-s</a> , <a href="#">Luke Bearl</a> , <a href="#">Rich Bowen</a> , <a href="#">SimpleAnecdote</a>