2025/11/01 09:35 1/6 WordPressインストール

WordPressインストール

WordPressをRaspbian(buster)へ構築した際のメモ

参考情報 Raspberry Pi WordPressサーバの構築

PHP7.4インストール

WordPressのシステム要件であるPHP7.4をインストールする。ただし□Raspbian(buster)のリポジトリには7.3までしか存在しないため、SURY PPAリポジトリからインストールする。 まず□SURY PPAリポジトリを使用するため必要なパッケージをインストールする。

\$ sudo apt install lsb-release apt-transport-https ca-certificates

PHPのGPGキーを取得する。

```
$ sudo wget -0 /etc/apt/trusted.gpg.d/php.gpg
https://packages.sury.org/php/apt.gpg
```

SURY PPAリポジトリ情報を追加する。

\$ echo "deb https://packages.sury.org/php/ \$(lsb_release -sc) main" | sudo
tee /etc/apt/sources.list.d/php.list

/etc/apt/php.list

deb https://packages.sury.org/php/ buster main

aptのパッケージ情報を更新し、インストール可能なPHPバージョンを確認する。

```
$ sudo apt-get update -y
$ sudo apt-get upgrade
$ sudo apt -a list php7.4
```

PHP7.4をインストールする。

```
$ sudo apt-get install php7.4
$ php -v
```

PHPモジュールのインストール

WordPressに必要なPHPモジュールをインストールする。

\$ sudo apt-get install php7.4-curl php7.4-json php7.4-mbstring php7.4-mysql

```
php7.4-imagick php7.4-xml php7.4-zip php7.4-intl
```

```
$ sudo apt-get install php7.4-fpm
```

PHP設定

PHPの設定を行う。

```
$ vi /etc/php/7.4/apache2/php.ini
```

```
[mbstring]
; language for internal character representation.
; This affects mb_send_mail() and mbstring.detect_order.
; http://php.net/mbstring.language
mbstring.language = Japanese

memory_limit = 128M

post_max_size = 40M

upload_max_filesize = 30M
```

ApacheのPHPモジュール設定

Apacheで稼働しているPHPバージョンを、phpinfoで確認する。

phpinfo.php

```
<?php
phpinfo();
?>
```

Apacheで有効になっているモジュールを確認する。

\$ ls -l /etc/apache2/mods-available/php* /etc/apache2/mods-enabled/php*

古いバージョンが有効であれば無効化し、PHP7.4を有効化する。

```
$ sudo a2dismod php7.3.conf php7.3.load
$ sudo a2enmod php7.4.conf php7.4.load
$ ls -l /etc/apache2/mods-available/php* /etc/apache2/mods-enabled/php*
```

Apacheを再起動し、新しいモジュールを有効化する。

2025/11/01 09:35 3/6 WordPressインストール

\$ sudo systemctl restart apache2

\$ sudo systemctl status apache2

モジュールのバージョンは、phpinfoで確認する。

ServerNameディレクティブの設定

Apacheをマルチドメイン構成にしているので、該当のドメイン設定ファイルを編集する。

\$ su vi /etc/apache2/sites-available/domain2.com

domain2.com

ServerName domain2.com

DirectoryIndexディレクティブの設定

Apacheへのアクセスでファイル名が省略された時にデフォルトで**index.php**が選択されるように設定する。

\$ sudo vi /etc/apache2/mods-available/dir.conf

index.phpを最初に設定する。

Rewriteモジュールの有効化

Apacheの有効なモジュールにrewrite_moduleがあることを確認する。

\$ sudo apache2ctl -M
(snip)
rewrite_module (shared)

有効でない場合は有効化する。

\$ sudo a2enmod rewrite

新たに有効化した場合はApacheを再起動する。

\$ sudo systemctl restart apache2

\$ sudo systemctl status apache2

MariaDBのインストール

```
$ sudo apt install mariadb-server-10.3
```

MariaDBの初期設定

```
$ sudo mysql_secure_installation
(snip)
Set root password? [Y/n] Y
(snip)
Remove anonymous users? [Y/n] Y
(snip)
Disallow root login remotely? [Y/n] Y
(snip)
Remove test database and access to it? [Y/n] Y
(snip)
Reload privilege tables now? [Y/n] Y
(snip)
```

初期設定の変更

```
$ sudo mysql -u root -p
(snip)
MariaDB [(none)]> grant all privileges on *.* to root@localhost identified
by 'Password' with grant option;
(snip)
MariaDB [(none)]> flush privileges;
(snip)
MariaDB [(none)]> quit
```

DB接続を確認する。

```
$ mysql -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 2026
Server version: 10.3.38-MariaDB-0+deb10ul Raspbian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> quit
```

2025/11/01 09:35 5/6 WordPressインストール

```
Bye
$
```

データペースの作成

WordPressのインストール

WordPresssサイトから最新版をダウンロードし、Document Rootに展開します。

```
$ wget https://ja.wordpress.org/latest-ja.tar.gz
$ ls -l
$ tar tvzf latest-ja.tar.gz
$ tar xvzf latest-ja.tar.gz
$ cp -vR wordpress/* /var/www/html/wordpress/
$ sudo chown -R www-data:www-data /var/www/html/wordpress/
```

インストールスクリプトを使ってセットアップします。

```
https://ドメイン名/wp-admin/install.php
```

Apache2設定例

www.domain2.com

```
<VirtualHost *:80>
    ServerName www.domain2.com
```

```
Last
update:
18:09
```

```
ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html/www.domain2.com
        <Directory /var/www/html/www.domain2.com>
                Options FollowSymLinks
                AllowOverride All
                Require all granted
        </Directory>
        ErrorLog ${APACHE LOG DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined
RewriteEngine on
RewriteCond %{SERVER_NAME} =www.domain2.com
RewriteRule ^ https://%{SERVER NAME}%{REQUEST URI} [END,NE,R=permanent]
</VirtualHost>
```

www.domain2.com-le-ssl.conf

```
<IfModule mod_ssl.c>
<VirtualHost *:443>
        ServerName www.domain2.com
        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html/www.domain2.com
        <Directory /var/www/html/www.domain2.com>
                Options FollowSymLinks
                AllowOverride All
                Require all granted
        </Directory>
        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE LOG DIR}/access.log combined
SSLCertificateFile /etc/letsencrypt/live/www.domain2.com/fullchain.pem
SSLCertificateKeyFile /etc/letsencrypt/live/www.domain2.com/privkey.pem
Include /etc/letsencrypt/options-ssl-apache.conf
</VirtualHost>
</IfModule>
```

https://kijima.mydns.jp/dokuwiki/ - Kijima's private wiki

Last undate: 2025/08/30 18:09

