

サーバー日記

大滝 博祐

September 22, 2025

Ubuntu ってことが前提です。mac でも工夫すればできるのかな？ ちょっと面倒なところあるけど。Windows の wsl でもできるはずですよ。

0.1 Apache2 のインストール

```
sudo apt install apache2
sudo systemctl start apache2
```

これにより、`/var/www/html` が作成される。`sudo chown -R hoge:hoge /var/www/html`
`sudo chmod -R 775 /var/www/html`
で書き込み権限を与える

0.1.1 ファイアウォールの有効化

```
sudo ufw enable
```

0.2 転送先 PC の IP アドレスを確認

ルータはただのルータなので、ルータにきた通信をそのルータがつながっているローカルネットワーク内のどこかの PC に転送してもらって、そっちで処理してもらう必要がある。その PC を決めて `ip a` で IP アドレスを確認。

0.3 ポート開放, ポート転送

まず、ブラウザから家庭用ルータの設定画面に入る。初期ユーザ名、パスワードは大体 `user`, `user` である。(例) `http://192.168.3.1` BB ユニットの場合は、ルータ機能の設定 -> ポート転送設定より、WAN 側ポート、LAN 側ポートともに 80 番ポート (実際はどれでもよかったりする) を有効にする。転送先 IP アドレスに転送先 PC の IP アドレスを入力する。`□-□` という表記の場合は、`□` 番から `□` 番ポートという意味なので同じ値を入力すれば、実質 1 つの番号を指していることになる。

0.4 ファイアウォールもポート開放！

ルータをポート開放してもファイアウォールがブロックしちゃうから、ファイアウォールにも穴をあける。ssh 用 22 番ポート開放

```
sudo ufw allow 22/tcp
```

3000 番ポート開放

```
sudo ufw allow 3000/tcp 80 番ポート閉鎖
```

```
sudo ufw delete allow 80/tcp
```

0.5 ルータのグローバル IP アドレスの確認

```
curl -4 ifconfig.me
(例)126.77.251.164
```

0.6

(例) `http://126.77.251.164:80` で転送先 PC の `/var/www/html/index.html` が表示される。

0.7 80 番ポート以外で

80 -> 8080 の場合

`/etc/apache2/ports.conf` の `Listen 80` を `Listen 8080`

`/etc/apache2/sites-enabled/000-default.conf` の 80 も 8080 にする。

もちろんルータの転送設定も変える。`sudo systemctl restart apache2` でサーバーを再起動

0.8 確認コマンド

- `sudo ufw status verbose`
ファイアウォールの詳細ステータス
- `sudo lsof -i :3000`
3000 番ポートを使用しているプロセスを調べる。

0.9 注意！

同じネットワーク内でグローバル IP を使ってサーバーにアクセスしたり、ssh したりはできない！ なぜならば、ループバック非対応のルータが多いから。

Chapter 1

コメント欄を作る

1.1

Apache と相性がいいのは PHP なので、Apache に PHP をインストールする。

```
sudo apt install php libapache2-mod-php
```

```
sudo systemctl restart apache2
```

1.2 HTML ファイルではなく PHP ファイルに

PHP ファイルにこれを追加\\

```
<!-- コメント欄 -->
<div style="padding: 20px;">
  <h2>コメント欄</h2>
  <form id="commentForm" method="POST" action="submit_comment.php">
    date_default_timezone_set('Asia/Tokyo'); //タイムゾーン設定
    <input type="text" name="name" placeholder="名前" required><br><br>
    <textarea name="comment" placeholder="コメントを入力してください" required></textarea><br><br>
    <button type="submit">送信</button>
  </form>

  <hr>

  <div id="commentList">
    <?php include 'comments.php'; ?>
  </div>
</div>
```

submit_comment.php に

```
<?php
if ($_SERVER["REQUEST_METHOD"] === "POST") {
    $name = htmlspecialchars($_POST["name"]);
    $comment = htmlspecialchars($_POST["comment"]);
    $entry = [
        "name" => $name,
        "comment" => $comment,
        "time" => date("Y-m-d H:i:s")
    ];

    $file = "comments.json";
    $comments = [];

    if (file_exists($file)) {
        $json = file_get_contents($file);
        $comments = json_decode($json, true);
    }

    $comments[] = $entry;
    file_put_contents($file, json_encode($comments, JSON_UNESCAPED_UNICODE | JSON_PRETTY_PRINT));

    // リダイレクトしてフォーム再送防止
    header("Location: " . $_SERVER["HTTP_REFERER"]);
}
```

```
    exit;
}
?>
```

comments.php に

```
<?php
$file = "comments.json";

if (file_exists($file)) {
    $comments = json_decode(file_get_contents($file), true);
    if ($comments) {
        // 新しいコメントが上に来るように逆順で表示
        foreach (array_reverse($comments) as $c) {
            echo "<p><strong>" . htmlspecialchars($c["name"]) . "</strong> (" . $c["time"] . ")<br>";
            echo nl2br(htmlspecialchars($c["comment"])) . "</p><hr>";
        }
    } else {
        echo "<p>コメントはまだありません。</p>";
    }
} else {
    echo "<p>コメントはまだありません。</p>";
}
?>
```

comments.json に最初は

```
[]
```

chmod 666 comments.json

でユーザーに書き込み権限を与えないとコメントできない。

コメントされると、これに付け足されていく。逆に、これを削除したら対応するコメントが消えるし、これを変えたらコメントの内容を改ざんできる。

1.3 index.php を表示させる

/etc/apache2/mods-enabled/dir.conf に、表示させる優先順位が書かれているので、一番目を index.php にする。

Chapter 2

ドメイン

家庭用回線のグローバル IP アドレスは動的なので、定期的に変更されるのでこれをサーバーとして使うのはあまりにも無責任なので、DNS を使って、ドメイン名と最新の IP アドレスを紐付けさせる。家庭用サーバーに今回は Duck DNS を使ってドメインを取得する。hirosuke.duckdns.org というドメインを取得したなら、<http://hirosuke.duckdns.org:3000> をブラウザで開く (3000 番ポートの場合)。

Chapter 3

sshポート設定

釧路行ってからも家のルータでサーバー管理しなきゃいけないでしょ？ 遠く離れた釧路からこっちのルータに ssh したら、ルータが転送先 PC に転送してその PC に ssh される仕組み。その PC とサーバーの転送先 PC を同じにしたら、ssh でサーバー管理できる。サーバーと呼べるような大したサーバーじゃないけど。

ssh 用のポートは 22 番ポート。22 番ポートを開放して、家の PC に転送されるようにすれば、curl -4 ifconfig.me でルータのグローバル IP を確認して、ssh ユーザー名@グローバル IP でその PC に ssh できる。

3.0.1 注意！

リモートかローカルかをちゃんと意識して、mv とかじゃなくて scp キーが変更された時は

3.1 ssh 鍵認証

1. ssh-keygen -t ed25519 -f ~/.ssh/ 【鍵ファイル名】
2. ssh-copy-id -i ~/.ssh/ 【公開鍵ファイル名】 【ユーザー名】 @ 【接続先のホスト名または IP アドレス】
3. ~/.ssh/config に
Host 【エイリアス名】
HostName 【接続先のホスト名または IP アドレス】
User 【ユーザー名】
IdentityFile ~/.ssh/ 【秘密鍵ファイル名】
4. ssh 【エイリアス名】

3.1.1 local と global どちらからも

```
Host hoge-local
HostName 192.168.3.15
User yuko
IdentityFile ~/.ssh/yuko_ubuntu

Host hoge-global
HostName 126.77.251.164
User yuko
IdentityFile ~/.ssh/yuko_ubuntu
```

#できないときは、再確認してsotaの cat yuko.pubをyukoの ~/.ssh/authorized_keys にコピー

3.2 scp (セキュアコピー)

scp [コピー元パス] [ユーザー名@IP: 相手のパス]
ディレクトリを scp するときは scp -r [コピー元パス] [ユーザー名@IP: 相手のパス]

3.3 鍵認証を用いた scp

scp -i <秘密鍵ファイル> [オプション] <コピー元パス> <ユーザー名@IP: コピー先パス>

3.3.1 関数で作っちゃおう (.zshrc)

```
grasscp() {  
    scp -i ~/.ssh/raspi "$1" hirosuke@126.77.251.164:"$2"  
}
```

実行例

```
grasscp fuga.txt "~"
```

~はこっち側で展開されないようにクォートで囲む

(例) scp -i ~/.ssh/raspi ./hoge.txt hirosuke@126.77.251.164:

3.4 差分コピー (rsync) アールシンク、remote synchronization(リモートシンクロ)

rsync [オプション] コピー元 コピー先

(例) rsync -av [コピーもと] [コピー先]

a ディレクトリ (archive) (例) rsync -av -e ssh コピー元 ユーザー@IP: コピー先 (例) rsync -av -e ssh /local/path/ user@remote-host:/remote/path/