

サーバ構築の手順書

作成日：2024 年度 11 月 27 日

作成者：中尾 晃太郎

目次

1 はじめに.....	2
2 環境.....	2
3 導入.....	5
4 ソフトウェアのインストール.....	6
5 Apache の設定	7
6 ファイアウォールの設定.....	10

1 はじめに

本資料は、サーバ構築に関する手順書である。主に、これからサーバを構築する初学者を対象としており、まっさらな PC に対して Web サーバを立ち上げる方法を解説する。Linux OS を利用し、特定のディストリビューションには依存せず、幅広い環境での適用が可能な内容とした。また、サーバのセットアップから、実際に Web サーバを構築するまでの手順を具体的に示すことで、実践的なスキルを習得できるように配慮している。これにより、初学者が自身の環境でサーバを構築し、運用するための基盤を築くことを目的としている。

2 環境

今回、デスクトップパソコンを使用する。使用する OS、バージョンは以下に示す。

表 1 使用する OS

OS	バージョン
Ubuntu	22.04

Ubuntu をサイトでダウンロードする。

[Ubuntu Desktop 日本語 Remix のダウンロードを開く。](#)



図 1 Ubuntu Desktop 日本語 Remix のダウンロード

ミラーサイト(ここでは [KDDI 研究所](#))から「ubuntu-ja-22.04-desktop-amd64.iso」(3.2GB)をダウンロードする

Index of /Linux/packages/ubuntu-jp/release-cd/releases/jammy





Name	Last modified	Size	Description
 Parent Directory		-	
 SHA256SUMS	2022-05-06 12:08	208	
 ubuntu-ja-22.04-desktop-amd64.iso	2022-05-02 12:06	3.2G	
 ubuntu-ja-22.04-desktop-amd64.iso.torrent	2022-05-06 12:07	256K	

図 2 KDDI 研究所のダウンロードサイト

[Rufus のサイト](#)を開き、**rufus-4.5.exe**(タイプ:標準 プラットフォーム:Windows x64 サイズ:1.4MB)をダウンロードする
バージョンが最新になっていたら[他のバージョン](#)の所からダウンロードする

最新リリース:				
リンク	タイプ	プラットフォーム	サイズ	日付
rufus-4.5.exe	標準	Windows x64	1.4 MB	2024.05.22
rufus-4.5p.exe	Portable	Windows x64	1.4 MB	2024.05.22
rufus-4.5_x86.exe	標準	Windows x86	1.5 MB	2024.05.22
rufus-4.5_arm64.exe	標準	Windows ARM64	4.8 MB	2024.05.22
rufus-4.6_BETA.exe	BETA	Windows x64	1.5 MB	2024.10.10
他のバージョン (GitHub)				
他のバージョン (FossHub)				

図 3 Rufus のダウンロードサイト

exe を起動し、USB(16GB)を挿入する。次のように設定し、ダウンロードを開始する。

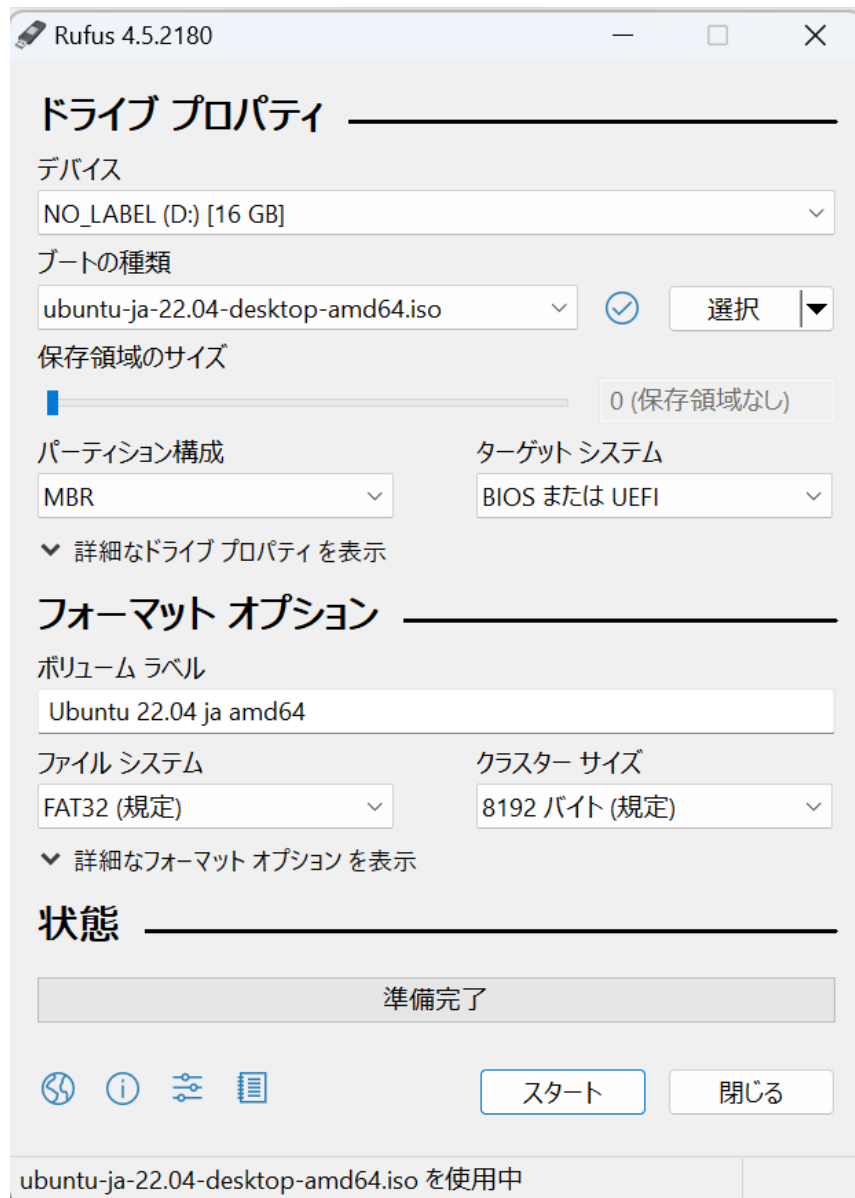


図 4 Rufus の起動

3 導入

デスクトップパソコンがシャットダウンしていることを確認してから、先ほど Ubuntu をダウンロードした USB を挿して起動する。起動したら BIOS(ここでは UEFI)画面になるようにして、USB からブートできるようにする。

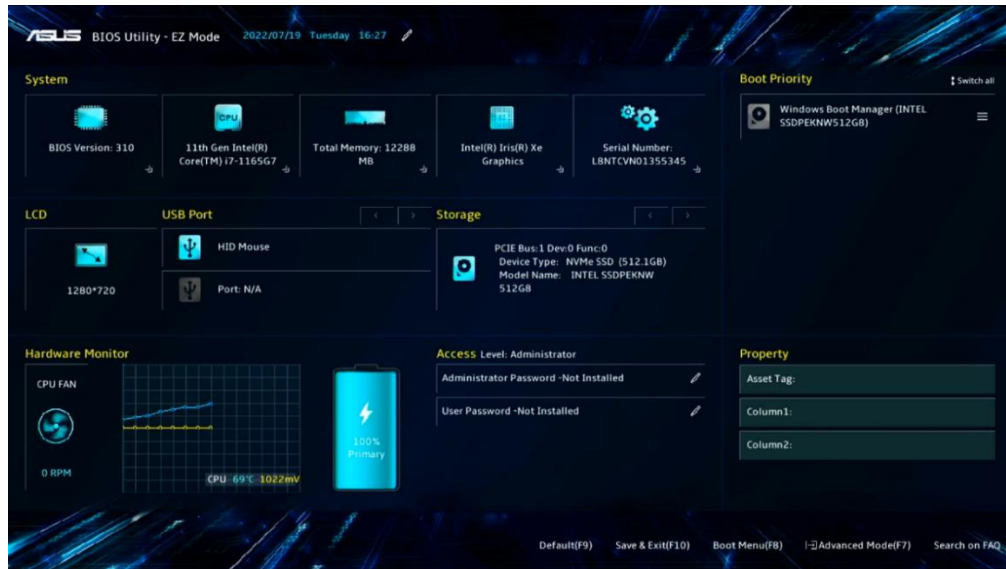


図 6 BIOS(UEFI)の画面

「GNU GRAB」の画面で Ubuntu が選択されていることを確認して進む。起動したら Ubuntu をインストールする操作をする。住んでいる場所や名前、パスワードなどの情報を入力し、インストールを開始する。再起動してログイン出来たら完了。



図 7 Ubuntu のインストール画面

4 ソフトウェアのインストール

ここからは Apache をインストールし、サーバ構築を試みる。Apache はオープンソースソフトウェアであり、Linux 環境における Web サーバとして高いシェアを持っている。

「端末」を開き、以下のコマンドを実行して Apache をインストールする。

[Apache のインストール]

```
$ sudo apt update    #パッケージの更新  
$ sudo apt -y install apache2    #Apache HTTP Server のインストール
```

インストールした後は Apache を起動し、サーバのステータスを確認する。正常に起動できていれば **Active : active(running)** と表示される。

[Apache の起動]

```
$ sudo systemctl start apache2    #Apache の起動  
$ sudo systemctl enable apache2    #Apache の自動起動の有効化  
$ sudo systemctl status apache2    #Apache のステータス
```

その後、ブラウザから「https://<サーバの IP アドレス>」にアクセスし、Apache のウェルカムページが表示されれば完了である。サーバ上で Web ページにアクセスする場合、「http://localhost」とすると表示できる。

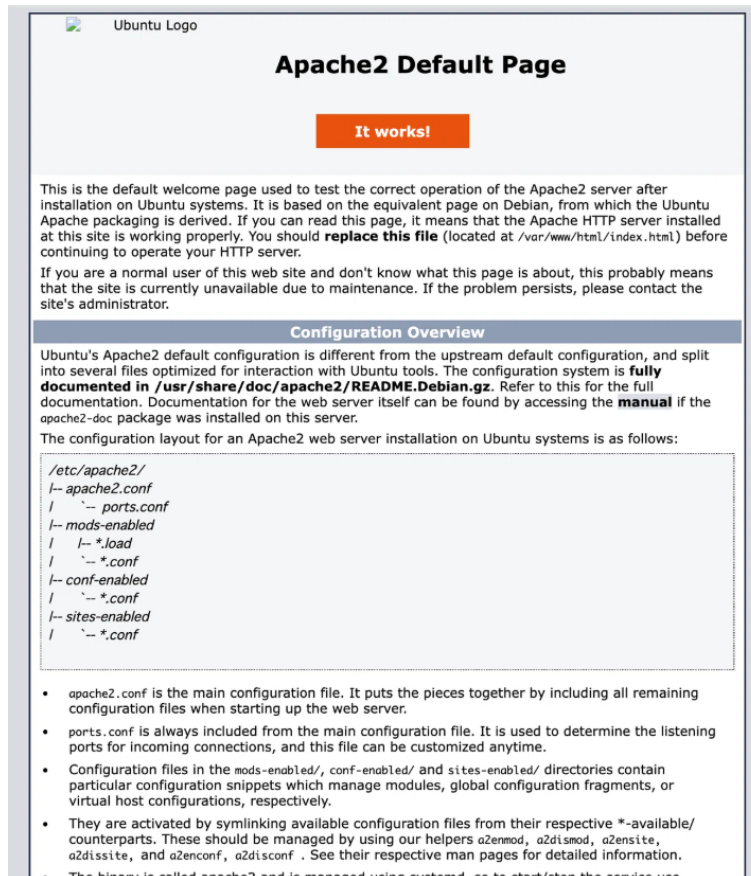


図 8 Apache のウェルカムページ

5 Apache の設定

Apache の設定ファイルは「`/etc/apache2/apache2.conf`」や「`/etc/apache2/sites-available/000-default.conf`」で管理される。

Web ページに関する設定をする場合は、基本的に「`000-default.conf`」を変更、もしくは新たに設定ファイルを作成する。

Apache の設定ファイルで記述可能な設定項目として、`DocumentRoot`、`DirectoryIndex` がある。これらは一般的に Web サーバの設定を変更する可能性がある項目であるため、以降では設定方法を紹介する。

Apache をインストールした直後は、「`/etc/apache2/sites-available/000-default.conf`」がデフォルトの仮想ホストとなっている。デフォルトの設定ファイルを残したまま異なる仮想ホストを作りたい場合は「`/etc/apache2/sites-available/`」直下に新たな設定ファイルを作成する。

ファイル名を決め、拡張子を.conf として作成する。

[設定ファイルの作成]

```
$ sudo touch /etc/apache2/sites-available/ファイル名.conf
```

その後、作成したファイルに以下の内容を記述する。**ディレクトリパス**には自身がコンテンツをデプロイするディレクトリを、**ファイル名**にはトップページとして表示したいファイルを指定する。

[設定ファイルの編集]

```
<VirtualHost *:80>
    ServerAdmin webmaster@localhost
    DocumentRoot ディレクトリパス
    DirectoryIndex ファイル名
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

指定したディレクトリやファイルが存在しない場合は以下の様にして作成する。

[ディレクトリの作成]

```
$ sudo touch ディレクトリパス/ファイル名
$ sudo vi ディレクトリパス/ファイル名
```

<--ファイルに記述する例-->

```
<html>
  <body>
    <h1>Welcome to my page</h1>
  </body>
</html>
```

作成した仮想ホストを反映させるためには、仮想ホストの有効化をする必要がある。

[仮想ホストの有効化]

```
$ sudo a2ensite ファイル名.conf #仮想ホストの有効化
$ sudo systemctl reload apache2 #有効化した仮想ホストをサーバに反映
```

a2ensite コマンドは、Apache HTTP サーバの設定で使用するコマンドの一つであり、仮想ホストの有効化を行う。**reload** は Apache の再起動を行うコマンドである。これにより、新しい仮想ホストを反映することが出来る。

一方、既にエントリしている仮想ホストを除外したい場合は、**a2dissite** コマンドを利用する。作成した仮想ホストが正しく動作しているのかを確認するために、デフォルトの仮想ホストを無効化する。

[デフォルトの仮想ホストの無効化]

```
$ sudo a2dissite 000-default.conf #仮想ホストの無効化
$ sudo systemctl reload apache2 #有効化した仮想ホストをサーバに反映
```

サーバにアクセスし、正常に表示されていたら成功。

6 ファイアウォールの設定

Ubuntu を含む Linux システムではセキュリティを強化するためにファイアウォールが必要になる。ファイアウォールは、許可されていないトラフィックをブロックし、攻撃や不正アクセスからシステムを保護してくれる。

Ubuntu では、ファイアウォールの設定に `ufw` コマンドを利用する。構築した Web サーバへのアクセスを許可するには以下のコマンドを実行する。

[ファイアウォールの有効化とポート開放]

```
$ sudo ufw enable  #ファイアウォールの有効化
$ sudo ufw allow  'Apache'  #80 番ポートの開放
$ sudo ufw status
Status: active

To Action From
--
Apache ALLOW Anywhere
Apache (v6) ALLOW Anywhere (v6)
```

正常に実行できれば、ステータスを確認した際に上記のような結果が得られる。

これでまっさらな PC に Web サーバを立ち上げる事が出来た。Web の外観などは作成したディレクトリの中のファイルを編集すれば変えることが出来る。

7 参考文献

Ubuntu20.04 : インストールと設定

<https://demura.net/misc/20879.html>(2024-11-27)

Ubuntu22.04 : インストールと設定

<https://demura.net/misc/23443.html>(2024-11-27)

Windows 10 と Ubuntu 18.04 デュアルブートする方法

<https://www.pc-koubou.jp/magazine/35542>(2024-11-27)

Ubuntu の apt を調べたのでまとめておく

<https://zenn.dev/ikkik/articles/ffa45ed1fe5a1a>(2024-11-27)