

これから始めるUbuntu入門

～Ubuntuのインストールとパッケージ管理

日本仮想化技術株式会社

水野 源



- 日本仮想化技術株式会社 技術部所属
 - クラウドを利用したDevOps案件等を担当
- Ubuntu Japanese Team Member
- Ubuntu.com Member
- Debian Package Maintainer
- 日経Linux誌で「Linux100%活用ガイド」連載中



本セミナーの想定受講者

- これからLinuxを学習しようと考えている方。
- Ubuntu、Debian GNU/Linux以外のディストリビューションの利用経験があり、Ubuntuについて知りたい方。



本セミナーのゴール

- Ubuntuの特徴を理解することができる。
- Ubuntuを使った基本的なシステムの構築ができる。



本日のアジェンダ

1. Ubuntuの特徴
2. Ubuntuのリリース
3. Ubuntuのインストール
4. Ubuntuのパッケージ管理
5. 【実例】UbuntuでWebサーバーを構築する
6. 参考資料

Ubuntuの特徴



Ubuntuとは

- Debian開発者であったMark Shuttleworthによって作られた、Debianベースの派生ディストリビューション。
- 名前の由来はズールー語で「思いやり」。
- 最初は使いやすいデスクトップOSを作ること为目标として開発開始。
- 最初のリリースは2004年10月。



Mark Shuttleworth by Martin Schmitt
from Wikimedia



Ubuntuは無償で利用可能

- Ubuntuの理念として「人はソフトウェアを自分たちの言語で、あらゆる障害に関わらず、無償で使用できるべきである」という考えがある。
- OSとしてのすべての機能は無償で使用できる。
 - もちろん営利目的での利用も可能。
- Ubuntuはプロジェクトが存続する限り、永久にUbuntuを無償で利用できることと、「エンタープライズ版」のような有償バージョンを作ることにはしないことを宣言している。
 - ただし有償でのエンタープライズ向けサポートは存在する。



- Ubuntuの開発の主体は**Ubuntuコミュニティ**。
- それとは別に、MarkがUbuntuをサポートするために設立した企業がCanonical。
 - CanonicalはフルタイムでUbuntuを開発する開発者を雇用し、セキュリティ修正などを行っている。
 - Canonicalはコミュニティに資金的な援助を行うなど、パトロン的存在でもある。
 - 勘違いされやすいが、UbuntuはCanonicalの製品というわけではない(Red HatとRHELのような関係ではない)。
- 緊急時に備え、Ubuntu財団という財団に運転資金がプールされている(初期投資で1000万ドル)。
- コミュニティによる開発体制と、企業による資金、技術的なサポートが両立しているところがUbuntuの特徴のひとつ。



- デスクトップとして
 - 個人、企業、学校などでも広く利用されている。
 - 開発者向けにも。
- サーバーやコンテナのベースOSとして
 - Linuxを利用していると判明しているWebサイトのうち45.7%がUbuntu。
 - 各リリースには最新のOpenStackが含まれている。
 - 3DCG映画のレンダリングクラスターでの大規模利用実績あり。
- IoT分野で
 - Raspberry Piも公式にサポート。
 - ROS/ROS 2が公式にサポートしているLinuxもUbuntu。
 - ロボット関係でも標準的に使われている。
- デスクトップ、サーバー、クラウド、コンテナ、IoTまで幅広く利用されているプラットフォーム。



エンタープライズ向け有償サポート

- OS自体は無償でフル機能が使えるが、それとは別にエンタープライズ向けのサポートが、Canonicalから有償で提供されている。
- 有償サポートパッケージ「Ubuntu Advantage」
 - 多数のUbuntuサーバーを管理するツール「Landscape」
 - 再起動なしにカーネルにパッチを当てる「Livepatch」
 - Extended Security Maintenanceによる延長サポート
 - Canonicalのナレッジベースへのアクセス
 - 24時間の電話、オンライン窓口……など
- 一番安いEssentialプランは年間225USD/物理サーバーだが、AWSのMarket Placeでは1時間単位での時間課金でも購入できる。



- コミュニティ主体の自由でオープンな開発体制。
- 幅広い用途に対応できる汎用性。
- エンタープライズレベルの有償サポート。
- 資金を含む企業による支援=プロジェクトの継続性。
- Debian由来の豊富なソフトウェア資産。
- 世界的に大きいシェア=サードパーティの対応。
- 半年ごとのリリースによる先進的な機能と、2年ごとにリリースされる長期サポートの両立。
- 一言で言えば、Linuxを使う多く場面において、使いやすくて都合がよいディストリ。

Ubuntuのリリース



Ubuntuのリリースの種類

- Ubuntu Desktop
 - デスクトップ向け。
- Ubuntu Server
 - サーバー向け。標準でGUIを持たない。
- Ubuntu Core
 - 主にIoT向け。
 - セキュリティ重視。
 - Snapパッケージをベースに構築されたUbuntu。
- Ubuntu Cloud
 - 各種クラウド向けイメージ。
 - クラウドベンダーごとに特化したカーネルなどを搭載。



Ubuntuの公式フレーバー

- Ubuntuには味付けの違う「フレーバー」が存在する。
 - Ubuntuをベースに、デフォルトでインストールされるパッケージセットを変更したもの。
 - 使用してるリポジトリやパッケージ自体は同じもの。
- Canonicalではなくコミュニティによるサポート。
 - そのためサポート期間が異なる(短い場合がある)ので注意。
- フレーバーー覧
 - Kubuntu
 - Xubuntu
 - Lubuntu
 - Ubuntu MATE
 - Ubuntu Budgie
 - UbuntuStudio
 - UbuntuKylin



Ubuntuのリリースサイクル

- 半年に一度のタイムベースリリースを採用。
 - 4月と10月の第三木曜日リリース。
- バージョン番号は西暦下2桁.月2桁で表現される。
 - 例: 2020年4月リリース=20.04
- リリースごとにコードネームをつけて管理。
 - 頭韻を踏んだ英単語で「形容詞+動物名」をアルファベット順につけるのがならわし。
 - 開発者間の会話やリポジトリのディレクトリ名などで、コードネームの形容詞部分が使われる。
 - 例: 20.04 = コードネームFocal Fossa = Focal
 - 固有名詞であることを理解していないと、何を言っているか理解できない事もよくある。



Ubuntuのサポート期間

- 通常のサポート期間は9ヶ月。
- 2年に1度、長期サポート版(LTS)がリリースされる。
 - LTSのサポート期間は5年
 - 前述のフレーバーは、サポート期間が異なることがある。
- Extended Security Maintenance(ESM)
 - LTSにのみ、重要なセキュリティ修正だけを提供する延長サポート)が有償で提供されている。
 - 通常サポート5年+ESMサポート5年=10年
- サーバー用途であれば**Ubuntu Serverの最新のLTSを第一候補とするのがよい。**



Ubuntuのリリースとサポート期間

Ubuntu 22.04 LTS

Ubuntu 21.10

Ubuntu 21.04

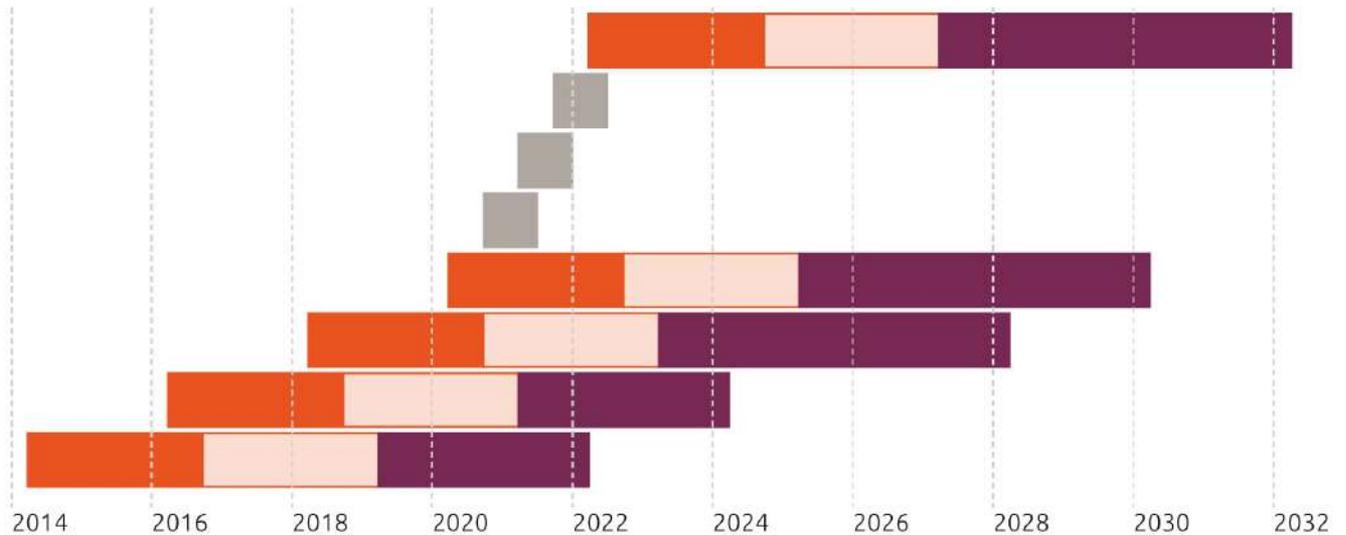
Ubuntu 20.10 (v5.8)

Ubuntu 20.04 LTS (v5.4)

Ubuntu 18.04 LTS

Ubuntu 16.04 LTS

Ubuntu 14.04 LTS



- Hardware and maintenance updates
- Maintenance updates
- Interim release Standard Support
- Extended Security Maintenance (ESM)

from <https://ubuntu.com/about/release-cycle>



LTSとポイントリリース

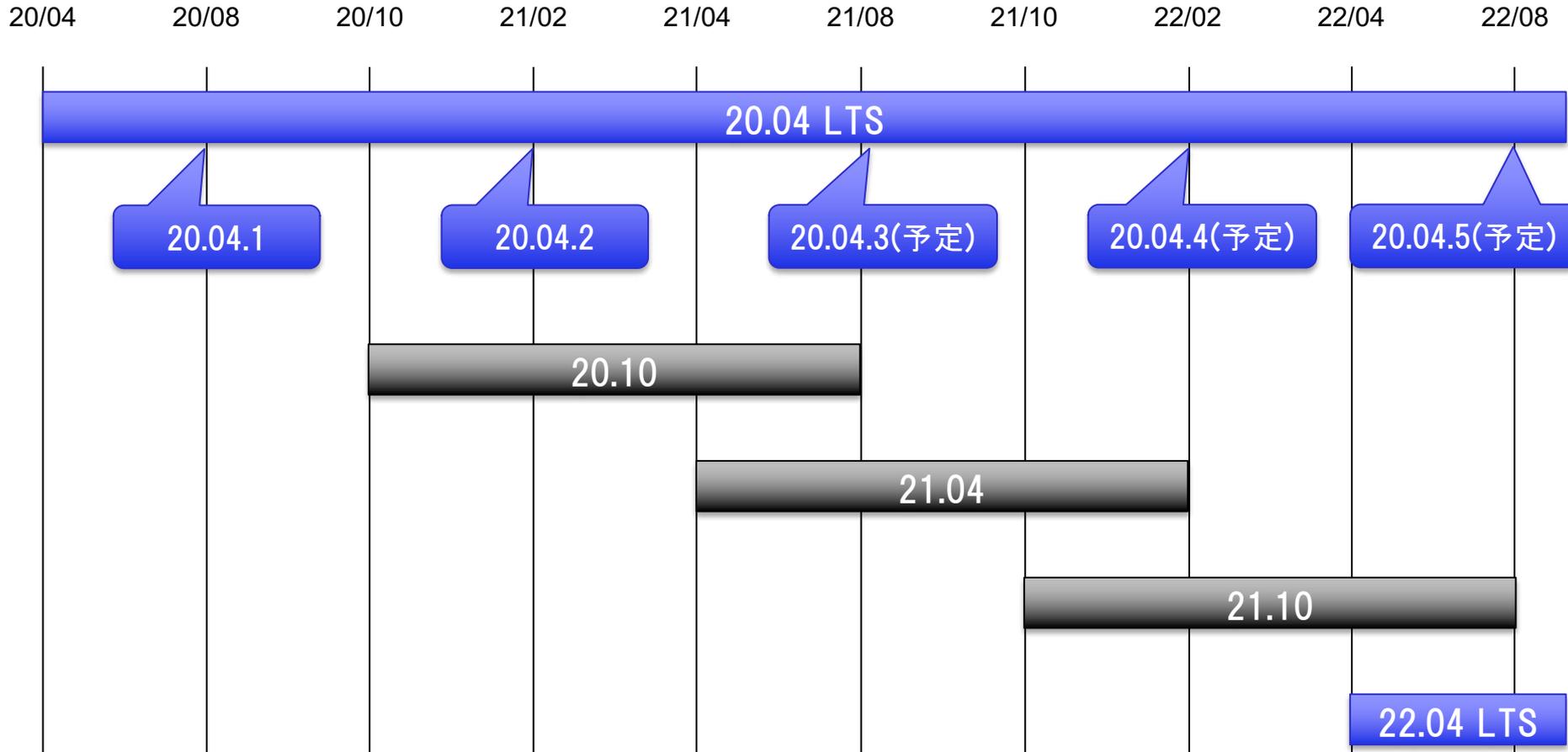
- LTSは5年使えるが、リリースから時間がたつと、新しいハードウェアにインストールできないことがある。
 - インストールしようとしたら、ストレージを認識しない、など。
- アップデートと、最新のUbuntuからバックポートした新しいカーネルを含んだインストールメディアをリリースしよう。
 - これがポイントリリース。
 - 20.04.Xのように、末尾にポイントリリース番号がつく。
- LTSリリースから数ヶ月後に最初のポイントリリースがリリースされる。
 - それから半年ごと(Ubuntuの次のリリースが出る度)に、新しいカーネルをバックポートしたポイントリリースがリリースされていく。
- 次のLTSのカーネルがバックポートされるまでポイントリリースは続く。
 - ポイントリリースはリリースから約2年半(YY.MM.5)で打ち止め。



- ポイントリリースはインストールメディアがリリースされるだけなので、**運用中のOSをアップデートし続けていけばポイントリリースからインストールした環境と同等になる。**
 - 再インストールや、特別なアップグレードは不要。
 - CentOSのマイナーリリースと違い、**新機能追加などは行われない。**
- もしインストールしたいUbuntuのバージョンに、最新のポイントリリースが存在するなら、そっちのメディアからインストールしよう。
 - もちろん古いメディアからインストール可能なら、アップデート後の差異はないので、それほど気にしなくてもいい。



LTSとポイントリリース



Ubuntuのインストール



用意するもの

- Ubuntu Server 20.04.2 LTSのインストールメディア
 - <https://releases.ubuntu.com/20.04/>
- 以下のスペックを満たすサーバー
 - 1GHz以上のIntel/AMD 64bit CPU
 - arm64/ppc64el/s390xにも対応
 - RAM: 1GB以上
 - DISK: 2.5GB以上

ubuntu releases



Select an image

Ubuntu is distributed on four types of images described below.

Desktop image

The desktop image allows you to try Ubuntu without changing your computer at all, and at your option to install it permanently later. This type of image is what most people will want to use. You will need at least 1024MiB of RAM to install from this image.

64-bit PC (AMD64) desktop image

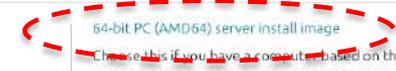
Choose this if you have a computer based on the AMD64 or EM64T architecture (e.g., Athlon64, Opteron, EM64T Xeon, Core 2). Choose this if you are at all unsure.

Server install image

The server install image allows you to install Ubuntu permanently on a computer for use as a server. It will not install a graphical user interface.

64-bit PC (AMD64) server install image

Choose this if you have a computer based on the AMD64 or EM64T architecture (e.g., Athlon64, Opteron, EM64T Xeon, Core 2). Choose this if you are at all unsure.





インストールの流れ

1. 言語の選択
2. キーボードの選択
3. ネットワークの設定
4. プロキシの設定
5. アーカイブサーバーの設定
6. ストレージの設定
7. 初期ユーザーの作成
8. SSHの設定
9. Snapパッケージのインストール



- カーソルキー ↑ ↓ : 項目の選択
 - フォーカスの移動も可能
- TABキー: フォーカスの移動(正順)
 - シフト+TABキーで逆順移動も可能
- スペースキー: 選択肢の選択
- Enterキー: 選択肢の選択
- ドロップダウンリストの操作
 - TABキーでドロップダウンリストをフォーカス
 - Enterキーでリストを開く
 - カーソルキー



- 起動した直後に表示される画面
 - この状態で何もせず待つとインストーラーが起動する。





言語の選択

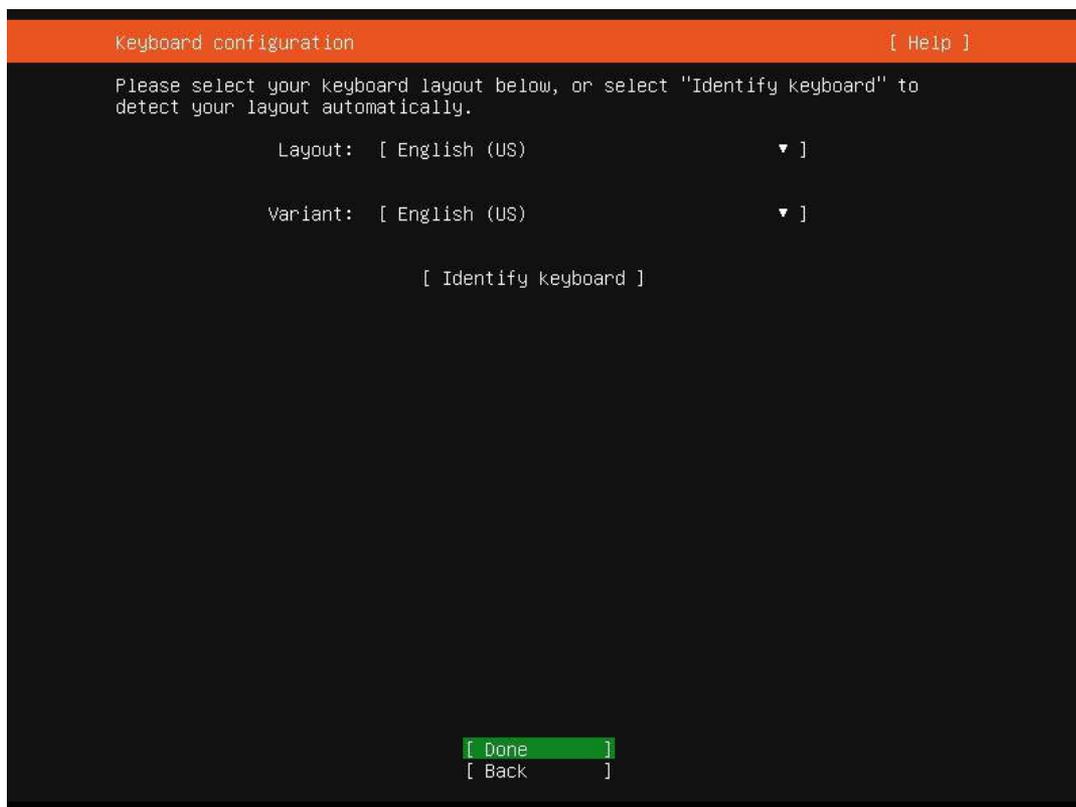
- キーボードのカーソルキーで選択してEnterで決定。
- 日本語はない。
 - サーバーのコンソールで日本語は使いづらいため、Englishをおすすめ。

```
Willkommen! Bienvenue! Welcome! Добро пожаловать! Willkommen! [ Help ]  
  
Use UP, DOWN and ENTER keys to select your language.  
  
[ Asturianu ]>  
[ Bahasa Indonesia ]>  
[ Català ]>  
[ Deutsch ]>  
[ English ]>  
[ English (UK) ]>  
[ Español ]>  
[ Français ]>  
[ Hrvatski ]>  
[ Latviski ]>  
[ Lietuviškai ]>  
[ Magyar ]>  
[ Nederlands ]>  
[ Norsk bokmål ]>  
[ Polski ]>  
[ Suomi ]>  
[ Svenska ]>  
[ Čeština ]>  
[ Ελληνικά ]>  
[ Беларуская ]>  
[ Русский ]>  
[ Српски ]>  
[ Українська ]>
```



キーボードの選択

- 言語でEnglishを選択したため、キーボードも英語になっている。
 - 日本語キーボードを使っているならJapaneseに変更する。
 - ドロップダウンリストなのでLayoutをフォーカスしてスペースキー。





ネットワークの設定

- デフォルトはDHCPになっている。
- ネットワークに接続できない状態でもインストールは可能。

```
Network connections [ Help ]

Configure at least one interface this server can use to talk to other machines,
and which preferably provides sufficient access for updates.

NAME    TYPE  NOTES
[ enp0s3 eth - ]
DHCPv4  10.0.2.15/24
08:00:27:9d:5d:d6 / Intel Corporation / 82540EM Gigabit Ethernet Controller
(PRO/1000 MT Desktop Adapter)

[ Create bond ▶ ]

[ Done ]
[ Back ]
```



IPアドレスの手動設定

- 設定を変更したいNICを選択し、「Edit IPv4IP」で手動設定が可能。

```
Network connections [ Help ]

Configure at least one interface this server can use to talk to other machines,
and which preferably provides sufficient access for updates.

NAME      TYPE  NOTES
[ enp0s3  eth  - ]
  DHCPv4  10.0.2.15/24
           08:00:27:9d:5d:d6 / Intel Corpor
           (PRO/1000 MT Desktop Adapter)
[ Create bond ▶ ]

[ Done ]
[ Back ]
```

Context menu for enp0s3:

- (close)
- Info
- Edit IPv4
- Edit IPv6
- Add a VLAN tag



IPアドレスの手動設定

- IPv4 Methodを「Manual」に変更。
- AddressやSubnetなどを入力。
 - ここで設定したIPアドレスはインストール後も引き継がれる。

The screenshot shows a terminal window titled "Network connections" with a "[Help]" link in the top right. Below the title bar, there is a text instruction: "Configure at least one interface this server can use to talk to other machines, and which preferably provides sufficient access for updates." Below this is a table with columns "NAME", "TYPE", and "NOTES".

The main content is a dialog box titled "Edit enp0s3 IPv4 configuration". It contains the following fields and controls:

- IPv4 Method: [Manual ▼]
- Subnet: [Input field]
- Address: [Input field]
- Gateway: [Input field]
- Name servers: [Input field]
IP addresses, comma separated
- Search domains: [Input field]
Domains, comma separated

At the bottom of the dialog box are two buttons: [Save] and [Cancel].

Below the dialog box, in the terminal window, are two buttons: [Done] and [Back].



- インターネットアクセスにプロキシが必要な場合は設定する。
 - プロキシが不要な場合は空欄のままでよい。

```
Configure proxy [ Help ]

If this system requires a proxy to connect to the internet, enter its details
here.

Proxy address: 

If you need to use a HTTP proxy to access the outside world,
enter the proxy information here. Otherwise, leave this blank.

The proxy information should be given in the standard form of
"http://[[user]][:pass]@host[:port]/".

[ Done ]
[ Back ]
```



アーカイブサーバーの設定

- パッケージをダウンロードするアーカイブサーバーを設定する。
 - IPアドレスから地理的に近いミラーが自動的に選択される。

```
Configure Ubuntu archive mirror [ Help ]

If you use an alternative mirror for Ubuntu, enter its details here.

Mirror address: http://jp.archive.ubuntu.com/ubuntu
                You may provide an archive mirror that will be used instead of
                the default.

[ Done ]
[ Back ]
```



- 使用するディスク、LVM、暗号化の有無などを選択する。
- 手動でパーティションやRAIDを設定することも可能。

```
Guided storage configuration [ Help ]
Configure a guided storage layout, or create a custom one:
( X ) Use an entire disk
      [ VBOX_HARDDISK_VBe28e8d7a-e894ba67 local disk 10.000G ▼ ]
      [ X ] Set up this disk as an LVM group
            [ ] Encrypt the LVM group with LUKS
                  Passphrase:
                  Confirm passphrase:
( ) Custom storage layout

[ Done ]
[ Back ]
```



ストレージの設定の確認

- デフォルトではLVMを使用する。
- 暗号化はなし。

```
Storage configuration [ Help ]

FILE SYSTEM SUMMARY

MOUNT POINT      SIZE  TYPE  DEVICE TYPE
[ /               8.996G new ext4 new LVM logical volume ▶ ]
[ /boot          1.000G new ext4 new partition of local disk ▶ ]

AVAILABLE DEVICES

No available devices

[ Create software RAID (md) ▶ ]
[ Create volume group (LVM) ▶ ]

USED DEVICES

DEVICE                                TYPE                SIZE
[ ubuntu-vg (new)                      LVM volume group   8.996G ▶ ]
ubuntu-lv    new, to be formatted as ext4, mounted at / 8.996G ▶ ]

[ VBOX_HARDDISK_VBe28e8d7a-e894ba67    local disk         10.000G ▶ ]
partition 1 new, bios_grub     1.000M ▶ ]
partition 2 new, to be formatted as ext4, mounted at /boot 1.000G ▶ ]
partition 3 new, PV of LVM volume group ubuntu-vg 8.997G ▶ ]

[ Done ]
[ Reset ]
[ Back ]
```



ストレージの変更の確認

- Continueを選択すると、ストレージに変更が書き込まれる。
- この時点でなら、インストールの中断が可能。

```
Storage configuration [ Help ]

FILE SYSTEM SUMMARY

MOUNT POINT  SIZE  TYPE  DEVICE TYPE
[ /           8.996G new ext4 new LVM logical volume ▶ ]
[ /boot      1.000G new ext4 new partition of local disk ▶ ]

AVAILABLE DEVICES

----- Confirm destructive action -----

Selecting Continue below will begin the installation process and
result in the loss of data on the disks selected to be formatted.

You will not be able to return to this or a previous screen once the
installation has started.

Are you sure you want to continue?

          [ No          ]
          [ Continue    ]

[ Done      ]
[ Reset     ]
[ Back      ]
```



初期ユーザーの作成

- ユーザー名、サーバー名、パスワードを設定する。
- ここで作成したユーザーがサーバーの管理者(sudo可能)となる。

```
Profile setup [ Help ]

Enter the username and password you will use to log in to the system. You can
configure SSH access on the next screen but a password is still needed for
sudo.

Your name: _____

Your server's name: _____
The name it uses when it talks to other computers.

Pick a username: _____

Choose a password: _____

Confirm your password: _____

[ Done ]
```



SSHの設定

- UbuntuはデフォルトでSSHがインストールされない。
- GitHubやLaunchpadからSSH公開鍵のインポートが可能。

```
SSH Setup [ Help ]

You can choose to install the OpenSSH server package to enable secure remote
access to your server.

[ ] Install OpenSSH server

Import SSH identity: [ No          ▼ ]
                    You can import your SSH keys from Github or Launchpad.

Import Username:

[X] Allow password authentication over SSH

[ Done      ]
[ Back     ]
```



Snapパッケージのインストール

- Snapとは、Canonicalが開発しているユニバーサルパッケージ。
- 代表的なSnapパッケージをインストールできる。
 - 不要であれば省略可。

```

Featured Server Snaps [ Help ]

These are popular snaps in server environments. Select or deselect with SPACE,
press ENTER to see more details of the package, publisher and versions
available.

[ ] microk8s      Lightweight Kubernetes for workstations and appliance
[ ] nextcloud    Nextcloud Server - A safe home for all your data
[ ] wekan        Open-Source Kanban
[ ] kata-containers  Lightweight virtual machines that seamlessly plug int
[ ] docker       Docker container runtime
[ ] canonical-livepatch  Canonical Livepatch Client
[ ] rocketchat-server  Group chat server for 100s, installed in seconds.
[ ] mosquitto     Eclipse Mosquitto MQTT broker
[ ] etcd          Resilient key-value store by CoreOS
[ ] powershell   PowerShell for every system!
[ ] stress-ng     A tool to load, stress test and benchmark a computer
[ ] sabnzbd       SABnzbd
[ ] wormhole      get things from one computer to another, safely
[ ] aws-cli       Universal Command Line Interface for Amazon Web Servi
[ ] google-cloud-sdk  Command-line interface for Google Cloud Platform prod
[ ] slcli         Python based SoftLayer API Tool.
[ ] doctl         The official DigitalOcean command line interface
[ ] conjure-up    Package runtime for conjure-up spells
[ ] minidlna-escoand  server software with the aim of being fully compliant
[ ] postgresql10  PostgreSQL is a powerful, open source object-relation
[ ] heroku        CLI client for Heroku
[ ] keepalived    High availability VRRP/BFD and load-balancing for Lin
[ ] prometheus    The Prometheus monitoring system and time series data
[ ] juju          A model-driven operator lifecycle manager

[ Done ]
[ Back ]

```



インストールの完了

- ログが表示されるので、完了まで待つ。
- Reboot Nowと表示されたらインストールは完了なので再起動する。
 - running 'curtin hook' の右側でクルクル回り続けるのはバグです。

```
Install complete! [ Help ]

running '/snap/bin/subiquity.subiquity-configure-apt
/snap/subiquity/2280/usr/bin/python3 true'
curtin command apt-config
curtin command in-target
running 'curtin curthooks'
curtin command curthooks
  configuring apt configuring apt
  installing missing packages
  configuring iscsi service
  configuring raid (mdadm) service
  installing kernel
  setting up swap
  apply networking config
  writing etc/fstab
  configuring multipath
  updating packages on target system
  configuring pollinate user-agent on target
  updating initramfs configuration
  configuring target system bootloader
  installing grub to target devices
finalizing installation
  running 'curtin hook' \
final system configuration
  configuring cloud-init
  installing openssh-server
  restoring apt configuration
  downloading and installing security updates
subiquity/Late/run

[ View full log ]
[ Reboot Now ]
```



Ubuntuのログイン画面

- あとは一般的なLinuxと同じ。

```
Ubuntu 20.04.2 LTS focal tty1
focal login:
```

Ubuntuのパッケージ管理



このセクションの内容

1. Ubuntuにおけるrootユーザーの扱い
2. Ubuntuのパッケージ形式についての概要
3. APTパッケージマネージャーの使い方
4. aptコマンドとapt-getコマンドの違い
5. Ubuntuのカーネルパッケージの仕組み
6. パッケージのアップデート方針
7. Ubuntuのリポジトリの構成



Ubuntuにおけるrootユーザー

- パッケージ操作にはroot権限が必要。
- ただしUbuntuではrootを直接使用しない。
 - rootアカウントは直接ログインできないよう、デフォルトで無効。
 - インストール時に作成したユーザーで都度sudoするのが基本。
 - 追加したユーザーも、sudoグループに所属させることでsudo可能に。
- rootでログインしたい/シェルを取りたいんだけど？
 - `sudo -i`もしくは`sudo -s`を使ってください。
 - `sudo su -`とかする必要はない。
- rootにパスワードを設定してはいけない。
 - rootにパスワードを設定し直すことで、rootアカウントを有効にして、CentOSのように使うこと自体は可能。
 - ただし、Ubuntuはrootが無効になっていることを前提に設計されているため、**セキュリティ的に脆弱な部分を作り出す可能性がある。**



Ubuntuのパッケージ

- Linuxディストリビューションでは、なんらかのパッケージ管理システムを使って、ソフトウェアをパッケージとして扱うのが一般的。
- UbuntuではDebian由来の**Debパッケージ**と、**ユニバーサルパッケージであるSnap**が利用されている。
 - ここではDebについて解説。



Debパッケージとは

- Debian由来のパッケージフォーマット。
- ファイルの実体はar形式のアーカイブ。
 - tarではない。
- 中身はシステムにインストールするファイルそのもの、インストール/アンインストール時に実行される処理用のスクリプト、パッケージ自身に関するメタ情報など。
- RPMと基本はいっしょ。



Debパッケージの管理

- APTというパッケージマネージャーを使って行う。
 - APT=YUM/DNFみたいなもの。
- APTはリポジトリからパッケージの検索、ダウンロード、署名検証、インストールなどを行う。
- パッケージはリポジトリサーバーに集められていて、そこからダウンロードするのもCentOSといっしょ。



aptコマンドの使い方

- パッケージの操作に使うコマンドは「apt」。
 - 昔からあるapt-getとapt-cacheが統合された新しいコマンド。
 - apt-getは今でも使えるけど、apt一本に絞った方が楽。
 - aptitudeは忘れていいです。
- aptは様々なサブコマンドと組み合わせて使う。
 - apt updateでパッケージ情報を更新。
 - apt upgradeでパッケージ本体を更新。
 - yumと異なり、update & upgradeの二段構え。
 - apt-getとはupgradeの挙動が異なることに注意(後述)。
- apt full-upgradeで全パッケージを完全に更新。
 - こちらはapt-get dist-upgradeと同等(後述)。



よく使うaptのサブコマンド

サブコマンド	用途
update	パッケージリストを更新する。
upgrade	すべてのパッケージを最新のものに更新する。ただしパッケージの削除が発生する場合は操作を保留する。
full-upgrade	upgradeで保留される処理も含めて、すべてのパッケージを更新する。
install	パッケージをインストールする。
remove	パッケージをアンインストールする。
purge	パッケージをアンインストールし、設定ファイルも削除する。
autoremove	不要になったパッケージをアンインストールする。
search	キーワードをもとにパッケージを検索する。
list	指定した条件を満たすパッケージの一覧を表示する。
show	パッケージの詳細情報を表示する。
edit-source	/etc/apt/sources.listを編集する。構文チェック機能つき。



aptとapt-getの違い

- apt full-upgrade/apt-get dist-upgradeはどちらも完全なアップグレードを行う。
 - 依存関係の変更によって、新規パッケージがインストールされたり、古いパッケージが削除されるとしても、すべて実行する。
- apt upgradeは**パッケージ削除を伴うアップグレードを保留する**。
- apt-get upgradeは削除に加え、**新規パッケージのインストールも保留する**。
 - これが問題となるのは、カーネルの更新。

コマンド	サブコマンド	パッケージの更新	パッケージのインストール	パッケージの削除
apt	upgrade	する	する	しない
	full-upgrade	する	する	する
apt-get	upgrade	する	しない	しない
	dist-upgrade	する	する	する

カーネルが更新
されない

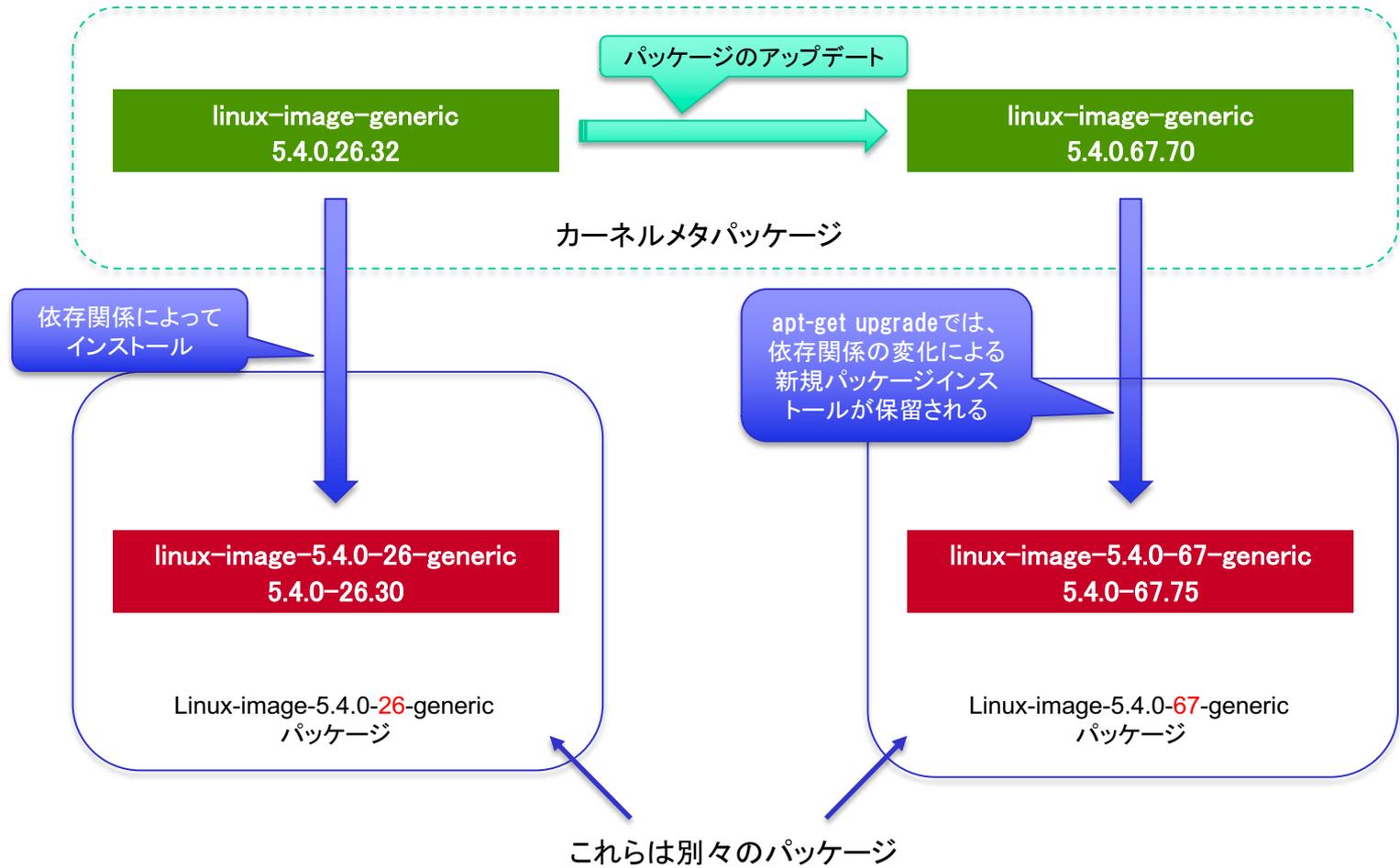


Ubuntuのカーネルパッケージ

- `apt-get upgrade`ではカーネルを更新できない。
- カーネルは複数のバージョンを同時にインストールして、ブートローダーで切り替えたい場合がある。
 - 通常のパッケージの作りでは、アップデートで中身が入れ替わってしまう。
 - 複数のカーネルを共存させるには、バージョンごとに別パッケージで提供する必要がある。
- つまりカーネルの更新とは、新規パッケージのインストール
 - パッケージのインストールが発生するため、`apt-get upgrade`では操作が保留されてしまう。
 - `apt-get`を使う場合はこの違いに注意すること。



Ubuntuのカーネルアップデート





パッケージのアップデート方針

- 一度リリースされたパッケージについては、日々のアップデートではバージョンを上げないのが基本ポリシー。
 - PHPがいきなり7.3→7.4になったりとかはしない。
- 共有ライブラリをはじめ、複数のパッケージが依存することでOSが作られているため、リリース後に迂闊にバージョンを変更すると影響が読めないのが理由。
 - そのため、LTSでは5年間同じバージョンを使い続けることになる。
 - とはいえ、開発元のサポートが切れたバージョンでも、5年以上サポートされ、セキュリティ修正が提供されるというのはメリットでもある。
- とはいえバージョンアップの激しいフロント寄りのアプリでは不便。
- 最近では、アプリの挙動が変わったり、致命的なバグが発生しないことが保証されている場合は、バージョンアップが許容されるようになっている。
 - 上げざるを得ないFirefoxとかがよい例。



Ubuntuのリポジトリとアップデート

- 20.04(focal)を例とすると、パッケージは以下のリポジトリで提供される。
 - ベースとなるパッケージと、アップデートはリポジトリ的に分離されている。
 - リリース名部分はバージョンによって異なる(例: 20.10の場合はgroovy)
- focal
 - リリース時のパッケージ。
- focal-updates
 - 一般的な、重要ではないアップデート。
 - アプリのバグ修正など。
- focal-security
 - 重要なセキュリティアップデート。
- focal-backports
 - 新しいUbuntuのリリースからバックポートされたパッケージ。
- focal-proposed
 - アップデートが正式にリリースされる前のテスト用リポジトリ。
 - 開発者向け。



Ubuntuのリポジトリの分類

- Ubuntuのリポジトリは大きくmain、universe、restricted、multiverseの4つのコンポーネントに分かれている。

- main
 - CanonicalによってサポートされるOSS。
 - セキュリティアップデートが保証されている。

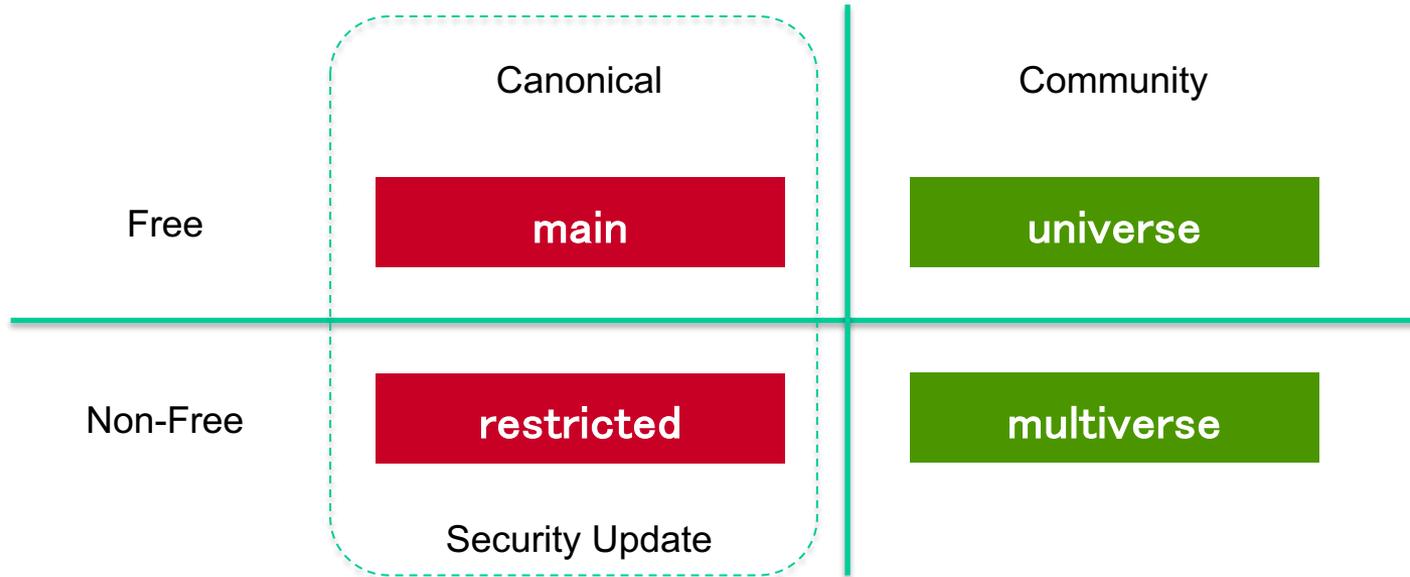
- universe
 - UbuntuコミュニティによってサポートされるOSS。
 - 多くがDebianからインポートしたパッケージで構成される。
 - セキュリティアップデートが保証されない。

- restricted
 - Canonicalによってサポートされる非OSS。

- multiverse
 - サポートが保証されない非OSS。
 - 利用は自己責任で。

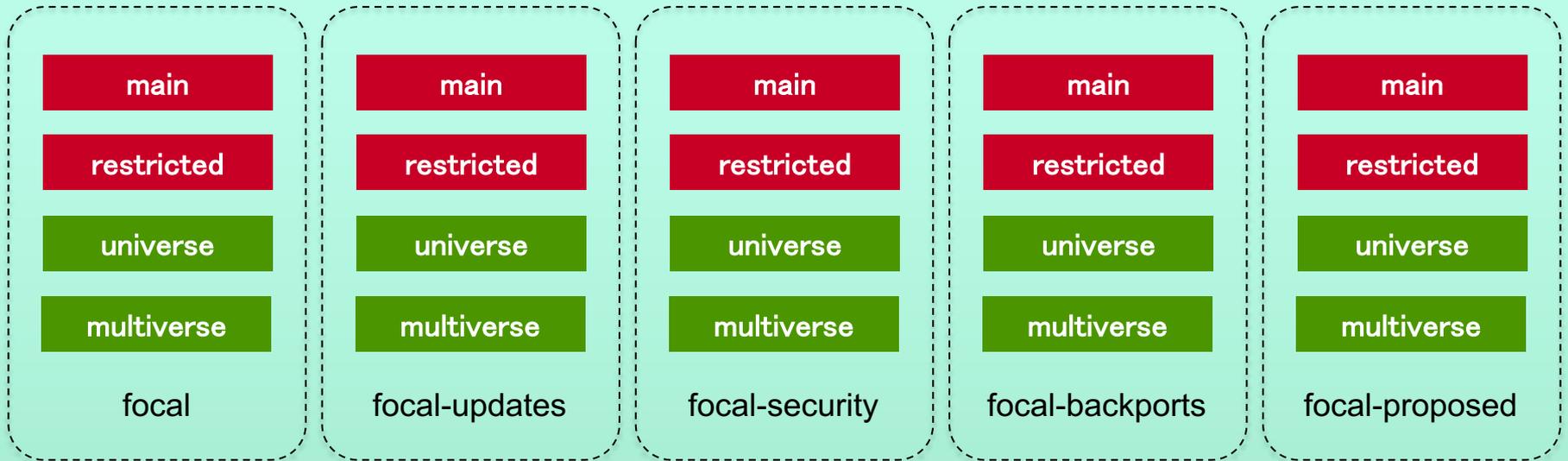


Ubuntuのリポジトリの分類





Ubuntu 20.04(focal)



【実例】UbuntuでWebサーバーを構築する



Webサーバーを動かすまで

1. IPアドレスの設定
2. Webサーバーのインストール
3. Webサーバーの設定
4. ファイアウォールの設定



- サーバーとして運用するなら、固定IPを設定するのが基本。
 - インストール時に設定したIPを自動で引き継ぐ。
 - インストール時にDHCPを利用した場合は、別途固定IPを設定しておくといよい。
- Ubuntu Serverではnetplan.ioを使ってネットワークの設定を行う。



- YAMLベースのネットワーク抽象化レンダラー
- /etc/netplan以下にYAMLファイルを作成して設定する。

/etc/netplan/99-config.yaml の記述例

```
network:
  version: 2
  ethernets:
    対象のネットワークインターフェイス名 (例 enp0s3):
      dhcp4: no
      dhcp6: no
      addresses:
        - 付与したいIPアドレス/ネットマスク
      gateway4: ゲートウェイのIPアドレス
      nameservers:
        addresses:
          - DNSサーバーのIPアドレス
          - アドレスは複数列挙できる
```



- netplan applyコマンドを実行する。
 - SSH経由でIPアドレスを変更する場合は、接続断に注意。

```
$ sudo netplan apply      # 設定変更を適用

$ ip a                    # IPアドレスを確認
(... 略...)
2: enp0s3: <BROADCAST, MULTICAST, UP, LOWER_UP> mtu 1500 qdisc fq_codel
state UP group default qlen 1000
    link/ether 08:00:27:9d:5d:d6 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.99/24 brd 192.168.1.255 scope global enp0s3
        valid_lft forever preferred_lft forever
(... 略...)
```



- apache2パッケージをインストールする。
- インストールするだけでサービスが動作する。

```
$ sudo apt update # 事前にパッケージリストを更新する
$ sudo apt install -y apache2 # UbuntuのApacheのパッケージ名はapache2

$ systemctl status apache2.service # ステータスを確認する
● apache2.service - The Apache HTTP Server
  _ Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
  _ Active: active (running) since Wed 2021-03-17 07:34:23 UTC; 4min 31s ago
  _ Docs: https://httpd.apache.org/docs/2.4/
  Process: 623 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
```



ubuntu

Apache2 Ubuntu Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

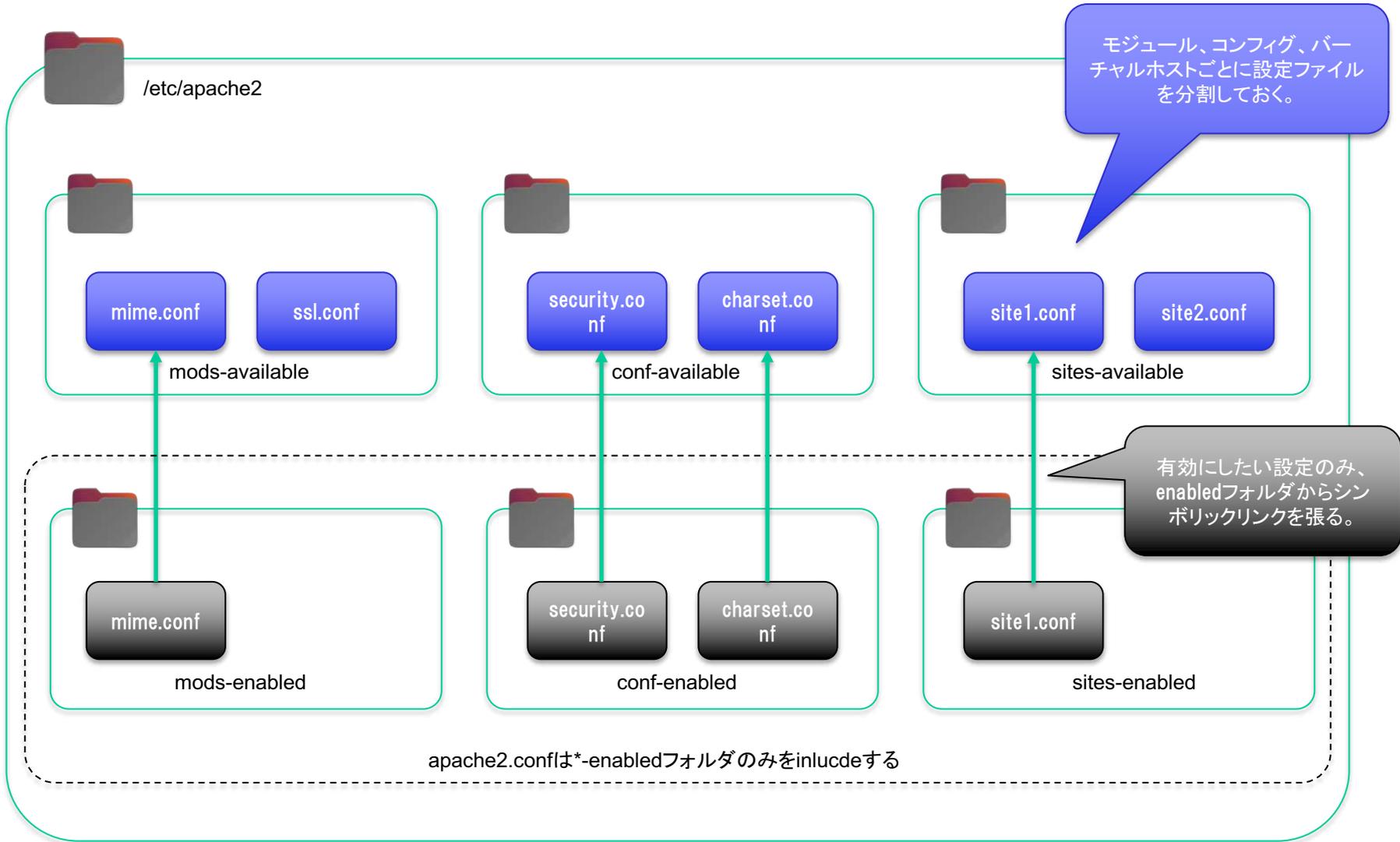
```
/etc/apache2/  
|-- apache2.conf  
|   |-- ports.conf  
|-- mods-enabled  
|   |-- *.load  
|   |-- *.conf  
|-- conf-enabled  
|   |-- *.conf  
|-- sites-enabled  
|   |-- *.conf
```



- UbuntuのApacheはCentOSと異なり、設定ごとにファイルが分割されている構成。
 - /etc/apache2/{conf,sites,mods}-availableにファイルの実体を置く。
 - /etc/apache2/{conf,sites,mods}-enabledに有効にしたい設定へのシンボリックリンクを置く。
 - リンクのON/OFFで設定、モジュール、バーチャルホストの有効、無効を切り替えられる。



Webサーバーの設定





- リンクのON/OFFで設定、モジュール、バーチャルホストの有効、無効を切り替えられる。
- シンボリックリンクを張る、切るを行うフロントエンドがa2{en,dis}系コマンド。

```
$ sudo a2enmod userdir           # UserDirモジュールを有効化
$ sudo a2dismod userdir          # UserDirモジュールを無効化

$ sudo a2enconf security         # Security設定を有効化
$ sudo a2disconf security        # Security設定を無効化

$ sudo a2ensite default-ssl      # default-ssl|バーチャルホストを有効化
$ sudo a2dissite default-ssl     # default-ssl|バーチャルホストを無効化
```



- Ubuntuではデフォルトでファイアウォールは動作していない。
 - これはデフォルトで外向きに開放されているサービスが存在しないため。
 - SSHサーバーすら、明示的に指定しない限りインストールされない。
- UbuntuではiptablesのフロントエンドであるUFWを使う。
 - Uncomplicated FireWallの略。
 - Ubuntu FireWallではない。



- コマンド名は「ufw」。
- aptと同様、サブコマンドを組み合わせて使う。

```
$ sudo ufw status      # UFWの状態を表示
Status: inactive

$ sudo ufw enable      # UFWを有効化
Firewall is active and enabled on system startup
```



■ 「ufw allow」で許可ルールを追加。

```
$ sudo ufw allow http # HTTP (TCP:80) を全体に対して開放
Rule added
Rule added (v6)

$ sudo ufw allow https # HTTPS (TCP:443) を全体に対して開放
Rule added
Rule added (v6)

$ sudo ufw status # UFWの状態を表示
Status: active
```

To	Action	From
--	-----	----
80/tcp	ALLOW	Anywhere
443/tcp	ALLOW	Anywhere
80/tcp (v6)	ALLOW	Anywhere (v6)
443/tcp (v6)	ALLOW	Anywhere (v6)

参考資料



■ Ubuntu Server Guide

- <https://ubuntu.com/server/docs>

■ Ubuntu Weekly Topics

- <https://gihyo.jp/admin/clip/01/ubuntu-topics>

■ Ubuntu Weekly Recipe

- <https://gihyo.jp/admin/serial/01/ubuntu-recipe>

■ 各種メーリングリスト

- <https://lists.ubuntu.com/>



■ Ubuntu Security Notices

- <https://ubuntu.com/security/notices>

■ CVE Reports

- <https://ubuntu.com/security/cve>

■ Planet Ubuntu

- <https://planet.ubuntu.com/>

■ Ubuntu Weekly Newsletter

- <https://wiki.ubuntu.com/UbuntuWeeklyNewsletter/>

■ Full Circle Magazine

- <https://fullcirclemagazine.org/>



■ 『LinuC通信』

- Linuxオープンソース関連ニュース
- Linuxの利用に役立つTipsやテクニク
- LinuC合格者の声

■ 『LinuC Level2・Level3を受けてみよう！』

- Level2・Level3の例題解説
- Level2・Level3を取得するメリット、活用の事例など
- Level1をお持ちでない方にも役立ちます

<http://www.lpi.or.jp/mail/>