

LinuCレベル2 201試験概要解説 サーバー構築のできるLinuxエキスパートへの道

2024/08/17 (Sta)

LPI-Japanプラチナスポンサー 株式会社ゼウス・エンタープライズ
鯨井 貴博 (LinuCエヴァンジェリスト)



Who are you?

鯨井貴博

LPI-Japan プラチナスポンサー 株式会社ゼウス・エンタープライズ
LinuCエヴァンジェリスト

大学時代 Unixの存在を知り、日経Linuxを読み始める。
2000年にVine Linux 2.0で一度挫折を経験。
その悔しさを忘れきれず、2007年 他業種からIT業界に転職しLinuxに再チャレンジ。

SE・商用製品サポート・インストラクター・プロジェクト管理などを経験し、現在に至る。
自分自身が学習で苦労した経験から、初心者を含む受講者に分かりやすい講義を行うように心がけている。

また、興味の向くIT技術・オープンソースソフトウェアなどについて、
Opensourcetechブログ (<https://www.opensourcetech.tokyo/>) で執筆中。
実際に自分でやってみる/使ってみる・開発者本人から話を聞いてみることを大切にしています。



Linus Torvaldsさん(Linux開発者)



Igor Sysoevさん(nginx開発者)



Alexei Vladishevさん(Zabbix開発者)



Who are you?

私が進んできた道



2007

2024



- https://www.cisco.com/c/ja_jp/training-events/training-certifications/certifications/associate/ccna.html
- <https://linuc.org/>
- <https://www.ipa.go.jp/shiken/index.html>
- <https://www.idla.org/certificate/general/>
- <https://www.accel-exam.jp/>
- <https://html5exam.jp/>
- <https://training.linuxfoundation.org/ja/certification/certified-kubernetes-administrator-cka/>



Who are you?

Open Source Summit Japan 2023



THE LINUX FOUNDATION
OPEN SOURCE SUMMIT
JAPAN

Innovation Happens Here.

- AUTOMOTIVE LINUX SUMMIT
- CLOUDOPEN
- CONTAINERCON
- CRITICAL SOFTWARE SUMMIT
- EMBEDDED IOT SUMMIT
- EMERGING OS FORUM
- LINUXCON
- OPEN SOURCE LEADERSHIP SUMMIT
- OPERATIONS MANAGEMENT SUMMIT
- OSPO CON
- SUPPLYCHAIN SECURITYCON

2023年12月5日・6日
DECEMBER 5-6, 2023
東京 | TOKYO, JAPAN
#OSSUMMIT



Open Source Summit Japan 2023: ボランティアリーダーの体験記
<https://www.opensourcetech.tokyo/entry/20231224/1703429785>





Who are you?

Open Source Summit Japan 2024

THE LINUX FOUNDATION
OPEN SOURCE SUMMIT
JAPAN

2024 OCTOBER 28-29
2024年 10月28・29日
TOKYO, JAPAN | 東京
#OSSummit

REGISTER SPONSOR

会場のご案内



虎ノ門ヒルズフォーラム

〒105-6305
東京都港区虎ノ門1-23-3
虎ノ門ヒルズ森タワー5階

<https://events.linuxfoundation.org/open-source-summit-japan/>



LinuC について



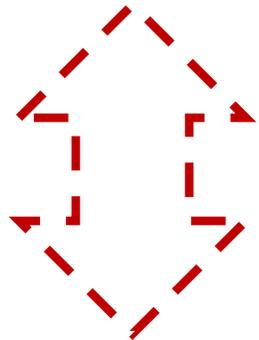
■LinuCとは

クラウド時代の即戦力エンジニアであることを証明するLinux技術者認定

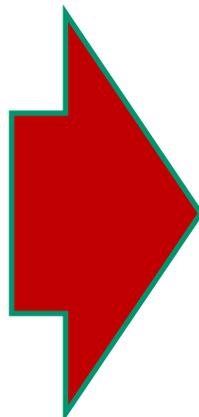
- ✓現場で「今」求められている新しい技術要素に対応
 - ・ オンプレミス／仮想化・コンテナを問わず様々な環境下でのサーバー構築
 - ・ 他社とのコラボレーションの前提となるオープンソースへの理解
 - ・ システムの多様化に対応できるアーキテクチャへの知見
- ✓全面的に見直した「今」身につけておくべき技術範囲を網羅
 - 今となっては使わない技術やコマンドの削除、アップデート、新領域の取り込み
- ✓Linuxの範疇だけにとどまらない領域までカバー
 - セキュリティや監視など、ITエンジニアであれば必須の領域もカバー



AWSなどの
パブリッククラウドを
活用するための技術



間が
欠けて
いる状態



AWSなどの
パブリッククラウドを
活用するための技術

仮想マシン/コンテナ技術、
クラウドセキュリティ、
アーキテクチャ、ほか

オンプレミスの
サーバーサイドLinux技術

オンプレミスの
サーバーサイドLinux技術

【今まで / その他】

LC **Linu**C Version10.0





- LinuCレベル2(201)の全体像を掴む
- LinuCレベル2(201)の学習方法を理解する
- LinuCレベル2(201)の受験計画を立てる



101試験

- 1.01 : Linuxのインストールと仮想マシン・コンテナの利用
 - 1.01.1Linuxのインストール、起動、接続、切断と停止
 - 1.01.2仮想マシン・コンテナの概念と利用
 - 1.01.3ブートプロセスとsystemd
 - 1.01.4プロセスの生成、監視、終了
 - 1.01.5デスクトップ環境の利用
- 1.02 : ファイル・ディレクトリの操作と管理
 - 1.02.1ファイルの所有者とパーミッション
 - 1.02.2基本的なファイル管理の実行
 - 1.02.3ハードリンクとシンボリックリンク
 - 1.02.4ファイルの配置と検索
- 1.03 : GNUとUnixのコマンド
 - 1.03.1コマンドラインの操作
 - 1.03.2フィルタを使ったテキストストリームの処理
 - 1.03.3ストリーム、パイプ、リダイレクトの使用
 - 1.03.4正規表現を使用したテキストファイルの検索
 - 1.03.5エディタを使った基本的なファイル編集の実行
- 1.04 : リポジトリとパッケージ管理
 - 1.04.1apt コマンドによるパッケージ管理
 - 1.04.2Debianパッケージ管理
 - 1.04.3yumコマンドによるパッケージ管理
 - 1.04.4RPMパッケージ管理
- 1.05 : ハードウェア、ディスク、パーティション、ファイルシステム
 - 1.05.1ハードウェアの基礎知識と設定
 - 1.05.2ハードディスクのレイアウトとパーティション
 - 1.05.3ファイルシステムの作成と管理、マウント

<https://linuc.org/linuc1/range/101.html>

<https://linuc.org/linuc1/range/102.html>

102試験

- 1.06 : シェルおよびスクリプト
 - 1.06.1シェル環境のカスタマイズ
 - 1.06.2シェルスクリプト
- 1.07 : ネットワークの基礎
 - 1.07.1インターネットプロトコルの基礎
 - 1.07.2基本的なネットワーク構成
 - 1.07.3基本的なネットワークの問題解決
 - 1.07.4クライアント側のDNS設定
- 1.08 : システム管理
 - 1.08.1アカウント管理
 - 1.08.2ジョブスケジューリング
 - 1.08.3ローカライゼーションと国際化
- 1.09 : 重要なシステムサービス
 - 1.09.1システム時刻の管理
 - 1.09.2システムのログ
 - 1.09.3メール配送エージェント(MTA)の基本
- 1.10 : セキュリティ
 - 1.10.1セキュリティ管理業務の実施
 - 1.10.2ホストのセキュリティ設定
 - 1.10.3暗号化によるデータの保護
 - 1.10.4クラウドセキュリティの基礎
- 1.11 : オープンソースの文化
 - 1.11.1オープンソースの概念とライセンス
 - 1.11.2オープンソースのコミュニティとエコシステム

Linuxの基礎知識や操作方法の取得



201試験

- 2.01 : システムの起動とLinuxカーネル
 - 2.01.1 ブートプロセスとGRUB
 - 2.01.2 システム起動のカスタマイズ
 - 2.01.3 Linux カーネルの構成要素
 - 2.01.4 Linuxカーネルのコンパイル
 - 2.01.5 カーネル実行時における管理とトラブルシューティング
- 2.02 : ファイルシステムとストレージ管理
 - 2.02.1 ファイルシステムの設定とマウント
 - 2.02.2 ファイルシステムの管理
 - 2.02.3 論理ボリュームマネージャの設定と管理
- 2.03 : ネットワーク構成
 - 2.03.1 基本的なネットワーク構成
 - 2.03.2 高度なネットワーク構成
 - 2.03.3 ネットワークの問題解決
- 2.04 : システムの保守と運用管理
 - 2.04.1 makeによるソースコードからのビルドとインストール
 - 2.04.2 バックアップとリストア
 - 2.04.3 ユーザへの通知
 - 2.04.4 リソース使用状況の把握
 - 2.04.5 死活監視、リソース監視、運用監視ツール
 - 2.04.6 システム構成ツール
- 2.05 : 仮想化サーバー
 - 2.05.1 仮想マシンの仕組みとKVM
 - 2.05.2 仮想マシンの作成と管理
- 2.06 : コンテナ
 - 2.06.1 コンテナの仕組み
 - 2.06.2 Dockerコンテナとコンテナイメージの管理

202試験

- 2.07 : ネットワーククライアントの管理
 - 2.07.1 DHCPサーバーの設定と管理
 - 2.07.2 PAM認証
 - 2.07.3 LDAPクライアントの利用方法
 - 2.07.4 OpenLDAPサーバーの設定
- 2.08 : ドメインネームサーバー
 - 2.08.1 BINDの設定と管理
 - 2.08.2 ゾーン情報の管理
 - 2.08.3 セキュアなDNSサーバーの実現
- 2.09 : HTTPサーバーとプロキシサーバー
 - 2.09.1 Apache HTTPサーバーの設定と管理
 - 2.09.2 OpenSSLとHTTPSの設定
 - 2.09.3 nginxの設定と管理
 - 2.09.4 Squidの設定と管理
- 2.10 : 電子メールサービス
 - 2.10.1 Postfixの設定と管理
 - 2.10.2 Dovecotの設定と管理
- 2.11 : ファイル共有サービス
 - 2.11.1 Sambaの設定と管理
 - 2.11.2 NFSサーバーの設定と管理
- 2.12 : システムのセキュリティ
 - 2.12.1 iptables や firewalld によるパケットフィルタリング
 - 2.12.2 OpenSSH サーバーの設定と管理
 - 2.12.3 OpenVPNの設定と管理
 - 2.12.4 セキュリティ業務
- 2.13 : システムアーキテクチャ
 - 2.13.1 高可用システムの実現方式
 - 2.13.2 キャパシティプランニングとスケーラビリティの確保
 - 2.13.3 クラウドサービス上のシステム構成
 - 2.13.4 典型的なシステムアーキテクチャ



- ①出題範囲の内容について調べてみる
公式ドキュメント・技術書など
- ②実際に操作してみる
これが大事！
- ③学習の補助教材などを利用する
 - ・メールマガジン
 - ・標準教科書
 - ・過去のセミナー資料
 詳細は、<https://lpi.or.jp/learning/>



メールマガジンでコツコツと

学習に役立つメールマガジン

LPI-Japanでは、試験レベルごとの例題解説など、学習に役立つメールマガジンを無料でお届けしています。

LPI-Japan LinuC通信「レベル2・レベル3を受けてみよう！」でサンプル問題作ってるので、よかったら登録してください！

過去のメールマガジンの例題解説をまとめています。

LPI-Japanでは、試験レベルごとの例題解説など、学習に役立つメールマガジンを無料でお届けしています。



LPI-Japanが開発した大人気の教科書でLinuxを効率的に学ぶ

- Linux 標準教科書
- Linux サーバー標準教科書
- 高信頼システム構築標準教科書
- Linux セキュリティ標準教科書
- Linux システム管理標準教科書

Linux豆知識
Linuxを学習する上で出てくる素朴な疑問や便利なテクニックなどを紹介しています。

Linux初心者のための入門編と中級者向けのネットワーク編のLinux解説コラム

- Linux 道場入門編
- Linux 道場ネットワーク編
- Linux 道場 Linux学習環境構築編



人気の技術解説無料セミナーも活用して

LPI-Japanでは、『LinuCレベル1～新出題範囲における受験準備とポイント解説』など、レベル別の技術解説無料セミナーを開催しています。学習の仕方ですら迷ったら是非足を運んでみてください。他の受験者の方と意見交換もでき、モチベーションもあがります！

過去のセミナー資料のダウンロードはこちら



④過去セミナーの動画

<https://www.youtube.com/user/LPIJapan>

検索

open your NEXT future

LinuC OSS-DB Silver/Gold HTML5 Professional Certification OPCEL

頼られるための、頼れる資格

Linux技術者認定「LinuC」

LPI-Japan
@LPIJapan チャンネル登録者数 3890人 79本の動画

LPI-Japanチャンネルは、LinuC (Linux) やOSS-DB (PostgreSQL)、HT...

ホーム 動画 再生リスト コミュニティ チャンネル 概要

人気の動画 ▶ すべて再生

動画タイトル	バージョン	技術解説無料セミナー	視聴回数	投稿日時
仮想マシン・コンテナの概念と利用	LinuCレベル1 Version10.0	1:07:35	9775	2年前
システムの起動とLinuxカーネル	LinuCレベル2 Version10.0	1:06:36	8487	2年前
ブートプロセスとsystemd	LinuCレベル1 Version10.0	1:15:40	8087	2年前
シェルスクリプトとジョブスケジューリング	LinuCレベル1 Version10.0	1:15:44	5822	2年前
ネットワークの基礎	LinuCレベル1 Version10.0	1:25:44	5757	2年前





学習の具体的な進め方(2~3か月程度)

学習開始(0%)

試験範囲の確認(LinuC HP)

●月頃、受験しようという目標を立てる

↓
LinuC認定教材の購入・1週目読込 <https://lpi.or.jp/linuc1/book.shtml>
<https://lpi.or.jp/linuc2/book.shtml>

↓
LinuC認定教材・Webサイトを参考に、実機操作(サーバ構築やコマンド操作)を試す
※操作やトラブルシュートで力が身に付く!

1ヶ月

↓
LinuC認定教材 2週目読込

●月●日頃、受験しようとする

↓
問題集やメルマガサンプル問題で理解力確認 <https://linuc.org/study/samples/>
※理解不足箇所の洗い出し

2ヶ月

↓
LinuC認定教材 3週目
※弱点の補強

仕上げ
(80%)

↓
受験申込

受験日の変更も可能なので安心

↓
問題を8~9割以上、正解となるまで繰り返し解く
苦手な部分を重点的に復習

- ・受験まで継続して学習すること
- ・繰り返し学習し、理解度/問題正解率を高めた状態で受験すること

2.5ヶ月

完全に理解した!
(100%)

↓
受験



- 2.01 : システムの起動とLinuxカーネル
 - 2.01.1 ブートプロセスとGRUB
 - 2.01.2 システム起動のカスタマイズ
 - 2.01.3 Linux カーネルの構成要素
 - 2.01.4 Linuxカーネルのコンパイル
 - 2.01.5 カーネル実行時における管理とトラブルシューティング
- 2.02 : ファイルシステムとストレージ管理
 - 2.02.1 ファイルシステムの設定とマウント
 - 2.02.2 ファイルシステムの管理
 - 2.02.3 論理ボリュームマネージャの設定と管理
- 2.03 : ネットワーク構成
 - 2.03.1 基本的なネットワーク構成
 - 2.03.2 高度なネットワーク構成
 - 2.03.3 ネットワークの問題解決
- 2.04 : システムの保守と運用管理
 - 2.04.1 makeによるソースコードからのビルドとインストール
 - 2.04.2 バックアップとリストア
 - 2.04.3 ユーザへの通知
 - 2.04.4 リソース使用状況の把握
 - 2.04.5 死活監視、リソース監視、運用監視ツール
 - 2.04.6 システム構成ツール
- 2.05 : 仮想化サーバー
 - 2.05.1 仮想マシンの仕組みとKVM
 - 2.05.2 仮想マシンの作成と管理
- 2.06 : コンテナ
 - 2.06.1 コンテナの仕組み
 - 2.06.2 Dockerコンテナとコンテナイメージの管理

- ✓ Linus Torvaldsが作成し、コミュニティが成長させてきたLinuxカーネル
- ✓ Linuxのファイルシステム・ネットワーク構成
- ✓ Linuxの保守・運用
- ✓ Linuxを用いたハイパーバイザー(KVM)
- ✓ コンテナ(Docker)



資格を取得する、しない！？(取得するメリット)

資格取得 × スキル保有、両方がある方がいい

技術の前では、みんな平等！

いつから始めても遅いということはない！

組織にいい流れをもたらす



2.01.1 ブートプロセスとGRUB

2.01.2 システム起動のカスタマイズ

2.01.3 Linux カーネルの構成要素

2.01.4 Linuxカーネルのコンパイル

2.01.5 カーネル実行時における管理とトラブルシューティング



2.01 : システムの起動とLinuxカーネル

2.01.1 ブートプロセスとGRUB

重要度 4

概要 ブートプロセス中およびリカバリモードのLinuxシステムを適切に操作する。関連するブートローダはGRUB バージョン2 である。BIOS と UEFI のシステム両方をカバーする。

- 詳細**
- BIOS と UEFI
 - Master Boot Record (MBR)
 - EFI System Partition (ESP), efibootmgr, UEFI shell
 - ブートローダの標準的なオプションの変更とブートローダのシェル(コマンドライン)の使用。
 - ブートローダの開始とカーネルへの引継ぎ。
 - GRUB の起動メニューを生成する。
 - grub-mkconfig, grub2-mkconfig
 - カーネルロードモジュールのロード。
 - initrd, initramfs
 - ハードウェアの初期化と設定。
 - デバイス、モジュール
 - ファイルシステムのチェックとマウント。
 - デモン / サービスの初期化と設定。
 - systemctl, systemd.unit
 - ハードディスクまたはリムーバブルデバイスにおけるブートローダのインストール場所を知っている。
 - grub-install, grub2-install
 - /boot/, /boot/grub/, /boot/grub2/ および /boot/efi/ の内容

2.01.2 システム起動のカスタマイズ

重要度 3

概要 さまざまなターゲットのシステムサービスの動作を照会および変更する。systemd と Linux のブートプロセスについての十分な理解が必要である。これには、systemd ターゲットの操作も含まれる。

- 詳細**
- systemd
 - /usr/lib/systemd/, /etc/systemd/, /run/systemd/, systemctl, systemd-delta
 - systemctl status, systemctl list-units, systemctl start/stop, systemctl enable/disable, systemctl mask/unmask
 - systemd のrescue モードと emergency モード。

2.01.3 Linux カーネルの構成要素

重要度 2

概要 特定のハードウェア、ハードウェアドライバ、システムリソース、およびさまざまな要求に必要なカーネルの構成要素を利用する。これには、異なる種類のカーネルイメージを実装すること、安定版および開発版のカーネルとパッチを区別すること、カーネルモジュールを利用することなども含まれる。

- 詳細**
- Kernel の配布形式。
 - bzImage , xz データ圧縮
 - Kernelのモジュールとドキュメント。
 - /usr/src/linux/
 - /usr/src/linux/Documentation/



2.01 : システムの起動とLinuxカーネル

2.01.4 Linuxカーネルのコンパイル

重要度 2

概要 Linuxカーネルの特定の機能を必要に応じて取り込んだり無効化するために、カーネルを適切に構成できる。また、必要に応じてLinuxカーネルをコンパイルし、新しいカーネルに変更点を書き込み、initrdイメージを作成し、新しいカーネルをインストールできる。

- 詳細**
- /usr/src/linux/
 - /usr/src/linux/.config
 - カーネルの Makefile
 - Kernel 2.6.x、3.x、4.x、5.x のmakeのターゲット。
 - all, config, xconfig, menuconfig, gconfig, oldconfig, mrproper, bzImage, modules, modules_install, rpm-pkg, binrpm-pkg, deb-pkg
 - カーネル構成をカスタマイズする。
 - 新しいカーネルおよび適切なカーネルモジュールを構築する。
 - /lib/modules/kernel-version/, gzip, bzip2
 - 新しいカーネルおよび必要なモジュールをインストールする。
 - module tools, depmod
 - ブートマネージャが新しいカーネルおよび関連付けられたファイルを探せるようにする。
 - モジュールの構成ファイル
 - DKMS を使用してカーネルのモジュールをコンパイルする。
 - dkms
 - initrd を構成する。
 - Dracut, mkinitrd, mkinitramfs

2.01.5 カーネル実行時における管理とトラブルシューティング

重要度 3

概要 2.6.x、3.x、4.x、5.x カーネルとそのロード可能なモジュールについての管理や照会ができる。ブートおよび実行時の一般的な問題を特定および修正することができる。udevを使用したデバイスの検知と管理について理解する。これには、udevルールのトラブルシューティングが含まれる。

- 詳細**
- コマンドラインユーティリティを使用して、現在実行中のカーネルおよびカーネルモジュールに関する情報を取得する。
 - depmod, modinfo
 - 手作業でカーネルモジュールをロードおよびアンロードする。
 - modprobe, insmod, lsmod, rmmod
 - モジュールをアンロードできるタイミングを判断する。
 - モジュールが受け取るパラメータを判断する。
 - /etc/modprobe.d/
 - カーネルのバージョンを確認する。
 - uname
 - /lib/modules/kernel-version/modules.dep
 - モジュールをファイル名ではなく別の名前でもロードできるようにシステムを設定する。
 - /proc ファイルシステム
 - /proc/sys/kernel/
 - /etc/sysctl.conf, /etc/sysctl.d/, /sbin/sysctl
 - /sys ファイルシステム (sysfs)
 - /bootおよび /lib/modules の内容
 - 利用可能なハードウェアに関する情報を分析するツール
 - dmesg, lspci, lsdev, lsusb, journalctl
 - udevルール
 - udevmonitor, udevadm monitor, /etc/udev/
 - /etc内のモジュール設定ファイル
 - カーネルダンプを取得する。設定は含まない。
 - kdump, kexec



2.01 : システムの起動とLinuxカーネル

The video player shows a thumbnail with the LinuC-2 logo and the title '主題2.01 : システムの起動とLinuxカーネル' (Topic 2.01: System Startup and Linux Kernel). The subtitle is 'ブートプロセスとGRUB、カーネルの構成要素、他' (Boot process and GRUB, kernel components, etc.). The instructor is identified as '鯨井 貴博 氏' (Mr. Takahiro Kamei) from '株式会社ゼウス・エンジニアリング' (Zeus Engineering Co., Ltd.). The video player includes a progress bar at 0:00 / 1:06:35 and a 'LinuC open your NEXT future LPI-JAPAN' banner at the bottom.

主題2.01 : システムの起動とLinuxカーネル
ブートプロセスとGRUB、
カーネルの構成要素、他

講師
株式会社ゼウス・
エンジニアリング
鯨井 貴博 氏

0:00 / 1:06:35 ・スタート >

LinuC open your NEXT future
LPI-JAPAN

システムの起動とLinuxカーネル (Linux学習)

LPI-Japan
5.56K subscribers

157 | Share | Clip | Save

12,855 views Oct 7, 2020

- 00:00 スタート
- 14:29 Linuxカーネルとは?
- 26:51 Linuxの起動プロセス
- 36:23 Linuxカーネルの再構築
- 51:10 Appendix
- 56:27 Q&A (ライブ回答)
- 1:06:08 Q&A (いただいた質問全てへの回答)

<https://www.youtube.com/watch?v=vCBgwYSymRA>



2.01 : システムの起動とLinuxカーネル

The Linux Kernel Archives

About Contact us FAQ Releases Signatures Site news

Protocol Location

HTTP <https://www.kernel.org/pub/>

GIT <https://git.kernel.org/>

RSYNC <rsync://rsync.kernel.org/pub/>

Latest Release
6.10.4 ↓

mainline:	6.11-rc3	2024-08-11	[tarball]	[patch]	[inc. patch]	[view diff]	[browse]		
stable:	6.10.4	2024-08-11	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
stable:	6.9.12 [EOL]	2024-07-27	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	6.6.45	2024-08-11	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	6.1.104	2024-08-11	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	5.15.164	2024-07-27	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	5.10.223	2024-07-27	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	5.4.281	2024-07-27	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	4.19.319	2024-07-27	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
linux-next:	next-20240809	2024-08-09						[browse]	

Other resources

Git Trees Patchwork Mirrors

Documentation Wikis Linux.com

Kernel Mailing Lists Bugzilla Linux Foundation

Social

Site Atom feed Releases Atom Feed Kernel Planet

mainline (開発版)
約3か月のインターバルで新機能を開発する
x.y-rc1 → x.y-rc2 → x.y-rc3 x.y-rc12

stable (安定版)
開発完了した新機能・バグ修正が行われる
x.y → x.y.1 → x.y.2

longterm (長期保守版)
重要なバグ修正のみが行われる
x.y → x.y.1 → x.y.2



2.01 : システムの起動とLinuxカーネル



index : kernel/git/stable/linux.git

Linux kernel stable tree

summary refs log **tree** commit diff stats

path: [root/rust/kernel/task.rs](#)

blob: 55dff7e088bf5fc96cc45246cfcccbcb663b3e2 (plain)

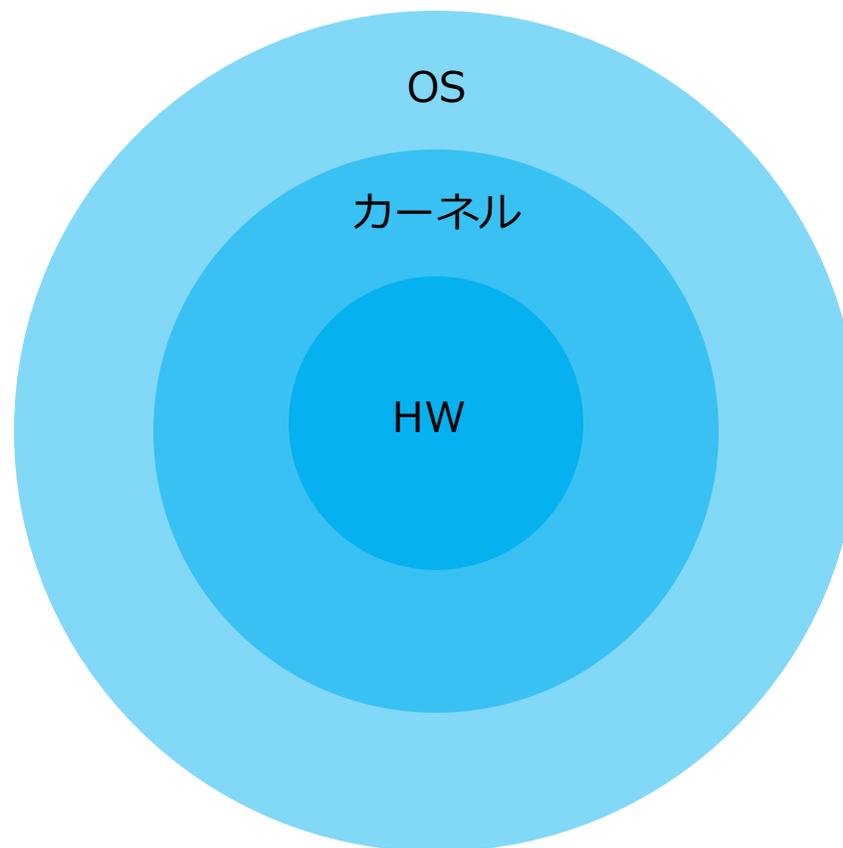
```
1 // SPDX-License-Identifier: GPL-2.0
2
3 //! Tasks (threads and processes).
4 //!
5 //! C header: [`include/linux/sched.h`](srctree/include/linux/sched.h).
6
7 use crate::types::Opaque;
8 use core::{
9     ffi::{c_int, c_long, c_uint},
10    marker::PhantomData,
11    ops::Deref,
12    ptr,
13 };
14
15 /// A sentinel value used for infinite timeouts.
16 pub const MAX_SCHEDULE_TIMEOUT: c_long = c_long::MAX;
17
18 /// Bitmask for tasks that are sleeping in an interruptible state.
19 pub const TASK_INTERRUPTIBLE: c_int = bindings::TASK_INTERRUPTIBLE as c_int;
20 /// Bitmask for tasks that are sleeping in an uninterruptible state.
21 pub const TASK_UNINTERRUPTIBLE: c_int = bindings::TASK_UNINTERRUPTIBLE as c_int;
22 /// Convenience constant for waking up tasks regardless of whether they are in interruptible or
23 /// uninterruptible sleep.
24 pub const TASK_NORMAL: c_uint = bindings::TASK_NORMAL as c_uint;
25
26 /// Returns the currently running task.
27 #[macro_export]
```

<https://www.kernel.org/>



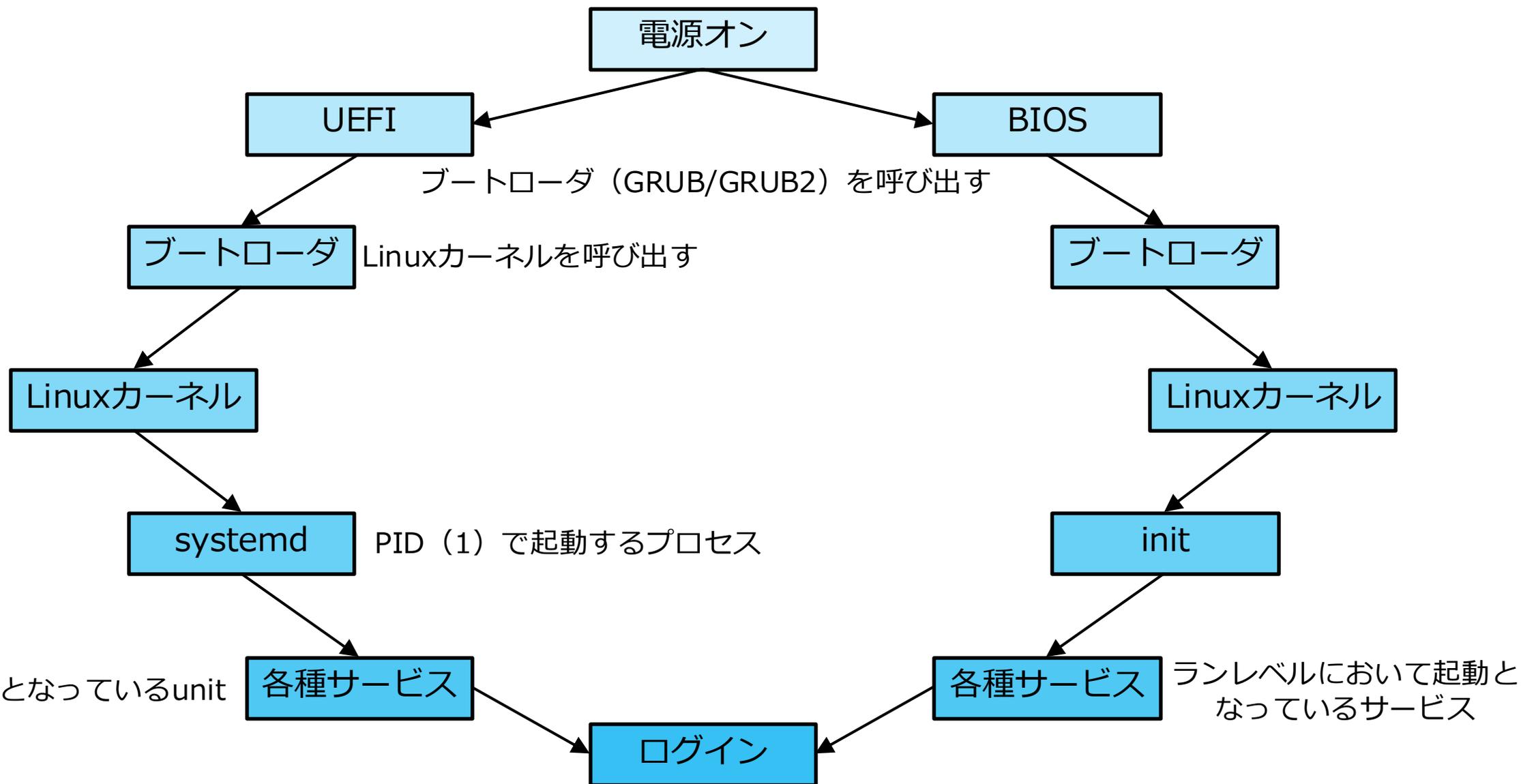
カーネル = OSの核となるプログラム

- ハードウェア制御
- メモリマネージメント
- プロセス処理
- ファイルシステム
- アプリケーション実行などを担当する。





2.01 : システムの起動とLinuxカーネル





2.02 : ファイルシステムとストレージ管理

2.02.1 ファイルシステムの設定とマウント

2.02.2 ファイルシステムの管理

2.02.3 論理ボリュームマネージャの設定と管理



2.02.1 ファイルシステムの設定とマウント

重要度 3

概要 標準的なLinuxファイルシステムの適切な設定と操作ができる。これには、各種ファイルシステムの設定およびマウントも含まれる。

- 詳細
- fstab設定の概念
 - /etc/fstab
 - ファイルシステムのマウントとアンマウント
 - mount, umount, /etc/mtab, /proc/mounts, systemd
 - マウントの順番
 - スワップパーティションおよびファイルを操作するツール
 - swapon, swapoff, mkswap
 - ファイルシステムを特定しマウントするための UUID の使用
 - blkid, lsblk

2.02.2 ファイルシステムの管理

重要度 4

概要 システムユーティリティを使用して、Linuxファイルシステムを適切に保守できる。これには、標準的なファイルシステムの操作およびSMARTデバイスの監視が含まれる。

- 詳細
- ファイルシステムを操作するツール
 - mkfs (mkfs.*), fsck (fsck.*)
 - ext4を操作するツール
 - tune2fs, dumpe2fs, dump, restore
 - XFS を操作するツール
 - xfs_info, xfs_check, xfs_repair, xfsdump, xfsrestore
 - subvolumes と snapshots を含む、基本的な Btrfs を操作するツール
 - ext4 ファイルシステムをBtrfsに変換したり操作する。
 - btrfs, btrfs-convert
 - HDD, SSD の健康状態を監視できる。
 - smartd, smartctl



2.02.3 論理ボリュームマネージャの設定と管理

重要度 3

概要 論理ボリューム、ボリュームグループ、および物理ボリュームの作成および削除ができる。これには、スナップショットと論理ボリュームのサイズ変更が含まれる。

- 詳細
- LVM のツールと設定ファイル
 - lvm, lvm.conf
 - 物理ボリュームの作成および削除方法
 - pv* コマンド
 - ボリュームグループの作成/削除、物理ボリュームの追加/削除、名前変更、有効化/無効化方法
 - vg* コマンド
 - 論理ボリュームの作成/削除、サイズ変更、名前変更、有効化/無効化方法
 - lv* コマンド



2.02 : ファイルシステムとストレージ管理

The image shows a YouTube video player interface. The video title is '主題2.02 : ファイルシステムとストレージ管理' (Topic 2.02: File System and Storage Management). The main content area has a red background with white text: 'ファイルシステムとストレージ管理' (File System and Storage Management) and 'LinuCレベル2 技術解説セミナー' (LinuC Level 2 Technical Explanation Seminar). The channel name is 'LPI-Japan' with 5.56K subscribers. The video has 4,726 views and was posted on Dec 5, 2020. The video player shows a progress bar at 0:00 / 1:12:25. The video player interface includes a play button, volume control, and a progress bar. The video player interface also includes a channel name 'LPI-Japan' with 5.56K subscribers, a 'Subscribed' button, and a list of video controls: Like (65), Comment, Share, Clip, Save, and a menu icon. The video player interface also includes a list of video chapters: 00:00 スタート, 05:33 ファイルシステムとは, 10:41 VFS (Virtual File System) とは, 13:48 i-node (index-node) とは, 21:22 マウントとは, 34:48 ファイルシステムの保守, 48:04 LVM (Logical Volume Manager) とは, 1:03:19 本日のまとめ, 1:07:38 Q&A (ライブ回答).

LinuC-2

講師
エスディーテック株式会社
末永 貴一 氏

主題2.02 : ファイルシステムとストレージ管理
ファイルシステムと
ストレージ管理

LinuCレベル2 技術解説セミナー

LC LinuC open your NEXT future
LPI-JAPAN

0:00 / 1:12:25 · スタート ▶

ファイルシステムとストレージ管理 (Linux学習)

LPI-Japan
5.56K subscribers

Subscribed

65 | | Share | Clip | Save | ...

4,726 views Dec 5, 2020

- 00:00 スタート
- 05:33 ファイルシステムとは
- 10:41 VFS (Virtual File System) とは
- 13:48 i-node (index-node) とは
- 21:22 マウントとは
- 34:48 ファイルシステムの保守
- 48:04 LVM (Logical Volume Manager) とは
- 1:03:19 本日のまとめ
- 1:07:38 Q&A (ライブ回答)

<https://www.youtube.com/watch?v=Yfm29vHFU7s&feature=youtu.be>



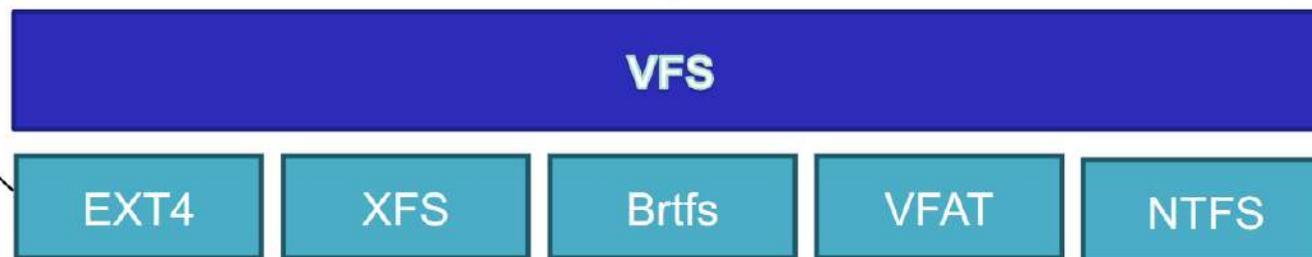
■ VFSのアクセスイメージ



#LinuC学習中

ファイル操作のコマンド
ls,cd,cp,等

同一の方法で異なるターゲット
に対してアクセスできる



異なるファイル
システム群



デバイスの相違を
意識しない

https://linuc.org/docs/seminar/20201205_linuc2.pdf



■ファイルシステムの保守

✓ ファイルシステムが適切に運用されているかを管理

- ストレージの状態確認
- 残容量のチェック
- ファイル数のチェック
- ファイルシステム破損時の対応

✓ ファイルシステム操作コマンド

- mkfs : ファイルシステム作成
- fsck : ファイルシステムチェック
- tune2fs : ファイルシステムのパラメータ操作
- dumpe2fs : ファイルシステムの情報表示
- df : ファイルシステムディスク情報の表示

https://linuc.org/docs/seminar/20201205_linuc2.pdf



#LinuC学習中



■LVM(Logical Volume Manager)とは



#LinuC学習中

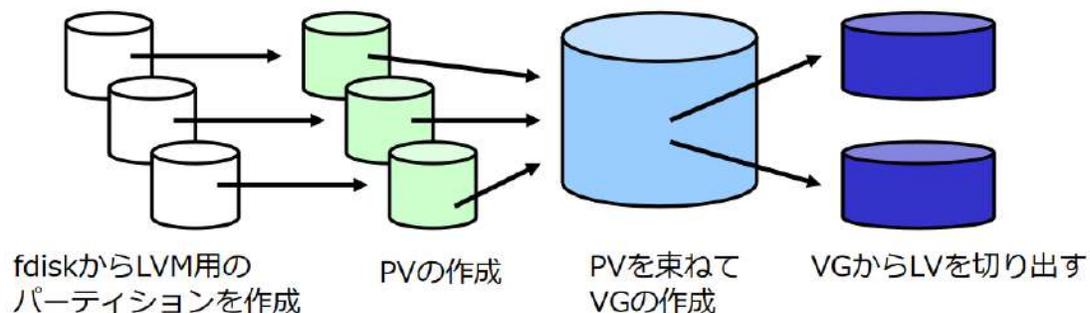
- ✓ 複数の異なるパーティション、物理ストレージを論理的に束ねるシステム
 - 動的なサイズ変更が可能
 - 物理的に異なるストレージ領域を1つの領域として扱うことができる

✓ LVMの構成要素

PV(Physical Volume) : LVM用に構成された、物理的なボリューム

VG(Volume Group) : 1つ以上のPVを束ねて構成される、論理的なボリューム

LV(Logical Volume) : VGから実際に利用する領域を切り出した論理的ボリューム



https://linuc.org/docs/seminar/20201205_linuc2.pdf



■ LVMのコマンド



#LinuC学習中

✓ PVの作成

- # pvcreate [PVにするデバイス ...]
- Ex. # pvcreate /dev/sdc1 /dev/sdc2 /dev/sdc3

✓ VGの作成

- # vgcreate [VG名] [VGに加えるPVデバイス]
- Ex. # vgcreate sample_vg /dev/sdc1 /dev/sdc2 /dev/sdc3

✓ LVの作成

- # lvcreate -n[LV名] -L[サイズ] [VG名]
- Ex. # lvcreate -n sample_lv -L 3g sample_vg

https://linuc.org/docs/seminar/20201205_linuc2.pdf



2.03 : ネットワーク構成

2.03.1 基本的なネットワーク構成

2.03.2 高度なネットワーク構成

2.03.3 ネットワークの問題解決



2.03.1 基本的なネットワーク構成

重要度 3

概要 ネットワークデバイスを設定し、有線または無線のローカルネットワークと広域ネットワークに接続できる。

詳細

- Ethernetネットワークインターフェイスを設定および操作する。デフォルトルートの設定を含む。
 - ip, ifconfig, route, arp, nmcli
- 無線ネットワークを構成する。
 - iw, iwconfig, iwlist

2.03.2 高度なネットワーク構成

重要度 3

概要

- 複数のサブネットへの経路設定ができる。これにはルータ機能の設定も含まれる。
- ネットワークの状態を監視できる。
- ネットワークのデバイス、通信状態などを分析できる。

詳細

- ルーティングテーブルを操作するユーティリティ
 - ip, route
- IPフォワードを設定してルータ機能を実装する。
 - /etc/sysctl.conf, sysctl
- ネットワークデバイスの状態を分析するユーティリティ
 - ip, ifconfig
- TCP/IPの通信状態やトラフィックを監視および分析するユーティリティ
 - ping, ping6, netcat(nc, ncat), tcpdump, nmap, ss, netstat

2.03.3 ネットワークの問題解決

重要度 3

概要 一般的なネットワーク設定に関する問題を特定して解決できる。これには、基本的な設定ファイルの位置とコマンドに関する知識も含まれる。

詳細

- ネットワークの設定に関する情報を取得する。
 - hostname, /etc/hostname, /etc/hosts, /etc/resolv.conf, nmcli, ip
- ネットワークの通信経路の問題を特定して解決する。
 - traceroute, traceroute6, ip, route, mtr
- ハードウェアの認識と利用に関する情報を取得する。
 - dmesg, /var/log/syslogおよび/var/log/messagesなどのシステムのログファイルおよび systemd のジャーナル
- システムの初期化ファイルとその内容 (systemd)
- NetworkManagerおよびそれがネットワーク設定に及ぼす影響について知っている。
 - /etc/network/, /etc/sysconfig/network-scripts/



2.03 : ネットワーク構成

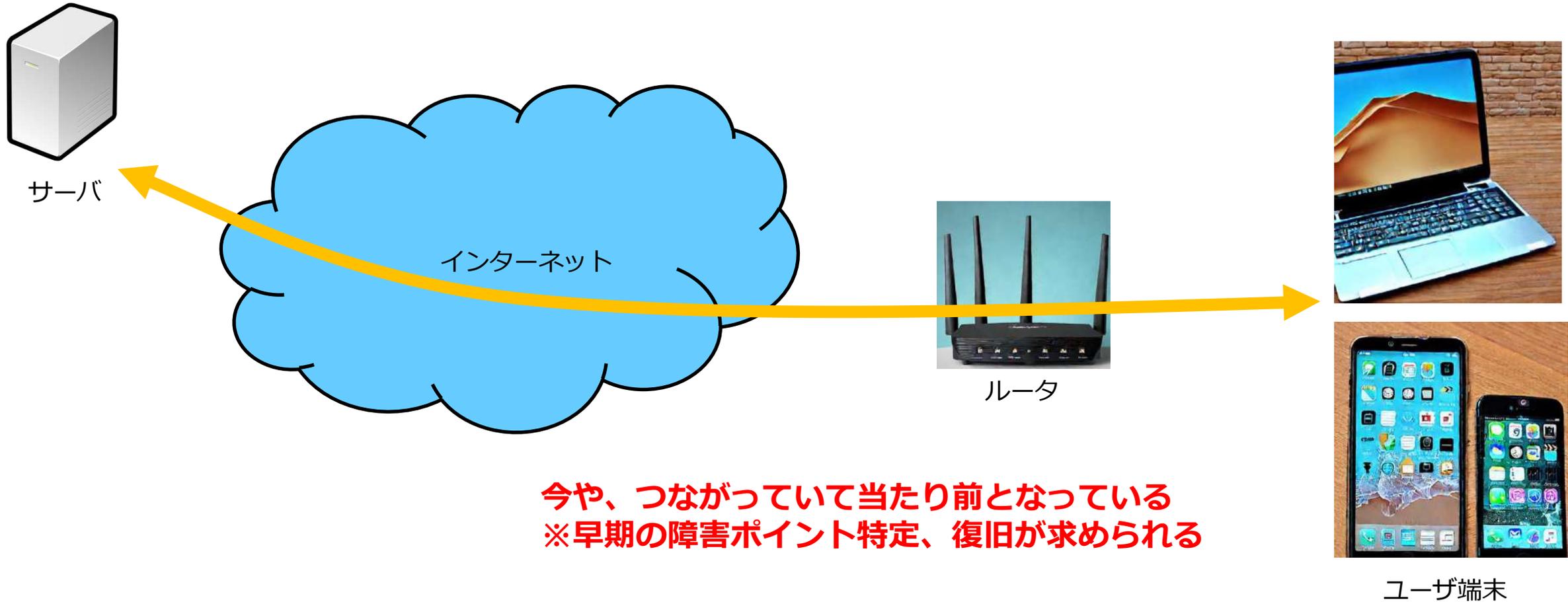
The video player shows a thumbnail with the LinuxC-2 logo and the title 'LinuxC-2 ネットワーク構成'. The video title is '主題2.03 : ネットワーク構成' and the main title is 'ネットワーク構成'. The instructor is identified as '講師 株式会社ゼウス・エンジニアリング 鯨井 貴博 氏'. The video player includes a progress bar at 0:00 / 1:04:32 and a banner for 'LinuxC open your NEXT future LPI-JAPAN'. Below the video, the channel name 'LPI-Japan' with 5.56K subscribers is shown, along with engagement icons for likes (22), shares, clips, and saves. A table of contents is visible at the bottom of the player.

Time	Content
00:00	スタート
04:30	アジェンダ~LinuxCの紹介
13:59	ネットワーク構成
23:27	基本的なネットワーク構成
31:25	高度なネットワーク構成
41:29	ネットワークの問題解決
50:37	wireshark (パケットキャプチャツール) のご紹介
1:04:32	Q&A

<https://www.youtube.com/watch?v=EXSCcit4HAc&feature=youtu.be>



ネットワークの重要性





トラブルシューティングの進め方(OSI参照モデル)

階層	階層名	取り扱う情報など
Layer7	アプリケーション層	アプリケーションデータ
Layer6	プレゼンテーション層	
Layer5	セッション層	
Layer4	トランスポート層	ポート番号(TCP/UDPなど)
Layer3	ネットワーク層	IPアドレス
Layer2	データリンク層	MACアドレス(ARP)
Layer1	物理層	物理メディア



**トラブルシューティングは
下の階層から！**



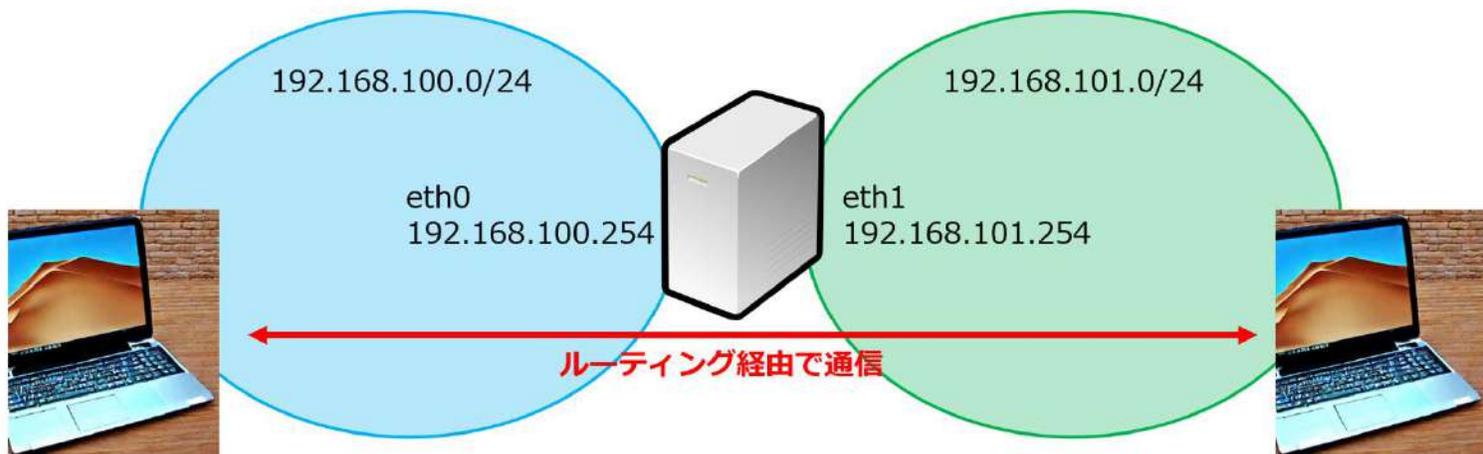
各種コマンドや各種ファイルの理解

- ✓ ip
- ✓ ifconfig
- ✓ route
- ✓ arp
- ✓ nmcli
- ✓ iw
- ✓ iwconfig
- ✓ iwlist
- ✓ /etc/sysconfig/network-scripts/ifcfg-xxxxxx



2.03 : ネットワーク構成

Linuxをルータとして動作させる



192.168.1.0/24 複数のPCを連携させるためのNW



内部アプリケーションを動作させるためのNW
Podネットワーク
10.0.0.0/16



2.04 : システムの保守と運用管理

2.04.1 makeによるソースコードからのビルドとインストール

2.04.2 バックアップとリストア

2.04.3 ユーザへの通知

2.04.4 リソース使用状況の把握

2.04.5 死活監視、リソース監視、運用監視ツール

2.04.6 システム構成ツール



2.04.1 makeによるソースコードからのビルドとインストール

重要度 3

概要 ソースコードから実行プログラムをビルドしてインストールできる。これには、ソースファイルの展開も含まれる。

- 詳細**
- gitを使ってソースコードを入手する。
 - git clone, git tag -l, git checkout
 - 一般的な圧縮およびアーカイブユーティリティを使用して、ソースコードを展開する。
 - gzip, gunzip, bzip2, xz, tar, unzip
 - ソースコードにパッチを適用する。
 - patch
 - configureスクリプトにパラメータを適用する。
 - configure
 - プログラムをビルドするmakeの実行について基本を理解する。
 - make, make install

2.04.2 バックアップとリストア

重要度 3

概要

- 重要なシステムデータをバックアップするシステムツールを使用できる。
- Linuxシステムのバックアップ計画を立てられる。

- 詳細**
- バックアップに含める必要があるディレクトリについて決定できる。
 - Amanda, Bacula, Bareos, BackupPCなどのネットワークバックアップソリューションについて知っている。
 - バックアップ対象として、ファイル、ブロック、イメージを使い分けられる。
 - テープ、ハードディスク、光メディアまたはその他のバックアップメディアを選択できる。
 - 容量、保存期間、ライトワンス、シーケンシャル/ランダムアクセス
 - 完全バックアップ、差分バックアップ、増分バックアップを使い分けられる。
 - バックアップファイルの整合性を確認する。
 - バックアップを部分的または完全に復元する。
 - dd, tar, /dev/st*, /dev/nst*, mt, rsync

2.04.3 ユーザへの通知

重要度 1

概要 現在発生しているシステム関連の問題についてユーザに通知できる。

- 詳細**
- ログオンメッセージを使用してユーザへの通知を自動的に行う。
 - /etc/issue, /etc/issue.net, /etc/motd
 - アクティブなユーザにシステムの保守を通知する。
 - wall, shutdown, systemctl



2.04.4 リソース使用状況の把握

重要度 3

概要 ハードウェアリソースとネットワーク帯域幅の使用率を測定でき、リソースの問題を解決できる。

- 詳細
- CPU使用率を測定する。
 - top, htop, ps, sar
 - メモリ使用量を測定する。
 - vmstat, free, sar
 - ディスクI/Oを測定する。
 - iostat, iotop, sar
 - I/O 待ちのプロセス, blocks in, blocks out
 - ネットワークI/Oを測定する。
 - netstat, iftop, ss, sar
 - システムがオープンしているファイルやポートを表示する。
 - lsof
 - ファイアウォール機能とルーティングスループットを測定する。
 - クライアントの帯域幅使用率をマップする。
 - ネットワーキングを含むシステムにおいてスループットを推定し、ボトルネックを見つけ出す。
 - iptraf

2.04.5 死活監視、リソース監視、運用監視ツール

重要度 2

- 概要
- システム監視の重要性を理解している。
 - 監視対象に応じて適切な監視手法を選択できる。
 - 運用監視ツールの導入メリットと種類を理解し、適切にシステムを監視できる。

- 詳細
- システムの障害と予兆についての関連性を知っている。
 - リソースの枯渇、過負荷、異常停止、結果異常、OOMKiller
 - システムの死活、サービスの死活
 - 死活監視の対象と手法
 - レスポンス、ログ監視（サーバー、サービス、プロセス、ネットワーク）
 - リソース監視の対象と手法
 - ログ監視、使用率監視（CPU、メモリ、ストレージ、通信量）、SNMP
 - 運用監視ツールによる監視作業の標準化と自動化
 - 標準管理項目、標準のしきい値、標準のアラート方式
 - 監視方式のパッケージ化、複数台の監視
 - 主要な監視ツールを知っている。
 - Icinga2、Nagios、collectd、MRTG、Cacti、Zabbix



2.04.6 システム構成ツール

重要度 4

概要

- システム構成ツールの必要性を理解している。
- システム構成ツールを使って、対象ホストの設定が行える。

詳細

- システム構成ツールの機能、メリットを知っている。
 - 自動化の標準化, 効率化, スケール可能, 冪等性
- Ansible の構成要素を理解している。
 - インベントリ、モジュール、Playbook
- システムの構成変更を自動化する。
 - 仮想サーバー・コンテナの払い出し
 - アプリケーションのリリース
 - ネットワーク機器のステータス取得・設定変更
- 自動化のためのファイルとツール
 - Playbook, YAML
 - ansible, ansible-playbook



2.04 : システムの保守と運用管理



講師

福田 浩之 氏

主題2.04 : システムの保守と運用管理

死活監視、リソース監視、 運用監視ツール

LinuCレベル2 技術解説セミナー



講師

株式会社カサレアル
植草 克友 氏

主題2.04 : システムの保守と運用管理

システム構成ツール

LinuCレベル2 技術解説セミナー



システムの保守と運用管理 / リソース監視 (Linux学習)

LPI-Japan 5.56K subscribers

3,829 views Dec 19, 2020

00:00 スタート

07:21 本日のテーマのご紹介

10:44 リソース使用率の測定 (topコマンド)

22:47 リソース使用率の測定 (CPU使用率、メモリ使用率、他)

29:03 ネットワークトラフィックの測定 (netstat/ss)

33:00 死活監視と運用監視ツール

36:11 SNMPの紹介

44:33 Zabbixの概要とデモ

55:45 Q&A (ライブ回答)

システム構成ツール / Ansibleとは (Linux学習)

LPI-Japan 5.56K subscribers

3,775 views Jul 1, 2020

00:00 スタート

06:50 Infrastructure As Code

18:59 Ansibleとは

27:28 デモ : Ansibleのインストール

44:43 Terraformとの連携

50:58 Q&A (ライブ回答)

1:03:27 Q&A (いろいろな質問への全ての回答)

<https://www.youtube.com/watch?v=zQO4Q3GaxOI>

<https://www.youtube.com/watch?v=ck3XyhfIBvo>



■ システムの監視

• 死活監視とは

- サーバーやネットワーク機器、アプリケーション（サーバーサービス）が正常に動作するか監視する。
- 障害発生をいち早く検知することができる。

• リソース監視

- CPU/メモリ/ディスク/ネットワークなどのリソースの使用状況を確認することにより、システムやパフォーマンスが十分であるか監視する。
- サーバーやネットワーク機器、NASなどはSNMPエージェントとして動作していることが多く外部からSNMPコマンドを利用して情報収集することが可能。

• アプリケーションの監視

- サーバーで稼働しているアプリケーションのリソースの使用状況やステータスを監視する。



#LinuC学習中

https://linuc.org/docs/seminar/20201219_linuc2.pdf

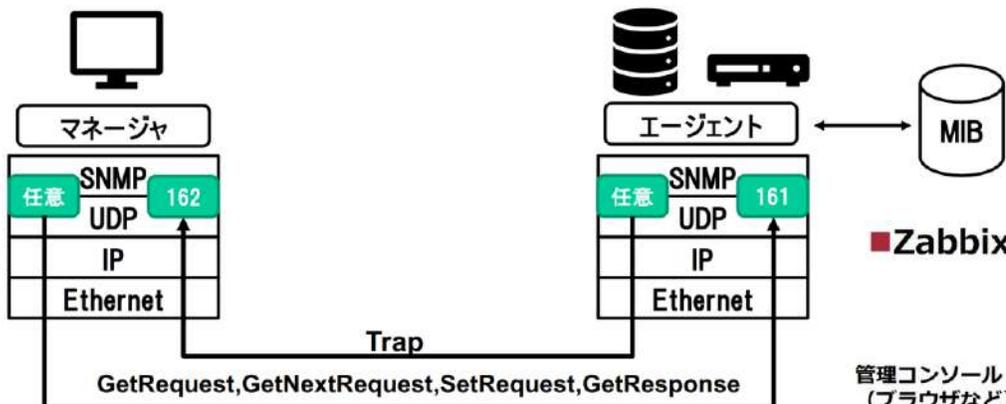


2.04 : システムの保守と運用管理

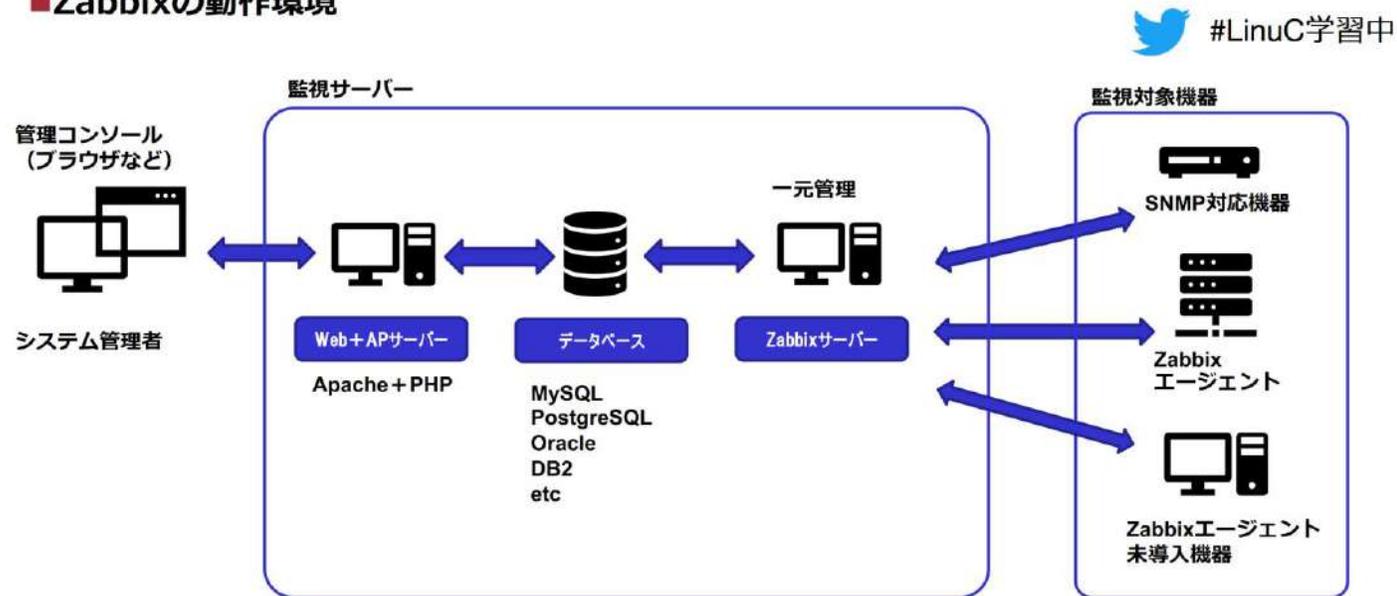
■SNMP概要

- 管理装置であるマネージャと管理対象機器であるエージェント間で情報の交換を行うプロトコル。
- 主にUDPポート161、162番が使用される。
- エージェントはMIBと呼ばれるデータベースにおいて情報を蓄積し、マネージャの要求 (Request) に応答 (Response) する。

#LinuC学習中



■Zabbixの動作環境



#LinuC学習中

https://linuc.org/docs/seminar/20201219_linuc2.pdf



■「Infrastructure As Code」とは…

コードによるインフラストラクチャの構成と変更



#LinuC学習中

■背景(課題)

- インフラストラクチャの複雑化
 - 台帳管理 (IT資産の構成、更新履歴など)
 - 実態と台帳の不整合 (「冪等性」が確保されていない)
- 適用対象の増加 (ソフトウェアのアップデートなど)
 - 速やかに適用していく必要がある一方、適用後に正しく動作する事の確認も必要
 - 上記の適用および確認作業を、個別に手作業で実施することの限界

べきとうせい

「冪等性」とは…

ある操作を何度繰り返し実行しても、結果は変わらないといった性質のこと

■対策のアプローチ

- 台帳管理 → 構成や定義のコード化 (※コードのバージョン管理が前提)
- 人手作業 → 自動化

<https://www.youtube.com/watch?v=ck3XyhfIBvo>

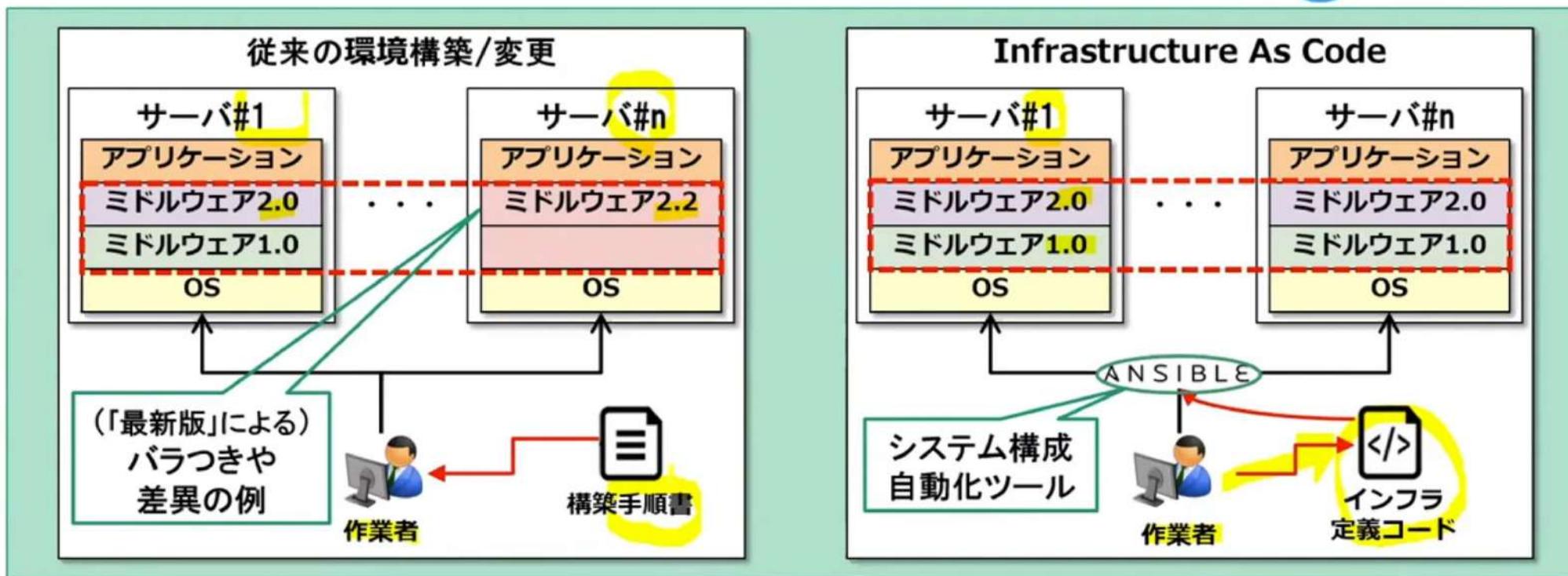


2.04 : システムの保守と運用管理

■従来の環境構築/変更と「Infrastructure As Code」の対比



#LinuC学習中



• 「Infrastructure As Code」による環境構築/変更

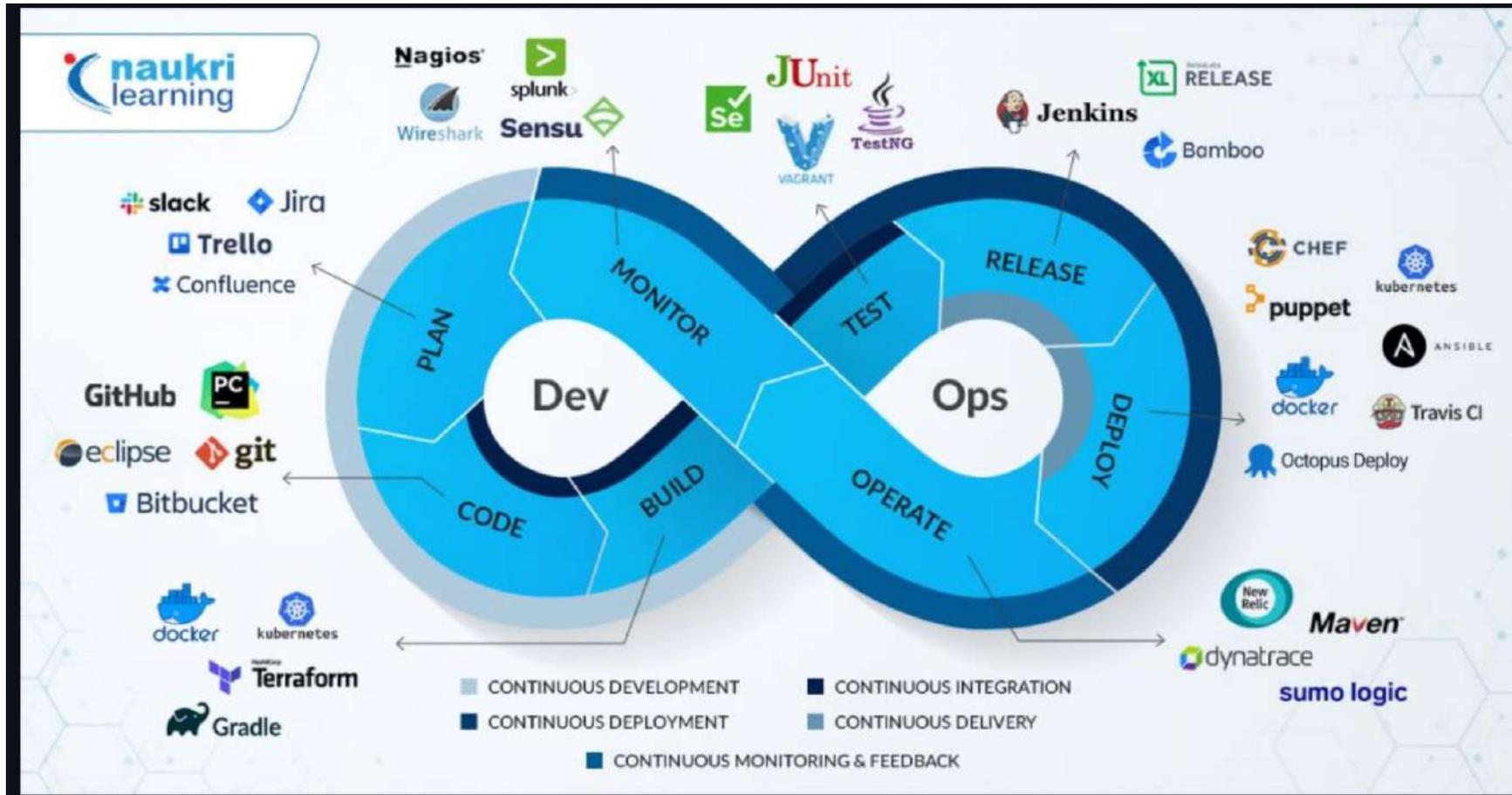
- コード化による実態(実際の構成や定義)と管理台帳間に生じる不整合の回避/低減
- 自動化によるヒューマンエラーの回避/低減

<https://www.youtube.com/watch?v=ck3XyhfIBvo>



2.04 : システムの保守と運用管理

DevOps(Sec)を理解するためには必須



<https://github.com/NotHarshhaa/devops-tools>



2.05 : 仮想化サーバー

2.05.1 仮想マシンの仕組みとKVM

2.05.2 仮想マシンの作成と管理



2.05.1 仮想マシンの仕組みとKVM

重要度 3

概要 基本的な仮想マシンの仕組みについて理解し、KVMを導入して仮想マシンの実行環境を構築できる。

- 詳細
- 仮想マシンの基本について知っている。
 - ホスト型とハイパーバイザー型 (KVM, VirtualBox, Xen)
 - コンピュータリソース (CPU、メモリ、ストレージ、ネットワーク) の仮想化
 - KVMを導入し、仮想マシンが稼働するための環境構築および設定ができる。
 - QEMU
 - 仮想化支援技術 (vmxとsvm)
 - /proc/cpuinfo, lscpu
 - KVMモジュール (kvm-intelとkvm-amd)
 - libvirtd
 - virt-manager
 - ネットワークの構成 (bridge-utils)

2.05.2 仮想マシンの作成と管理

重要度 3

概要 各種ツールを使って、仮想マシンを作成し、起動や停止ができる。

- 詳細
- 仮想マシンを作成し、OSをインストールする。
 - virt-manager, virt-install, 完全仮想化と準仮想化 (virtio)
 - 仮想マシンを起動、停止する。
 - virt-manager, virsh
 - パフォーマンスを監視する。
 - virt-manager



2.05 : 仮想化サーバー

The image shows a YouTube video player interface. On the left, the channel logo for 'LinuxC-2' is displayed, featuring a green circle with 'LC' and the text 'LinuC-2' below it. The instructor's name '末永 貴一 氏' (Mr. Kinoshita Takashi) and company 'エスディーテック株式会社' (Esdee Tech Co., Ltd.) are listed. The video title is '主題2.05 : 仮想化サーバー' (Topic 2.05: Virtualization Server) and '仮想マシンの概念、種類、利用シーン' (Concepts, Types, and Use Cases of Virtual Machines). Below the title, it says 'LinuCレベル2 技術解説セミナー' (LinuC Level 2 Technical Explanation Seminar). The video player shows a play button and a progress bar at 0:00 / 1:26:18. At the bottom, the channel name 'LPI-Japan' with 5,56K subscribers is shown, along with a 'Subscribed' button and engagement icons for likes (59), shares, clips, and saves. A table of contents is visible at the bottom of the player.

Time	Content
00:00	スタート
08:59	仮想マシンの基礎
22:06	KVMとは
32:30	QEMUとは
35:38	KVM環境構築
51:05	準仮想化
57:02	まとめ
59:25	Q&A (ライブ回答)
1:25:48	Q&A (頂いた質問への全ての回答)

https://www.youtube.com/watch?v=a_508d1h4TQ&feature=youtu.be



■仮想マシンとは・・・



#LinuC学習中

- ✓コンピュータ（機械）の動作をエミュレーションするもの
 - 機械（ハードウェア）自体の動作をソフトウェア的に再現する。
 - 動作しているOS（システム）上に別のコンピュータを実現させる。
- ✓同一コンピュータ上に複数のコンピュータ・OSを動作可能とする
 - 一般的にはOS上に異なるOSを複数動かせることができる環境
 - 同一コンピュータ上に同時に複数のOS（ゲストOS）を動作させることができる。



以上のような仮想マシンを実現させるために必要な演算管理、メモリ管理、入出力管理等を行う制御プログラムを**ハイパーバイザー**（仮想化モニター、仮想化OS）と呼ぶ。

参考：

シミュレータは外から見た振る舞いを再現するもの。エミュレータは中身の動作まで再現するもの。

https://www.youtube.com/watch?v=a_508d1h4TQ&feature=youtu.be



2.05 : 仮想化サーバー

■ハイパーバイザーの種類



✓ Type1 ハイパーバイザー

- ハードウェア上で直接動作する。
- ゲストOSはこのハイパーバイザー上で動作する。
- ベアメタルハイパーバイザー、ネイティブハイパーバイザーとも呼ばれる。
- Xen、KVM等

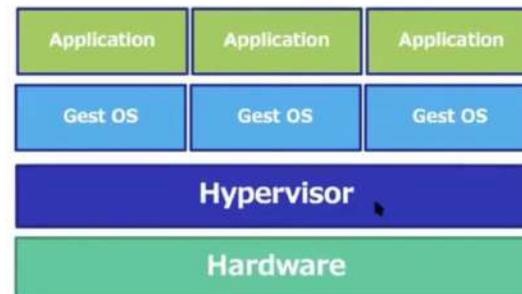
✓ Type2 ハイパーバイザー

- ハードウェア上で動作するOS上でアプリケーションとして動作する。
- ハイパーバイザーアプリケーション上でゲストOSは動作する。
- ホストハイパーバイザーとも呼ばれるが、厳密にはハイパーバイザーではない。
- VirtualBox、QEMU等

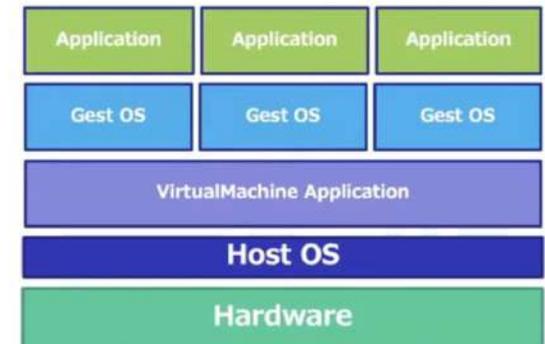
■ハイパーバイザーの種類



Type1



Type2





VMware製品ライセンス変更問題を追う

+ 特集をフォロー

第4回

どうする「脱VMware」、ユーザー企業が検討する主な移行先は3種類

渥美 友里 日経クロステック/日経コンピュータ

2024.06.27

有料会員限定



全3758文字

情報セキュリティ戦略セミナー2024～AI時代のサイバーセキュリティ対策 PR
<特別企画>五輪女子メダリストの共演！伊藤美誠×藤澤五月のスペシャルトーク PR
スキル、リスク、コスト……生成AIの導入を阻む3つの“逆風”を跳ね返す PR

「今年度中にVMware製品から移行してほしい」――。仮想化環境の構築・運用を担うITベンダーのA社の社員は、顧客からこう言われて頭を抱えている。これまで使っていなかったVMware以外の製品の知識や運用ノウハウを学ぶため、「移行先のハイパーバイザーに詳しい協力会社を探している」と話す。

米Broadcom（ブロードコム）によるVMware製品ライセンスの変更によって大幅な値上げに直面したユーザー企業が、「脱VMware」を検討し始めている。ITベンダーは顧客からの問い合わせに追われている状況だ。

ハイパーバイザーの観点で見れば、「VMware ESXi」の移行先は大きく3種類ある。Windows Server標準の「Hyper-V」、Linux標準の「KVM」、米Nutanix（ニュータニックス）の「Nutanix AHV」だ。ただし仮想化環境の構成要素はハイパーバイザーだけではなく、運用管理機能なども含めた「仮想化プラットフォーム」の観点で、移行先を考える必要がある。

VMware製品の移行先となる主な仮想化製品		
社名	製品名	ハイパーバイザー
米Microsoft	Windows Server	Hyper-V
米Nutanix	Nutanix Cloud Infrastructure	Nutanix AHV
米Red Hat	OpenShift Virtualization	KVM

VMware製品からの移行先の候補となる主な仮想化製品

(出所：日経クロステック)



2.06 : コンテナ

2.06.1 コンテナの仕組み

2.06.2 Dockerコンテナとコンテナイメージの管理



2.06.1 コンテナの仕組み

重要度 2

概要 基本的なコンテナの仕組みについて理解している。

- 詳細
- 物理マシン、仮想マシン、コンテナの特徴と違いを理解している。
 - コンテナのファイルシステムとイメージの関係を知っている。
 - コンテナを実現する技術の概念を知っている。
 - 名前空間, cgroups

2.06.2 Dockerコンテナとコンテナイメージの管理

重要度 3

- 概要
- Dockerを導入してコンテナ実行環境を構築できる。
 - Dockerコンテナを実行できる。
 - コンテナイメージを管理できる。

- 詳細
- Dockerを導入して、ネットワークを構成する。
 - ポート変換, フラットL2ネットワーク
 - Dockerコンテナを実行して、停止する。
 - `docker ps/stats`, `docker run/create/restart`, `docker pause/unpause`, `docker stop/kill`, `docker rm`
 - Dockerコンテナに接続してプロセスを実行する。
 - `docker attach`, `docker exec`
 - コンテナイメージを管理する。
 - Dockerレジストリ: `docker images`, `docker pull`, `docker rmi`, `docker import`
 - Dockerfile: `docker build`, `docker commit`



2.06 : コンテナ

LinuC-2

講師
株式会社ゼウス・
エンタープライズ
鯨井 貴博 氏

主題2.06 : コンテナ
コンテナ技術・Docker/
Kubernetesについて

LinuCレベル2 技術解説セミナー

LC LinuC open your NEXT future LPI-JAPAN

コンテナ技術・Dockerの解説 (Linux学習)

LPI-Japan 5.56K subscribers

3,450 views Oct 12, 2021

- 00:00 スタート
- 03:39 アジェンダ〜LinuCの紹介
- 11:08 コンテナ技術の解説
- 12:15 物理マシン・仮想マシン・コンテナの違い
- 25:06 Dockerの解説
- 48:08 コンテナ操作のデモ (Katacodaを使用)
- 54:47 Kubernetesとは
- 1:02:36 本日のまとめ
- 1:03:10 Q&A
- 1:19:23 Q&A (質問のテキストと全ての回答)

LinuC-2

講師
株式会社ゼウス・
エンタープライズ
鯨井 貴博 氏

主題2.06 : コンテナ
コンテナの仕組み、
Dockerコンテナの管理方法

LinuCレベル2 技術解説セミナー

LC LinuC open your NEXT future LPI-JAPAN

コンテナの仕組み (Linux学習)

LPI-Japan 5.56K subscribers

3,622 views Jun 4, 2020

- 00:00 スタート
- 00:23 講師紹介
- 04:41 LinuCの紹介
- 10:20 コンテナの仕組み (2.06.1)
- 21:27 Dockerコンテナとコンテナイメージの管理
- 44:14 Dockerコンテナの操作デモ
- 55:30 Q&A (ライブ回答)
- 1:11:17 Q&A (頂いた質問への全ての回答)

<https://www.youtube.com/watch?v=gSTGXuj-2wM>

<https://www.youtube.com/watch?v=Gm5KYhMs20k>



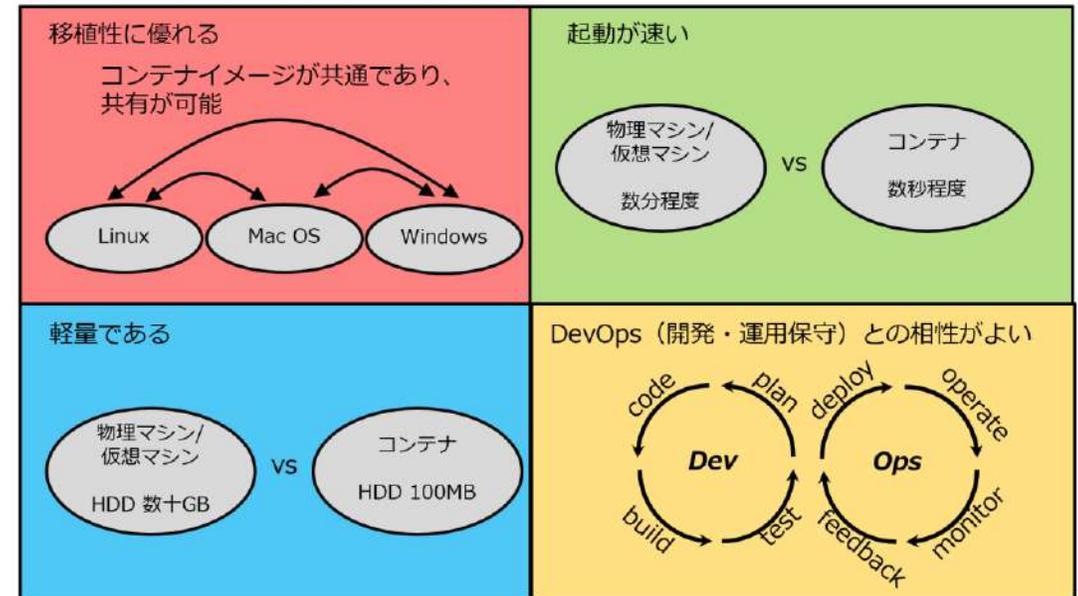
2.06 : コンテナ

コンテナの特徴

1HWを複数コンテナで共有 ⇔ 柔軟なリソース調整（増減）が可能
環境構築が容易（インストールの手間など）
軽量



コンテナの特徴

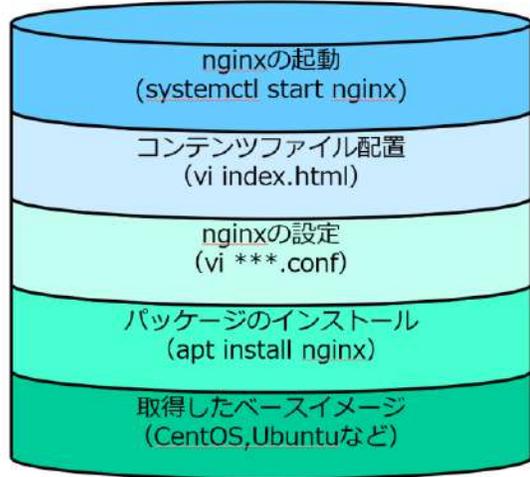


<https://xtech.nikkei.com/atcl/nxt/column/18/02864/062400006/>



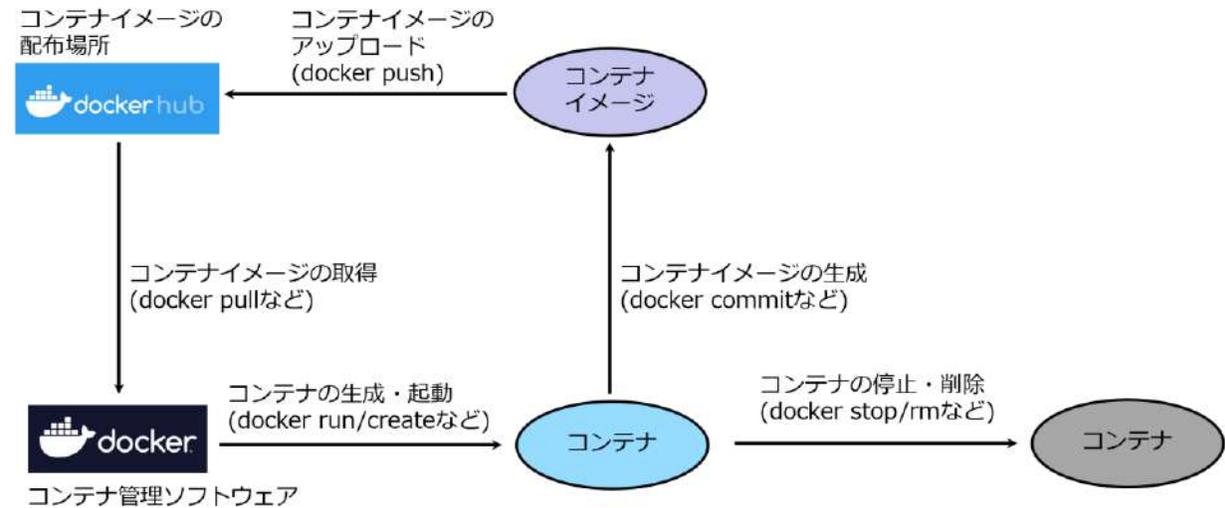
2.06 : コンテナ

ベースイメージを取得し、そこにイメージの層（レイヤー）を重ねて利用する



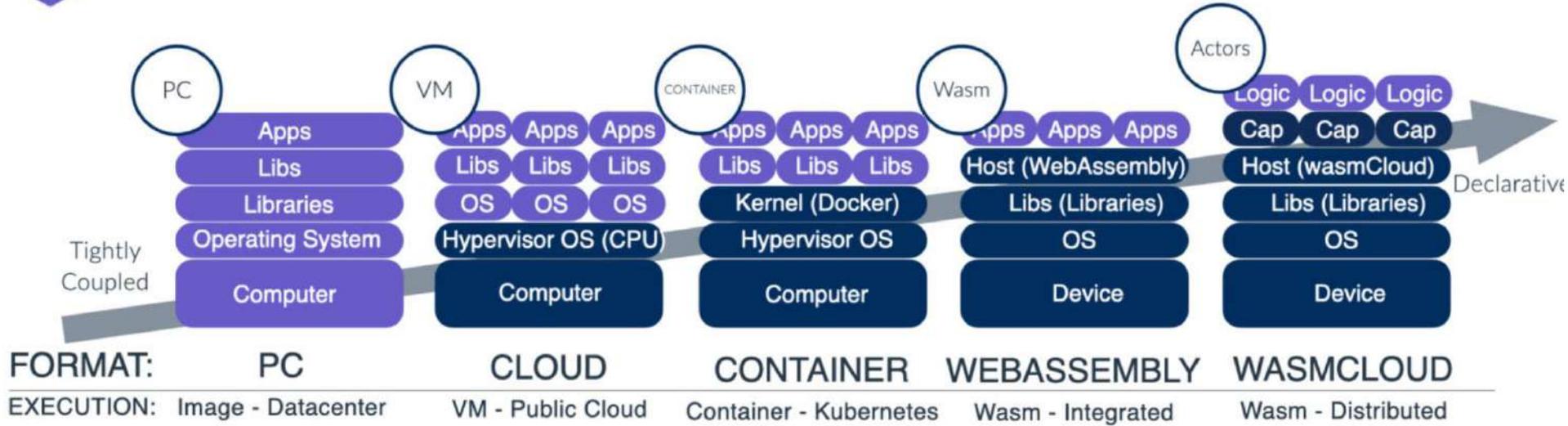
ベースイメージに対して、
コマンドを実施しサービスを構築

Overlay Filesystemという仕組みを使っている





2.06 : コンテナ



FORMAT: PC CLOUD CONTAINER WEBASSEMBLY WASMCLOUD
 EXECUTION: Image - Datacenter VM - Public Cloud Container - Kubernetes Wasm - Integrated Wasm - Distributed

Developer	Full	OS, App, Lib	App, Lib	Wasm	Business Logic
Abstraction	-	CPU	Linux Kernel	Secure Sandbox	Sandbox + Capabilities
Compatability	All	Most	Most	Most	Most
Size	Large	Med	Small	Tiny	Minuscule
Portability	-	Low	Med (CPU, Linux)	High	Highest
Security	System	OS	Process Boundary	Capability	Actor
Where		Proprietary Cloud & Edge	Dev, Edge, Cloud, K8s	Dev, App, Edge, Cloud, K8s, Browser, Devices	Dev, App, Edge, Cloud, K8s, Browser, Devices

Legend: Developer Provided
Device / Computer

Copyright © 2021 Cosmonic Corp.

www.Cosmonic.com

<https://b-nova.com/en/home/content/how-containerless-works-thanks-to-web-assembly-runtimes/>





Cloud Native Community Japan - Wasm Japan Meetup #1

Cloud Native Community Japan

Jul 22, 6:00 – 9:00 PM

In-person event



About this event

近年Wasm (WebAssembly)の注目が高まっており、CNCFでもWasmを専門に扱うイベントが行われるなど非常に活発です。

今回は、国内で早くから活動をされているWasm Night、ミドクラさんにもご協力頂き、Wasmについて、その入門から、エッジ・クラウドでの活用、LF・CNCFでの最新動向など広く共有します。

参加登録方法

「RSVPにログイン」より、LF Accountを作成し、ログインください。ログインし、「出欠確認」より参加登録できます。

本イベントのオーガナイザー

- Ryu Ishimoto, Midokura
- Chikoski, Wasm Night
- Yuichi Nakamura, Evangelist, Linux Foundation Japan/Chief OSS Strategist, Hitachi, Ltd.
- Toru Komatsu, Engineer, Preferred Networks, Inc
- Kohei Ota, Senior Field Engineer, Apple

<https://community.cncf.io/events/details/cncf-cloud-native-community-japan-presents-cloud-native-community-japan-wasm-japan-meetup-1/>



9/29に第2弾(202の解説)があります！

LinuCレベル2

Linux研修【オンライン】

LinuCレベル2 202試験概要解説

サーバー構築のできるLinuxエキスパートへの道

いいね! 0

シェアする

✕ ポスト

B! 0

セミナー開催日

2024年9月29日（日）13時00分～14時15分

<https://linuc.org/study/seminar/5730/>



提供するITサービス

Network Engineering Service

高い専門スキルを有するエンジニア集団だから可能な質の高いソリューション

ゼウス・エンタープライズは、時代変革の要となるネットワーク・セキュリティ分野に特化したエンジニア集団として、顧客のニーズや課題に迅速かつ確実に応え、満足度の高いIT支援サービスを提供しています。情報通信・官公庁・金融・製造などの様々なクライアント先にてTCP/IPスタックの機器や、Linuxにおける豊富な経験と高度な技術を活用し、ネットワークやセキュリティ分野のパフォーマンスを最大限に引き出します。

主な業務としては、小規模LANから大規模WANまでのネットワーク構築や運用支援。各種アプリケーションの実行基盤やデータベースなど業務サーバーの構築や運用支援。また、オンプレ環境やクラウド環境、ハイブリッドクラウドの環境においても、セキュリティを重視した構築や運用支援を提供しています。

そして、当社は活躍する社員一人ひとりの能力を昇華させるべく、「ゼウスITトレーニングセンター」という教育機関を併設しており、ネットワークやLinuxを中心に、時代のトレンドに沿ったインフラ教育を行っています。

日々変革を遂げるIT業界に伴い、研修にて社員のスキルを底上げし、ネットワーク・セキュリティに特化したスペシャリスト集団として、クライアントの課題解決に貢献いたします。



<https://www.zeus-enterprise.co.jp/solution/service/>

リナックス・ネットワークに強いITスクール

IT Training

未経験者を戦力に育て上げた独自のカリキュラムに定評があるITキャリアスクールです

LPI-Japanのアカデミック認定校であるITキャリアスクール「ゼウスITトレーニングセンター」を運営し、リナックスとネットワークに強いエンジニアを育成します。当社社員の研修カリキュラムを基にしているため、プロの技術者だけでなく未経験者までも現場ですぐに活躍できる人材に育てます。セキュリティ分野から開発分野まで、クライアントの要望に合わせて講座を選択できるよう、幅広いコースを展開しています。



コース紹介

Course introduction

- > Linux & ネットワーク 講座
- > AWS 講座
- > Web 資格対策講座
- > Linux & Network in English 講座
- > 情報セキュリティ 講座
- > MS Office 講座
- > パソコン 講座
- > Kubernetes 研修

<https://www.it-training.tokyo/>



技術を楽しみながら、一緒に働ける仲間を募集中！



RECRUIT

採用情報

トップ > 採用情報

整った職場環境、充実した制度で 「一生働ける会社」を実現します

ゼウス・エンタープライズは、全社員の終身雇用を目指し、働きやすい社内環境作りに力を入れています。「ゼウスITトレーニングセンター」を研修拠点として、経験の有無を問わず、着実な技術の習得に導きます。また、独自の福利厚生制度を導入し、社員の健康的・文化的な生活を支援します。「安心できる環境で長く働きたい」「確かなスキルを身につけて活躍したい」「ワークライフバランスを大切にしたい！」という方のエントリーをお待ちしています。

Pick-up	現在募集中の	①ネットワークエンジニア <small>新卒</small> <small>キャリア</small>
	職種	②バイリンガルコーディネータ <small>新卒</small> <small>キャリア</small>

<p>新卒採用応募事項</p> <p>ENTRY →</p>	<p>キャリア採用応募事項</p> <p>ENTRY →</p>
---------------------------------------	---

会社説明会 →

<https://www.zeus-enterprise.co.jp/recruit/>



Thank you for joining my session!





Q and A