

LinuC レベル2 技術解説 オンラインセミナー

～受験に向けての準備とポイント解説～

2020/10/04 (Sun) 13:00-14:00

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 エバンジェリスト)



Who are you? (講師紹介)

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エバンジェリスト

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)



提供するITサービス

リナックス・ネットワークグループ



リナックスに特化したエンジニアとヘルプデスクの常駐型支援サービスを行っています。研修は、エンジニアの育成に特化したLPI-Japanのアカデミック認定校「Zeus Linux Training Center / Zeus Network Training Center」で行います。資格取得の支援、新しい知識の習得に努め、つねに高水準のサービスの提供をめざしています。

コンピュータシステムの企画・開発・保守

請負業務

特定労働者派遣

・ヘルプデスク

・エンジニア



アプリケーショングループ



Webアプリケーションやお客様のシステムなど、プログラミング言語を用いた開発支援サービスを行っています。

また、「失敗しないお店選び」をモットーとしたグルメサイト「Gourmet True」を2020年1月にリリース。これを皮切りに様々なインターネットサービスも開発しています。

アプリケーション開発

システム開発



クロスボーダーSE&ITソリューショングループ

語学力堪能なエンジニア(クロスボーダーSE)とヘルプデスク(バイリンガルコーディネータ)による支援サービスを行っています。

技術用語を多く含む会話の通訳、ドキュメントの翻訳という専門性を活かし、オフショア開発のプロジェクト、外国人技術者を多数かかえる現場などにありがたい意思疎通面でのデメリット回避などに力を発揮します。

コンピュータシステムの企画・開発・保守

翻訳業務 通訳業務

積算代行 与信代行サービス 特定労働者派遣

・バイリンガルコーディネータ

・クロスボーダーSE



Zeus Linux Training Center / Zeus Network Training Center

当社で培われた技術者向け研修カリキュラムを、一般のお客様や企業様向けに提供することを目的としたITキャリアスクール「Zeus Linux Training Center / Zeus Network Training Center」を運営しています。

当スクールはLPI-Japanのアカデミック認定校です。未経験者からプロの技術者養成まで、「現場ですぐに活躍できる人材」の育成をめざした研修は、技術者を求める企業からも高い評価をいただいています。

教育事業 (BtoB、BtoC)

IT・英会話・マナー

社員教育

試験会場

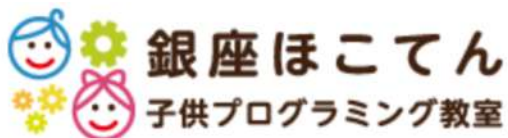


<https://www.zeus-enterprise.co.jp/solution/service>





提供するITサービス



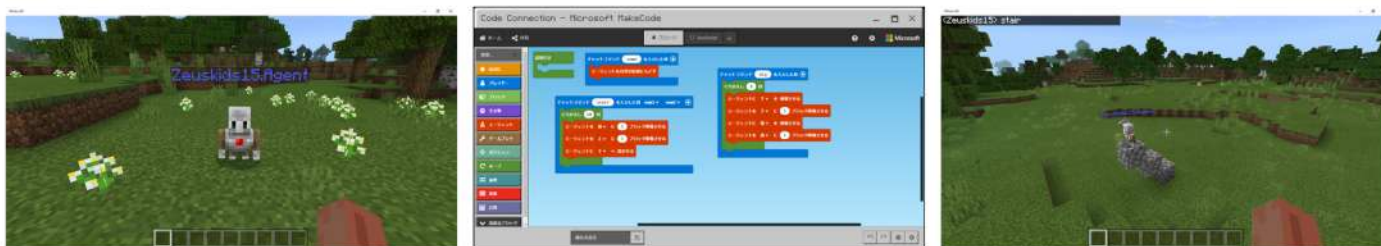
COURSE ～コース紹介～

Minecraftコース

大人気ゲーム！Minecraftを使って楽しくプログラミング！

「Minecraft MakeCode」では、通常のMinecraftとは違い、エージェントという小さなロボットをプログラムによって操ることで、プレイヤーの代わりに様々な作業をととても短い時間で行わせることができます！プログラミング的手法を使い、Minecraftの世界を自由に作り上げましょう！

→各コマ、集合型レッスン定員12名・オンラインレッスン定員3名



※ 保護者同伴可能

オンラインレッスン対応!!



当スクールでは通常のレッスンをオンラインでもご受講いただけます！
インターネット環境とPCをお持ちでしたら、Zoomを使用し
オンラインコースでもプログラミングを学べます！

<https://www.it-training.tokyo/kids/index.html>



会場

銀座ほこてん子供プログラミング教室
〒104-0061 東京都中央区銀座5丁目8-20 銀座コア8階

対象年齢

小学校3年～6年

持ち物

筆記用具

月謝

6,000円～（税込） / 月2回～（1コマ60分）

教材費

Minecraftのライセンス代：3300円（税込） / テキスト
代：2530円（税込）

無料体験 期間

3回目のレッスンまで無料で受講いただけます！



1. LinuC Level2試験とは
 - 試験概要と特徴
2. 技術解説
 - 主題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
 - 2.01.3 Linuxカーネルの構成要素
 - 2.01.4 Linuxカーネルのコンパイル
3. Q&A



2020年4月1日 LinuC レベル1・2 新バージョン Version10.0リリース！

LinuCレベル1 /レベル2 Version 10.0は、クラウド時代に活躍する全てのIT技術者が身につけておきたい内容を網羅すべく生まれ変わった、新しい「LinuC」です。

1. 現場で求められている技術要素に対応

- ・クラウドを支える仮想化技術領域に拡大
- ・オープンソースの文化への理解を追加
- ・システムアーキテクチャの要素を導入

2. 全面的に見直した出題範囲

新しい技術要素の追加はもちろん、古い技術の削除や入れ替え、出題構成の見直し、出題範囲の記述の詳細化と平易な説明文への変更と併せて、学びやすさ教えやすさを実現

3. Linuxにとどまらない認定領域

Linuxシステムの構築・運用に関わるLinux技術者にとどまらず、クラウドシステムや各種アプリケーション開発に携わるIT技術者にとっても有効な技術認定へ発展

旧バージョンとの差分については、こちら。
https://linuc.org/docs/linuc_ver10.pdf



仮想環境を含むLinuxのシステム設計、ネットワーク構築において、アーキテクチャに基づいた設計、導入、保守、問題解決ができる技術者を認定する試験

- ✓ Linuxシステムの設計、構築、監視、トラブルシューティングができる。
- ✓ 仮想マシンやコンテナの仕組みを理解し、その管理と運用ができる。
- ✓ セキュリティとシステムアーキテクチャの基本を理解し、サービスの設計、構築、運用・管理ができる。

201試験/202試験の両方に合格することでレベル2に認定
1試験あたり 15,000円（消費税別）
ピアソンVUEにおけるCBT or PBT（団体受験用）

詳細は、<https://linuc.org/linuc2/>





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

<https://linuc.org/linuc2/range/201.html>

<https://linuc.org/linuc2/range/202.html>



- 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。

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

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。



人気の技術解説無料
セミナーも活用して

LPI-Japanでは、『LinuxC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



LinuC学習のコツ

④過去セミナーの動画

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

open your NEXT future

LinuC OSS-DB Silver/Gold HTML5 Professional Certification Apache Cloud Stack Certification Exam OPCEL Certification Exam

頼られるための、頼れる資格 LPI-JAPAN

LPI-Japan
チャンネル登録者数 401人

チャンネル登録

ホーム 動画 再生リスト チャンネル フリートーク 概要

LinuC 技術解説無料セミナー ▶ すべて再生

テーマ	バージョン	タイトル	長さ	視聴回数	投稿日時
システムの保守と運用管理 (Ansibleの概要/ Infrastructure As Codeの説明、他)	LinuCレベル2 Version10.0	技術解説無料セミナー「システム構成...」	1:04:05	288 回視聴	2 か月前
仮想化サーバー	LinuCレベル2 Version10.0	技術解説無料セミナー「仮想化サー...	1:26:19	322 回視聴	2 か月前
クラウドセキュリティの基礎 (Linuxインスタンスの構成やSSHによる公開鍵の利用など)	LinuCレベル1 Version10.0	技術解説無料セミナー「クラウドセキ...	1:04:11	392 回視聴	2 か月前
仮想マシン・コンテナの 概念と利用	LinuCレベル1 Version10.0	技術解説無料セミナー「仮想マシン・...	1:07:35	1097 回視聴	3 か月前
コンテナ (コンテナの仕組み/ Dockerコンテナとコンテナイメージの管理)	LinuCレベル2 Version10.0	技術解説無料セミナー「コンテナ...	1:12:17	453 回視聴	3 か月前

HTML5 技術解説無料セミナー

[2020/07/12]HTML5-1技術解説セミナー「オフライン・ストレージ系API概要」





主題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

カーネルロードブル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。

デーモン / 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



2.01.3 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となる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/

2.01.4 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



What's Linux kernel ?



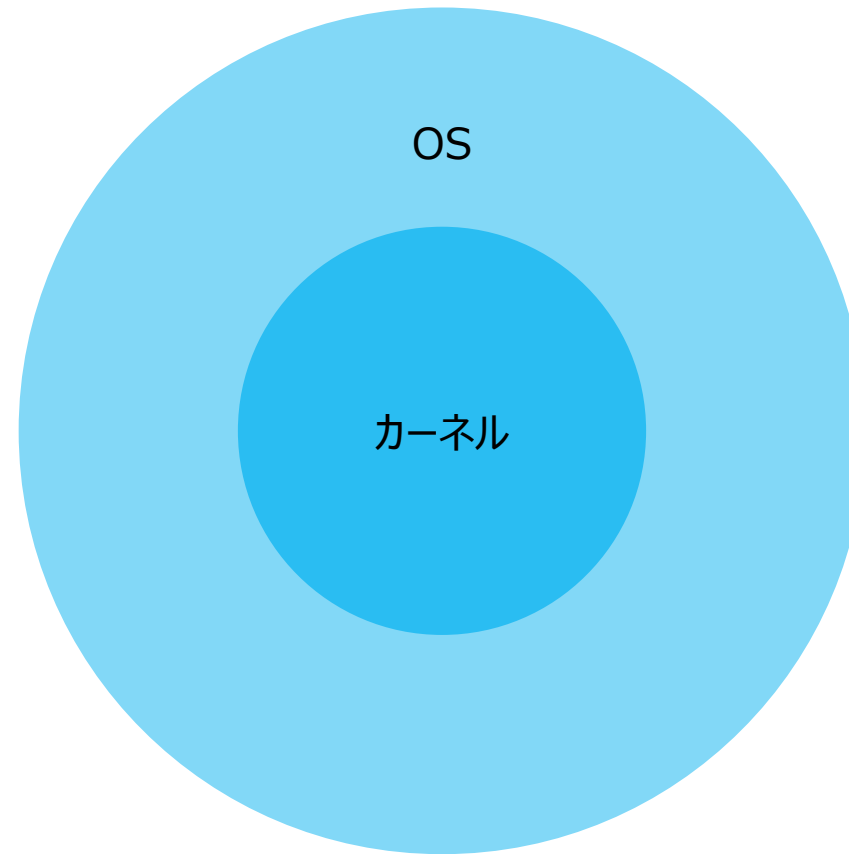
<https://www.kernel.org/>



What's Linux kernel ?

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。





What's Linux kernel ?

Linuxカーネルの歴史（1991年8月25日）

Hello everybody out there using minix -

I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things).

I've currently ported bash(1.08) and gcc(1.40), and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. Any suggestions are welcome, but I won't promise I'll implement them :-)

Linus (torv...@kruuna.helsinki.fi)

PS. Yes - it's free of any minix code, and it has a multi-threaded fs. It is NOT protable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-).

<https://groups.google.com/forum/#!original/comp.os.minix/dlNtH7RRrGA/SwRavCzVE7qJ>



What's Linux kernel ?

Linuxカーネルの歴史

Linuxカーネルの歴史をたどる 2020 Linux Kernel History Report を公開

By The Linux Foundation | 8月 25, 2020

Download the 2020 Linux Kernel History Report

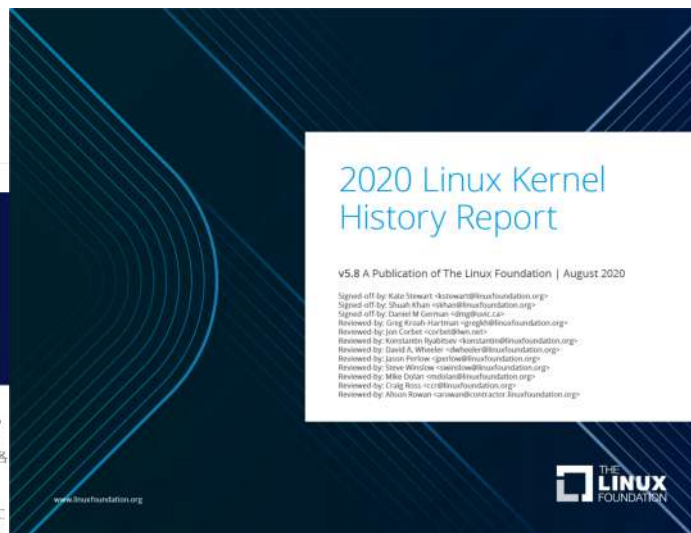
過去数十年にわたりLinuxは着実に成長し、最も広く使われているオペレーティング システム カーネルになりました。センサーからスーパーコンピューターまで、宇宙船、自動車、スマートフォン/ウォッチなど、さまざまなデバイスでLinuxが使用されています。Linux Foundationは、2008年に Linux Kernel Development Reports (Linuxカーネル開発レポート) の発行を開始して以来、各時点における進捗を観察してきました。

1991年の最初のリリース以来、Linuxは2万人を超えるコントリビューターを擁する歴史上最も成功したコラボレーションの一つになりました。バージョン5.8の最近の発表がこれまで最も大きいカーネルの一つであることを考えると、最新のリリースで1時間あたり10+コミットという新記録を示すなど、開発がスローダウンする兆しは見えません。

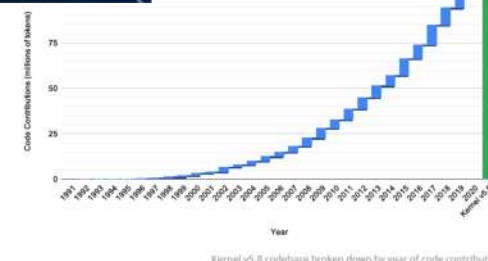
本レポートは、Linuxの全歴史をカバーしています。Linuxの分析は、初期リリースおよび1991年9月17日の最初のカーネルリリースから2020年8月2日までの期間のBitKeeperとgitの開発者コミュニティによるコミットに基づいています。2020年8月2日にLinux 5.8がリリースされ、5.9のマージウィンドウが完了した現在、過去29年間記録されたLinuxカーネルヒストリーの100万件を超えるコミットを分析することが可能になりました。

レポートでは、Linuxカーネルの歴史と、最も重要なソフトウェアコラボレーションの一つであるLinux開発を可能にするために出現したベストプラクティスとツーリング インフラストラクチャの影響について振り返ります。

1991年 0.01リリース
1994年 1.0リリース
1996年 2.0リリース
2003年 2.6リリース
2011年 3.0リリース
2015年 4.0リリース
2019年 5.0リリース



As Table 1 illustrates, over half of the code in the 5.8 kernel was written in the last seven years, but traces exist from all the prior years contribute to the code in version 5.8 even today.



Kernel v5.8 codebase broken down by year of code contribution

Year	# of Tokens	Proportion	Accumulated
1991	2,964	0.00%	0.00%
1992	30,740	0.03%	0.03%
1993	46,910	0.04%	0.07%
1994	38,443	0.03%	0.11%
1995	95,498	0.09%	0.19%
1996	204,573	0.19%	0.38%
1997	370,193	0.34%	0.72%
1998	436,941	0.40%	1.11%
1999	775,706	0.70%	1.81%
2000	1,465,450	1.33%	3.15%
2001	828,530	0.75%	3.90%
2002	2,475,002	2.24%	6.14%
2003	1,310,598	1.19%	7.33%
2004	2,023,705	1.83%	9.16%
2005	2,575,195	2.33%	11.49%
2006	2,445,966	2.22%	13.71%
2007	3,118,829	2.83%	16.54%
2008	4,410,921	4.00%	20.54%
2009	5,461,240	4.95%	25.49%
2010	4,805,525	4.35%	29.84%
2011	5,859,906	5.31%	35.15%
2012	6,072,494	5.50%	40.65%
2013	6,440,436	5.84%	46.49%
2014	6,091,508	5.52%	52.01%
2015	8,968,298	8.13%	60.14%
2016	7,622,786	6.91%	67.04%
2017	11,028,463	9.99%	77.04%
2018	9,248,269	8.38%	85.42%
2019	10,843,056	9.83%	95.24%
2020*	5,248,662	4.76%	100.00%

Table 1. Year of Origin of Tokens in Linux v5.8

*as of 8/2/2020

The Linux Foundation

<https://www.linuxfoundation.jp/blog/2020/08/download-the-2020-linux-kernel-history-report/>
https://www.linuxfoundation.jp/wp-content/uploads/2020/08/2020_kernel_history_report_082720.pdf



2020 Linux Kernel History Report



What's Linux kernel ?

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

5.8.10

Download icon

mainline:	5.9-rc5	2020-09-13	[tarball]	[patch]	[inc. patch]	[view diff]	[browse]		
stable:	5.8.10	2020-09-17	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
stable:	5.7.19 [EOL]	2020-08-27	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	5.4.66	2020-09-17	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	4.19.146	2020-09-17	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	4.14.198	2020-09-12	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	4.9.236	2020-09-12	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	4.4.236	2020-09-12	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
linux-next:	next-20200918	2020-09-18					[browse]		

Other resources

[Cgit](#)
[Bugzilla](#)
[Mirrors](#)

[Documentation](#)
[Patchwork](#)
[Linux.com](#)

[Wikis](#)
[Kernel Mailing Lists](#)
[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.....

<https://www.kernel.org/>
<https://www.kernel.org/category/releases.html>



What's Linux kernel ?

Linuxカーネル



index : kernel/git/torvalds/linux.git

Linux kernel source tree

master switch

Linus Torvalds

about summary refs log **tree** commit diff stats

log msg search

Mode	Name	Size			
-rw-r--r--	.clang-format	16544	log	stats	plain
-rw-r--r--	.cocciconfig	59	log	stats	plain
-rw-r--r--	.get_maintainer.ignore	71	log	stats	plain
-rw-r--r--	.gitattributes	62	log	stats	plain
-rw-r--r--	.gitignore	1852	log	stats	plain
-rw-r--r--	.mailmap	17556	log	stats	plain
-rw-r--r--	COPYING	496	log	stats	plain
-rw-r--r--	CREDITS	99788	log	stats	plain
d-----	Documentation	3324	log	stats	plain
-rw-r--r--	Kbuild	1327	log	stats	plain
-rw-r--r--	Kconfig	555	log	stats	plain
d-----	LICENSES	141	log	stats	plain
-rw-r--r--	MAINTAINERS	570461	log	stats	plain
-rw-r--r--	Makefile	63210	log	stats	plain
-rw-r--r--	README	727	log	stats	plain
d-----	arch	836	log	stats	plain
d-----	block	2787	log	stats	plain
d-----	certs	374	log	stats	plain
d-----	crypto	5623	log	stats	plain
d-----	drivers	4494	log	stats	plain

<https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/tree/?h=v5.9-rc6>



What's Linux kernel ?

ソースコードディレクトリから見るLinuxカーネルの持つ役割

```
[root@centos8test src]# ls
-rw-r--r--. 1 root root 108M  7月 22 16:44 linux-5.7.10.tar.xz
[root@centos8test src]# cd linux-5.7.10
[root@centos8test linux-5.7.10]# ls
COPYING      Kbuild  MAINTAINERS  arch  crypto  include  kernel  net  security  usr
CREDITS      Kconfig Makefile  block  drivers  init  lib  samples  sound  virt
Documentation LICENSES  README    certs  fs      ipc  mm  scripts  tools
```

```
.
linux-5.7.10/kernel/dma/
linux-5.7.10/kernel/dma/Kconfig
linux-5.7.10/kernel/dma/Makefile
.
```

```
.
linux-5.7.10/arch/riscv/
linux-5.7.10/arch/riscv/Kbuild
linux-5.7.10/arch/riscv/Kconfig
linux-5.7.10/arch/riscv/Kconfig.debug
linux-5.7.10/arch/riscv/Kconfig.socs
linux-5.7.10/arch/riscv/Makefile
.
```

```
.
linux-5.7.10/fs/ext4/
linux-5.7.10/fs/ext4/Kconfig
linux-5.7.10/fs/ext4/Makefile
.
```

```
.
linux-5.7.10/drivers/gpu/drm/radeon/r600.c
linux-5.7.10/drivers/gpu/drm/radeon/r600_blit_shaders.c
linux-5.7.10/drivers/gpu/drm/radeon/r600_blit_shaders.h
linux-5.7.10/drivers/gpu/drm/radeon/r600_cs.c
linux-5.7.10/drivers/gpu/drm/radeon/r600_dma.c
linux-5.7.10/drivers/gpu/drm/radeon/r600_dpm.c
linux-5.7.10/drivers/gpu/drm/radeon/r600_dpm.h
linux-5.7.10/drivers/gpu/drm/radeon/r600_hdmi.c
linux-5.7.10/drivers/gpu/drm/radeon/r600_reg.h
linux-5.7.10/drivers/gpu/drm/radeon/r600d.h
.
```

security (セキュリティ)
virt (仮想化関連)
mm (メモリ管理)
lib (ライブラリ)
crypto (暗号化)
sound (サウンド)
Documentation
(ドキュメント)
など



What's Linux kernel ?

unameコマンド (Linuxカーネルバージョンの確認)

```
[root@centos8test ~]# uname -r  
5.7.10
```

```
[root@centos8test ~]# uname -a  
Linux centos8test 5.7.10 #2 SMP Thu Jul 23 23:00:48 JST 2020 x86_64 x86_64 x86_64 GNU/Linux
```

```
[root@centos8test boot]# ls -l /boot  
合計 210472
```

```
lrwxrwxrwx. 1 root root    23  7月 24 10:08 System.map -> /boot/System.map-5.7.10  
-rw-  
-----. 1 root root 3909996  5月  8 20:07 System.map-4.18.0-193.el8.x86_64  
-rw-r--r--. 1 root root 4600996  7月 24 10:08 System.map-5.7.10  
-rw-r--r--. 1 root root  187648  5月  8 20:07 config-4.18.0-193.el8.x86_64  
drwxr-xr-x. 3 root root   4096  7月 23 11:14 efi  
drwx-  
-----. 4 root root   4096  7月 23 11:17 grub2  
-rw-  
-----. 1 root root 54334239  7月 23 11:16 initramfs-0-rescue-ab3015baa6bb48818a10a264e966f18c.img  
-rw-  
-----. 1 root root 29419458  7月 23 11:18 initramfs-4.18.0-193.el8.x86_64.img  
-rw-  
-----. 1 root root 19588423  7月 23 11:24 initramfs-4.18.0-193.el8.x86_64kdump.img  
-rw-  
-----. 1 root root 77062114  7月 24 10:09 initramfs-5.7.10.img  
drwxr-xr-x. 3 root root   4096  7月 23 11:15 loader  
drwx-  
-----. 2 root root   16384  7月 23 11:13 lost+found  
lrwxrwxrwx. 1 root root    20  7月 24 10:08 vmlinuz -> /boot/vmlinuz-5.7.10  
-rwxr-xr-x. 1 root root 8913656  7月 23 11:16 vmlinuz-0-rescue-ab3015baa6bb48818a10a264e966f18c  
-rwxr-xr-x. 1 root root 8913656  5月  8 20:07 vmlinuz-4.18.0-193.el8.x86_64  
-rw-r--r--. 1 root root 8542464  7月 24 10:08 vmlinuz-5.7.10
```



What's Linux kernel ?

Linux = Linuxカーネル ?

Linux (狭義) =

Linuxカーネル

Linux (広義、ディストリビューション) =

シェル・
アプリケーションなど

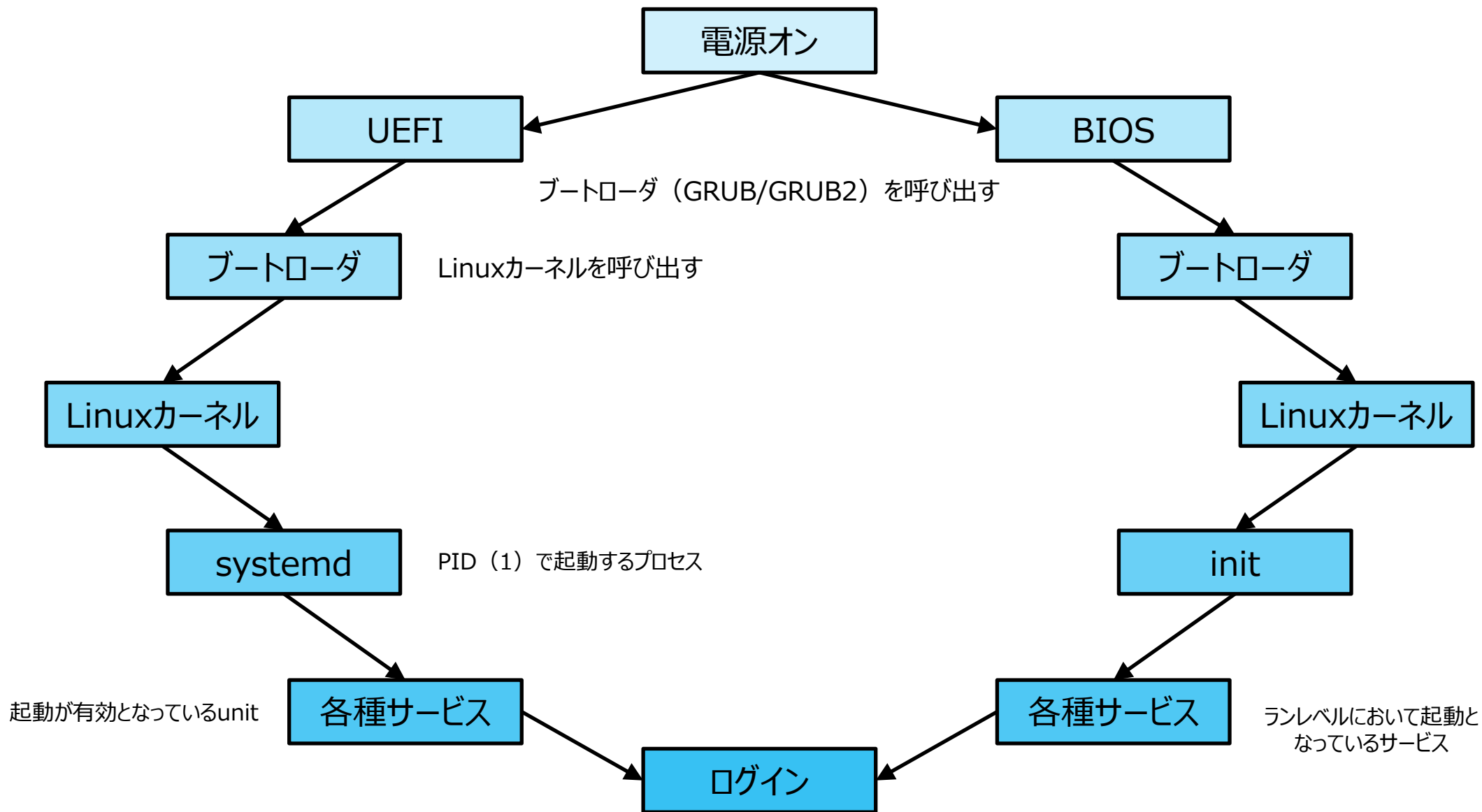
Linuxカーネル



Linuxの起動プロセス



Linuxの起動プロセス





UEFI (Unified Extensible Firmware Interface)
BIOS (Basic Input Output System)

UEFI/BIOSの役割

- ハードウェア初期化
- 自己診断処理
- メモリチェック
- ハードウェア設定読込
- 起動デバイスのチェック
- 起動デバイスにあるブートローダ (GRUB/GRUB2) を探し起動

UEFI/BIOSの違い

- BIOSはMBR (Master Boot Record) 、UEFIはGPT (GUID Partition Table) でパーティションを管理
- BIOSは2.2TB以上のディスクが非サポート、UEFIはそれ以上をサポート
- UEFIは、GUIなどでの操作可能



```
takahiro@ubuntu2004server:~$ sudo parted -l
[sudo] password for takahiro:
Model: ATA ST250DM000-1BD14 (scsi)
Disk /dev/sda: 250GB
Sector size (logical/physical): 512B/4096B
Partition Table: gpt
Disk Flags:
```

Number	Start	End	Size	File system	Name	Flags
1	1049kB	2097kB	1049kB		bios_grub	
2	2097kB	250GB	250GB	ext4		

```
takahiro@ubuntu2004server:~$ sudo gdisk -l /dev/sda
GPT fdisk (gdisk) version 1.0.5
```

Partition table scan:
MBR: protective
BSD: not present
APM: not present
GPT: present

Found valid GPT with protective MBR; using GPT.
Disk /dev/sda: 488397168 sectors, 232.9 GiB
Model: ST250DM000-1BD14
Sector size (logical/physical): 512/4096 bytes
Disk identifier (GUID): F5262966-F739-4E41-AEDD-D3F685D7A60A
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 488397134
Partitions will be aligned on 2048-sector boundaries
Total free space is 4397 sectors (2.1 MiB)

Number	Start (sector)	End (sector)	Size	Code	Name
1	2048	4095	1024.0 KiB	EF02	
2	4096	488394751	232.9 GiB	8300	



```
takahiro@ubuntu2004server:~$ sudo dmidecode -t 0
```

```
[sudo] password for takahiro:
```

```
# dmidecode 3.2
```

```
Getting SMBIOS data from sysfs.
```

```
SMBIOS 2.7 present.
```

```
Handle 0x0000, DMI type 0, 24 bytes
```

```
BIOS Information
```

```
Vendor: Dell Inc.
```

```
Version: A12
```

```
Release Date: 01/10/2013
```

```
Address: 0xF0000
```

```
Runtime Size: 64 kB
```

```
ROM Size: 12288 kB
```

```
Characteristics:
```

```
PCI is supported
```

```
PNP is supported
```

```
BIOS is upgradeable
```

```
BIOS shadowing is allowed
```

```
Boot from CD is supported
```

```
Selectable boot is supported
```

```
EDD is supported
```

```
5.25"/1.2 MB floppy services are supported (int 13h)
```

```
3.5"/720 kB floppy services are supported (int 13h)
```

```
3.5"/2.88 MB floppy services are supported (int 13h)
```

```
Print screen service is supported (int 5h)
```

```
8042 keyboard services are supported (int 9h)
```

```
Serial services are supported (int 14h)
```

```
Printer services are supported (int 17h)
```

```
ACPI is supported
```

```
USB legacy is supported
```

```
BIOS boot specification is supported
```

```
Function key-initiated network boot is supported
```

```
Targeted content distribution is supported
```

```
UEFI is supported
```

UEFIが使われている

```
BIOS Revision: 4.6
```




systemd/init

- PIDが1（一番最初に起動するプロセス）
- 各種サービスを起動する
 - systemdの場合 = 各unitの設定
 - initの場合 = 各ランレベルの設定

```
[root@centos8test ~]# pstree -p
systemd(1)─NetworkManager(791)─[NetworkManager](794)
                                └─[NetworkManager](795)
    ─agetty(806)
    ─auditd(736)─sedispatch(738)
                └─[auditd](737)
                  └─[auditd](739)
    ─crond(805)
    ─dbus-daemon(763)─[dbus-daemon](770)
    ─firewalld(780)─[firewalld](1007)
    ─nginx(1018)─nginx(1019)
    ─polkitd(761)─[polkitd](772)
                  └─[polkitd](773)
```

```
[root@db01 ~]# systemctl --all
UNIT                                LOAD    ACTIVE SUB    JOB    DESCRIPTION
proc-sys-fs-binfmt_misc.automount loaded  active waiting Arbitrary
Executable File Form
dev-block-8:2.device               loaded  active plugged LVM PV OgMPDL-
wIE0-OxZd-bJos-q
dev-cdrom.device                   loaded  active plugged VBOX_CD-ROM
dev-centos-root.device              loaded  active plugged /dev/centos/root
dev-centos-swap.device              loaded  active plugged /dev/centos/swap
```

```
[root@db01 ~]# systemctl list-unit-files --type service
```

UNIT FILE	STATE
arp-ethers.service	disabled
auditd.service	enabled
auth-rpcgss-module.service	static
autovt@.service	enabled
blk-availability.service	disabled
brandbot.service	static
chrony-dnssrv@.service	static

```
[root@CentOS6 ~]# chkconfig --list | grep -E "(httpd|mysqld)"
httpd      0:off  1:off  2:on   3:on   4:on   5:on   6:off
mysqld     0:off  1:off  2:on   3:on   4:on   5:on   6:off
```



dmesgコマンド（カーネル起動中のメッセージ確認）

```
[root@centos8test ~]# dmesg
[ 0.000000] Linux version 5.7.10 (root@centos8test) (gcc version 8.3.1 20191121 (Red Hat 8.3.1-5) (GCC), GNU ld version 2.30-73.el8) #2 SMP Thu Jul 23
23:00:48 JST 2020
[ 0.000000] Command line: BOOT_IMAGE=(hd0,msdos1)/vmlinuz-5.7.10 root=/dev/mapper/cl-root ro crashkernel=auto resume=/dev/mapper/cl-swap
rd.lvm.lv=cl/root rd.lvm.lv=cl/swap
.
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009fbff] usable
.
[ 0.073034] Kernel command line: BOOT_IMAGE=(hd0,msdos1)/vmlinuz-5.7.10 root=/dev/mapper/cl-root ro crashkernel=auto resume=/dev/mapper/cl-swap
rd.lvm.lv=cl/root rd.lvm.lv=cl/swap
.
[ 0.076940] Memory: 261120K/2096696K available (12292K kernel code, 2138K rwd data, 4456K rodata, 2336K init, 6156K bss, 147040K reserved, 0K cma-
reserved)
.
[ 0.356231] TCP: Hash tables configured (established 16384 bind 16384)
[ 0.356231] UDP hash table entries: 1024 (order: 3, 32768 bytes, linear)
.
[ 1.553830] Serial: 8250/16550 driver, 4 ports, IRQ sharing enabled
.
[ 1.554196] ehci_hcd: USB 2.0 'Enhanced' Host Controller (EHCI) Driver
.
[ 1.624187] rtc_cmos rtc_cmos: registered as rtc0
.
[ 1.845319] systemd[1]: Running in initial RAM disk.
[ 1.846688] systemd[1]: Set hostname to <centos8test>.
.
[ 3.143944] e1000: Intel(R) PRO/1000 Network Driver - version 7.3.21-k8-NAPI
[ 3.144184] e1000: Copyright (c) 1999-2006 Intel Corporation.
.
[ 5.533112] audit: type=1404 audit(1600700457.406:2): enforcing=1 old_enforcing=0 auid=4294967295 ses=4294967295 enabled=1 old-enabled=1
lsm=selinux res=1
[ 5.700566] SELinux: Permission watch in class filesystem not defined in policy.
```



Linux kernelの再構築



- ①ソースコードのダウンロード
- ②ダウンロードファイルの解凍
- ③カーネル設定(.config)の生成
- ④カーネルに組み込む機能のコンパイル
- ⑤動的モジュールとして組み込む機能のコンパイル
- ⑥動的モジュールのインストール
- ⑦カーネルのインストール
- ⑧再起動

Opensourcetechnologyブログ

OpensourcetechnologyによるNGINX/Zabbix/Neo4j/Linuxなどオープンソース技術に関するブログです。

2020-07-24

CentOS8のカーネル再構築(Linux kernel 5.7.10)

Linux

kernel

カーネル

CentOS

CentOS8

make

make config

make menuconfig

make bzImage

make modules

make modules_install

make install

.config

リンク

当サイトのプライバシーポリシーについて

LinuxCエヴァンジェリストの鯨井貴博@opensource-techです。

Linux Kernelの再構築方法の紹介です。
CentOS8のデフォルトカーネル(4.18.0-193.el8.x86_64)を新しいカーネルにしてみます。
<https://www.kernel.org/>

The Linux Kernel Archives

AboutContact usFAQReleasesSignaturesSite news

Protocol	Location
HTTP	https://www.kernel.org/pub/
GIT	https://git.kernel.org/
RSYNC	rsync://rsync.kernel.org/pub/

Latest Release
5.7.10

mainline:	5.8-rc6	2020-07-19	[tarball]	[patch]	[inc. patch]	[view diff]	[browse]
stable:	5.7.10	2020-07-22	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff] [browse] [changelog]
longterm:	5.4.53	2020-07-22	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff] [browse] [changelog]
longterm:	4.19.134	2020-07-22	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff] [browse] [changelog]
longterm:	4.14.189	2020-07-22	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff] [browse] [changelog]
longterm:	4.9.231	2020-07-22	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff] [browse] [changelog]
longterm:	4.4.231	2020-07-22	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff] [browse] [changelog]
linux-next:	next-20200722	2020-07-22					[browse]

CentOS8のカーネル再構築(Linux kernel 5.7.10)
<https://www.opensourcetechnology.tokyo/?page=1596349636>





ソースコードのダウンロード(wgetなど)
解凍(tar/xzなど)

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
5.8.10 

mainline:	5.9-rc5	2020-09-13	[tarball]	[patch]	[inc. patch]	[view diff]	[browse]		
stable:	5.8.10	2020-09-17	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
stable:	5.7.19 [EOL]	2020-08-27	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	5.4.66	2020-09-17	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	4.19.146	2020-09-17	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	4.14.198	2020-09-12	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	4.9.236	2020-09-12	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	4.4.236	2020-09-12	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
linux-next:	next-20200918	2020-09-18						[browse]	

Other resources

Cgit	Documentation	Wikis
Bugzilla	Patchwork	Kernel Mailing Lists
Mirrors	Linux.com	Linux Foundation

Social

[Site Atom feed](#)
[Releases Atom Feed](#)
[Kernel Planet](#)

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



カーネル設定(.config)の生成

[root@centos8test linux-5.7.10]# **make help** ※一部ターゲットを抜粋

Configuration targets:

- config** - Update current config utilising a line-oriented program ...テキスト形式でパラメータを選択
- nconfig** - Update current config utilising a ncurses menu based program
- menuconfig** - Update current config utilising a menu based program...メニュー形式でパラメータを選択
- xconfig** - Update current config utilising a Qt based front-end
- gconfig** - Update current config utilising a GTK+ based front-end
- oldconfig** - Update current config utilising a provided .config as base...既存設定を使用
- localmodconfig** - Update current config disabling modules not loaded
- localyesconfig** - Update current config converting local mods to core
- defconfig** - New config with default from ARCH supplied defconfig
- savedefconfig** - Save current config as ./defconfig (minimal config)
- allnoconfig** - New config where all options are answered with no
- allyesconfig** - New config where all options are accepted with yes
- allmodconfig** - New config selecting modules when possible
- alldefconfig** - New config with all symbols set to default
- randconfig** - New config with random answer to all options
- yes2modconfig** - Change answers from yes to mod if possible
- mod2yesconfig** - Change answers from mod to yes if possible
- listnewconfig** - List new options
- helpnewconfig** - List new options and help text
- olddefconfig** - Same as oldconfig but sets new symbols to their...既存設定を使用 + 追加されているパラメータを選択
default value without prompting

各コマンド実行時、
不足するパッケージがあれば追加しましょう。



make menuconfig (メニュー形式)

```
.config - Linux/x86 5.7.10 Kernel Configuration

Linux/x86 5.7.10 Kernel Configuration
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----).
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes
features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ]
excluded <M> module < > module capable

*** Compiler: gcc (GCC) 8.3.1 20191121 (Red Hat 8.3.1-5) ***
General setup --->
[*] 64-bit kernel
Processor type and features --->
Power management and ACPI options --->
Bus options (PCI etc.) --->
Binary Emulations --->
Firmware Drivers --->
[*] Virtualization --->
General architecture-dependent options --->
[*] Enable loadable module support --->
-* Enable the block layer --->
IO Schedulers --->
Executable file formats --->
Memory Management options --->
[*] Networking support --->
Device Drivers --->
File systems --->
Security options --->
-* Cryptographic API --->
Library routines --->

1(+)
```



make config (テキスト形式)

```
[root@centos8test linux-5.7.10]# make config
HOSTCC scripts/kconfig/conf.o
HOSTLD scripts/kconfig/conf
scripts/kconfig/conf --oldaskconfig Kconfig
#
# using defaults found in /boot/config-4.18.0-193.el8.x86_64
#
/boot/config-4.18.0-193.el8.x86_64:1089:warning: symbol value 'm' invalid for NF_CT_PROTO_GRE
/boot/config-4.18.0-193.el8.x86_64:2949:warning: symbol value 'm' invalid for ISDN_CAPI
*
* Linux/x86 5.7.10 Kernel Configuration
*
*
* Compiler: gcc (GCC) 8.3.1 20191121 (Red Hat 8.3.1-5)
*
*
* General setup
*
Compile also drivers which will not load (COMPILE_TEST) [N/y/?]
```



make olddefconfig (既存設定を踏襲 + 追加された設定はデフォルト値)

```
[root@centos8test linux-5.7.10]# make olddefconfig
scripts/kconfig/conf --olddefconfig Kconfig
#
# using defaults found in /boot/config-4.18.0-193.el8.x86_64
#
/boot/config-4.18.0-193.el8.x86_64:1089:warning: symbol value 'm' invalid for NF_CT_PROTO_GRE
/boot/config-4.18.0-193.el8.x86_64:2949:warning: symbol value 'm' invalid for ISDN_CAPI
#
# configuration written to .config
#
```



make listnewconfig (追加されている設定をリストアップ)

```
[root@centos8test linux-5.7.10]# make listnewconfig
scripts/kconfig/conf --listnewconfig Kconfig
#
# using defaults found in /boot/config-4.18.0-193.el8.x86_64
#
/boot/config-4.18.0-193.el8.x86_64:1089:warning: symbol value 'm' invalid for NF_CT_PROTO_GRE
/boot/config-4.18.0-193.el8.x86_64:2949:warning: symbol value 'm' invalid for ISDN_CAPI
CONFIG_BUILD_SALT=""
CONFIG_SCHED_THERMAL_PRESSURE=n
CONFIG_IKHEADERS=n
CONFIG_UCLAMP_TASK=n
CONFIG_TIME_NS=y
CONFIG_BOOT_CONFIG=n
CONFIG_BPF_LSM=n
.
```



.config (設定ファイル) の内容

```
[root@centos8test linux-5.7.10]# cat -n .config
```

```
1 #
2 # Automatically generated file; DO NOT EDIT.
3 # Linux/x86 5.7.10 Kernel Configuration
4 #
5
6 #
7 # Compiler: gcc (GCC) 8.3.1 20191121 (Red Hat 8.3.1-5)
8 #
9 CONFIG_CC_IS_GCC=y ← カーネルに組み込む
10 CONFIG_GCC_VERSION=80301
11 CONFIG_LD_VERSION=230000000
12 CONFIG_CLANG_VERSION=0
13 CONFIG_CC_CAN_LINK=y
14 CONFIG_CC_HAS_ASM_GOTO=y
```

・
省略

```
8173 CONFIG_TEST_BPF=m ← 動的モジュールとして利用する
8174 # CONFIG_TEST_BLACKHOLE_DEV is not set
```



make bzImage (カーネルに組み込む機能のコンパイル)

※.configで「パラメータ名=y」となっている項目

```
[root@centos8test linux-5.7.10]# make bzImage
SYSTBL arch/x86/include/generated/asm/syscalls_32.h
SYSHDR arch/x86/include/generated/asm/unistd_32_ia32.h
SYSHDR arch/x86/include/generated/asm/unistd_64_x32.h
SYSTBL arch/x86/include/generated/asm/syscalls_64.h
HYPERCALLS arch/x86/include/generated/asm/xen-hypercalls.h
SYSHDR arch/x86/include/generated/uapi/asm/unistd_32.h
SYSHDR arch/x86/include/generated/uapi/asm/unistd_64.h
SYSHDR arch/x86/include/generated/uapi/asm/unistd_x32.h
HOSTCC arch/x86/tools/relocs_32.o
.
.
CC arch/x86/boot/video-vesa.o
CC arch/x86/boot/video-bios.o
LD arch/x86/boot/setup.elf
OBJCOPY arch/x86/boot/setup.bin
OBJCOPY arch/x86/boot/vmlinux.bin
HOSTCC arch/x86/boot/tools/build
BUILD arch/x86/boot/bzImage
Setup is 15068 bytes (padded to 15360 bytes).
System is 8328 kB
CRC da731b84
Kernel: arch/x86/boot/bzImage is ready (#2)
```




make modules (動的モジュールとして利用する機能のコンパイル)

※.configで「パラメータ名=m」となっている項目

```
[root@centos8test linux-5.7.10]# make modules
```

```
CALL scripts/checksyscalls.sh
```

```
CALL scripts/atomic/check-atomics.sh
```

```
DESCEND objtool
```

```
CHK include/generated/compile.h
```

```
CC [M] arch/x86/events/amd/power.o
```

```
CC [M] arch/x86/events/intel/uncore.o
```

```
CC [M] arch/x86/events/intel/uncore_nhmex.o
```

```
CC [M] arch/x86/events/intel/uncore_snb.o
```

```
CC [M] arch/x86/events/intel/uncore_snbep.o
```

```
LD [M] arch/x86/events/intel/intel-uncore.o
```

```
CC [M] arch/x86/events/intel/cstate.o
```

```
LD [M] arch/x86/events/intel/intel-cstate.o
```

```
CC [M] arch/x86/events/rapl.o
```

```
CC [M] arch/x86/kernel/cpu/mce/inject.o
```

```
LD [M] arch/x86/kernel/cpu/mce/mce-inject.o
```

```
AS [M] arch/x86/crypto/twofish-x86_64-asm_64.o
```

```
CC [M] arch/x86/crypto/twofish_glue.o
```

```
LD [M] arch/x86/crypto/twofish-x86_64.o
```

```
.
```

```
.
```



make modules_install (make modulesでコンパイルした動的モジュールのインストール)

```
[root@centos8test linux-5.7.10]# make modules_install
INSTALL arch/x86/crypto/blowfish-x86_64.ko
INSTALL arch/x86/crypto/camellia-aesni-avx-x86_64.ko
INSTALL arch/x86/crypto/camellia-aesni-avx2.ko
INSTALL arch/x86/crypto/camellia-x86_64.ko
INSTALL arch/x86/crypto/cast5-avx-x86_64.ko
INSTALL arch/x86/crypto/cast6-avx-x86_64.ko
INSTALL arch/x86/crypto/chacha-x86_64.ko
INSTALL arch/x86/crypto/crc32-pclmul.ko
INSTALL arch/x86/crypto/crc32c-intel.ko
INSTALL arch/x86/crypto/crct10dif-pclmul.ko
INSTALL arch/x86/crypto/des3_edc-x86_64.ko
INSTALL arch/x86/crypto/ghash-clmulni-intel.ko
INSTALL arch/x86/crypto/poly1305-x86_64.ko
INSTALL arch/x86/crypto/serpent-avx-x86_64.ko
INSTALL arch/x86/crypto/serpent-avx2.ko
INSTALL arch/x86/crypto/serpent-sse2-x86_64.ko
INSTALL arch/x86/crypto/sha512-ssse3.ko
.
```



make install (make bzImageでコンパイルしたもののインストール)

```
[root@centos8test linux-5.7.10]# make install
sh ./arch/x86/boot/install.sh 5.7.10 arch/x86/boot/bzImage ¥
  System.map "/boot"
```

```
[root@centos8test boot]# ls -l /boot
合計 210472
lrwxrwxrwx. 1 root root    23  7月 24 10:08 System.map -> /boot/System.map-5.7.10
-rw-----. 1 root root 3909996  5月  8 20:07 System.map-4.18.0-193.el8.x86_64
-rw-r--r--. 1 root root 4600996  7月 24 10:08 System.map-5.7.10
-rw-r--r--. 1 root root  187648  5月  8 20:07 config-4.18.0-193.el8.x86_64
drwxr-xr-x. 3 root root   4096  7月 23 11:14 efi
drwx-----, 4 root root   4096  7月 23 11:17 grub2
-rw-----. 1 root root 54334239  7月 23 11:16 initramfs-0-rescue-ab3015baa6bb48818a10a264e966f18c.img
-rw-----. 1 root root 29419458  7月 23 11:18 initramfs-4.18.0-193.el8.x86_64.img
-rw-----. 1 root root 19588423  7月 23 11:24 initramfs-4.18.0-193.el8.x86_64kdump.img
-rw-----. 1 root root 77062114  7月 24 10:09 initramfs-5.7.10.img
drwxr-xr-x. 3 root root   4096  7月 23 11:15 loader
drwx-----, 2 root root  16384  7月 23 11:13 lost+found
lrwxrwxrwx. 1 root root    20  7月 24 10:08 vmlinuz -> /boot/vmlinuz-5.7.10
-rwxr-xr-x. 1 root root 8913656  7月 23 11:16 vmlinuz-0-rescue-ab3015baa6bb48818a10a264e966f18c
-rwxr-xr-x. 1 root root 8913656  5月  8 20:07 vmlinuz-4.18.0-193.el8.x86_64
-rw-r--r--. 1 root root 8542464  7月 24 10:08 vmlinuz-5.7.10
```



再起動（reboot）後、GRUBの起動メニューに構築したカーネルが選択されるようになっている。

```
CentOS Linux (5.7.10) 8 (Core)
CentOS Linux (4.18.0-193.el8.x86_64) 8 (Core)
CentOS Linux (0-rescue-ab3015baa6bb48818a10a264e966f18c) 8 (Core)

Use the ↑ and ↓ keys to change the selection.
Press 'e' to edit the selected item, or 'c' for a command prompt.
```

“e”をタイプすると、
起動メニューの編集が可能



起動時に使用されるパラメータを変更して、「Ctrl+x」でそのパラメータで起動

```
load_video
set gfx_payload=keep
insmod gzio
linux ($root)/vmlinuz-5.7.10 root=/dev/mapper/cl-root ro crashkernel=auto resume=/dev/mapper/cl-swap rd.lvm.lv=cl/root rd.lvm.lv=cl/swap
initrd ($root)/initramfs-5.7.10.img $tuned_initrd
```

Press Ctrl-x to start, Ctrl-c for a command prompt or Escape to discard edits and return to the menu. Pressing Tab lists possible completions.

```
load_video
set gfx_payload=keep
insmod gzio
linux ($root)/vmlinuz-5.7.10 root=/dev/mapper/cl-root rw init=/bin/sh crashkernel=auto resume=/dev/mapper/cl-swap rd.lvm.lv=cl/root rd.lvm.lv=cl/swap
initrd ($root)/initramfs-5.7.10.img $tuned_initrd
```

例：起動出来ないトラブルなどの際、
シングルユーザモードにする

CentOS7をシングルユーザモードで起動する方法

<https://www.opensourcetech.tokyo/entry/20200308/1583595134>



grub2-mkconfig

全体設定 (/etc/default/grub) とカスタム設定 (/etc/grub.d/配下の設定ファイル) からGRUB設定 (grub.cfg) を生成するコマンド
標準出力に出力されるが、-oオプションでファイル出力可能

```
[root@centos8test ~]# ls -l /etc/default/grub
-rw-r--r--. 1 root root 311  7月 23 11:17 /etc/default/grub
[root@centos8test ~]# ls -l /etc/grub.d/
合計 84
-rwxr-xr-x. 1 root root 8958  4月 14 23:53 00_header
-rwxr-xr-x. 1 root root 1043 12月 12 2019 00_tuned
-rwxr-xr-x. 1 root root 1240  4月 14 23:53 01_menu_auto_hide
-rwxr-xr-x. 1 root root  232  4月 14 23:53 01_users
-rwxr-xr-x. 1 root root 13434  4月 14 23:53 10_linux
-rwxr-xr-x. 1 root root 11696  4月 14 23:53 20_linux_xen
-rwxr-xr-x. 1 root root 2559  4月 14 23:53 20_ppc_terminfo
-rwxr-xr-x. 1 root root 10670  4月 14 23:53 30_os-prober
-rwxr-xr-x. 1 root root 1412  4月 14 23:53 30_uefi-firmware
-rwxr-xr-x. 1 root root  214  4月 14 23:53 40_custom
-rwxr-xr-x. 1 root root  216  4月 14 23:53 41_custom
-rw-r--r--. 1 root root  483  4月 14 23:53 README
[root@centos8test ~]# grub2-mkconfig -o /var/tmp/grub.cfg
Generating grub configuration file ...
done
[root@centos8test ~]# ls -l /var/tmp/
合計 44
-rw-r--r--. 1 root root 36617  9月 22 00:08 dmesg.log
-rw-r--r--. 1 root root  5121  9月 26 23:20 grub.cfg
```




Appendix



Linuxカーネルに関する用語集

用語	説明
DKMS	Dynamic Kernel Module Support カーネルのソースツリーの外にソースが存在する Linux カーネルモジュールの生成を可能にするプログラム/フレームワーク https://wiki.archlinux.jp/index.php/Dynamic_Kernel_Module_Support
Dracut	initramfsを作成などするツール https://access.redhat.com/documentation/ja-jp/red_hat_enterprise_linux/6/html/deployment_guide/sec-verifying_the_initial_ram_disk_image
initramfs	Linuxカーネルが起動時に使用する仮想ルートファイルシステム (initial ram filesystem) https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/tree/Documentation/filesystems/ramfs-rootfs-initramfs.rst?h=v5.9-rc7
initrd	initramfsが使われる前から利用されていた仮想ルートファイルシステム (initial ram disk) https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/tree/Documentation/filesystems/ramfs-rootfs-initramfs.rst?h=v5.9-rc7
vmlinuz	圧縮されたbzImageフォーマットのLinuxカーネルイメージ (vmlinux)
圧縮・アーカイブ形式	Linuxカーネル関連では、xz/zip/bzip2/cpio/tarなどが使用されている



Dracut (initramfsを作成する)

```
[root@centos8test tmp]# dracut "initramfs-$(uname -r).img" $(uname -r)
Warning no default label for /var/tmp/initramfs-5.7.10.img
[root@centos8test tmp]# ls -l /var/tmp/initramfs-5.7.10.img
-rw-----. 1 root root 77062113  9月 27 00:29 /var/tmp/initramfs-5.7.10.img
[root@centos8test tmp]# lsinitrd /var/tmp/initramfs-5.7.10.img
Image: /var/tmp/initramfs-5.7.10.img: 74M
=====
Version: dracut-049-70.git20200228.el8
```

Arguments:

dracut modules:

- bash
- systemd
- systemd-initrd
- nss-softokn
- rngd
- i18n
- network-legacy
- network
- ifcfg
- prefixdevname
- dm
- kernel-modules
- kernel-modules-extra
- kernel-network-modules



systemd (systemctlコマンド)

コマンド	説明
systemctl start nginx	サービスの起動
systemctl stop nginx	サービスの停止
systemctl restart nginx	サービスの再起動
systemctl reload nginx	サービス設定の再読み込み
systemctl enable nginx	サービス自動起動設定の有効化
systemctl disable nginx	サービス自動起動設定の無効化
systemctl is-enabled nginx	サービス自動起動設定の確認
systemctl --all	全情報を表示
systemctl list-unit-files --type service	インストール済みのユニットファイル表示

<https://www.opensourcetechnologytokyo.com/entry/20200717/1594960896>

https://access.redhat.com/documentation/ja-jp/red_hat_enterprise_linux/7/html/system_administrators_guide/sect-managing_services_with_systemd-services

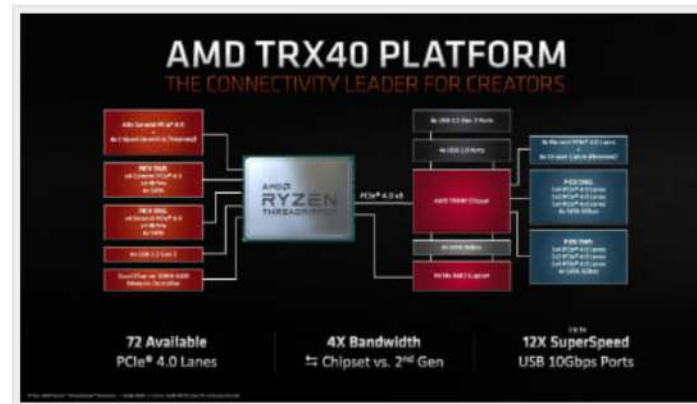


Linus Torvalds氏、メインマシンにRyzen Threadripper 3970Xを導入

～15年ぶりの非インテル

劉 堯 2020年5月27日 15:04

ツイート リスト B! 54 Pocket 13 いいね! 77 シェア



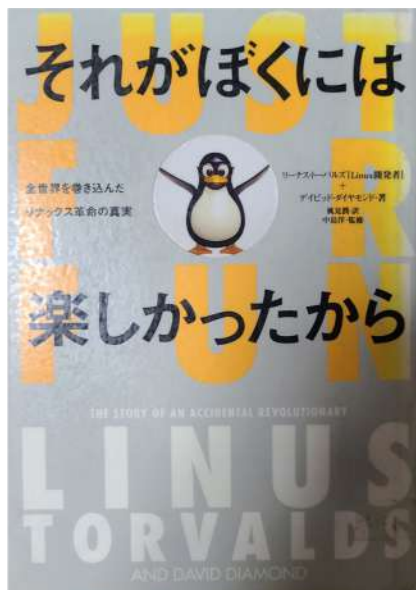
Linuxの生みの親であるLinus Torvalds氏は24日、Linuxカーネル開発などに関する非公式メーリングリスト「LKML.ORG」において、自身のPCのCPUとしてRyzen Threadripper 3970Xを導入したことを明らかにした。

メールの内容自体はLinux 5.7-rc7についてのものだが、後段で「私にとって今週の最大の興奮はメインマシンをアップグレードしたことです。15年ぶりにIntelベースではない自分のデスクトップとなりました。まだArmに切り替えたわけではないが、AMDの(Ryzen)Threadripper 3970Xとなりました。(これにより)「allmodconfig」テストビルドは、これまで使っていたものの3倍の速度となりました」と報告している。

<https://pc.watch.impress.co.jp/docs/news/yajiuma/1255074.html>
<https://lkml.org/lkml/2020/5/24/407>



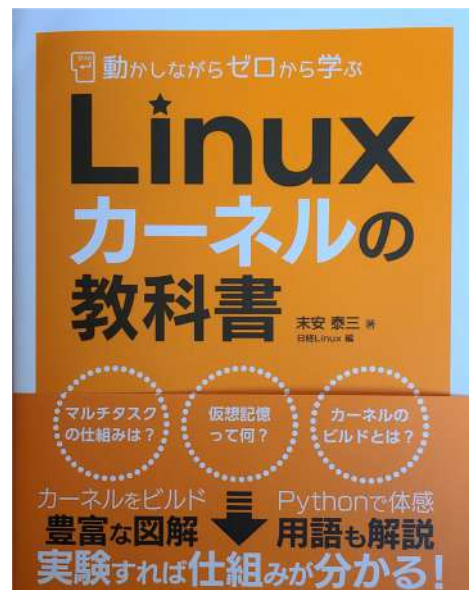
Linuxカーネルを知るための書籍・イベント



それがぼくには楽しかったから
全世界を巻き込んだリナックス革命の真実



改訂3版 Linuxエンジニア養成読本
(Software Design plus)



動かしながらゼロから学ぶ
Linuxカーネルの教科書



Open Source Summit Japan
<https://events.linuxfoundation.org/open-source-summit-japan/>



本日のまとめ



- Linuxカーネルとは何か
 - Linuxカーネルの歴史
 - Linuxカーネルの機能/役割
- Linuxの起動プロセス
- Linuxカーネルの再構築



Just for fun !
実機を操作して
Linuxを学んでね♪



Q & A



Thank you for join today's seminar!



<https://www.OpensourceTech.tokyo/>

https://twitter.com/matt_zeus

<https://www.facebook.com/takahiro.kujirai.1>