Engineer

オペレーティングシステム

ほかのプログラムのためにハードウェアを管理するソフトウェア

どのようなものか

オペレーティングシステムは、コンピューターで最初に起動し、ほかのすべての下で動き続けるソフトウェアです。プロセッサー、メモリー、ストレージ、ネットワークをプログラム間で分け、ファイルの読み取り、装置の利用、ネットワークへの接続に、共通の方法を与えます。通常OSと略します。Windows、macOS、Android、iOS、DebianやUbuntuなどのLinuxディストリビューションは、OSです。

NISTの用語集には、組み合わせて理解できる二つの説明があります。一つは、コンピューターを動かす主制御アプリケーションとし、「コンピューターの電源を入れたとき、最初に読み込まれるプログラム」と呼びます。主な構成要素のカーネルは常にメモリー内にあり、ウェブサーバーなど、「すべてのアプリケーションプログラムの標準を定める」と説明します。もう一つは、運用技術のセキュリティーに関するNISTのガイドから取り、「プログラムの実行順序を監督する、サービスルーチンの統合された集合」と呼びます。入出力制御、資源のスケジューリング、データ管理を行う場合があり、アプリケーションに、コンピューター制御の基本コマンドを提供します。

カーネルは中核ですが、OSはカーネルだけではありません。周りには、プログラムが呼ぶシステムライブラリー、シェルと日常のコマンドラインツール、起動時にサービスを立ち上げて再起動も行う管理機構、ソフトウェアをインストール、更新するパッケージマネージャーがあります。有効にする機能、ログの行き先、ログインを許す相手を決める、大量の既定設定も含みます。Linuxディストリビューションは、それをLinuxカーネルの周りに組み立てます。そのため、カーネルが同じでも、動作は大きく違うことがあります。

OSがプログラムに提供するものの多くは、ハードウェアの代わりとなる、少数の抽象概念です。Unix系システムの基になる標準POSIXは、それを厳密に定義します。プロセスは、一つ以上のスレッドが動くアドレス空間と、そのスレッドに

必要なシステム資源を合わせたものです。ファイルは、書き込み、読み取り、または両方ができるオブジェクトで、アクセス権と種類を持ちます。通常ファイル、ディレクトリー、シンボリックリンク、FIFO、ソケット、装置を表すキャラクター特殊ファイルかブロック特殊ファイルです。ブロック特殊ファイルは、ハードウェアの特性を意識させない形で、装置へのアクセスを提供します。装置自身の定義は、「コンピューターの周辺機器、またはアプリケーションからそのように見えるオブジェクト」です。POSIXが掲げる目的は、ソースコードの段階でアプリケーションを移植可能にすることです。このインターフェースに合わせたプログラムは、別の適合システムでも、コンパイルして動かせます。

メモリーも、同じように扱います。Linuxカーネルのドキュメントは、物理メモリーが限られた資源であることから始め、仮想メモリーを、その詳細をアプリケーションソフトウェアから抽象化する層と説明します。各プログラムには、専用のアドレス空間が見えます。OSは、その時々に、どのプログラムのどの部分を実メモリーに置き、どこを圧縮し、ディスクへ書き出し、あるいは破棄して後から読み直すかを決めます。

概念を判断に生かす

「オペレーティングシステムとは？」を実際の判断に生かしましょう。その概念を、開発または購入するシステムに結び付けて考えるお手伝いをします。 サーバー、メール、アカウント、更新を日々管理し、誰もが頼る仕組みを平穏に動かし続けます。

サービスを見る お問い合わせ →

用語の背後にある前提を検証する

解説とリンク先の実績を参考に、プロジェクトについての質問を整理してください。ツールを選ぶ前に、前提を一緒に検討できます。 UbuntuでPostgreSQL・MS SQLサーバー10台を運用 — 10 導入し正常に保ったDBサーバー

UbuntuでPostgreSQL・MS SQLサーバー10台を運用 お問い合わせ →

なぜ重要なのか

OSを選ぶと、多くのことを一度に決めます。動かせるソフトウェア、運用者がすでに知っているツール、セキュリティー修正の提供者と期間、契約する場合のサポート費用です。後で変えるには、その上に作ったすべてのプログラムと習慣を移す必要があり、その選択は、最初に使ったハードウェアより長く残りがちです。

一台のマシン内では、主な信頼境界でもあります。ユーザー、グループ、ファイル権限、プロセス間の分離は、すべてOSの機能で、一つのプログラムの欠陥が、どこまで届くかを決めます。全サービスが完全な権限で動くマシンに残る壁は、ネットワークの境界だけです。

既定値は、ほかの誰かが決めたことです。新規インストールでも、有効なサービス、決まった場所のファイル、選択済みの設定があります。大半は妥当ですが、仕事に合わないものも少しあり、それがどれかは教えてもらえません。適切な運用には、どの既定値を受け入れたかを知ることも含まれます。

コードをどこまで移せるかも決めます。移植可能なインターフェースに従うプログラムは、再ビルドで別のシステムへ移れます。一つのシステムの拡張に頼るプログラムやスクリプトは、その拡張が使える場所にしか移れません。

実際の活用例

あるノートパソコンでは、2026-09-01に、36 GBのマシンで、macOSアプリの診断レポートに、111 GBの圧縮ページが記録されました。別のメモリーフットプリントのサンプルでは、アプリケーションのコアイベントを出す接続リトライのループ中に、毎秒約41 MBの増加が記録されました。別々の観測であり、一つの連続した測定ではありません。イベント経路を変更し、バッファーに上限を設け、イベント種別で購読を絞り、ログ出力のレートに許容枠を設けました。テスト一式の実行結果を比較すると、コアのログ行数は610,996から1,411へ減りました。テストは上限を設けた設計を守りますが、この比較は、本番のベンチマークでも、修正後の実測メモリー値でもありません。別に起きた終了時のタイムアウトには、独自の修正が必要でした。

自社のホストでは、数か月にわたり、自動化ロールが毎回SSHの設定を書いていましたが、アクセスのレポートが、デーモン自身の実効設定を読み、二つの不一致を見つけました。三段階で検証するSSHの強化は、原因を説明します。OSが、コメントアウトしない自前の既定値を、追加設定ファイルの読み込み位置より先に配置し、その二つの設定では、最初の指定が優先していました。ベンダーのファイルより先に並ぶ、ファイル名のプレフィックスで直し、現在は、デーモンが返す値に対して24のディレクティブを検査します。

同じホストでは、侵入者を遮断するサービスを設定した際にも、OS側の選択に出会いました。ログ取得元はdetectに設定します。OSが従来型の認証ログを用意しないためで、選択を誤ると、現れないファイルを見張り、静かに失敗します。

問題が起きるところ

作成元のシステムの習慣を、スクリプトが引きずります。 Macで書いたシェルスクリプトをLinuxサーバーで、またはその逆で動かすと、同じコマンド名でもオプションが違います。POSIXの `sed` は `-E` と `-n` を認めます。 `-i` で元ファイル

を直接編集するのは拡張で、よく使う二つの版で、指定方法が異なります。

```
sed -i 's/old/new/' notes.txt     # GNU sed。多くのLinuxシステムで使われます
sed -i '' 's/old/new/' notes.txt  # BSD sed。macOSで使われます
```

どちらの書き方も、もう一方では失敗します。足りるならPOSIXの定義内にとどめ、本番で動かすOSでスクリプトを実行します。拡張に価値がある場合は、両側に同じツールを入れます。

同じOSの二台でも違います。 Debianという名は、リリースの系列を指します。両方Debianと説明されるホストでも、リリース、カーネル、パッケージの版が何年も離れ、導入済みパッケージやローカルの変更も違う場合があります。比較する前に、各マシンの実体を読み取ります。

```
cat /etc/os-release
uname -srm
```

最初は、ディストリビューションの名前とリリースを、次は、カーネルの名前とリリース、マシンのプロセッサー種別を出します。文書化した同じ構成定義から、すべてのホストを構築し、インベントリーにリリースを残します。

管理者としてサービスを動かします。 Unix系システムでrootとして動くサービスは、すべてのファイルを読み、すべての設定を変えられます。その欠陥は、マシン全体の欠陥です。サービスごとに、書き込む場所だけに書き込み権限を持つ、専用の非特権ユーザーを与えます。追加の権限は、必要だと分かったときに、一つずつ認めます。

セキュリティーサポートを過ぎても、リリースを使います。 どのリリースも、公開された予定に従って、修正の提供を終えます。Debianのセキュリティーチームなら、各安定版を公開日から三年間サポートします。古いままのホストも動き続けるため、忘れられます。他の更新日と並べて、各ホストのサポート終了日を記録し、十分前からアップグレードを計画します。

参考資料

- オペレーティングシステム：NIST CSRC用語集 — 主制御アプリケーションからサービスルーチンの集合まで、二つの定義です。

- POSIX.1-2024、基本定義：定義 — プロセス、ファイル、装置など、OSを構成する言葉の厳密な定義です。
- POSIX.1-2024：sed — 適合するすべての `sed` が受け入れるオプションと、移植可能なスクリプトが使える範囲です。
- Linuxカーネルのドキュメント：概念の概要 — カーネル自身が説明する仮想メモリー、ページキャッシュ、メモリー回収です。
- DebianセキュリティーFAQ — セキュリティー修正の扱いと、安定版への提供期間です。

用語集

実際の制約に合わせて選ぶ

デプロイの流れ、運用上のリスク、緊急対応の担当者をお知らせください。検査、復旧手段、明確な責任の所在を伴う変更の範囲を定められます。

料金を見る お問い合わせ →

用語を実務に結び付ける

「オペレーティングシステムとは？」がお客様の組織にどう当てはまるかをご相談ください。現在のシステムの具体例が一つあれば、話を始められます。

サービスを見る

welcome@engineer.company +45 50 10 61 84 デンマーク、コペンハーゲン

このページを持ち歩く 連絡先カード

心を込めて丁寧に制作： D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

著作権 © 2025 – 現在 · Engineer ApS · すべての権利を留保します。

印刷元： https://engineer.company/ja/glossary/operating-system/