eBPF FOUNDATION

eBPF の現状

In partnership with

THE LINUX FOUNDATION | Research

2024 年 1 月

eBPF の現状

Linuxカーネルの3000万行のコードは重要な機能を実現しているが、新機能の追加には数年かかることもある。

eBPFは、パケット フィルタリングをはるかに超えて進化し、**カーネル内の汎用コンピューティングマシン** になっている。

eBPFを使用すると、**エンジニアはカーネル内でカスタムvプログラムを迅速に構築できる。**
コミュニティ全体が変更を受け入れる必要はない。

クラウド ネイティブ ワークロードのニーズと並行して、**eBPFは機能をサポートし、パフォーマンスを向上させ、簡潔さを促進する。**

eBPFを使用すると、Linuxシステムの **可観測性、ネットワーキング スタック** の一部の書き換えやバイパス、**脆弱性の迅速な修正** が可能になる。

Google、Meta、Netflix
などの大手ハイテク企業は何年も前からデータセンターでeBPFを活用している

多くのアプリケーションがすでにeBPFを使用し、
継続的プロファイリング、監視サーバー、可観測性プラットフォーム、パフォーマンス監視ツールを実現している。

イノベーションはeBPFの中心であり、
より高速で安全で柔軟性の高いイテレーション サイクルを作成する。

検証機とJITコンパイラは、eBPFの展開において **安全性とパフォーマンスの利点** を提供する。

eBPFのその他の課題には、**パフォーマンスと機能のトレードオフ、ツールの共存や相互運用性、**
プログラムの記述に必要なカーネルの専門知識などがある。

eBPF Foundationと運営委員会は、**技術的な方向性** を提供し、テクノロジーのロードマップに関する **コラボレーションを最適化** する。

eBPFはクラウド ネイティブ インフラ スタックの重要な層になりつつあるため、すべてのオペレーティング環境の命令セットを中心とした **標準化がeBPFエコシステムの課題。**

目次

はじめに

Linux オペレーティング システムは、誕生からわずか 30 年で、普遍的なコンピューティング オペレーティング システムに成長しました。Android OS、世界のトップ 100 万 Web サーバーの 93%、IOT デバイス、車載システム、テレビ、スマートウォッチ、世界最速のスーパー コンピューター トップ 500 など、何十億ものマシンが **Linux で動作しています**。

この広範な展開により、最先端の機能よりも安定性を求めるユーザーがカーネルを変更することが難しくなっています。 しかし、現在、急速に拡大しているテクノロジーによって、Linux カーネルがプログラム可能になり、イノベーションのサイクルに革命が起こりました。 これは、リスクやコストのかかる変更をカーネルに加えることなく、Linux の動作をリアルタイムで変更できることを意味します。

そのテクノロジーは eBPF です。

eBPF を使用すると、ユーザーは Linux カーネル内でカスタムプログラムを実行できます。これにより、カーネルの動作が変更され、私たちのコンピューティング生活を機能させる主要部分の実行が**最大 10 倍高速**かつ効率的になります。これには、エンジニアがシステムのどこで問題が発生しているかを確認し、修正を迅速に見つけることを可能にする可観測性、電子メールの移動速度から計算の実行速度まで、すべてに関係するネットワーキング、デジタルラ イフとインフラストラクチャをサイバー脅威から安全に保つセキュリティが含まれます。

Netflix 時代に eBPF に取り組み始め、現在は Intel のフェローである Brendan Gregg は、「eBPF は、これまで固定されていたコンピューティングの領域に柔軟性を提供し、可観測性、セキュリティ、およびネットワーキングの新しいテクノロジーを強化します。」と述べています。

eBPF のイノベーションはまた、企業がより優れたパフォーマンスを達成するために必要なハードウェアが少なくなり、同じ機能を実行するための消費電力が少なくなることを意味します。 これにより、運用のコスト効率、エネルギー効率、持続可能性が向上し、株主、消費者、コミュニティの期待に応えるための必要性が高まっています。

eBPF がリリースされてからの 10 年間で、主要なユースケースとして、可観測性、ネットワーキング、セキュリティが浮上してきました。 しかし、昨年、Meta は新しい eBPF ベースの中央処理装置スケジューラーによって大きな成果を上げ、Meta のいくつかの最大のアプリケーションで 5% の CPU 帯域幅の向上を実現しました。「これは非常に大きな数字です。なぜなら、当社の CPU が 5% 増えるのと基本的に同じことだからです」と、Meta のソフトウェアエ ンジニアリング担当ディレクターの Dan Kelley は言います。「eBPF でできることを根本的に増やそうとしています。新しい CPU スケジューラーを使用することで、より迅速に反復処理を行うことができ、生産性とパフォーマンスにおける大きな成果は、コンピューターに費やす必要のないコストになります。これは、Meta が行っているカーネル プロジェクトという意味では、ここ 4、5 年で最もエキサイティングな開発の 1 つです。」

新しい eBPF ベースの中央処理装置スケジューラーによって、Meta のいくつかの最大級のアプリケーションで 5% の CPU 帯域幅の向上を実現しました。.

eBPF は最近まで Linux のみのテクノロジーでした。 2021 年、**Microsoft** は eBPF プログラムを Windows OS 上で実行できるようにするために、eBPF for Windows プロジェクトを作成しました。これは、eBPF が業界全体のインフラストラクチャ言語として標準化されるための基礎を築いたと Graf は述べています。 統一された基盤インフラストラクチャがあれば、企業はいずれかの OS にロックインされるリスクなしに、その上で好きなようにイノベーションを行うことができます。

ブラウザからデータベース、クラウドに至るまで、このベンダーロックインの欠如は、歴史的にイノベーションの増加、コストとパフォーマンスの面での競争を促進してきました。これは、Linux と eBPF の両方を推進するオープンソースの精神の基盤となる信条です。

さらに、Linux と Windows のどちらの世界の開発者も、カーネル拡張機能を記述することができます。Microsoft のプリンシパル ソフトウェア デザイン エンジニアである Alan Jowett は次のように述べています。「これは、より多くのアプリケーションがより早く市場に提供され、より大きな市場に対応できることを意味します。」

誰かが「**アイデアを思いついて、(eBPF) プログラムを書けば**」それを数日以内に「**エンドユーザー**（多くの場合は企業や組織）**に届けることができる**」と、Thomas Graf（Isovalent の CEO で eBPF エコシステムの先駆者）は、「**Unlocking the Kernel**」（eBPF の開発と成長を追ったドキュメンタリー）の中で述べています。

また、Microsoft が eBPF エコシステムに加わったことは、「業界を永遠に変える」テクノロジーとして確立するための「最後の大きなマイルストーン」だったと Graf は述べています。

eBPF の進化

eBPF のような「仮想マシン」を Linux カーネルに組み込もうとする試みは以前にも行われていましたが、混乱を招きすぎるのではないかという懸念から常に却下されてきました。

代わりに、開発者は回避策を見つけました。そのほとんどはカーネル モジュールの形で行われましたが、モジュールにバグがあるとカーネル全体がクラッシュする可能性があるため、リスクが高いものでした。

もう 1 つの選択肢は、カーネルに実際の変更を求めることでした。これは、前述のように、エンド ユーザー環境のサポート リリースに到達するまでに数年かかる可能性があります。 数十年前のツールを使用したり、カーネルの外部でコンテキストを切り替えたりするには、時間と処理能力が必要です。

その後、エンジニアの Alexi Starovoitov は、新しい方法で Daniel Borkmann と協力し始めました。 Linux カーネル内にまったく新しい仮想マシンを挿入しようとするのではなく、彼は既存のものに目を向けました。それは、BPF (Berkeley Packet Filter) として知られるパケット フィルターを移動する仮想マシンです。 彼らは BPF の機能を「レゴ ブロックを積み上げるように」拡張し、単なるネットワークをはるかに超えて、カーネル内の汎用コンピューティング マシンになるまで拡張しました。

このレポートでは、eBPF の進化、eBPF が生み出した革命、現在 eBPF を使用して構築されているもの、課題、および方向性について説明します。

出典：x.com/breakawaybilly

eBPF は「ブラックボックス」を開く

オペレーティング システムのカーネルは、ハードウェアと直接やり取りし、デバイス ドライバーからユーザー アプリケーションまでのマシン コンポーネントと対話するソフトウェアです。 カーネルは、アプリケーションが行うすべての興味深いことに関与します。 アプリケーションがネットワーク メッセージを送信したり、メモリを割り当てたり、その他いくつかのことを行う場合、カーネルを経由してハードウェアに関与します。Linux カーネルには **3000 万行のコード** があり、変更を加えると、それをマージするだけで数ヶ月から数年かかることもあります。

eaBPF は、Linux カーネルの仮想マシンのようなものです。 eBPF を使用すると、開発者は小さな専用プログラムを実行するための eBPF 命令を記述します。 それらは eBPF の「検証ツール」に送られ、プログラムがカーネルに追加しても安全であり、バグをもたらしたりカーネルをクラッシュさせたりしないことをチェックします。 プログラムは JIT コンパイルされてマシン コードになり、実行されてイベント ターゲットにアタッチされます。つまり、プログラムはファイルを開くなどのイベントによってアクティブ化されます。

eBPF はプラットフォームの機能を拡張し、パフォーマンスを向上させ、複雑さを軽減するため、次世代のクラウドネイティブ ワークロードを触媒します。

Red Hat のシニア プリンシパル カーネル エンジニアである Toke Hoiland-Jorgensen は、「eBPF は、これまで非常に困難だった、あるいは事実上不可能だったことを可能にするという意味で、非常に重要です」と述べています。「eBPF のパワーは、この一般性、つまり関数を指定しないという事実から来ています。特定の場所に独自の機能を実装することができ、必要なことは何でも実行できます。」

eBPF はクラウド ネイティブとともに成長し、その逆も同様

Meta は現在、eBPF ベースのスケジューラーを運用しています。 Google は 2024 年初頭に独自の eBPF ベースのスケジューラのテストを開始する予定だと Kelley は述べています。 これは、eBPF がすでにあらゆる業界の企業で広く使用されているにもかかわらず、その影響の初期段階にあることを強調しています。

そして、ますますクラウド ネイティブ化が進む今日のコンピューティングの世界にぴったりのタイミングです。 **Gartner** は、2025 年までに「新しいデジタル ワークロードの 95% 以上がクラウド ネイティブ プラットフォームで展開される」と推定しています。 これは 2021 年の 30% から増加しています。

クラウド ネイティブ開発アプローチにより、企業はより効率的かつ信頼性の高いアプリケーションを構築して提供できるようになります。 eBPF は、プラットフォームの機能を拡張し、パフォーマンスを向上させ、複雑さを軽減するため、次世代のクラウド ネイティブ ワークロードを促進します。

Linux と Kubernetes の組み合わせは、事実上のクラウド OS であり、クラウド ネイティブ開発の基盤です。 確かな基盤ですが、Kubernetes は非常に複雑です。 また、Linux カーネルに変更を加えるには何年もかかることがあります。ほとんどの組織には時間がありません。 これらの要因は、組織のイノベーション能力を低下させます。 これが、eBPF がクラウド ネイティブの世界で広く急速に受け入れられている理由です。

「企業は自社のソフトウェアをクラウドに移行しようとしており、それを支援する可観測性ツールやネットワーキング ツール、セキュリティ ツールを求めています。eBPF は、これらのツールを作成するための優れたプラットフォームであり、非常に革新的なインフラストラクチャ技術になります」と、eBPF の先駆者である Isovalent のチーフ オープンソース オフィサーであり、O'Reilly レポート"**What is eBPF**" の 著者である Liz Rice は言います。eBPF がカーネルに組み込まれていることで、「マシン全体で何が起こっているかを観察し、影響を与えることができる、これらの信じられないほど強力なツールを持つことができます。これが eBPF がクラウド ネイティブの世界で離陸した理由です」と Rice は言います。

おもなユースケース：可観測性、ネットワーキング、セキュリティ

5 年以上前から、eBPF は世界中の何百万ものデバイスとサーバーで動作しています。 ほとんどの人は、企業が eBPF を使用することによってすでに影響を受けていますが、おそらくそれを知らないでしょう。

米国のハイパースケーラーの多く—Meta、Google、Netflix—は、本番環境で eBPF を使用しています。 すべての Android 携帯電話は、トラフィックを監視するために eBPF を使用しています。 Meta データセンターを出入りするすべてのパケットは、eBPF によって処理されます。ソフトウェア、クラウド サービス、金融サービス、通信、メディアとエンターテイメント、e コマース、コンサルティング、セキュリティなど、さまざまな業界の企業が、eBPF テクノロジーを使用して、より多くのことをより速く、時間とコストを節約し、パフォーマンスを向上させるようになっています。 ここでは、これまでのところ、本番環境での eBPF の 3 つの主要なユースケースの内訳を示します。

可観測性

多くの企業にとって、可観測性の分野が eBPF が最初に飛躍し、最大の影響を与えた場所です。 Graf はコンピューティングにおける可観測性を、部屋の電気を消して掃除をすることにたとえます。 電気をつけると、掃除がずっと早く進み、良い結果が得られます。

eBPF の可観測性ツールは電気をつけます。

「カーネル開発者でない人にとって、Linux カーネルはブラックボックスのようなものです。」と、Hoiland-jorgensen は言います。「eBPF はブラックボックスを開き、システムがどのように機能しているかについて、以前は得られなかった情報を得ることを可能にします。

こで時間を費やしているか、どこでボトルネックが発生しているか、CPU (カーネルとアプリケーションを実行する中央処理装置) の動作速度、どこでサイクルが回転しているかをより簡単かつ完全に把握し、どのコード部分が誤動作している可能性があるかをより迅速に見つけることができます。 企業が数百または数千台の巨大なサーバー クラスタを導入していて、何か問題が発生した場合、問題の原因を特定するのに数日、数週間、場合によっては数か月かかることがあります。

米国のハイパースケール企業の多く (Meta、Google、Netflix) は、本番環境で eBPF を使用しています。すべての Android スマートフォンは、トラフィックを監視するために eBPF を使用しています。Meta データセンターを出入りするすべてのパケットは、eBPF によって処理されます。.

Gregg は、今日の主要な eBPF ベースの可観測性ツールの多くと、数十の高度な eBPF トレース ツールを作成して、サイクルを数日、数時間、または数分に短縮し、CPU、メモリ、ディスク、ファイル システム、ネットワーク、言語、アプリケーションなどの分析を含むシステムの動作に関する独自の洞察を提供しています。 eBPF プログラムをファイル オープンなどのイベントにアタッチすることで、ユーザーはシステムで何が起こっているかを驚くほど可視化するメトリックを取得できます。

特定のレベルを達成するには、パフォーマンスやインストルメンテーションのボトルネックのためにコストがかかりすぎるため、企業は長い間、可観測性に関する選択を行う必要がありました。 Gregg の仕事と他の人たちの努力のおかげで、eBPF はより大きな可観測性をより効率的かつ簡単に追加できるようにし、根本原因となるエラーの可視性を大幅に向上させるツールを構築できるようになりました。

「Brendan の eBPF での作業は可観測性を破壊しました。彼が Netflix で作ったものはすべて主流になり、彼の仕事は私たちに多くのツールをもたらしました。」と、Netflix のプラットフォーム ネットワーキングのディレクターである Shweta Saraf は述べています。

現在、Netflix は、クラウド エコシステムを可視化するために、1 時間あたり数十億の eBPF「フローログ」を取り込み、強化しています。 強化されたデータにより、Netflix はネットワークの可用性、パフォーマンス、セキュリティを分析し、アプリケーションがグローバルに分散したクラウドベースのエコシステム全体にデータを配信できることを確認できます。 eBPF はそれを「非常に効率的」にする、と Saraf は付け加えました。

Netflix は、毎時数十億の eBPF「フローログ」を取り込み、充実させることで、そのクラウド エコシステムの可視性を確保しています。 この充実したデータにより、Netflix はネットワークの可用性、パフォーマンス、およびセキュリティを分析し、アプリケーションが世界中に分散したクラウドベースのエコシステムを通じてデータを確実に配信できるようにしています。

ネットワーキング

ネットワーキングの分野は、eBPF が速度とパフォーマンスを向上させる方法の好例です。

Linux ネットワーク スタックの多くの部分は、IP とポート範囲がコンテナごとに変更されるのではなく、スプレッドシートで追跡できた数十年前に書かれたものです。 eBPF を使用すると、プログラマーはネットワーク スタックを書き直したり、必要な部分のみを利用したり、完全にスキップして時間と処理能力を節約したりできます。 不要なものをバイパスしたり、ソフトウェアを構築する新しい方法に基づいて機能を書き直したりすることで、ネットワークのパフォーマンスが劇的に向上します。

Cilium のレイヤー 4 **ロード バランサー XDP** は、Seznam.cz のスループットを倍増させ、CPU 消費を 72 分の 1 に削減しました。Meta は 2018 年に Katran を**オープンソース化**しました。データセンターを出入りするすべてのパケットは、パフォーマンス上の利点から eBPF によって処理されます。 2022 年、Walmart は eBPF を使用して Kernel Function as a Service を提供する L3AF プロジェクトをオープンソース化し、eBPF プログラムのマーケットプレースを作成しました。ユーザと開発者は自分の署名した eBPF プログラムを共有し、他の人から eBPF プログラムをダウンロードできます。 iptables はカーネル コミュニティと eBPF によって**置き換えられ**、ネットワークをほぼライン レートにしています。

セキュリティ

eBPF によって強化された可観測性によって、カーネル内だけでなく Kubernetes やクラウド ネイティブ環境全体を含むセキュリティ攻撃を検出して防止する能力が向上します。

「見えないものを安全にすることはできません」と語るのは、クラウド ネイティブなアプリケーションにおけるセキュリティ侵害を防止、検出、軽減するためのセキュリティ プラットフォームを提供する**Tigera** のチーフ プロダクト オフィサーである Amit Gupta です。「セキュアなアーキテクチャの最初のステップは、完全でセキュアな可視性を得ることです。eBPF はその可視性を生み出すのに役立ちます。eBPF データがユーザー空間に戻ってきたら、発生している他のすべてのものと調整して ... 悪意のあるアクティビティの可能性があるかどうかを判断できます。」と Gupta は言います。 攻撃者を見つけ出すのに何カ月もかかったり、完全に見失ったりする代わりに、eBPF ツールは「数秒以内に」アクションを実行できると Gupta は言います。

eBPF はまた、セキュリティ ポリシーを分散環境に適用し、リアルタイムで実装できるようにします。 例えば、カーネルに脆弱性が発生した場合、カーネル コードを変更することなく eBPF を介して迅速な修正を行うことができ、セキュリティ アップデートをその場で行うことができます。

これらの機能により、「eBPF はソフトウェア サプライ チェーンのセキュリティ保護において非常に大きな役割を果たすでしょう」と Gupta は付け加えています。

「ソフトウェアが世界を侵食している」と言う人もいるが、私はそう言いたい。「eBPF がソフトウェアを侵食している」と、グローバル クラウド プラットフォームである Cloudflare の**ブログ**で述べられています。同社は DDoS 軽減策やその他の製品を構築するために eBPF を使用しています。

eBPF は、同じプログラムを複数のバージョンの Linux カーネルで実行できるように記述することもできます。 これは、企業が新しいカーネル バージョンを採用する割合が異なるため、重要です。eBPF は複数のバージョンで実行できるため、企業は新しいセキュリティ機能を採用する可能性が高くなります。なぜなら、古いカーネル バージョンとの相互運用性があることがわかっているからです。

たとえば、カーネルに脆弱性が発生した場合、カーネル コードを変更することなく eBPF を介して迅速な修正を行うことができるため、セキュリティ更新をその場で行うことができます。

eBPF の普及

当初、eBPF は Linux 開発者だけが利用できる技術でした。非常に複雑で、Linux カーネルの専門知識を持つ人がほとんどいなかったからです。

その後、Google、Facebook、Netflix などのハイパースケール企業が参入しました。 彼らには、この技術を開発するためのエンジニアリング力と、それを利用し、展開し、改善するためのシステムの専門知識と制御力がありました。 彼らはまた、本番環境で eBPF を主導してきました。これは、おそらく今日のほぼすべてのデジタル消費者の生活に影響を与えていることを意味します。

2017 年以来、Meta はデータセンターに入るすべてのパケットを eBPF で処理しており、Google もデータセントラフィックのほとんどを eBPF で処理しています。 Meta は、**CPU スケジューラーの効率性**を向上させたことでの成果を文書化しています。 Google はセキュリティの監視と実施の両方に eBPF を使用しています。 Isovalent の Bill Mulligan と Daniel Borkman は **InfoQ** で、**Android スマートフォン** システムのすべてのネットワーキング パケットが eBPF にヒットすると報告しています。Netflix のプラットフォーム ネットワーキングのスタッフ ソフトウェア エンジニアである Arthur Gonigberg は、2 億人を超える顧客から Netflix に送られるすべてのトラフィックとすべてのサーバーが「どこかの eBPF を経由する」と述べています。

.

図 1

EBPF FOUNDATION メンバー

プラチナ

CROWDSTRIKE
Google
HUAWEI
intel
ISOVALENT
Meta
Microsoft
Red Hat

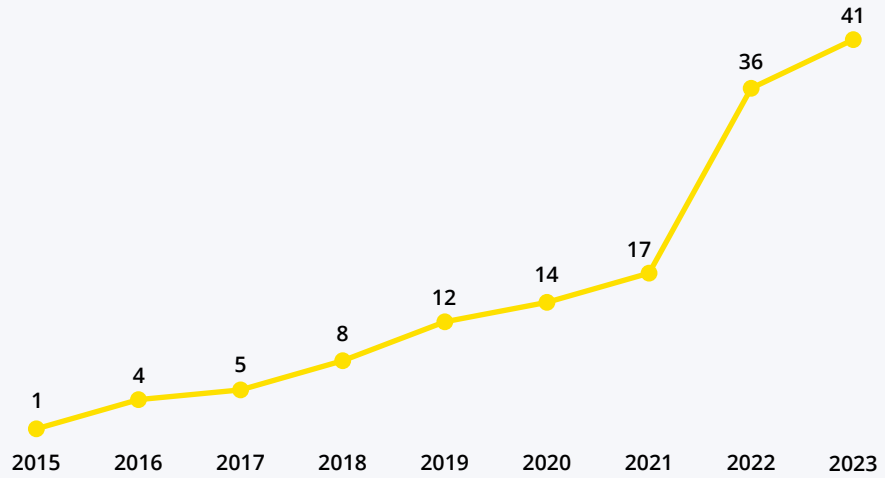
シルバー

DATADOG
FUTUREWEI
NETFLIX

出典：eBPF Foundation

図 2

アクティブな EBPF ランドスケープ プロジェクト（年度別）



出典 : GitHub

私たちのための eBPF

ハイパースケール企業や大企業には、ほとんどの企業にはないものがあります。それは、ソフトウェア エンジニアのチームです。eBPF をより多くの企業に普及させるために、このテクノロジーをより利用しやすく、すぐに使えるようにするオープンソース プロジェクトが生まれました。

eBPF を搭載した Cilium は、他のすべてをリードしています。2016 年に設立された Cilium は、最初はネットワークに焦点を当てていたが、eBPF を使用した接続性、可観測性、セキュリティに拡大しました。Cilium は、Kubernetes、Envoy、Prometheus と並んで、Cloud Native Computing Foundation の中で最も急成長しているプロジェクトの 1 つです。

100 社以上の企業が **Cilium** を採用したことを公表しており、このプロジェクトは世界最大の Kubernetes クラスタのいくつかを動かしています。エンド ユーザーは、スタートアップから大手金融機関、通信会社、The New York Times、Bloomberg、Bell Canada などの企業にまで及んでいます。

Isovalent が開発のために設立した Cilium は、企業がクラウド ネイティブ環境で必要とするソリューションを構築し、Kubernetes を既存のレガシー インフラストラクチャに接続する機能を構築します。その一部はクラウド ネイティブになることはありません。さらに、主要なクラウド プロバイダーが Kubernetes ソリューションに Cilium を統合し、クラウド ネイティブ エコシステム全体の「デフォルト」にしています。

「AWS はオンプレミスの Kubernetes サービス EKS Anywhere の舞台裏でネットワーキングとセキュリティ機能を提供するために Cilium を使っています。Google Cloud は Google Kubernetes Engine のネットワーキング コンポーネントを強化するために Cilium を使っている」と **The Stack** は報告しています。 S&P Global のネットワーク エンジニアは、Cilium を使った eBPF ベースのネットワーキングを活用して「マルチクラウド接続スーパーハイウェイ」を構築しました。

「Goldman Sachs は eBPF を直接使うことはありません。Cilium のような eBPF ベースのソリューションを購入するでしょう」と Graf は言います。

Red Hat は eBPF を早期に採用した企業の1つでもあり、同社のエンタープライズ Linux ディストリビューションで eBPF を早い段階から実現し、サポートしています。同社の顧客は、さまざまなプラットフォームで Red Hat Enterprise Linux (RHEL) を実行しています。 RHEL における eBPF の初期の用途の 1 つは、**高速データパス** (XDP) サブシステムです。これにより、柔軟でプログラマブルなネットワーク パケット処理を Linux カーネル データ パスに直接統合し、非常に高いパフォーマンスを実現できます。XDP には、サーバーがデータ パケットを処理するとすぐに実行される eBPF フックがあるため、eBPF はパケットを確認して、Linux カーネルが単独で動作している場合よりも高速にパケットを送信できます。 eBPF プログラムは、プログラムされていることは何でも実行します。それは単に DDoS 攻撃から保護するためにパケットをドロップすることにすぎないかもしれません。Red Hat は、複数の XDP プログラムを順番に実行するためのライブラリも開発しています。したがって、あるベンダーがユーザにファイアウォールを提供し、別のベンダーが同じ顧客にパケット ロガーを提供する場合、顧客はそれらを一緒に実行する方法を知る必要があります。 libXDP ライブラリは、それらが一緒に正しく動作することを確認します。

稼働・開発中の eBPF アプリケーション

その他の**主要な eBPF アプリケーション**:

- **Bcc**: 効率的な eBPF カーネル トレースのためのツールキットとライブラリ。
- **Bpftrace**: Linux eBPF の高レベル トレース言語。
- **Falco**: アプリケーションの異常なアクティビティを検出するように設計された動作アクティビティ モニター。 eBPF の助けを借りて、Falco は「Linux カーネル層でシステムを監査する」。
- **Katran**: ハイ パフォーマンスのレイヤ 4 ロードバランサー。
- **Pixie**: eBPF を使用してテレメトリ データを自動的にキャプチャする Kubernetes アプリケーション用のオープンソースの可観測性ツール。
- **Calico**: コンテナと Kubernetes のためのプラグ可能な eBPF ベースのネットワークとセキュリティ。
- **Tetragon**: eBPF ベースの透過的なセキュリティ可観測性とリアルタイムのランタイム エンフォースメントを組み合わせて提供。

eBPF Foundation の **Web サイト**にリストされているように、他にも約 30 の新興アプリケーションが開発中です。これらのアプリケーションは、継続的なプロファイリング、疑わしい動作を検出するためのイベントの検出とフィルタリング、監視サーバー、可観測性プラットフォーム、パフォーマンス監視ツール、クラウド ネイティブ ワークロードのプラットフォーム、ロードバランサー、Kubernetes リソースのデバッグと検査のためのツール、サービス マップ、強制システムなどを可能にします。eBPF には、eBPF プログラムのデバッグ、ロード、コンパイルを支援する Rust、Go 言語、C++ で記述されたライブラリもあります。

直接的および間接的な方法でのイノベーション

eBPF は、多くの直接的および間接的な方法で、組織のイノベーション能力を向上させます。

最も直接的には、たとえば Meta の CPU スケジューリングに関する作業のように、開発ライフサイクルを加速するいくつかのパフォーマンス向上を提供します。「eBPF が通常のカーネル開発と比べて非常に魅力的なのは、通常のカーネル開発では、仮説を立て、新しいカーネルを構築し、コンピューターを再起動し、仮説をテストしなければならないことです。遅いですね。」と、Meta の Kelley は説明します。「アイデアを持ってからテスト結果を得るまでの時間が短いほど、大きなアイデアをどれだけ早く実行できるか、どれだけ大きなアイデアを合理的に追求できるかが決まります。eBPF はそのすべてを縮小しています。イテレーション サイクルははるかに高速で、より多くの実験を実行できるため、それほど賢くなる必要はありません。短い反復時間で安全に行うことができれば、発見のプロセスを早めることができます。」

さらに、eBPF が Linux カーネル リリースから切り離されているということは、開発者が必要とする新機能がメジャー ディストリビューションのリリースに反映されるまで、何か月も何年も待つ必要がないということです。 Linux カーネル リリースから切り離されていることは、Linux カーネル自体の「機能の忍び込み」とそれに伴うパフォーマンスの低下を防ぐのにも役立ちます。

同じように、eBPF は、カーネルにパッチを適用し、変更を段階的にフリートにデプロイするという時間のかかるプロセスをバイパスすることで、本番環境のフィードバック ループを Linux カーネル自体から切り離します。代わりに、変更を行ってその場でテストすることができます。eBPF は、ユーザー空間スタックで必要なものを活用することもでき、スタックの大部分を書き換えるのではなく、小さな変更を行うことができます。

これらすべてが、アプリケーションのモノリスの一部をマイクロサービスに分割するプロセスによく似ているように聞こえるなら、それは偶然ではありません。実際、eBPF はここ数年の最も重要なコンピューティング パラダイムのいくつかと一致しています。代表的な例 : エッジ コンピューティングが計算リソースとデータ ストレージ リソースをデータ生成のポイントに近づけるのと同じように、eBPF はデータ処理をパケット レベルから、特定のソケット フックや XDP (eXpress Data Path) などのイベント ソースにできるだけ近い場所に移動します。これにより、リソースの消費が削減され、データ処理の効率が向上します。 組織は eBPF を使用することで、ネットワーク、可観測性、セキュリティの効率とパフォーマンスをさらに向上させることができます。

eBPF が Linux カーネル リリースから切り離されているということは、開発者が必要とする新機能がメジャー ディストリビューションのリリースに反映されるまで、何か月も何年も待つ必要がないことを意味します。

eBPF は、同じプログラムを複数のバージョンの Linux カーネルで実行できるように記述することもできます。 これは、より迅速なイノベーションにも拍車をかけます。たとえば、Meta、Google、Amazon などの企業は何百万台ものコンピューターを所有していますが、そのすべてがまったく同じバージョンのカーネルを実行しているわけではありません。コンピューター全体をアップグレードするには長い時間がかかるためです。そのため、常に複数のバージョンの OS を同時に実行しています。フリート上で実行されている複数のバージョンのソフトウェアにそれぞれの eBPF プログラムを特化させなければならないとしたら、それは大きな税金になるでしょう。「異なるバージョンのカーネルで実行できることは、開発のイテレーション速度をスピードアップするもう 1 つの要因です」と Meta の Kelley は説明します。

より間接的には、eBPF が導入するガードレールは、開発者が面倒な (しかし重要な) 管理、可観測性、セキュリティ タスクではなく、イノベーションと継続的なイテレーションに集中するのに役立ちます。

たとえば、開発者がバグを探して修正するのに費やす時間は、収益を向上させるアプリケーションやサービスの構築に費やす時間ではありません。eBPF プログラムを使用してネットワーク パケットやその他のイベントをリアルタイムでトレースすることで、組織はシステムの動作を詳細に把握し、問題に迅速に対応できるようになります。 たとえば、それでもパケットが失われた場合は、eBPF ベースのオープンソース ツールである **Pwruc** があります。これは、Linux カーネル内のパケットをトレースして、「ネットワーク接続の問題のデバッグ」を迅速化するためのものです。

最も重要なのは、eBPF がカーネルを保護し、最終的にイノベーションを促進することです。

Linux 用の eBPF **検証**ツールは、2 段階のプロセスで eBPF プログラムの安全性を保証します。最初にループやその他の構成検証を禁止するチェックを行い、次にすべての命令の実行をシミュレートし、レジスタとスタックの状態変化を観察します。

簡単な作業ではありませんが、検証ツールで何かが合格すれば、誰もが eBPF ベースの新しい製品に自信を持つようになります。スタートアップは、Netflix のような大企業が eBPF ベースの製品を使用するように説得する可能性が高くなります。

Windows 側でも同じことが言えます。Linux とは異なる検証ツールを使用します。Windows の検証ツールは、最初に eBPF コードをオフラインでチェックし、すべてのコードが証明可能な安全性を備えていることと、Microsoft が安全と見なす人物によって署名されていることを要求します。これにより、コードの出所を確認できます。

従来の Windows ドライバーは、サードパーティによって Windows コードベースの外部で開発されており、Microsoft の開発者ほど Windows に関する深い知識を持っていないことが多いため、ドライバーの安定性が問題になることがありました。

eBPF はカーネルを保護し、最終的にはイノベーションの強化を促進します。

eBPF for Windows は、検証ツールを使用することで、これらの問題の一部をキャッチできる別のセキュリティ チェックを提供します。また、Microsoft の Jowett は次のように述べています。「Windows ドライバーやその他のカーネル エクステンションが安全であることを確認することで、Windows のエコシステム エクスペリエンスを大幅に向上させることができます。

“Fast by Friday”

テクノロジーは、多くの場合、物事をより速く、より効率的に進めます。これは、必要なハードウェアが少なく、消費電力が少ない環境に役立ちます。タスクの完了を高速化することで、イノベーションを促進します。企業がコンピューティング コストを抑制するのに役立ち、Web サイトの読み込みが高速なエンド ユーザーにもメリットがあります。

しかし、新世代のハードウェアとソフトウェアが優れたパフォーマンスを約束しているからといって、その可能性が常に実現されるとは限らない、と Gregg は 2023 年の eBPF サミットの**基調講演**で述べました。むしろ、ボトルネックは長い間謎のままであり、物事を遅らせます。 新しいソフトウェアとハードウェアのオプションは、エンジニアに時間がないため、常に評価されているわけではありません。コンピューターがますます複雑になるにつれて、すべてのコンピューティングの問題をより迅速に解決できるようにすることが課題となっています。

「あらゆるパフォーマンスを解決し」、エンジニアがあらゆるソフトウェアのパフォーマンス問題を修正でき、本番環境で「即座に」実行できるようにするには、「eBPF が不可欠だ」と Gregg は言います。

eBPF ツールは、CPU のフレーム グラフを使用して、CPU がブロックされている場所を確認し、レイテンシーの問題を掘り下げ、より効率的なツールを作成し、最終的には 0 インストルメンテーションを可能にします。彼は業界に対して、コンピューティングの問題を数週間や数か月ではなく、5 日以内に修正するための標準を設定するよう求めています。これが、彼の造語である「fast by Friday」です。

問題がより早く修正されれば、イノベーションも速く起こります。

魔法の妖精の粉ではない

eBPF には多くの利点がありますが、Red Hat の Hoiland-jorgensen は、eBPF は「何でもできるようにする魔法の妖精の粉」ではないと言います。「私たちはまだ物理法則に縛られています。CPU は 1 秒間に特定の数の処理しか実行できません。」

eBPF の最大の課題の 1 つは、ほとんどすべてのソフトウェア プログラムと同様に、セキュリティです。 eBPF は可観測性を高めるため、セキュリティを強化できます。しかし、どのようなプログラムでもセキュリティの課題が生じる可能性があります。 カーネル空間でコードを実行する場合、コンピューター上の通常のプログラムでは利用できない多くの機能にアクセスすることになります。そのため、そうすることで潜在的な害がない完全な未来はありません。eBPF 開発コミュニティは、ツールの能力を犠牲にすることなく、カーネル内の他の形式のコード実行よりも eBPF を安全にする方法を継続的に学習しています。

eBPF 開発コミュニティは、ツールの能力を犠牲にすることなく、カーネル内の他の形式のコード実行よりも eBPF を安全にする方法を継続的に学習しています。

eBPF のその他の課題は次のとおりです。

- **パフォーマンスのトレードオフ：**eBPF によって達成される高いパフォーマンスは、特定のユースケースに合わせた最適化の結果です。eBPF で多くのことを行いすぎると、利益を食べてしまう可能性があります。ユーザーは、パフォーマンスの向上を維持するために有効にする機能のサブセットを選択する必要がある場合があります。
- **共存：**カーネルの機能はまた、多くのユースケースをサポートしなければなりません。 つまり、ツールは共存しなければなりません。あるソフトウェアの eBPF 機能は、他のソフトウェアと組み合わせて動作しなければなりません。 カーネルに 2 つの eBPF プログラムがあり、両方が同じパケットで動作したい場合、どちらが先に行くかを決める必要があります。 現在、これはほとんどが手動で行われていますが、eBPF プログラムがお互いに干渉しないようにするための作業が行われています。
- **カーネルの深い専門知識：**eBPF は Linux カーネルの内部をアプリケーションに公開します。そうすることで、アプリケーションとカーネルの間の境界のために、アプリケーションが伝統的に保護されてきたカーネルの詳細の多くを公開します。これは、eBPF を効果的にプログラミングするには、カーネルの深い専門知識が必要であることを意味します。これは、eBPF の初期の採用のほとんどが、カーネル開発者も雇用している組織によって行われている理由の 1 つです。

Gregg は、ソフトウェア開発者の大多数が eBPF コードを書かないとしても、eBPF は広く展開できると主張します。 Netflix に 3,000 人のエンジニアがいるとすれば、eBPF コードを書くことができるのは約 10 人だけです。しかし、誰もがその存在を知り、必要なプログラムを手に入れるためにその 10 人のところに行くだろうと、Gregg は **YOW! Perth** で語りました。

「企業が eBPF を最大限に活用するためには、eBPF プログラミングを扱うことができるシニア エンジニアが少なくとも 1 人必要であり、彼らは彼らのニーズに最適なオープンソース、商用、または社内ソリューションを見つけるために彼らの会社を導くことができます。」と Gregg は言います。「eBPF は難しい技術かもしれません。適切なソリューションがまだ存在しない場合は、制限された環境でカーネル エンジニアリングとアセンブリ プログラミングを組み合わせて直接プログラミングする必要があり、その価値を完全に実現するには、オペレーティング システムの内部構造に対する確かな理解と想像力が必要です。」

プラットフォーム エンジニアも重要な役割を果たすことになります。eBPF 開発者がオンデマンドで確実かつ安全にコードをデプロイするために必要なハードウェアとソフトウェアをキュレートするだけでなく、Linux カーネルの eBPF 拡張によって、新しい抽象化とビルディング ブロックの作成が可能になります。プラットフォーム エンジニアは、開発者のためにキュレートする標準化されたツールセットに、時間をかけて追加することができます。

- **データが多すぎます。** eBPF 技術のおかげですべてを見ることができれば、「すべてを見ることができます」と、eBPF Summit の**基調講演**で Postman のプロダクト オブザーバビリティ責任者である Jean Yang は述べました。そのため、システムエンジニアはネットワーク データに圧倒される可能性があります。eBPF で収集されたデータをより適切な形式で表示できるようにするには、フィルタリングが必要です。

eBPF がインフラ ソフトウェアの世界で最も影響力のある技術の 1 つになるにつれて、プロジェクト間のコラボレーションを最適化し、「eBPF のコアを適切に維持し、明るい未来に向けた明確なロードマップとビジョンを備えていることを確認する」ことが重要になります。

- **Interoperability. 相互運用性。** 一部の ISV は eBPF ベースのソフトウェアを出荷し始めています。より多くのソフトウェアベンダーが eBPF を組み込むにつれて、相互運用性はさらに大きな必須事項になります。これには、相互運用性のための技術的ソリューション、コミュニティ規範、および優れた eBPF アプリケーション市民であるためのベストプラクティスの組み合わせが必要になります。

eBPF Foundation の役割

eBPF がインフラストラクチャ ソフトウェアの世界で最も影響力のある技術の 1 つになるにつれて、プロジェクト間のコラボレーションを最適化し、「eBPF のコアを適切に維持し、将来の明るい未来に向けた明確なロードマップとビジョンを備えていることを確認する」ことが重要になると、Isovalent の Graf は eBPF Foundation と eBPF の技術的方向性とビジョンを担当する運営委員会を発表する**ブログ**で述べています。eBPF の Windows カーネルへの移植や、その他の新しいプラットフォームへの移植には、移植性とランタイム要件の調整も必要になります。

結論：eBPF が向かう先

eBPF はすでに広く展開されていますが、eBPF の専門家によると、これから解き放たれるイノベーションの大きな波はまだ始まったばかりだといいます。

カーネル開発に関して、Meta が CPU スケジューラーで達成した成果は、eBPF を「根本的に興味深い形で異なるもの」にしていると Meta の Kelley 氏は述べています。これは、これまで以上に迅速に「単に起動してテストする」ことができるという勝利の組み合わせを実証したからです。

eBPF が新しいクラウド ネイティブ インフラストラクチャ スタックの新しいレイヤになることは間違いなく、すべてのアプリケーションの可観測性、パフォーマンス、信頼性、ネットワーキング、セキュリティに影響を与えると支持者は言います。プラットフォーム エンジニアは、eBPF を利用したインフラストラクチャ ビルディング ブロックを組み合わせて、開発者がソフトウェアをデプロイできるプラットフォームを作成し、ビジネス ロジックを追加し、今日のデジタル化とクラウド ネイティブ化に対応できない古い Linux カーネル内部を置き換えます。

一方、Microsoft の eBPF への取り組みは、BSD や Apple の OS X などの「他のオペレーティング システム」が eBPF を使用するための道を開いていると、Gregg は Yow! Perth conference の**基調講演**で述べました。 Microsoft が行っている作業には、eBPF のための独立した基盤とバイトコード、変更を監視する委員会などが含まれます。

Microsoft の Jowett は、eBPF プログラムがすべての環境で同じように機能するようにするためには、命令セットを中心とした eBPF エコシステムの標準化が鍵になると述べています。これにより、アプリケーションの相互移植が容易になります。

「これは、eBPF を Linux 固有の技術から業界全体のプラットフォームに移行させるための基本的な部分です」と Jowett は言います。

Gregg は次のように付け加えています。「Microsoft は、これが Linux を超えて成長することを確実にする方法を推し進めています。」最終的には、eBPF はすべてのオペレーティング システムで利用できるようになると彼は述べています。

eBPF の成功はどのようなものになるのでしょうか？ Postman の Yang は、それは何にも見えないと言います。数年前、ハードウェアが技術スタックを支配していたとき、「Intel Inside」は自信を植え付けるフレーズでした。どのチップが何をしているのか、それ以上調べる人はほとんどいませんでした。

今日のソフトウェアとクラウド ファーストが定義された世界では、「eBPF inside」は同じ重要性を持ち、開発者もユーザーも誰もそれが存在することを知らないだろうと Yang は言います。それは私たちの周りのデジタル世界を動かしているだけになるでしょう。

謝辞

まず最初に、eBPF Foundation とプロジェクトのすべてのメンテナー、貢献者、メンバーに感謝の意を表します。

以下の方々が本稿の内容に貢献してくださいました。

- Dan Brown, eBPF Foundation
- Lisa Caywood, Red Hat
- Will Ferrier, Intel
- Arthur Gonigberg, Netflix
- Thomas Graf, Isovalent
- Brendan Gregg, Intel
- Amit Gupta, Tigera
- Daniel Haney, Microsoft
- Toke Hoiland Jorgensen, RedHat
- Alan Jowett, Microsoft
- Dan Kelley, Meta
- Bill Mulligan, Isovalent
- Sridhar Rao, eBPF Foundation
- Liz Rice, Isovalent
- Shweta Saraf, Netflix
- Chris Wright, RedHat

本訳文について

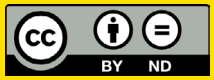
この日本語文書は、**The State of eBPF** の参考訳として、The Linux Foundation Japan が便宜上提供するものです。英語版と翻訳版の間で齟齬または矛盾がある場合（翻訳版の提供の遅滞による場合を含むがこれに限らない）、英語版が優先されます。

この日本語文書を引用する際には、下記の一文を記載してください。
引用：The State of eBPF 参考訳（The Linux Foundation Japan 提供）

翻訳協力：木下兼一

eBPF Foundation は、オープンソース エコシステムにある eBPF 関連プロジェクトのクロスプラットフォーム コミュニティを 1 つの独立したフォーラムにまとめます。このファウンデーションは、Kubernetes、オペレーティング システム カーネル、エンタープライズ コミュニティなどの個別のワークストリーム内で適用される共通の技術ビジョン、用語、セキュリティのベスト プラクティス、および一般的なロードマップについて共同作業するためのフォーラムを提供します。.

2021 年に設立された **Linux Foundation Research** は、拡大するオープン ソース コラボレーションを調査し、新たな技術トレンド、ベストプラクティス、オープンソース プロジェクトのグローバルな影響に関する洞察を提供してい ます。プロジェクトのデータベースやネットワークを活用し、定量的・定性 的手法のベストプラクティスに取り組むことで、Linux Foundation Research は、世界中の組織にとって有益なオープンソースの知見を提供す るライブラリを構築しています。



この著作物を引用する場合は、以下のように明記してください。
The State of eBPF," The Linux Foundation, January 2024.