

# 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、ネットワーク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。

