The eBPF Re-Platforming Thesis

An Investor's Due Diligence Guide

By Andrew Green

June 2026

Introduction

We cannot solve 2026 problems with a 2016 approach. The average enterprise's infrastructure stack is not equipped to address today's problems around GPU scaling, real-time inference, supply chain security, zero day exploits, semantic and intent analysis, and agent identities. The requirements have outgrown the infrastructure architectures that the legacy stack is built upon.

To modernize the stack, developers need a sanctioned, safe, and efficient way to modify infrastructure behavior directly in the operating system. eBPF does exactly that; it grants universal visibility and control over the entire system without modifying application code or installing traditional agents and it does it in a safe and performant manner.

Investors should not view eBPF as just an alternative backend implementation that solves the same problems as existing solutions. They should rather view it as a way of redefining how applications are architected from the ground up to tackle challenges of today and disrupt hundreds of billions of dollars in enterprise software and hardware spend.

Despite being one of the most consequential infrastructure technologies of the past ten years, eBPF remains an underexplored thesis for capital markets. But the window between "emerging technology" and "mainstream premium" is closing. The first acquisition wave of eBPF-based products (2020–2022) consisted of linear upgrades to point solutions. The current wave of acquisitions (2025–present) demonstrates that kernel-layer controls define how products are architected for today and the shape of tomorrow's markets.

This document provides the technical and market analysis framework to evaluate eBPF-native companies. It covers the following:

- How eBPF achieved real-world production results, such as 35% reduction in CPU usage in hyperscaler environments, reduction of cross-zone network traffic costs by 50%, reduction of log volume by 70%, 40% less memory usage, and 98% fewer restarts
- Why it's a fundamental block for new AI-related categories including agent monitoring, agentic sandboxes, and agentic runtime security
- How an accelerating rate of eBPF-specific acquisitions since 2020 demonstrates the evolution of the market and sentiment, from a linear upgrade of an existing category, to a replatform of infrastructure software and hardware architectures
- Why using eBPF is not just a binary Yes/No, where implementation is a differentiator in eBPF-based products

The companies that own that kernel layer and build upward into analytics, policy management, and workflow integration are structurally positioned to displace incumbents who cannot replicate kernel-level access through any amount of engineering effort.

eBPF based products and companies represent an attractive market for investment due to their ability to replatform billion dollar market segments with efficient go to market motions. This whitepaper will walk through:

- The eBPF Market Opportunity
- eBPF as an Economic Differentiator
- Current Ecosystem and Business Models
- Open Source as a Competitive Moat
- M&A Landscape and Exit Examples
- eBPF Due Diligence for Investors
- Market Outlook for 2026

The eBPF market opportunity

As eBPF is Turing-complete and operates in the kernel, it can be used in every system the operating system touches. It provides a horizontal re-platforming mechanism for existing infrastructure software and hardware systems.

To narrow down some of the most lucrative use cases, this paper will explore the segments where an incumbent product category is displaced or replaced by eBPF, due to its structural performance, cost, or visibility advantage.

These include:

- **Networking and network security**
- **Runtime Security**
- **Observability**
- **Hardware appliances**
- **Generative AI**

The aggregate revenue opportunity across just these five categories is substantial. Each one has several sub-categories that individually represent multi-billion-dollar markets. While eBPF does not replace the products themselves, it replaces the data collection and enforcement plane underneath them.

eBPF displacing incumbent spend

Networking and network security

eBPF improves both performance and scalability of virtualized networks by letting packet processing, policy enforcement, and routing logic happen directly in the kernel. These realize hardware level performance gains at the software level that are otherwise unachievable with traditional virtual, cloud, and container networking constructs.

The traditional network security and packet filtering stack (hardware firewalls, web application firewalls, CASB, SWG, DDoS protection, and SASE appliances) was designed for a perimeter model that no longer reflects how enterprise traffic flows. eBPF can process packets before they traverse the full network stack, delivering throughput improvements that make software-defined packet filtering competitive with dedicated hardware appliances for the first time. Cloudflare's deployment of eBPF-based filtering to handle terabit-scale DDoS attacks is the most public proof point and is now being productized by startups to eat into legacy firewall market share.

- **Network security market size was valued at USD 28.58 billion in 2025.** This number includes the following:
    - **The global WAF market size was valued at USD 8.60 billion in 2025**
    - **DDoS protection & mitigation security market valued at USD 5.80 billion in 2025**
- **SASE market estimated at USD 19.19 billion in 2026**

Runtime Security

eBPF replaces legacy security vendors that rely on easily evaded, resource-heavy user-space agents. It provides enforcement and visibility via dynamic, secure, low-overhead, and efficient programs that run directly in the kernel.

Most security markets are moving away from posture management approaches in favor of runtime enforcement focused on detection and response. This includes both a re-imagining of EDR/XDR products, as well as AI-for-security and Security-for-AI products. Some examples of categories that heavily use eBPF for runtime security are cloud detection and response (CDR) and application detection and response (ADR). eBPF allows these vendors to secure workloads across the whole stack, including networks, processes, identities, and supply chain.

- **Cloud workload protection market size is estimated at $9 billion in 2025**
- **XDR market size is estimated at USD 7.92 billion in 2025**

Observability

Observability consists of a wide range of categories, from APM to cloud cost management and runtime security which all rely on data collection. This observability infrastructure has historically required a choice between depth and performance. High quality sensors meant a lot of agent overhead; lightweight agents meant blind spots, and agentless collection relies on telemetry from the cloud provider.

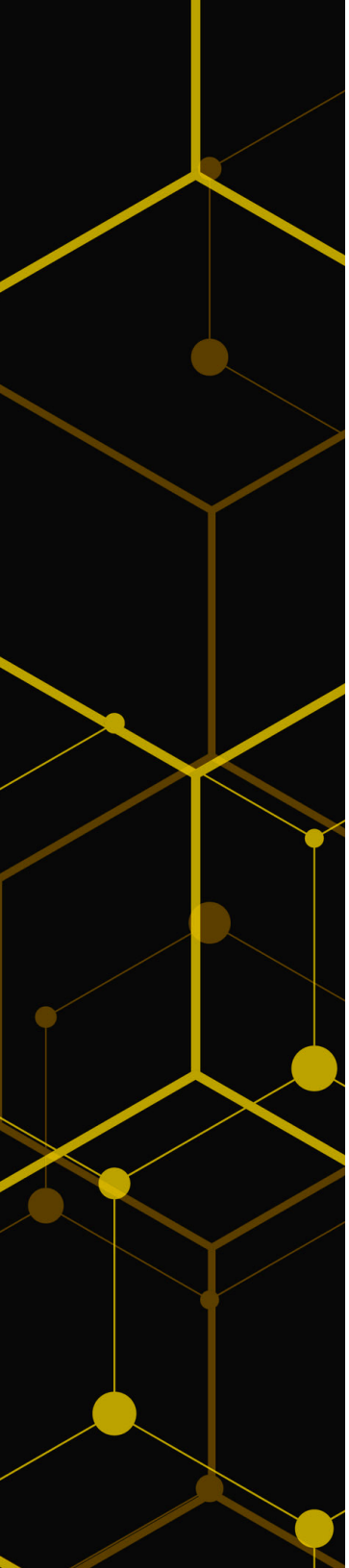
eBPF can collect high-fidelity data from the lowest level in the software stack. It gathers kernel-level data such as distributed traces, service mappings, protocol-specific usage without having to trade fidelity over performance overhead.

- **Application Performance Management Market estimated at USD 14.30 billion in 2026**
- **Cloud Monitoring Market estimated at USD 3.75 billion in 2025**

Hardware appliances

eBPF is becoming natively integrated in hardware appliances, as illustrated by **Cisco shipping Nexus switches with Tetragon**. eBPF is emerging as the universal instruction set to program silicon accelerators like SmartNICs and DPUs, hardware appliances, including CPU/GPU servers, storage arrays, IoT gateways, and devices. This means the TAM for an eBPF-native startup extends to every connected device and piece of silicon that runs a modern operating system.

- **Enterprise networking market size estimated at $86.89 billion in 2025**
- **Data processing unit (DPU) market, valued at USD 4.5 Billion in 2026**
- **Smart NIC Market is estimated to be valued at USD 1.03 Billion in 2026**



eBPF for greenfield generative AI infrastructure

eBPF is becoming the go-to approach for emerging AI infrastructure products. The examples listed below serve as examples for what is possible, but are a very small subset of the overall potential.

- **Agent monitoring** - eBPF's Kernel-level capture can observe where actions execute, parse protocols to extract semantics, stitch events into chains, evaluate workflow-level policies.

- **Agentic sandboxes** - eBPF can be used to define kernel-level policies such as read-only root filesystem, seccomp profiles to block exploitable syscalls, resource limits, and network policies to restrict egress to approved destinations.

- **Agentic runtime security** - eBPF can trace the entire model stack, including interactions with third-party models and communication with vector databases to define runtime enforcement policies.

eBPF as an economic differentiator – the black ink in the OpEx spreadsheet

For digital enterprises, infrastructure spend (compute, network bandwidth, and storage) is the largest driver of cost of goods sold (COGS). So while scalability in the cloud is technologically "infinite", it's not also financially sustainable.

eBPF directly tackles cloud-related COGS by optimizing how software interacts with hardware at the lowest possible level. Traditionally, organizations work with what the hyperscaler gives them - an all-purpose virtual machine or container, and a marketplace of third party applications. With eBPF, organizations regain some of the control surrendered to the cloud provider.

eBPF-native platforms represent a rare class of enterprise software that can promise a hard, immediate financial ROI through COGS reduction. Organizations can reduce the absolute amount of compute used by architecting their products with eBPF.

Some documented examples include:

Compute

- **Datadog** **35%** CPU usage reduction
- **Meta** **20%** CPU cycles reduction
  5% throughput improvement
- **Seznam** **72x** CPU reduction

Network

- **Polar Signals** **50%** cross-zone cost reduction

Operational

- **LinkedIn** **70%** log volume reduction
- **DoorDash** **40%** memory usage improvement
  98% fewer restarts
  80% faster deployments

The technical levers driving margin expansion

To understand how eBPF achieves these unit economic improvements, look at the below examples of how eBPF works under the hood.

- **It can increase virtualized network throughput to match physical network throughput** - Both virtualization and containerization introduce networking overhead. To improve network throughput to match that of the host, organizations can replace the native networking implementation with eBPF-projects such as Netkit, which eliminates the software abstraction overhead.

- **It lowers packet filtering latency for high-volume traffic** - XDP, an eBPF-based high-performance network data path, hooks at the lowest level before the network stack for coarse packet filtering. This is suitable for high-volume traffic filtering such as DDoS protection. Cloudflare uses eBPF to protect against terabit-per-second scale attacks where incoming packets can increase 40x with only a 2x increase in CPU usage.

- **It can lower overhead associated with running sensors and agents** - While user-space agents come with a high performance tax, eBPF-based sensors can dramatically reduce overhead by collecting metrics, logs, and traces directly in kernel space without expensive system calls. Some eBPF-based agents and sensors can introduce as little as a 2% CPU load increase relative to baseline levels, and near-zero memory increase.

- **It is used for observability and optimization of CPU cycles** - eBPF enables continuous profiling and optimization by tracking CPU cycles, function call frequencies, and hot code paths in real-time, allowing developers to identify and optimize bottlenecks. eBPF programs can also implement custom scheduling policies and load balancing algorithms directly in the kernel, reducing context switches and improving CPU utilization for request processing. With this visibility, Meta observed 20% performance gains after changing a single character in their code.

Who develops using eBPF and what is their route to monetization?

The eBPF ecosystem is uniquely validated from the top down with early hyperscaler exploration and deployments, while also being pushed forward from the bottom up by startups and their creative use cases.

Understanding the players clarifies their monetization path:

Hyperscalers, like Meta, Google, and AWS, usually develop for internal use cases, benefiting from large-scale efficiency gains. Their monetization route is OpEx-related, where 1% performance improvement across millions of servers translates directly to millions in operational cost savings. Meta, for example, developed an eBPF-based profiling orchestrator that collects observability data out-of-process. A single-character change delivered 15,000 servers' worth of annual capacity savings

Large technology providers use eBPF to enhance their products with more advanced features to defend their market share and improve their gross margins. For example, Datadog's Universal Service Monitoring uses eBPF to automatically discover, map, and monitor services and dependencies without requiring customers to instrument any code. Similarly, CrowdStrike's Runtime Cloud Data Protection uses eBPF to detect and block unauthorized data movements in real time. These are considerable differentiators and retention mechanisms for large providers.

However, even the large technology providers need to invest in skills and research to develop these features. To fast-track their access to these capabilities, large providers often acquire startups explicitly for their eBPF knowledge.

Startups have built whole products and categories with an eBPF-native approach. These offer higher performance and a lower footprint compared to non-eBPF approaches. Their primary route to monetization often follows an open source-to-enterprise distribution approach. For example, a monetization path is the open core model where startups give away the eBPF sensor and monetize the control plane. eBPF kernel instrumentation and basic data collection are given away for free which drives frictionless, bottom-up adoption by developers. The commercial entity monetizes via subscription fees for the proprietary control plane which may include things like multi-cluster scaling, compliance reporting, role-based access control (RBAC), data retention, and enterprise support. Other monetization playbooks will continue to develop as the ecosystem evolves.

Open source as a competitive moat

In modern infrastructure software, proprietary, closed-source technology is often a barrier to adoption. eBPF is natively open source. While there are in-house and closed source commercial solutions, the ecosystem mainly consists of open source projects deployed across the whole enterprise spectrum.

The most successful startups in this space rely on an open source strategy to create a flywheel that is difficult for proprietary competitors to replicate. A widely-deployed OSS project accumulates production insights from thousands of organisations, generating a breadth of real-world bug reports, edge case exposure, and contributor improvements that no internal engineering team can match. This dynamic builds impenetrable distribution moats with efficient go to market motions and broad customer and market feedback.

Cilium, for example, is now deployed as the default CNI in major managed Kubernetes offerings including Google GKE, Amazon EKS, and Azure AKS. This is a distribution footprint that Isovalent was unlikely to achieve through a purely commercial go to market motion. A widely deployed OSS project also accumulates production insights from thousands of organizations with a breadth of real-world stress testing, edge-case exposure, and community-driven feature improvements that no proprietary internal engineering team could afford to match. That installed base is what made the Cisco acquisition strategically compelling and helped justify a premium multiple.

Investors should investigate whether a target company holds a structurally advantageous position within an open source project. These include primary maintainership, committer status, roadmap influence, and the organisational credibility that comes from being the team that built the project and ecosystem around it. The following table represent the most significant open source projects in the eBPF ecosystem:

PROJECT	CATEGORY	KEY MAINTAINER(S)	COMMERCIAL DERIVATIVE
Cilium	Networking / Security	Isovalent (Cisco), Multi-vendor	Isovalent Enterprise Platform
Falco	Runtime Security	Sysdig + multi-vendor	Sysdig Platform
Pixie	Observability	New Relic	New Relic integration
Tracee	Runtime Security / Forensics	Aqua Security	Aqua Platform
Pyroscope	Continuous Profiling	Grafana Labs	Grafana Cloud Profiles
Inspektor Gadget	Debugging / Observability	ARMO / Microsoft	ARMO Platform (Kubescape)
Kubescape	Kubernetes Security	ARMO	ARMO Platform
OpenTelemetry	Telemetry Framework	CNCF / multi-vendor	Splunk, Datadog, Honeycomb distributions
Kubearmor	Runtime Security	AccuKnox	AccuKnox Platform

Visit **https://ebpf.io/applications/** for a more comprehensive list of projects.

LF Research's 2025 study of 800 VC-backed OSS companies found that community health metrics correlate directly with company valuations. For due diligence purposes, metrics such as contributor diversity, maintainer concentration, and issue cadence are more predictive of whether an open source moat is durable.

Investors can also call upon the OpenSSF Criticality Score, a composite of contributor diversity, commit frequency, dependent projects, and issue activity, as a predictive aggregate of company valuation. Investors conducting deeper diligence on eBPF-native targets should pull Criticality Scores via the OpenSSF dashboard as a screening signal.

M&A landscape and exit examples

The M&A landscape for eBPF reveals a highly liquid ecosystem driven by forced consolidation. Legacy infrastructure and security providers are burdened by user-space technical debt. To remain competitive, they are paying strategic premiums for eBPF-native startups. These acquisitions are rarely just about buying revenue. They are defensive maneuvers to leapfrog multi-year internal R&D cycles and capture scarce, highly specialized kernel-engineering talent.

The exit landscape can be mapped across three distinct waves of market maturity:

Wave 1 (2020–2022): Feature & Sensor Upgrades

The first wave of acquisitions consisted of performance-based observability point solution tools. For example, New Relic acquiring Pixie (2020), Elastic acquiring Optimyze (2021), and Datadog acquiring Seekret (2022). These startups offered an improved way of delivering application and infrastructure monitoring. For the first time, observability vendors were able to go beyond native hardware or cloud telemetry without using resource-intensive user-space sensors. The eBPF acquisition chain of events started when established monitoring vendors recognized that eBPF-based approaches offered fundamentally superior telemetry collection with lower overhead, no instrumentation requirements, and kernel-level fidelity.

Wave 2 (2023–2024): Platform & Community Land Grabs

The second wave shifted toward combining application logic with new data sources. Instead of using eBPF solely for collecting metrics in the context of APM, the kernel-level data can be used to build correlations for both security and performance monitoring. For example, Snyk acquired Helios' (2023) full-stack runtime data collection to integrate it with the wider Snyk Developer Security Platform. CrowdStrike also acquired Flow Security's eBPF-based DSPM product to integrate it into Falcon Cloud Security as part of the Falcon XDR platform.

Cisco's acquisition of Isovalent (2024) is perhaps the landmark eBPF deal. On the surface, Cisco acquired a networking, observability, and security platform. However, the wider implication of Isovalent's work also included the most successful open source eBPF project, Cilium, which is native in all hyperscalers, and some of the founders of eBPF itself.

Wave 3 (2025–Present): AI & Runtime Security Consolidation

The third wave consists of runtime and AI security. Zscaler acquiring Red Canary (2025), Palo Alto Networks acquiring Protect AI (2025), SentinelOne acquiring Prompt Security (2025), Cyera acquiring Otterize (2025), and Google finally acquiring Wiz (2026), all point to a market in which incumbent security platforms have concluded that eBPF-native runtime visibility and enforcement is a must-have capability, with acquisition being the fastest path to both market and talent. These are not simply improvements of otherwise existing products, but rather capabilities newly unlocked by eBPF.

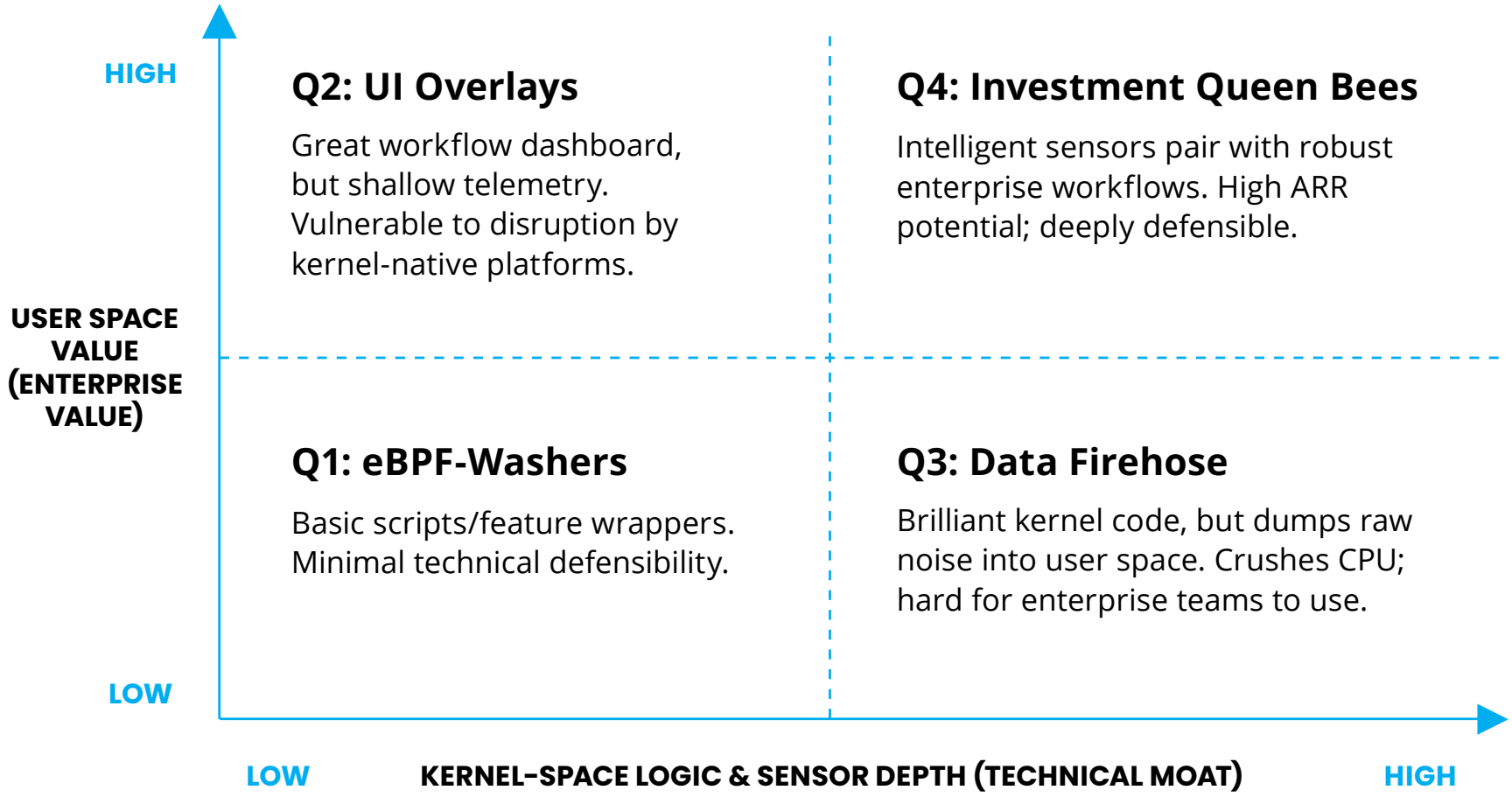
The Platform Phenomenon

Notably, the ecosystem has matured to the point where eBPF-native companies are now acquiring other eBPF startups. For example, Upwind, who, despite already having a comprehensive eBPF-based runtime solution, have acquired Nyx (2025). Nyx's solution is also eBPF-based and detects exploited vulnerabilities and behavioral anomalies.

This is a clear indicator that using eBPF is not just a binary Yes/No. The kernel instrumentation and application logic complement each other across deployments. **Platforms that have already established a strong eBPF foundation will aggressively acquire adjacent eBPF IP to expand their TAM, driving further liquidity in the market.**

Wave 1	2020	New Relic acquires Pixie	Observability
		Splunk acquires Flowmill	Observability
	2021	Red Hat acquires StackRox	Security
		Elastic acquires Cmd	Security
		Elastic acquires Optimyze	Observability
		Microsoft acquires Kinvolk	Observability
	2022	Datadog acquires Seekret	Observability
Wave 2	2023	Snyk acquires Helios	Security, Observability
		Postman acquires Akita	Observability, Security
		Grafana acquires Pyroscope	Observability
	2024	Cisco acquires Isovalent	Networking, Security, Observability
		SUSE acquires StackState	Observability
		CrowdStrike acquires Flow Security	Security
		Cisco acquires Splunk	Observability, Security
Wave 3	2025	Cyera acquires Otterize	Security
		Harness acquires Traceable	Security, Observability
		Upwind acquires Nyx	Security
		Zscaler acquires Red Canary	Security
		F5 acquires MantisNet	Networking, Observability
		Palo Alto Networks acquires Protect AI	AI security
		SentinelOne acquires Prompt Security	AI security
		A10 Networks acquires ThreatX Protect	Networking, Security
	2026	Google acquires Wiz	Security

eBPF due diligence for investors



eBPF due diligence for investors

As eBPF becomes table stakes in infrastructure and security, investors face a growing risk of "eBPF-washing." Simply using eBPF to collect some kernel-level data does not make a product better or create a competitive moat. The barrier to entry for writing a basic eBPF program has fallen dramatically and as always the true value lies in how that technology is architected at scale.

There are two points that can help differentiate between simplistic and comprehensive implementations. Sensor quality, or the actual eBPF program operating in kernel-space, and application logic, which is defined in user space.

- **Sensor quality (the kernel space moat)** - A good eBPF implementation needs to instrument the kernel at different points and capture a wide range of data without shipping everything to user space and overwhelming the CPU. It needs to perform kernel-level filtering based on allow/block rules, perform data extraction such as syscall arguments and network packet headers, and aggregate and count frequency of events. Pre-processed data can be then added to eBPF maps which makes it available to user space where it can be further analyzed by the rest of the application.
- **Application logic (the user space value capture)** - Raw system data is useless without context and actionable workflows. Application logic using data collected from the kernel should include correlation across multiple event types, enrichment with metadata and external data sources, behavioral analysis, communication with control planes, policy management and rule updates, and an alert generation system that a non-kernel engineer can easily operate.

When evaluating a pitch deck claiming eBPF capabilities, carefully consider how the founding team is capturing customer value in both kernel space with more intelligent eBPF programs and user space with a more automated user experience.

Outlook on the eBPF market

As eBPF enters its next phase of maturity, **several structural market shifts are accelerating its adoption and expanding the addressable market for venture backed startups**. For investors, the historical constraints of eBPF are rapidly dissolving, replaced by massive go to market tailwinds.

Increasing appetite for eBPF-based efficiency gains

The economics of infrastructure are shifting in eBPF's favor. Hardware costs are rising, GPU capacity is constrained, and cloud spend is a shared CFO/CTO concern. In this environment, the efficiency gains that eBPF delivers are investment justifications, as observed at Meta, Datadog, and DoorDash.

The appetite is especially acute for AI infrastructure operators running GPU clusters at scale, as well as AI consumers who require performance and security. For operators, network throughput and scheduling efficiency directly determine utilisation rates and therefore the cost per inference or training run. eBPF-based networking (Netkit replacing veth, XDP-based load balancing) and custom kernel schedulers (sched_ext) are increasingly being evaluated as cost reduction levers with quantifiable ROI.

Top-down and bottom-up GTM convergence

eBPF has historically been a technology known to kernel engineers and a small community of infrastructure specialists. That is changing rapidly and the commercial implications are significant. Awareness is now expanding across three audiences simultaneously.

Platform and DevOps engineers are encountering eBPF through the tools they already use. Cilium is the default CNI across major cloud providers and neo and sovereign clouds. OpenTelemetry is a consistent cross-vendor sensor. Engineers who deploy these platforms are increasingly curious about the underlying technology - a dynamic that creates bottom-up enterprise demand without requiring vendors to educate the market from scratch.

Security buyers are being introduced to eBPF through their incumbent vendors. CrowdStrike, SentinelOne, and Zscaler have all either acquired eBPF-native companies or incorporated eBPF capabilities into their marketing messaging in the past 24 months. When category-defining security platforms begin positioning eBPF as a differentiator, they educate the CISO buyer population at scale.

The developer talent pool is the third and most structurally important awareness vector. **The eBPF Foundation programs, a growing academic publication record, and CNCF-backed educational resources are producing engineers who treat eBPF as a first-class tool rather than an advanced specialisation.** This matters for investment because TAM in infrastructure software is frequently constrained not by buyer demand but by the availability of engineers capable of deploying and maintaining it. The talent pool has shifted from a niche specialty to a mainstream cloud native skillset, drastically lowering the R&D execution risk for early-stage companies.

Expansion beyond the Linux kernel

A commonly cited limitation of eBPF is that it is only scoped for the Linux kernel. While Linux was first to implement this mechanism for instrumenting the kernel, projects are expanding the scope to other operating systems.

The **eBPF for Windows** open source project allows existing eBPF toolchains and APIs familiar in the Linux ecosystem to be used on top of Windows. Vendors who have developed runtime security for Linux using eBPF are expanding to provide similar capabilities for other operating systems. For example, **Wiz** has recently released a Windows-compatible version of their runtime security sensor. This cross-platform evolution effectively doubles the endpoint security TAM and allows startups to sell into mixed-OS enterprise environments.

Why 2026 is the right time to invest in eBPF

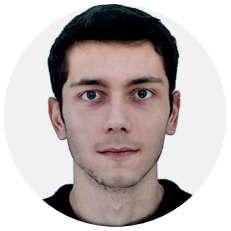
The venture capital window for eBPF is currently in its most lucrative phase. While early products were exploring how to provide performance improvements in existing product categories, better virtualized networks, higher fidelity data, smaller sensor footprint, new startups are developing complex application logic which would be unachievable without eBPF.

Wave 1 acquisitions were paying for proof-of-concept point solutions. For example, superior observability sensors, better profiling tools, or higher-fidelity network telemetry. The multiples reflected genuine but bounded value. eBPF as a capability upgrade to a specific product category.

The 2026 infrastructure problems around GPU scaling, real-time inference, supply chain security, zero day exploits, agent identities, semantic and intent analysis, are all being tackled with eBPF. Legacy user space solutions are structurally incapable of securing, observing, and routing these next-generation workloads. eBPF is the only viable kernel-level control plane that can deliver the required performance, visibility, and security.

> **The companies that dominate the eBPF ecosystem today will own the foundational data path of the AI and cloud native eras.**

Refer to the ebpf.foundation landscape for a current view of the commercial ecosystem.



About the Author

ANDREW GREEN

Andrew Green is an enterprise IT research analyst and writer covering security, networking, and infrastructure. Lately, he's been deciphering the AI market and its impact on enterprise products. Andrew has been a GigaOm analyst since 2020. He also works with technology vendors to produce technical content, competitive analysis, and market landscape assessments, and shares independent and non-sponsored content on Substack and LinkedIn.

The **eBPF Foundation** was created to advance eBPF as an open, shared technology for programmable infrastructure. It brings together a cross-platform community of maintainers and organizations working upstream to evolve eBPF's capabilities while ensuring its safety, security, and performance. Foundation members collaborate on common technical priorities, security best practices, community development, and promotional opportunities supporting eBPF across kernels, operating systems, and enterprise environments. Find further information here: https://www.ebpf.foundation