META'S STROBELIGHT LEVERAGES EBPF TO REDUCE CPU CYCLES AND SERVER DEMANDS BY UP TO 20%

OVERVIEW

Meta, a global leader in technology and social networking, faced the challenge of providing comprehensive profiling data for its backend services without disrupting performance. To address this, Meta developed Strobelight, a profiling orchestrator that leverages eBPF to collect observability data efficiently. This solution has driven measurable efficiency improvements across Meta's infrastructure, resulting in substantial capacity savings and operational benefits.

CHALLENGE

Meta needed a way to gather and normalize profiling data across its vast and varied backend services without introducing overhead that could impact performance. The key challenges included:

- Ensuring minimal disruption to live services while collecting performance data.
- Making profiling data uniform and easily interpretable.
- Preventing system overloads due to excessive data storage demands.
- Supporting multiple kernel versions across Meta's infrastructure.

SOLUTION

To overcome these challenges, Meta implemented Strobelight, a sophisticated profiling orchestrator that integrates multiple profiling tools, including eBPF. Strobelight enables engineers to collect detailed observability data out-of-process, covering:

- CPU time spent in function calls and execution paths.
- Call stacks for native and non-native languages (e.g., Python, Java, Erlang).
- Off-CPU time and service request latency analysis.
- AI/GPU profiling and memory tracking.

Strobelight's eBPF-powered profiling capabilities provide low-overhead data collection, avoiding additional instrumentation inside binaries and maintaining efficient performance. Additionally, given Meta's diverse infrastructure with multiple kernel versions, Strobelight was designed to ensure:

- Feature compatibility across different kernel versions, with appropriate fallbacks.
- Dynamic sampling to balance data collection rates and storage efficiency.
- Concurrency and queuing safeguards to prevent performance degradation.

RESULTS

The deployment of eBPF within Strobelight has led to significant efficiency gains, including:

- 15,000 servers' worth of annual capacity savings from a single one-character code change.
- 20% reduction in CPU cycles, equating to a 10-20% reduction in the number of required servers for Meta's top services.
- Faster debugging and performance analysis, allowing engineers to prevent regressions before they reach production.
- Dynamic sampling mechanisms, optimizing profiling rates without overloading storage systems.

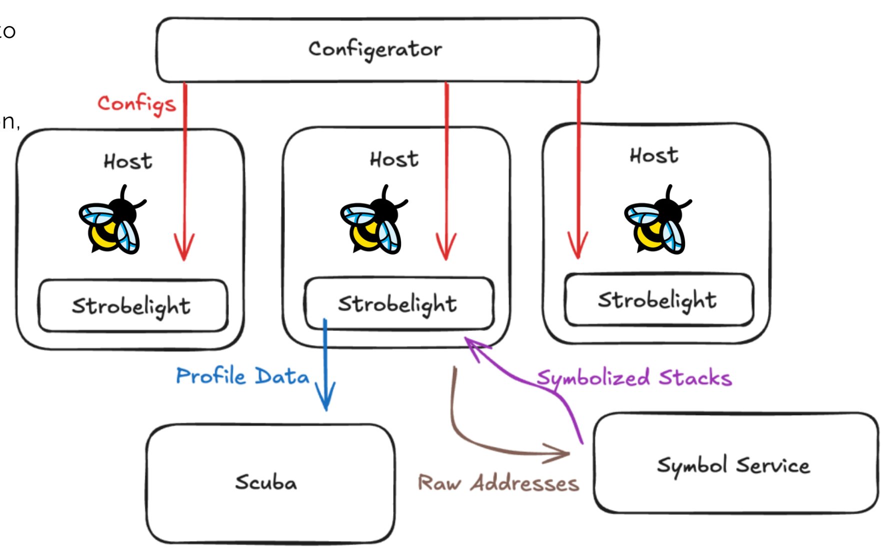
WHY EBPF?

eBPF was selected as the core technology due to its low overhead, which ensured minimal impact on targeted processes; flexibility, with numerous attach points and built-in kernel helpers; and lack of requirements for additional instrumentation, simplifying deployment.
Additionally, unlike traditional profiling tools that require added instrumentation inside binaries and impact runtime performance, Strobelight's eBPF-based approach enabled real-time profiling without modifying application code, broader observability across multiple languages and systems, and efficient data collection with minimal overhead.

NEXT STEPS

Meta continues to expand its use of eBPF to further enhance observability, particularly in:

- AI/ML workloads.
- Advanced memory tracking.
- More complex efficiency analyses for improved resource utilization.
- Open sourcing Strobelight's profilers and libraries for broader adoption within the open source community.



A simplified Strobelight service graph.