eBPF

eBPF is a technology that can run programs in a privileged context such as the operating system kernel.[5] It is the successor to the Berkeley Packet Filter (BPF, with the "e" originally meaning "extended") filtering mechanism in Linux and is also used in non-networking parts of the Linux kernel as well.

eBPF	
Original author(s)	Alexei Starovoitov, Daniel Borkmann[1][2]
Developer(s)	Open source community, Meta, Google, Isovalent, Microsoft, Netflix[1]
Initial release	2014[3]
Repository	Linux: git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/ (https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/) Windows: github.com/Microsoft/ebpf-for-windows/ (https://github.com/Microsoft/ebpf-for-windows/)
Written in	C
Operating system	Linux, Windows[4]
Type	Runtime system
License	Linux: GPL Windows: MIT License
Website	ebpf.io (https://www.ebpf.io/)

It is used to safely and efficiently extend the capabilities of the kernel at runtime without requiring changes to kernel source code or loading kernel modules.[6] Safety is provided through an in-kernel verifier which performs static code analysis and rejects programs which crash, hang or otherwise interfere with the kernel negatively.[7][8]

This validation model differs from sandboxed environments, where the execution environment is restricted and the runtime has no insight about the program.[9] Examples of programs that are automatically rejected are programs without strong exit guarantees (i.e. for/while loops without exit conditions) and programs dereferencing pointers without safety checks.[10]

Design

Loaded programs which passed the verifier are either interpreted or in-kernel just-in-time compiled (JIT compiled) for native execution performance. The execution model is event-driven and with few exceptions run-to-completion,[2] meaning, programs can be attached to various hook points in the operating system kernel and are run upon triggering of an event. eBPF use cases include (but are not limited to) networking such as XDP, tracing and security subsystems.[5] Given eBPF's efficiency and flexibility opened up new possibilities to solve production issues, Brendan Gregg famously dubbed eBPF "superpowers for Linux".[11] Linus Torvalds said, "BPF has actually been really useful, and the real power of it is how it allows people to do specialized code that isn't enabled until asked for".[12] Due to its success in Linux, the eBPF runtime has been ported to other operating systems such as Windows.[4]

History

eBPF evolved from the classic Berkeley Packet Filter (cBPF, a retroactively-applied name). At the most basic level, it introduced the use of ten 64-bit registers (instead of two 32-bit long registers for cBPF), different jump semantics, a call instruction and corresponding register passing convention, new instructions, and a different encoding for these instructions.[13]

Most significant milestones in the evolution of eBPF

Date	Event
April 2011	The first in-kernel Linux just-in-time compiler (JIT compiler) for the classic Berkeley Packet Filter got merged.[14]
January 2012	The first non-networking use case of the classic Berkeley Packet Filter, seccomp-bpf,[15] appeared; it allows filtering of system calls using a configurable policy implemented through BPF instructions.
March 2014	David S. Miller, primary maintainer of the Linux networking stack, accepted the rework of the old in-kernel BPF interpreter. It was replaced by an eBPF interpreter and the Linux kernel internally translates classic BPF (cBPF) into eBPF instructions.[16] It was released in version 3.18 of the Linux kernel.[17]
March 2015	The ability to attach eBPF to kprobes as first tracing use case was merged.[19] In the same month, initial infrastructure work got accepted to attach eBPF to the networking traffic control (tc) layer allowing to attach eBPF to the core ingress and later also egress paths of the network stack, later heavily used by projects such as Cilium.[20][21][22]
August 2015	The eBPF compiler backend got merged into LLVM 3.7.0 release.[23]
September 2015	Brendan Gregg announced a collection of new eBPF-based tracing tools as the bcc project, providing a front-end for eBPF to make it easier to write programs.[24]
July 2016	eBPF got the ability to be attached into network driver's core receive path. This layer is known today as eXpress DataPath (XDP) and was added as a response to DPDK to create a fast data path which works in combination with the Linux kernel rather than bypassing it.[25][26][27]
August 2016	Cilium was initially announced during LinuxCon as a project providing fast IPv6 container networking with eBPF and XDP. Today, Cilium has been adopted by major cloud provider's Kubernetes offerings and is one of the most widely used CNIs.[28][22][29]
November 2016	Netronome added offload of eBPF programs for XDP and tc BPF layer to their NIC.[30]
May 2017	Meta's layer 4 load-balancer, Katran, went live. Every packet towards facebook.com since then has been processed by eBPF & XDP.[31]
November 2017	eBPF becomes its own kernel subsystem to ease the continuously growing kernel patch management. The first pull request by eBPF maintainers was submitted.[32]
September 2017	Bpftool was added to the Linux kernel as a user space utility to introspect the eBPF subsystem.[33]
January 2018	A new socket family called AF_XDP was published, allowing for high performance packet processing with zero-copy semantics at the XDP layer.[34] Today, DPDK has an official AF_XDP poll-mode driver support.[35]
February 2018	The bpfilter prototype has been published, allowing translation of a subset of iptables rulesets into eBPF via a newly developed user mode driver. The work has caused controversies due to the ongoing nftables development effort and has not been merged into mainline.[36][37]
October 2018	The new bpftrace tool has been announced by Brendan Gregg as DTrace 2.0 for Linux.[38]
November 2018	eBPF introspection has been added for kTLS in order to support the ability for in-kernel TLS policy enforcement.[39]
November 2018	BTF (BPF Type Format) has been added to the Linux kernel as an efficient meta data format which is approximately 100x smaller in size than DWARF.[40]
December 2019	The first 880-page long book on BPF, written by Brendan Gregg, was released.[41]

March 2020	Google upstreamed BPF LSM support into the Linux kernel, enabling programmable Linux Security Modules (LSMs) through eBPF.[42]
September 2020	The eBPF compiler backend for GNU Compiler Collection (GCC) was merged.[43]
July 2022	Microsoft released eBPF for Windows, which runs code in the NT kernel.[4]

Branding

The alias eBPF is often interchangeably used with BPF,[2][44] for example by the Linux kernel community. eBPF and BPF is referred to as a technology name like LLVM.[2] eBPF evolved from the Berkeley Packet Filter as an extended version, but as its use cases outgrew networking, today "eBPF" is preferentially interpreted as a pseudo-acronym.[2]

The bee is the official logo for eBPF. At the first eBPF Summit there was a vote taken and the bee mascot was named "eBee".[45][46] The logo has originally been created by Vadim Shchekoldin.[46] Earlier unofficial eBPF mascots have existed in the past,[47] but have not seen widespread adoption.

Governance

The eBPF Foundation was created in August 2021 with the goal to expand the contributions being made to extend the powerful capabilities of eBPF and grow beyond Linux.[1] Founding members include Meta, Google, Isovalent, Microsoft and Netflix. The purpose is to raise, budget and spend funds in support of various open source, open data and/or open standards projects relating to eBPF technologies[48] to further drive the growth and adoption of the eBPF ecosystem. Since inception, Red Hat, Huawei, Crowdstrike, Tigera, DaoCloud, Datoms, FutureWei also joined.[49]

Adoption

eBPF has been adopted by a number of large-scale production users, for example:

- Meta uses eBPF through their Katran layer 4 load-balancer for all traffic going to facebook.com[50][51][52][31]
- Google uses eBPF in GKE, developed and uses BPF LSM to replace audit and it uses eBPF for networking[29][53][54][55]
- Cloudflare uses eBPF for load-balancing and DDoS protection and security enforcement[56][57][58][59][60]
- Netflix uses eBPF for fleet-wide network observability and performance diagnosis[61][62]
- Dropbox uses eBPF through Katran for layer 4 load-balancing[63]
- Android uses eBPF for NAT46 and traffic monitoring[64][65][66]
- Samsung Galaxy uses eBPF for Networking solutions [67]

- Yahoo! Inc uses eBPF through Cilium for layer 4 load balancing[68]
- LinkedIn uses eBPF for infrastructure observability[69]
- Alibaba uses eBPF for Kubernetes Pod load-balancing[70]
- Datadog uses eBPF for Kubernetes Pod networking and security enforcement[71][72][73]
- Trip.com uses eBPF for Kubernetes Pod networking[74][75]
- Shopify uses eBPF for intrusion detection through Falco[76]
- DoorDash uses eBPF through BPFAgent for kernel level monitoring[77]
- Microsoft ported eBPF and XDP to Windows[78][79][80]
- Seznam uses eBPF through Cilium for layer 4 load-balancing[81]
- DigitalOcean uses eBPF and XDP to rate limit access to internal services in their virtual network[82]
- CapitalOne uses eBPF for Kubernetes Pod networking[83]
- Bell Canada uses eBPF to moderize telco networking with SRv6[84]
- Elastic_NV uses eBPF for code profiling as part of their observability offering [85]
- Apple uses eBPF for Kubernetes Pod security[86]
- Sky uses eBPF for Kubernetes Pod networking[87]
- Walmart uses eBPF for layer 4 load-balancing[88][89]
- Huawei uses eBPF through their DIGLIM secure boot system[90]
- Ikea uses eBPF for Kubernetes Pod networking[91]
- The New York Times uses eBPF for networking[92]
- Red Hat uses eBPF at scale for load balancing and tracing in their private cloud
- Palantir Technologies uses eBPF to debug networking problems in large scale Kubernetes clusters[93]

Security

Due to the ease of programmability, eBPF has been used as a tool for implementing microarchitectural timing side-channel attacks such as Spectre against vulnerable microprocessors.[94] While unprivileged eBPF implemented mitigations against transient execution attacks,[95] unprivileged use has ultimately been disabled by the kernel community by default to protect from use against future hardware vulnerabilities.[96]

See also

- Express Data Path

References

1. "Meta, Google, Isovalent, Microsoft and Netflix Launch eBPF Foundation as Part of the Linux Foundation" (https://www.linuxfoundation.org/press-release/facebook-google-isovalent-microsoft-and-netflix-launch-ebpf-foundation-as-part-of-the-linux-foundation/) . *Linux Foundation*. 12 August 2021. Retrieved 1 July 2022.
2. "BPF Internals" (https://www.usenix.org/conference/lisa21/presentation/gregg-bpf) . *USENIX LISA 2021 conference*. 1 June 2021. Retrieved 1 July 2022.
3. "eBPF and Kubernetes: Little Helper Minions for Scaling Microservices" (https://kccnceu20.sched.com/event/ZemQ/ebpf-and-kubernetes-little-helper-minions-for-scaling-microservices-daniel-borkmann-cilium) . *CNCF KubeCon + CloudNativeCon Europe 2020*. 19 August 2020. Retrieved 1 July 2022.
4. "Making eBPF work on Windows" (https://cloudblogs.microsoft.com/opensource/2021/05/10/making-ebpf-work-on-windows/) . *Microsoft Open Source Blog*. 10 May 2021. Retrieved 1 July 2022.
5. "eBPF Documentation: What is eBPF?" (https://ebpf.io/what-is-ebpf) . *eBPF.io*. Retrieved 1 July 2022.
6. "eBPF - Rethinking the Linux Kernel" (https://www.infoq.com/presentations/facebook-google-bpf-linux-kernel/) . *QCon 2020*. Retrieved 1 July 2022.
7. "Safe Programs The Foundation of BPF" (https://www.youtube.com/watch?v=AV8xY318rtc) . *eBPF Summit 2021*. 8 November 2020. Retrieved 1 July 2022.
8. "BPF and Spectre: Mitigating transient execution attacks" (https://popl22.sigplan.org/details/prisc-2022-papers/11/BPF-and-Spectre-Mitigating-transient-execution-attacks) . *POPL 2022 conference*. 22 January 2022. Retrieved 1 July 2022.
9. "eBPF - The Silent Platform Revolution from Cloud Native" (https://conferences.sigcomm.org/sigcomm/2023/files/workshop-ebpf/1-CloudNative.pdf#page=20) (PDF). *SIGCOMM 2023, 1st Workshop on eBPF and Kernel Extensions*. 10 September 2023. Retrieved 5 October 2023.
10. Hedam, Niclas (26 May 2023). "eBPF - From a Programmer's Perspective" (https://hed.am/papers/2021-EBPF.pdf) (PDF). doi:10.13140/RG.2.2.33688.11529/4 (https://doi.org/10.13140%2FRG.2.2.33688.11529%2F4) .
11. "Linux BPF Superpowers" (https://www.brendangregg.com/blog/2016-03-05/linux-bpf-superpowers.html) . *Brendan Gregg's Blog*. 5 March 2016. Retrieved 1 July 2022.
12. "Linus Torvalds talks about coming back to work on Linux" (https://www.zdnet.com/article/linus-torvalds-talks-about-coming-back-to-work-on-linux/) . *zdnet Interview with Linus Torvalds*. 23 October 2018. Retrieved 1 July 2022.
13. "Classic BPF vs eBPF" (https://www.kernel.org/doc/html/v6.1/bpf/classic_vs_extended.html) . *LWN*. March 2014. Retrieved 6 January 2023.
14. "net: filter: Just In Time compiler" (https://lore.kernel.org/netdev/1301838968.2837.200.camel@edumazet-laptop/) . *lore.kernel.org*. April 2011. Retrieved 1 July 2022.
15. "Yet another new approach to seccomp" (https://lwn.net/Articles/475043/) . *LWN*. 1 January 2012. Retrieved 1 July 2022.
16. "BPF updates" (https://lore.kernel.org/netdev/1396029506-16776-1-git-send-email-dborkman@redhat.com/) . *lore.kernel.org*. March 2014. Retrieved 1 July 2022.

17. "Linux kernel 3.18, Section 1.3. bpf() syscall for eBFP virtual machine programs" (https://kernelnewbies.org/Linux_3.18#bpf.28.29_syscall_for_eBFP_virtual_machine_programs) . *kernelnewbies.org*. December 7, 2014. Retrieved September 6, 2019.
18. "Happy birthday BPF!" (https://lore.kernel.org/bpf/20210926203409.kn3gzz2eaodflels@ast-mbp.dhcp.thefacebook.com/) . *lore.kernel.org*. September 2014. Retrieved 1 July 2022.
19. "tracing: attach eBPF programs to kprobes" (https://lore.kernel.org/netdev/1425252465-27527-1-git-send-email-ast@plumgrid.com/) . *lore.kernel.org*. March 2015. Retrieved 1 July 2022.
20. "eBPF support for cls_bpf" (https://lore.kernel.org/netdev/cover.1425208501.git.daniel@iogearbox.net/) . *lore.kernel.org*. March 2015. Retrieved 1 July 2022.
21. "net, sched: add clsact qdisc" (https://lore.kernel.org/netdev/61198814638d88ce3555dbecf8ef875523b95743.1452197856.git.daniel@iogearbox.net/) . *lore.kernel.org*. January 2016. Retrieved 1 July 2022.
22. "eBPF-based Networking, Observability, Security" (https://cilium.io/) . *cilium.io*. January 2016. Retrieved 1 July 2022.
23. "LLVM 3.7 Release Notes" (https://releases.llvm.org/3.7.0/docs/ReleaseNotes.html#non-comprehensive-list-of-changes-in-this-release) . *releases.llvm.org*. August 2015. Retrieved 1 July 2022.
24. "bcc: Taming Linux 4.3+ Tracing Superpowers" (https://www.brendangregg.com/blog/2015-09-22/bcc-linux-4.3-tracing.html) . *brendangregg.com*. September 2015. Retrieved 1 July 2022.
25. "Add driver bpf hook for early packet drop and forwarding" (https://lore.kernel.org/netdev/1468955817-10604-1-git-send-email-bblanco@plumgrid.com/) . *lore.kernel.org*. July 2016. Retrieved 1 July 2022.
26. "eCHO episode 9: XDP and Load Balancing" (https://www.youtube.com/watch?v=OIyPm6K4ooY) . *youtube.com*. June 2021. Retrieved 1 July 2022.
27. Høiland-Jørgensen, Toke; Brouer, Jesper Dangaard; Borkmann, Daniel; Fastabend, John; Herbert, Tom; Ahern, David; Miller, David (December 2018). "The eXpress data path: Fast programmable packet processing in the operating system kernel". *Proceedings of the 14th International Conference on emerging Networking EXperiments and Technologies*. pp. 54–66. doi:10.1145/3281411.3281443 (https://doi.org/10.1145%2F3281411.3281443) . ISBN 9781450360807. S2CID 53779310 (https://api.semanticscholar.org/CorpusID:53779310) .
28. "Cilium - Fast IPv6 Container Networking with BPF and XDP" (https://www.slideshare.net/ThomasGraf5/cilium-fast-ipv6-container-networking-with-bpf-and-xdp) . *slideshare.net*. August 2016. Retrieved 1 July 2022.
29. "New GKE Dataplane V2 increases security and visibility for containers" (https://cloud.google.com/blog/products/containers-kubernetes/bringing-ebpf-and-cilium-to-google-kubernetes-engine) . *cloud.google.com*. May 2021. Retrieved 16 August 2022.
30. "nfp ring reconfiguration and XDP support" (https://lore.kernel.org/netdev/1478193129-23476-1-git-send-email-jakub.kicinski@netronome.com/) . *lore.kernel.org*. November 2016. Retrieved 1 July 2022.
31. "XDP 1.5 Years In Production. Evolution and Lessons Learned" (https://lpc.events/event/2/contributions/109/) . *lpc.events*. November 2018. Retrieved 16 August 2022.
32. "pull-request: bpf 2017-11-23" (https://lore.kernel.org/netdev/20171123120135.8371-1-daniel@iogearbox.net/) . *lore.kernel.org*. November 2017. Retrieved 1 July 2022.

33. "tools: add bpftool" (https://lore.kernel.org/netdev/20170926153522.31500-1-jakub.kicinski@netronome.com/) . *lore.kernel.org*. September 2017. Retrieved 1 July 2022.

34. "Introducing AF_XDP support" (https://lore.kernel.org/netdev/20180131135356.19134-1-bjorn.topel@gmail.com/) . *lore.kernel.org*. January 2018. Retrieved 1 July 2022.

35. "AF_XDP Poll Mode Driver" (https://doc.dpdk.org/guides/nics/af_xdp.html) . *doc.dpdk.org*. August 2022. Retrieved 16 August 2022.

36. "BPF comes to firewalls" (https://lwn.net/Articles/747551/) . *lwn.net*. February 2018. Retrieved 1 July 2022.

37. "Why is the kernel community replacing iptables with BPF?" (https://cilium.io/blog/2018/04/17/why-is-the-kernel-community-replacing-iptables/) . *cilium.io*. April 2018. Retrieved 1 July 2022.

38. "bpftrace (DTrace 2.0) for Linux 2018" (https://www.brendangregg.com/blog/2018-10-08/dtrace-for-linux-2018.html) . *brendangregg.com*. October 2018. Retrieved 16 August 2022.

39. "Combining kTLS and BPF for Introspection and Policy Enforcement" (http://vger.kernel.org/lpc_net2018_talks/ktls_bpf.pdf) (PDF). *vger.kernel.org*. November 2018. Retrieved 1 July 2022.

40. "BTF deduplication and Linux kernel BTF" (https://nakryiko.com/posts/btf-dedup/) . *nakryiko.com*. November 2018. Retrieved 1 July 2022.

41. "BPF Performance Tools (book)" (https://www.brendangregg.com/bpf-performance-tools-book.html) . *brendangregg.com*. December 2019. Retrieved 16 August 2022.

42. "MAC and Audit policy using eBPF (KRSI)" (https://lore.kernel.org/bpf/20200329004356.27286-1-kpsingh@chromium.org/) . *lore.kernel.org*. March 2020. Retrieved 16 August 2022.

43. "BPF in GCC" (https://lwn.net/Articles/831402/) . *lwn.net*. September 2020. Retrieved 16 August 2022.

44. Brendan Gregg (December 2019). *BPF Performance Tools*. Addison-Wesley. ISBN 978-0136554820.

45. "eBPF Summit Day Two" (https://cilium.io/blog/2020/10/29/ebpf-summit-day-2) . *cilium.io*. October 2020. Retrieved 1 July 2022.

46. "What is the bee named?" (https://ebpf.io/what-is-ebpf#what-is-the-bee-named) . *ebpf.io*. Retrieved 1 July 2022.

47. "eBPF: One Small Step" (https://www.brendangregg.com/blog/2015-05-15/ebpf-one-small-step.html) . *Brendan Gregg's Blog*. May 2015. Retrieved 1 July 2022.

48. "eBPF Foundation Charter" (https://ebpf.foundation/charter/) . *ebpf.foundation*. June 2021. Retrieved 16 August 2022.

49. "eBPF Foundation Governance" (https://ebpf.foundation/governance/) . *ebpf.foundation*. August 2022. Retrieved 16 August 2022.

50. "Open-sourcing Katran, a scalable network load balancer" (https://engineering.fb.com/2018/05/22/open-source/open-sourcing-katran-a-scalable-network-load-balancer/) . *fb.com*. May 2018. Retrieved 16 August 2022.

51. "BPF at Facebook" (https://www.youtube.com/watch?v=ZYBXZFKPS28) . *youtube.com*. December 2019. Retrieved 16 August 2022.

52. "From XDP to socket" (https://lpc.events/event/11/contributions/950/) . *lpc.events*. September 2021. Retrieved 16 August 2022.

53. "eCHO episode 29: BPF LSM with KP Singh" (https://www.youtube.com/watch?v=OBFYMBHrstI) . *youtube.com*. November 2021. Retrieved 16 August 2022.

54. "BPF security auditing at Google - Brendan Jackman/KP Singh" (https://www.youtube.com/watch?v=URm_q9ylxBk) . *youtube.com*. November 2021. Retrieved 16 August 2022.

55. "Replacing HTB with EDT and BPF" (https://legacy.netdevconf.info/0x14/session.html?talk-replacing-HTB-with-EDT-and-BPF) . *netdevconf.info*. July 2020. Retrieved 16 August 2022.

56. "Cloudflare architecture and how BPF eats the world" (https://blog.cloudflare.com/cloudflare-architecture-and-how-bpf-eats-the-world/) . *blog.cloudflare.com*. May 2019. Retrieved 16 August 2022.

57. "It's crowded in here!" (https://blog.cloudflare.com/its-crowded-in-here/) . *blog.cloudflare.com*. October 2019. Retrieved 16 August 2022.

58. "Production ready eBPF, or how we fixed the BSD socket API" (https://blog.cloudflare.com/tubular-fixing-the-socket-api-with-ebpf/) . *blog.cloudflare.com*. February 2022. Retrieved 16 August 2022.

59. "Live-patching security vulnerabilities inside the Linux kernel with eBPF Linux Security Module" (https://blog.cloudflare.com/live-patch-security-vulnerabilities-with-ebpf-lsm/) . *blog.cloudflare.com*. June 2022. Retrieved 16 August 2022.

60. "Unimog - Cloudflare's edge load balancer" (https://blog.cloudflare.com/unimog-cloudflares-edge-load-balancer/) . *blog.cloudflare.com*. September 2020. Retrieved 16 August 2022.

61. "How Netflix uses eBPF flow logs at scale for network insight" (https://netflixtechblog.com/how-netflix-uses-ebpf-flow-logs-at-scale-for-network-insight-e3ea997dca96) . *netflixtechblog.com*. June 2021. Retrieved 16 August 2022.

62. "Extending Vector with eBPF to inspect host and container performance" (https://netflixtechblog.com/extending-vector-with-ebpf-to-inspect-host-and-container-performance-5da3af4c584b) . *netflixtechblog.com*. February 2019. Retrieved 16 August 2022.

63. "Dropbox traffic infrastructure: Edge network" (https://dropbox.tech/infrastructure/dropbox-traffic-infrastructure-edge-network) . *dropbox.tech*. October 2018. Retrieved 16 August 2022.

64. "eBPF Traffic Monitoring" (https://source.android.com/docs/core/datausage/ebpf-traffic-monitor) . *source.android.com*. August 2022. Retrieved 16 August 2022.

65. "Extending the Kernel with eBPF" (https://source.android.com/docs/core/architecture/kernel/bpf) . *source.android.com*. August 2022. Retrieved 16 August 2022.

66. "NAT46 translation with BPF" (https://lore.kernel.org/bpf/20220407084727.10241-1-lina.wang@mediatek.com/) . *lore.kernel.org*. April 2022. Retrieved 16 August 2022.

67. "BPF for Android: How we leverage BPF for our networking solutions - Madhan Raj Kanagarathinam" (https://www.youtube.com/watch?v=unnVlwyevmU) . *www.youtube.com*. February 2024. Retrieved 19 February 2022.

68. *Software L4 Load Balancing for Kubernetes Services at Yahoo! – Karthikeyan Thangaraj, Verizon Media* (https://www.youtube.com/watch?v=-C86fBMcp5Q) , retrieved 2024-02-03

69. "Skyfall: eBPF agent for infrastructure observability" (https://www.linkedin.com/blog/engineering/infrastructure/skyfall-ebpf-agent-for-infrastructure-observability) . *www.linkedin.com*. Retrieved 2024-02-03.

70. "How Does Alibaba Cloud Build High-Performance Cloud-Native Pod Networks in Production Environments?" (https://www.alibabacloud.com/blog/how-does-alibaba-cloud-build-high-performance-cloud-native-pod-networks-in-production-environments_596590) . *alibabacloud.com*. September 2020. Retrieved 16 August 2022.

71. "Datadog on eBPF" (https://datadogon.datadoghq.com/episodes/datadog-on-ebpf/) . *datadogon.datadoghq.com*. February 2021. Retrieved 16 August 2022.

72. "Runtime Security Monitoring with eBPF" (https://www.sstic.org/media/SSTIC2021/SSTIC-actes/runtime_security_with_ebpf/SSTIC2021-Article-runtime_security_with_ebpf-fournier_afchain_baubeau.pdf) (PDF). *sstic.org*. February 2021. Retrieved 16 August 2022.

73. "Our eBPF Journey at Datadog - Laurent Bernaille & Tabitha Sable, Datadog" (https://www.youtube.com/watch?v=6mTVuZUHLBg) . *youtube.com*. November 2020. Retrieved 16 August 2022.

74. "User Story - How Trip.com uses Cilium" (https://cilium.io/blog/2020/02/05/how-trip-com-uses-cilium/) . *cilium.io*. February 2020. Retrieved 16 August 2022.

75. "Trip.com: Stepping into Cloud Native Networking Era with Cilium+BGP" (https://arthurchiao.art/blog/trip-stepping-into-cloud-native-networking-era/) . *arthurchiao.art*. November 2020. Retrieved 16 August 2022.

76. *Keynote: Open Source Intrusion Detection for Containers at Shopify - Shane Lawrence & Kris Nóva* (https://www.youtube.com/watch?v=6pVci31Mb6Q) , retrieved 2024-02-09

77. Rogers, Patrick (2023-08-15). "BPFAgent: eBPF for Monitoring at DoorDash - DoorDash Engineering Blog" (https://doordash.engineering/2023/08/15/bpfagent-ebpf-for-monitoring-at-doordash/) . *DoorDash Engineering Blog*. Retrieved 2024-02-09.

78. "Making eBPF work on Windows" (https://cloudblogs.microsoft.com/opensource/2021/05/10/making-ebpf-work-on-windows/) . *cloudblogs.microsoft.com*. May 2021. Retrieved 16 August 2022.

79. "Getting Linux based eBPF programs to run with eBPF for Windows" (https://cloudblogs.microsoft.com/opensource/2022/02/22/getting-linux-based-ebpf-programs-to-run-with-ebpf-for-windows/) . *cloudblogs.microsoft.com*. February 2022. Retrieved 16 August 2022.

80. "Progress on making eBPF work on Windows" (https://cloudblogs.microsoft.com/opensource/2021/11/29/progress-on-making-ebpf-work-on-windows/) . *cloudblogs.microsoft.com*. November 2019. Retrieved 16 August 2022.

81. "Cilium Standalone Layer 4 Load Balancer XDP" (https://cilium.io/blog/2022/04/12/cilium-standalone-L4LB-XDP/) . *cilium.io*. July 2022. Retrieved 16 August 2022.

82. *Rate limiting access to internal services in a virtual network – Nick Bouliane, DigitalOcean* (https://www.youtube.com/watch?v=gcHxfhDT-I4) , retrieved 2024-02-04

83. "Building a Secure and Maintainable PaaS - Bradley Whitfield, Capital One" (https://www.youtube.com/watch?v=hwOpCKBaJ-w) . *youtube.com*. November 2020. Retrieved 16 August 2022.

84. *Why eBPF is changing the telco networking space – Daniel Bernier, Bell Canada* (https://www.youtube.com/watch?v=fNtG0iHYne4) , retrieved 2024-02-09

85. *Elastic Universal Profiling* (https://www.elastic.co/observability/universal-profiling) , retrieved 2024-02-26

86. "Think eBPF for Kernel Security Monitoring - Falco at Apple- Eric Sage & Melissa Kilby, Apple" (https://www.youtube.com/watch?v=ZBlJSr6XkN8) . *youtube.com*. October 2021. Retrieved 16 August 2022.

87. "eBPF & Cilium at Sky – Sebastian Duff, Anthony Comtois, Jospeh [sic] Samuel, Sky" (https://www.youtube.com/watch?v=u-4naOMfs_w) . *youtube.com*. August 2021. Retrieved 16 August 2022.

88. "Running and orchestrating multiple XDP and TC programs – Brian Merrell, Walmart" (https://www.youtube.com/watch?v=Fu4L8ewcO70) . *youtube.com*. August 2021. Retrieved 16 August 2022.

89. "High Performance Load Balancing @Walmart – Kanthi Pavuluri & Karan Dalal, Walmart" (https://www.youtube.com/watch?v=thmAcyix8FM) . *youtube.com*. August 2021. Retrieved 16 August 2022.

90. "DIGLIM eBPF: secure boot at application level with minimal changes to distros - Roberto Sassu" (https://www.youtube.com/watch?v=iA7T4MAqKUc) . *youtube.com*. August 2022. Retrieved 16 August 2022.

91. "IKEA Private Cloud, eBPF Based Networking, Load Balancing, and Observability with... Karsten Nielsen" (https://www.youtube.com/watch?v=sg-F_R-ZVNc) . *youtube.com*. May 2022. Retrieved 16 August 2022.

92. *Panel Discussion: Is There Actually a Byte Behind All the Buzz? eBPF in Production!* (https://www.youtube.com/watch?v=qmrHONqsV2M) , retrieved 2024-02-09

93. *Using user-space tracing to solve DNS problems – Andrius Grabauskas, Palantir* (https://www.youtube.com/watch?v=0RDp1IPxbg0) , retrieved 2024-02-09

94. "Reading privileged memory with a side-channel" (https://googleprojectzero.blogspot.com/2018/01/reading-privileged-memory-with-side.html) . *googleprojectzero.blogspot.com*. 3 January 2018. Retrieved 16 August 2022.

95. "BPF and Spectre: Mitigating transient execution attacks" (https://popl22.sigplan.org/details/prisc-2022-papers/11/BPF-and-Spectre-Mitigating-transient-execution-attacks) . *popl22.sigplan.org*. Retrieved 16 August 2022.

96. "bpf: Disallow unprivileged bpf by default" (https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=8a03e56b253e9691c90bc52ca199323d71b96204) . *kernel.org*. Retrieved 16 August 2022.

Further reading

- Gregg, Brendan (December 2019). *BPF Performance Tools*. Addison-Wesley. ISBN 978-0136554820.
- David Calavera, Lorenzo Fontana (December 2019). *Linux Observability With BPF*. O'Reilly Media, Incorporated. ISBN 978-1492050209.
- Gregg, Brendan (December 2020). *Systems Performance, Second edition*. ISBN 978-0136820154.
- Rice, Liz (April 2022). *What Is eBPF?*. ISBN 978-1492097259.
- Rice, Liz (April 2023). *Learning eBPF: Programming the Linux Kernel for Enhanced Observability, Networking, and Security*. O'Reilly Media. ISBN 978-1098135126.

External links

- eBPF.io - Introduction, tutorials & eBPF community resources (https://ebpf.io/)
- eBPF.foundation - Linux Foundation's eBPF Foundation site (https://ebpf.foundation/)
- eBPF documentary - Documentary on the beginnings of eBPF (https://www.youtube.com/watch?v=Wb_vD3XZYOA)