

ISOVALENT
now part of cisco

eBPF Research: What's Going On In Academia?

Paul Chaignon
Isovalent at Cisco

06-10-2026



Who Am I?



Paul Chaignon

Software Engineer @ Cisco

Datapath team for Cilium

Lead organizer of eBPF Workshop

Following eBPF research since 2015

Type selector

☒ all ☒ foundation ☒ using ☒ analysis ☒ improving

Areas selector

☒ all ☒ verifier ☒ offload ☒ networking ☒ storage ☒ security
☒ tracing ☒ jit ☒ misc

Hide non-default conferences from CSRankings

☒ EuroSys, ATC, OOPSLA, ASE, ISSTA, ICDE

Selected 92 papers.

It's the Kernel's Fault! Custom Page Fault Handling With bpf_fault

SOSP'26 [Paper](#) Tal Zussman, Riju Dey, Hasan Zengin, Yiming Fang, David Hildenbrand, Asaf Cidon

using misc

Extends the Linux kernel to allow users to run custom eBPF-based page fault handlers.

Tamias: Feedback-Guided Systematic Concurrency Exploration for Transparent Distributed System Fuzzing

SOSP'26 [Paper](#) Congyu Liu, Yaoxu Song, Pedro Fonseca, Yongle Zhang

using misc security

Systematically explores the concurrency space during fuzzing using sched_ext to control the order of execution of threads.

Waiting at the front door: Continuous monitoring of latency in the host network stack

IMC'26 [Paper](#) Simon Sundberg, Anna Brunstrom, Simone Ferlin-Reiter, Jesper Dangaard Brouer, Toke Høiland-Jørgensen

using networking

Implements a tool to measure the host latency of every packet on the system with very low overhead using the fentry and fexit BPF attach points.

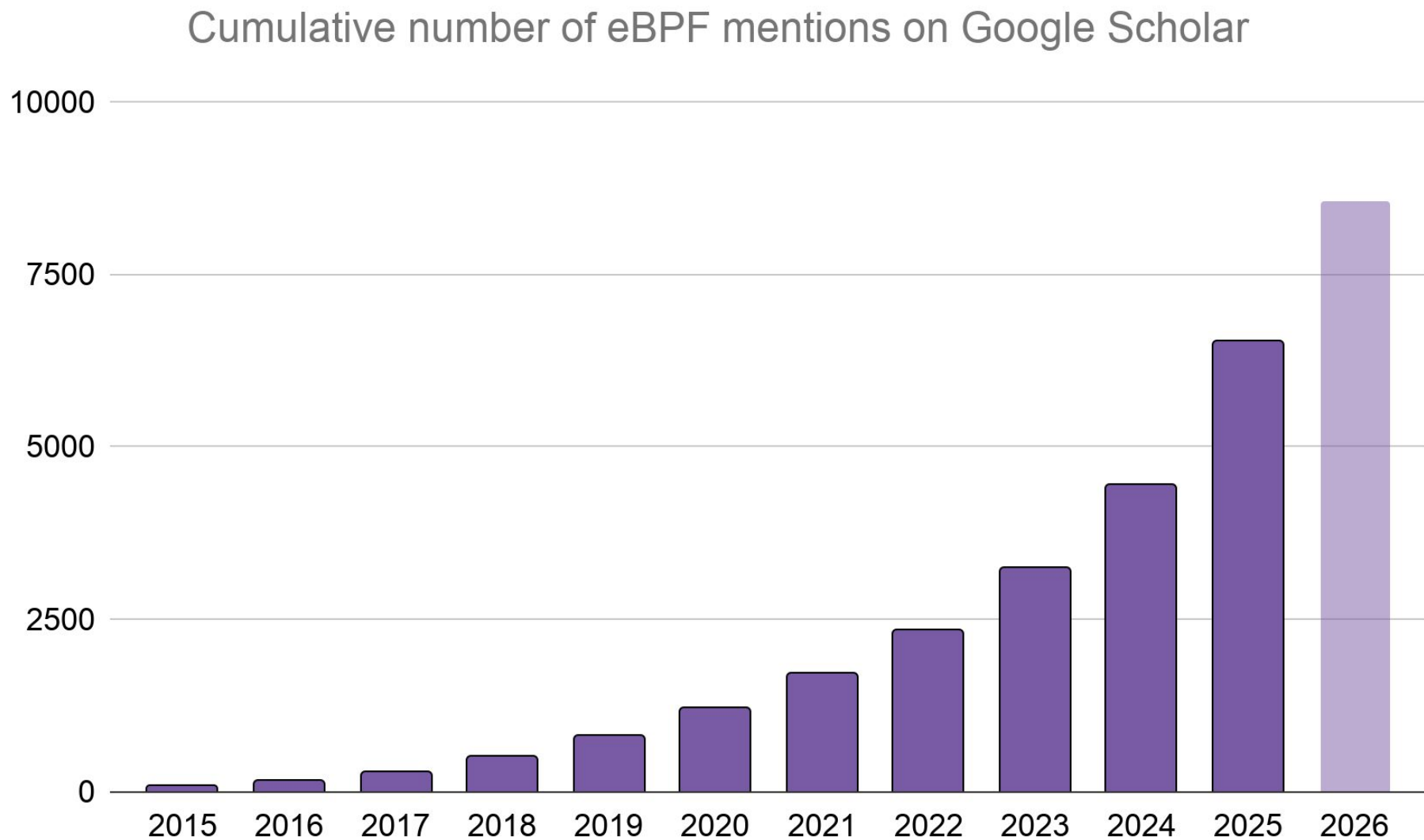
Efficient Build Dependency Verification Using eBPF and Incremental Analysis

ICSE'26 [Paper](#) Yuta Saito, Kazunori Sakamoto, Hironori Washizaki

using security

Uses BPF-based system call tracing to identify dependencies in build systems.

A Myriad of eBPF Papers



Goals for Today

- Share the trends for eBPF research
- Present the main eBPF research topics
- Discuss how to nudge academia towards specific topics

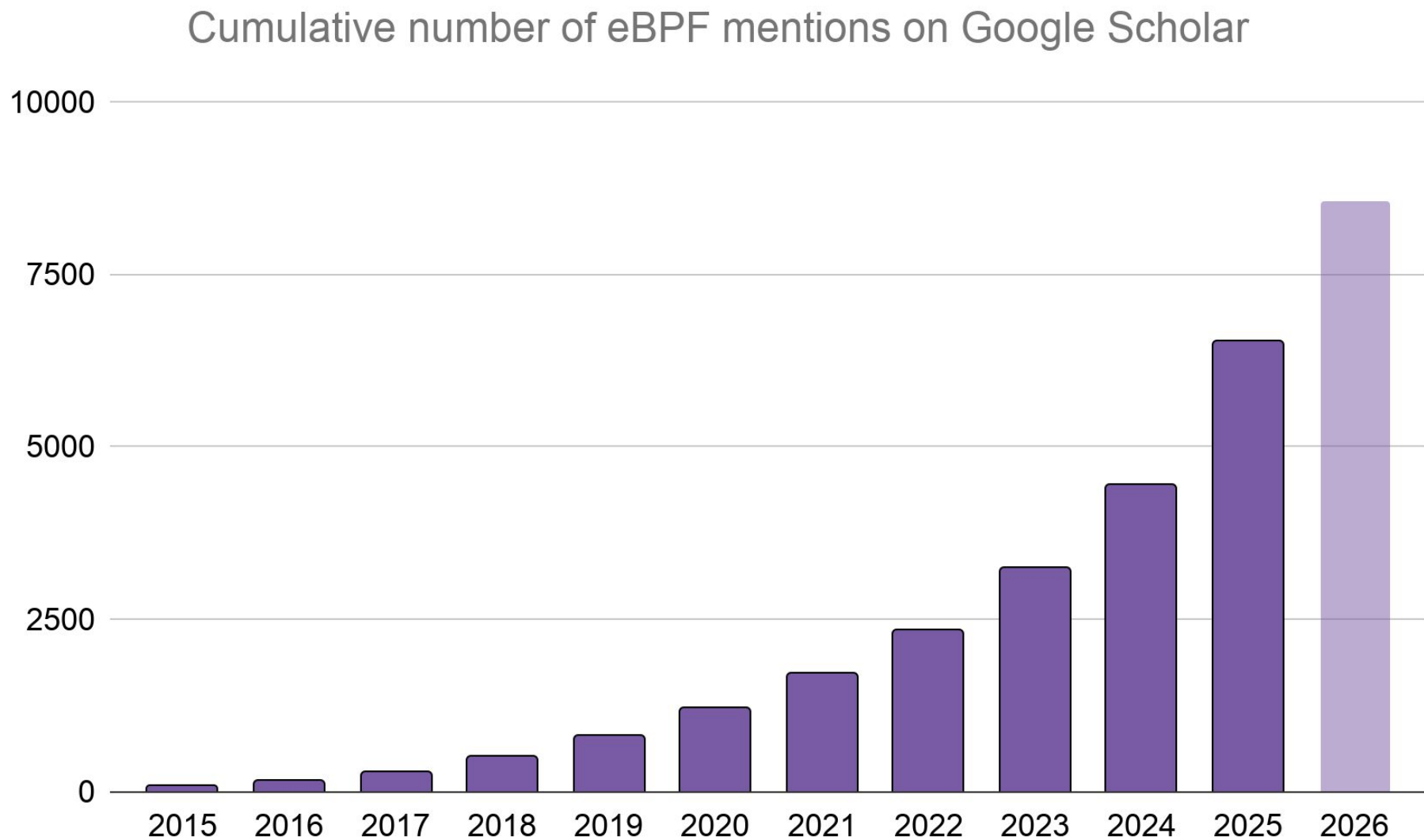


Goals for Today

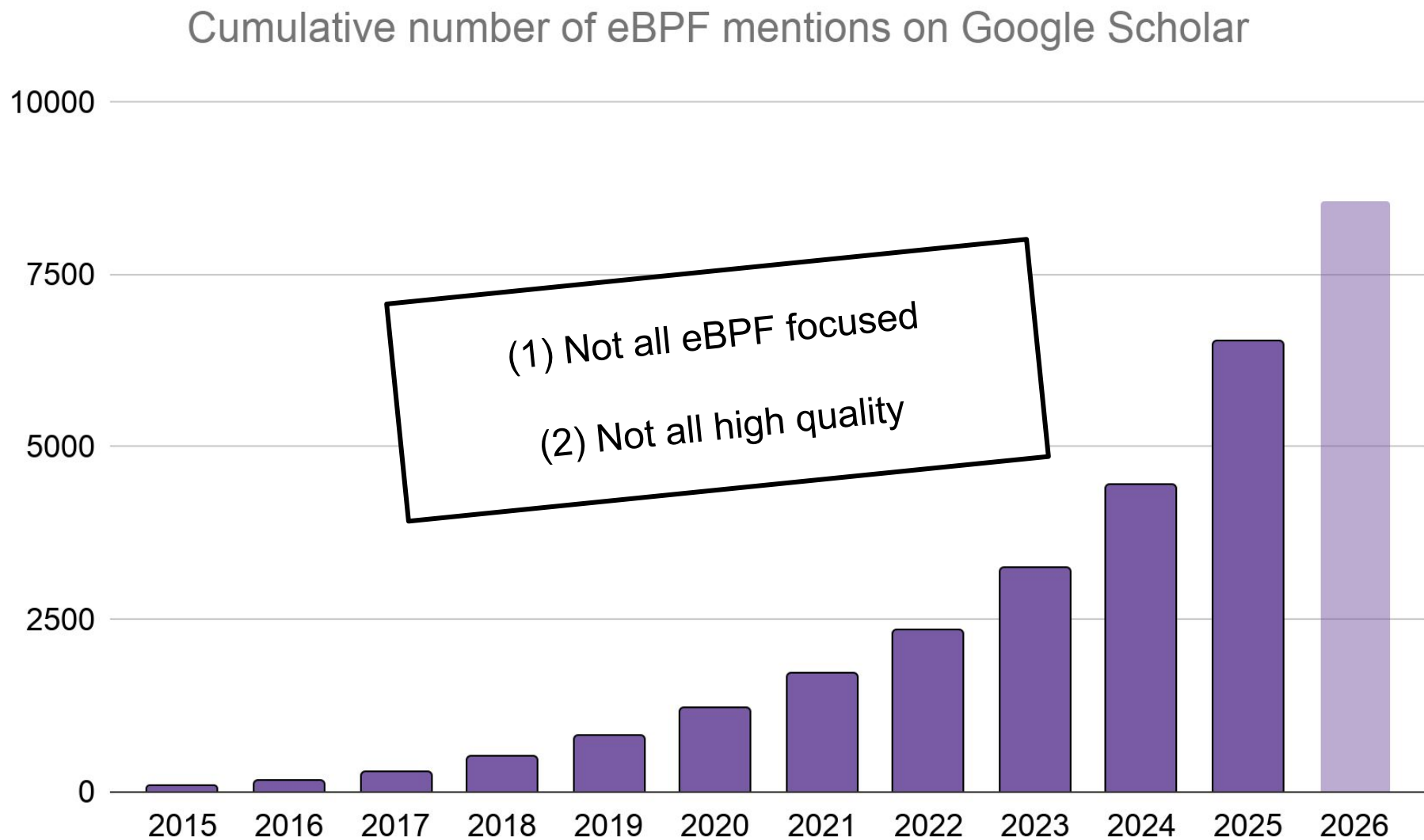
- Understand how to navigate through this
- Share the trends for eBPF research
- Present the main eBPF research topics
- Discuss how to nudge academia towards specific topics



How to Triage Papers?



How to Triage Papers?



How to Triage Papers?

- Use citation count?
 - Easy to cheat
 - Citations can mean many things

Retraction Watch

One way to boost your uni's ranking: Ask faculty to cite each other

STAT Reporting from the frontiers of health and medicine

Gaming the system, scientific 'cartels' band together to cite each others' work

By [Ivan Oransky](#) and [Adam Marcus](#) Jan. 13, 2017



Academia Obscura

@AcademiaObscura

Follow

Everyone's favourite co-author.

Google Scholar

Follow



et al.

The academic superstar everybody wants to be co-author with. See my home page for more info.
bibliometrics, scientometrics, citation analysis
No verified email - [Homepage](#)

Citation indices

	All	Since 2011
Citations	2415484	781837
h-index	333	275
i10-index	333	333

Co-authors View all...

Paul Erdős, A. Author

Title	1-20	Cited by	Year
Protein measurement with the Folin phenol reagent	OH Lowry, NJ Rosebrough, AL Farr, RJ Randall J biol Chem 193 (1), 265-275	195644	1951
Molecular cloning	J Sambrook, EF Fritsch, T Maniatis Cold spring harbor laboratory press 2, 14-9.23	176236	1989
Psychometric theory	JC Nunnally, IH Bernstein, JMF Berge McGraw-Hill	85826	1967
DNA sequencing with chain-terminating inhibitors	F Sanger, S Nicklen, AR Coulson Proceedings of the National Academy of Sciences 74 (12), 5463-5467	65990	1977

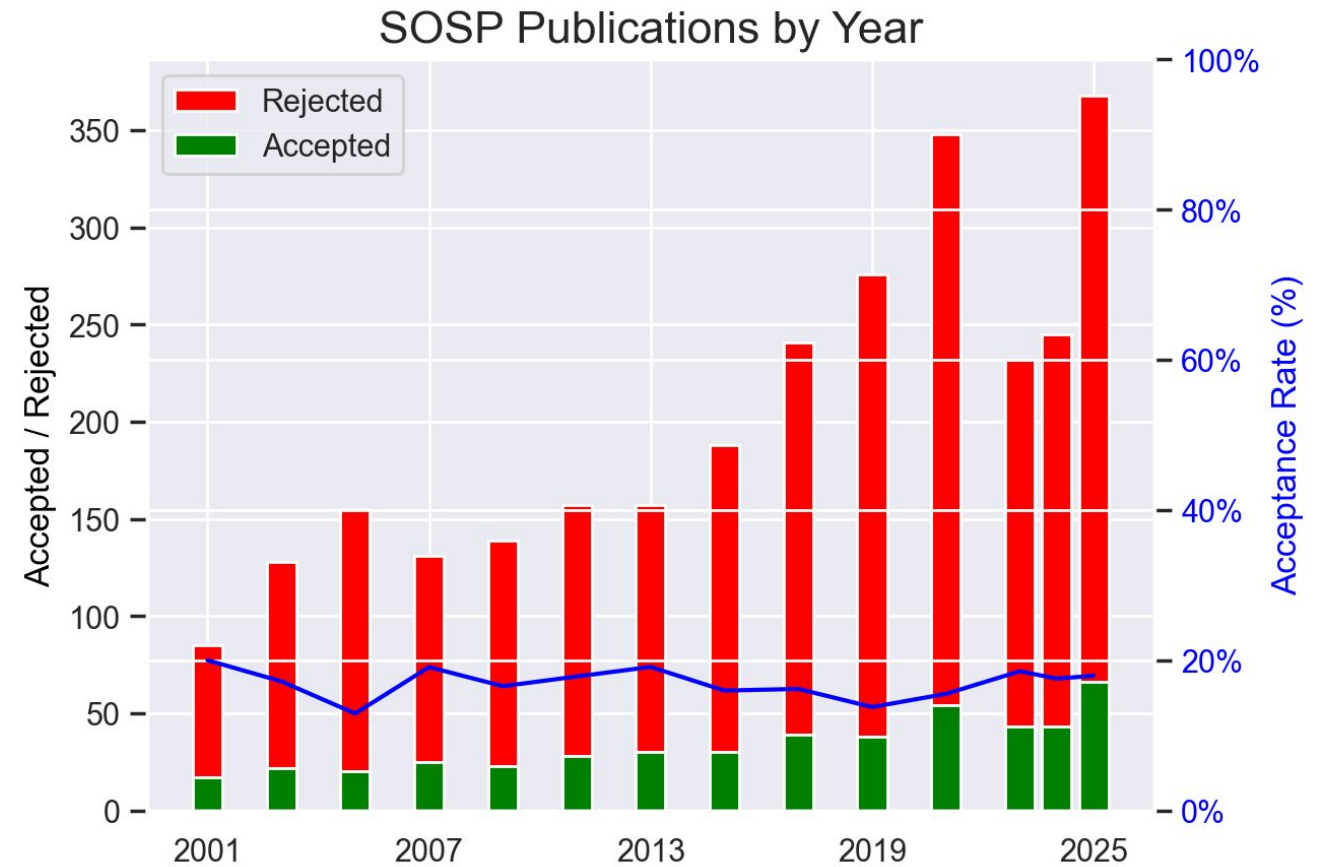
3:55 PM - 22 Aug 2017

354 Retweets 684 Likes



How to Triage Papers?

- Look for the top conferences instead
 - Hard to get into
 - Hard to cheat
- Approach taken by CSRankings
 - Already has the top conferences
 - Open to edits via GitHub



Building the List

- All BPF papers from CSRankings-listed conferences
- No journals, no workshops
- BPF needs to play a key role in the design
- Interactive selectors to filter
- Links to paper and video
- One line summary focused on BPF



Type selector

☒ all ☒ foundation ☒ using ☒ analysis ☒ improving

Areas selector

☒ all ☒ verifier ☒ offload ☒ networking ☒ storage ☒ security
☒ tracing ☒ jit ☒ misc

Hide non-default conferences from CSRankings

☒ EuroSys, ATC, OOPSLA, ASE, ISSTA, ICDE

Selected 92 papers.

It's the Kernel's Fault! Custom Page Fault Handling With bpf_fault

SOSP'26 [Paper](#) Tal Zussman, Riju Dey, Hasan Zengin, Yiming Fang, David Hildenbrand, Asaf Cidon

using misc

Extends the Linux kernel to allow users to run custom eBPF-based page fault handlers.

Tamias: Feedback-Guided Systematic Concurrency Exploration for Transparent Distributed System Fuzzing

SOSP'26 [Paper](#) Congyu Liu, Yaoxu Song, Pedro Fonseca, Yongle Zhang

using misc security

Systematically explores the concurrency space during fuzzing using sched_ext to control the order of execution of threads.

Waiting at the front door: Continuous monitoring of latency in the host network stack

IMC'26 [Paper](#) Simon Sundberg, Anna Brunstrom, Simone Ferlin-Reiter, Jesper Dangaard Brouer, Toke Høiland-Jørgensen

using networking

Implements a tool to measure the host latency of every packet on the system with very low overhead using the fentry and fexit BPF attach points.

Efficient Build Dependency Verification Using eBPF and Incremental Analysis

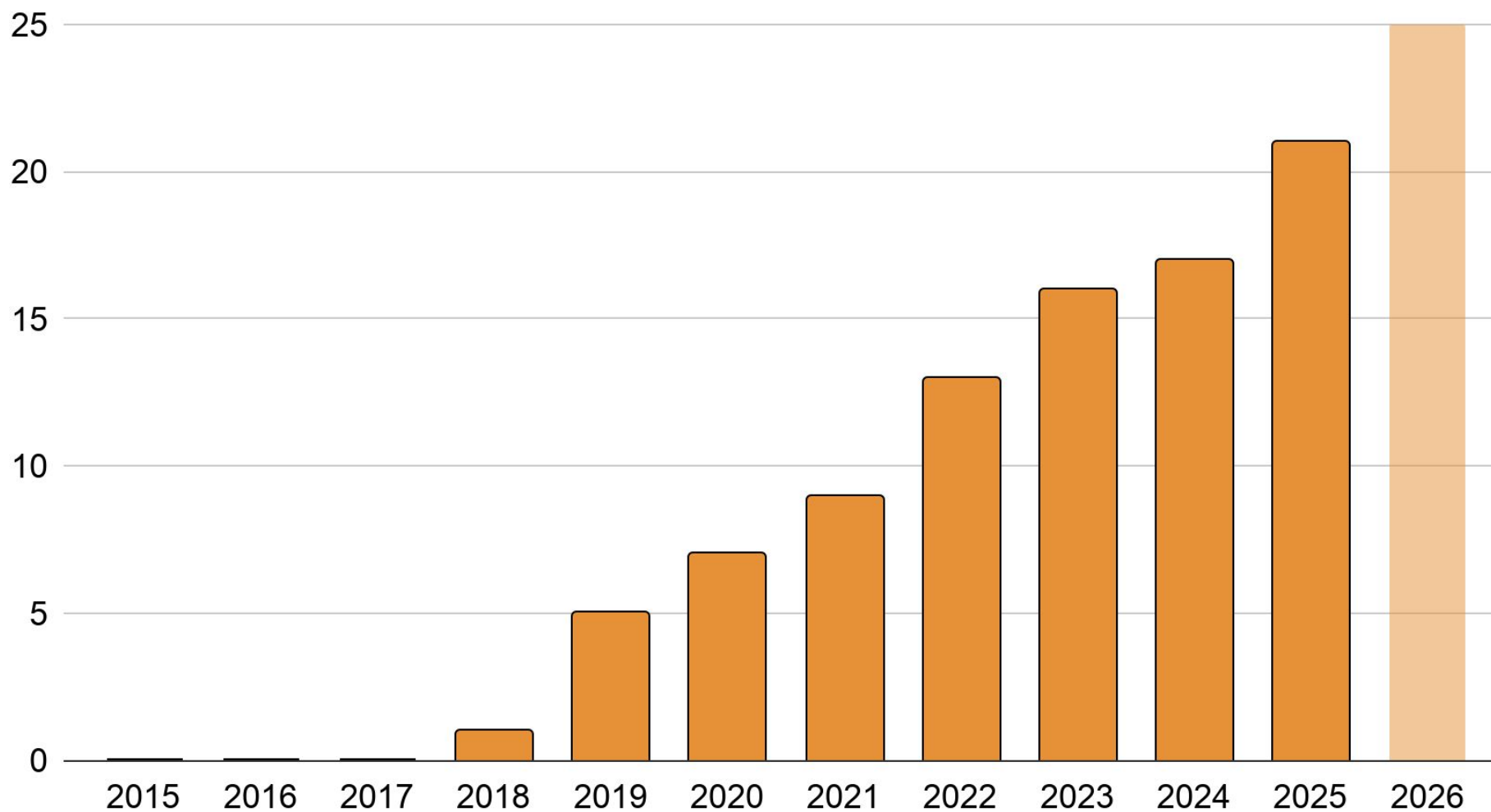
ICSE'26 [Paper](#) Yuta Saito, Kazunori Sakamoto, Hironori Washizaki

using security

Uses BPF-based system call tracing to identify dependencies in build systems.

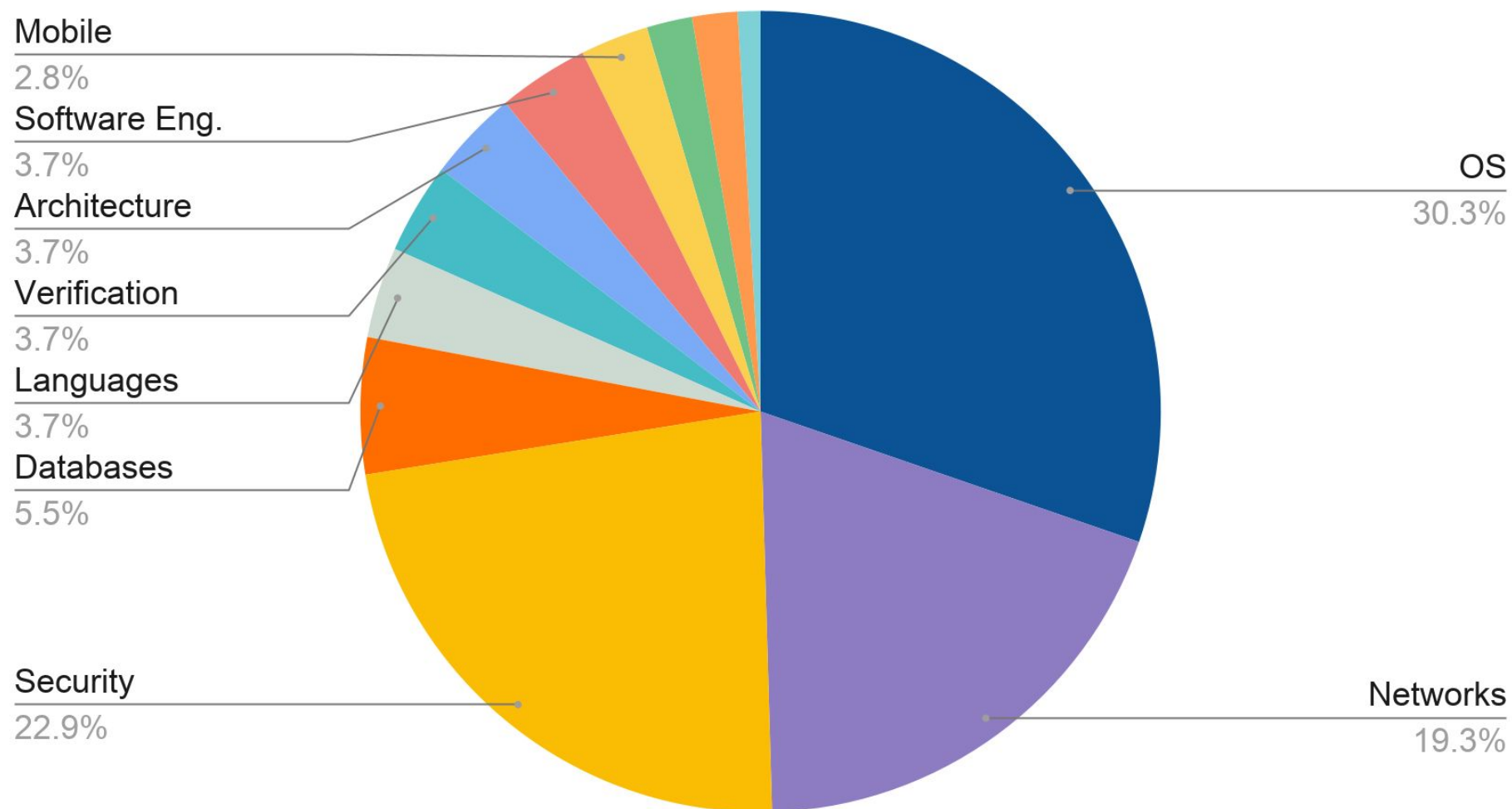
BPF Statistics

Cumulative number of conferences



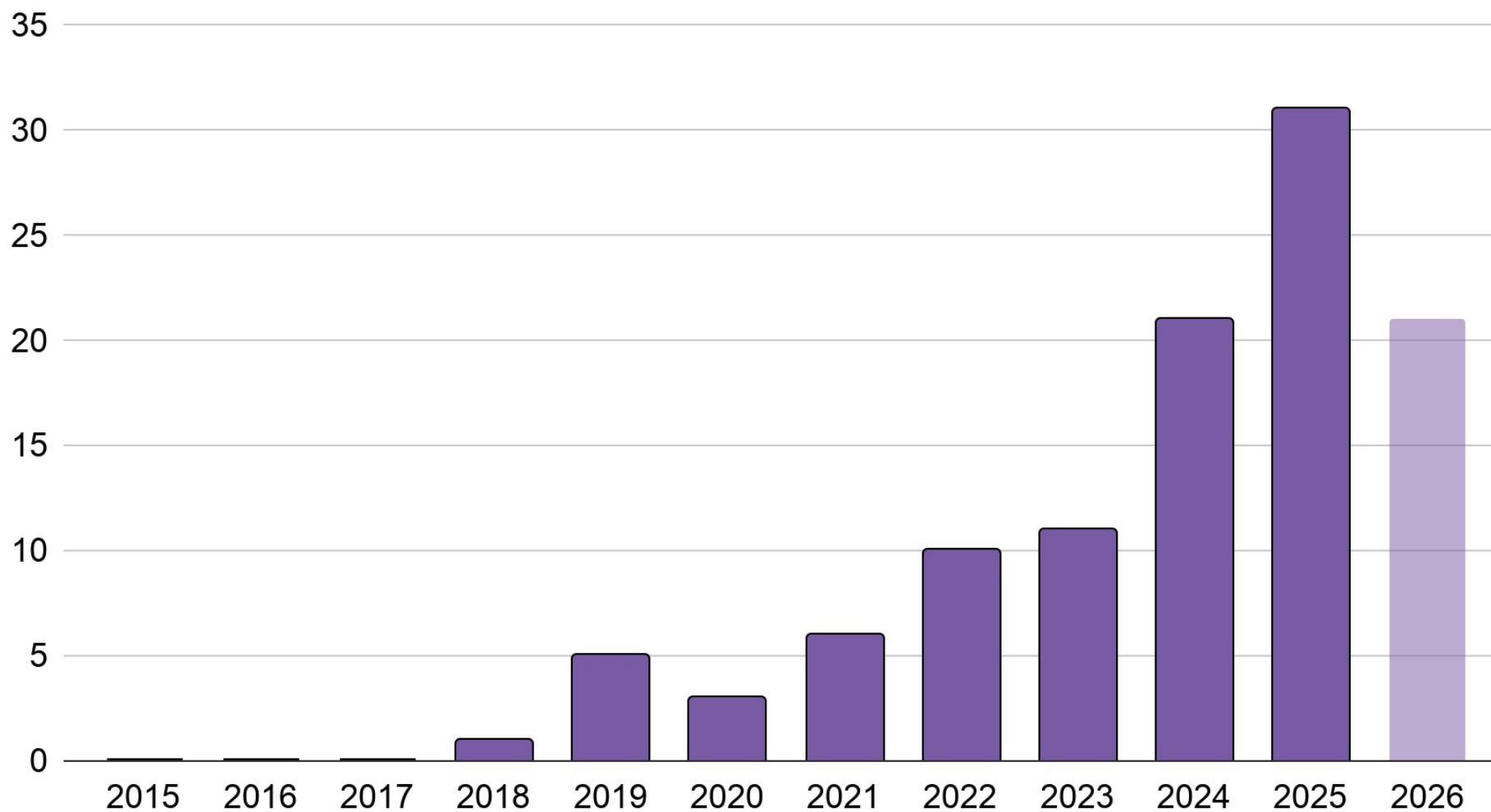
BPF Statistics

Number of papers per research domain



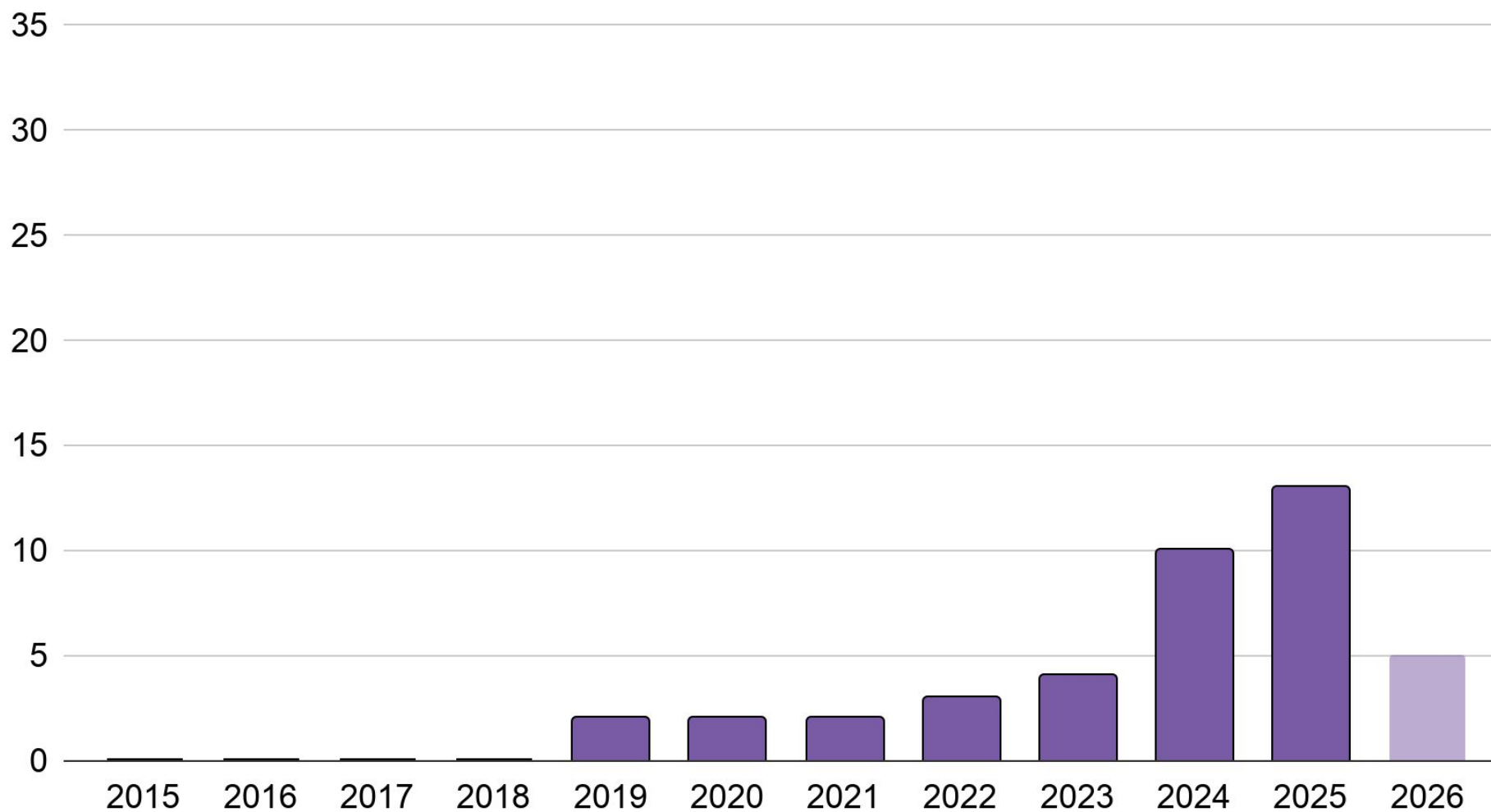
BPF Statistics

BPF Papers Published in CSRankings-listed Conferences

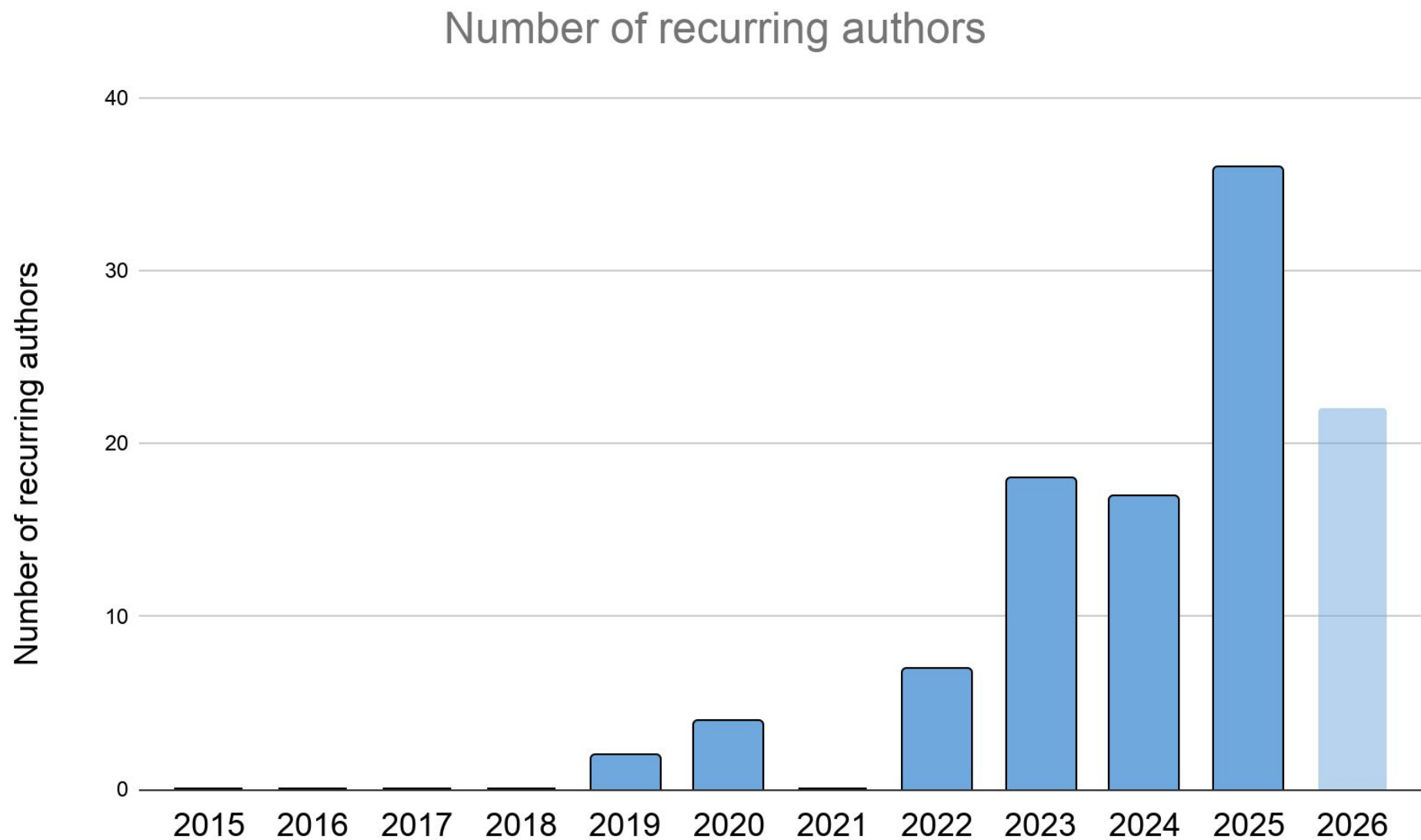


BPF Statistics

Papers Improving or Studying BPF



BPF Statistics



Research Topics: Using BPF

- BPF to extend:
 - Linux: Page fault handler, lock policies, prefetching, peripheral drivers, power management...
 - Other software: Hypervisors, DBMS, FUSE, RT OSes, BLE devices...
 - Protocols: QUIC and BGP.

Research Topics: Using BPF

- BPF to extend:
 - Linux: Page fault handler, lock policies, prefetching, peripheral drivers, power management...
 - Other software: Hypervisors, DBMS, FUSE, RT OSes, BLE devices...
 - Protocols: QUIC and BGP.
- BPF to offload:
 - To the kernel: Memcached, Paxos ops, DBMS proxies, L7 processing...
 - To hardware: FPGA NICs or storage devices.

Research Topics: Using BPF

- BPF to extend:
 - Linux: Page fault handler, lock policies, prefetching, peripheral drivers, power management...
 - Other software: Hypervisors, DBMS, FUSE, RT OSes, BLE devices...
 - Protocols: QUIC and BGP.
- BPF to offload:
 - To the kernel: Memcached, Paxos ops, DBMS proxies, L7 LB...
 - To hardware: FPGA NICs or storage devices.
- BPF to implement:
 - Networking: OVS, RSS replacement, service mesh, userspace stack...
 - Tracing: Distributed tracing, measurement systems, mobile system tracing...
 - Security: Audit systems, fuzzing-focused schedulers, IDPSes...

Research Topics: Using BPF

- BPF to extend:
 - Linux: **Page fault handler**, lock policies, prefetching, peripheral drivers, power management...
 - Other software: Hypervisors, DBMS, FUSE, RT OSes, BLE devices...
 - Protocols: QUIC and BGP.
- BPF to offload:
 - To the kernel: Memcached, Paxos ops, DBMS proxies, **L7 LB**...
 - To hardware: FPGA NICs or storage devices.
- BPF to implement:
 - Networking: OVS, RSS replacement, service mesh, userspace stack...
 - Tracing: Distributed tracing, measurement systems, mobile system tracing...
 - Security: Audit systems, fuzzing-focused schedulers, IDPSes...

Research Topics: Improving BPF

- Improving the verification:
 - Replacement: Proof-Carrying Code, PREVAIL, language-based (Rust)
 - Improvements: Hardware-based hardening and runtime checks

Research Topics: Improving BPF

- Improving the verification:
 - Replacement: Proof-Carrying Code, PREVAIL, language-based (Rust)
 - Improvements: Hardware-based hardening and runtime checks
- Finding verifier bugs:
 - Testing: Finding test oracles for the Linux verifier, fuzzing of userspace eBPF runtimes
 - Verification: Agni, RIOT kernel

Research Topics: Improving BPF

- Improving the verification:
 - Replacement: Proof-Carrying Code, PREVAIL, language-based (Rust)
 - Improvements: Hardware-based hardening and runtime checks
- Finding verifier bugs:
 - Testing: Finding test oracles for the Linux verifier, fuzzing of userspace eBPF runtimes
 - Verification: Agni, RIOT kernel
- Performance improvements: Compiler optimizations, synthesis-based compilation, extensions for prefetching and batching.

What Got “Upstreamed”?

- 2019–2021: Fixes in the RISC-V and arm64 JIT compilers found by formal verification
- 2021–2026: Lots of fixes in the verifier found by fuzzing and formal verification
- 2021: Precision and optimization improvements found by formal verification and program synthesis
- 2021: PREVAIL, the Windows verifier
- 2021: Speculative execution fixes
- 2023: Exception support
- 2024: Deadlock fixes
- 2026: Arena improvements

What Got “Upstreamed”?

- 2019–2021: Fixes in the RISC-V and arm64 JIT compilers found by formal verification
- 2021–2026: Lots of fixes in the verifier found by fuzzing and formal verification
- 2021: Precision and optimization improvements found by formal verification and program synthesis
- 2021: PREVAIL, the Windows verifier
- 2021: Speculative execution fixes
- 2023: Exception support
- 2024: Deadlock fixes
- 2026: Arena improvements

Upstreamed by: Luke Nelson, Harishankar Vishwanathan, Elazar Gershuni, Hao Sun, Kumar Kartikeya Dwivedi, Priya Govindasamy

What Got “Upstreamed”?

- 2019–2021: Fixes in the RISC-V and arm64 JIT compilers found by formal verification
- 2021–2026: Lots of fixes in the verifier found by fuzzing and formal verification
- 2021: Precision and optimization improvements found by formal verification and program synthesis
- 2021: PREVAIL, the Windows verifier
- 2021: Speculative execution fixes
- 2023: Exception support
- 2024: Deadlock fixes
- 2026: Arena improvements

Upstreamed by: **Luke Nelson, Harishankar Vishwanathan, Elazar Gershuni, Hao Sun, Kumar Kartikeya Dwivedi, Priya Govindasamy**

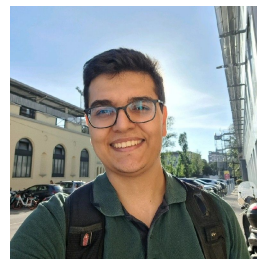
Which Research Teams?



**Harishankar
Vishwanathan**
(Rutgers University)



Hao Sun
(ETH Zurich)



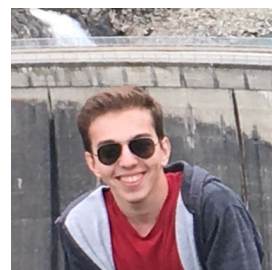
Farbod Shahinfar
(Politecnico di Milano)



Yusheng Zheng
(UC Santa Cruz)



**Kumar Kartikeya
Dwivedi**
(EPFL)



Tal Zussman
(Columbia University)

Conclusion

- eBPF research going well
 - Some push back on non-formal static analysis
- Do we want to encourage specific research topics?
- How?
 - 💰 eBPF Foundation cannot replace NSF
 - Convince the students!
 - Write it (or publish)



Thanks!

Questions?