



Extending BPF Signing into an IMA Integrity Lifecycle

Hongming (Henry) Qiu¹ · Avery Blanchard² · Gheorghe Almasi²
Hubertus Franke² · Tianyin Xu¹

¹ University of Illinois Urbana-Champaign

² IBM Research



LINUX
PLUMBERS
CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026

Why sign BPF programs?

- eBPF programs run in privileged context as part of the kernel.
- Major Linux distros ship signed LKMs but not eBPF programs
- Integrity, provenance, policy and enforcement.
- Legitimate software can contain vulnerable BPF, so enforcement is absolutely necessary.
 - CVE-2024-52529: Cilium L7 policy bypass.
 - CVE-2026-26963: Cilium host-firewall bypass.



LINUX

PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



BPF signing: where we are

- KP Singh at LPC 2022: flexibility
 - Separate policy specification from signature verification.
 - Talked about IMA integration but policy semantics left open.
- KP Singh at LPC 2025: signing support and deployment
 - Mentioned difficulties with runtime generation and relocations
 - Trusted userspace loaders can authorize unsigned BPF.
 - Trade-off: general-purpose loaders can admit arbitrary BPF.



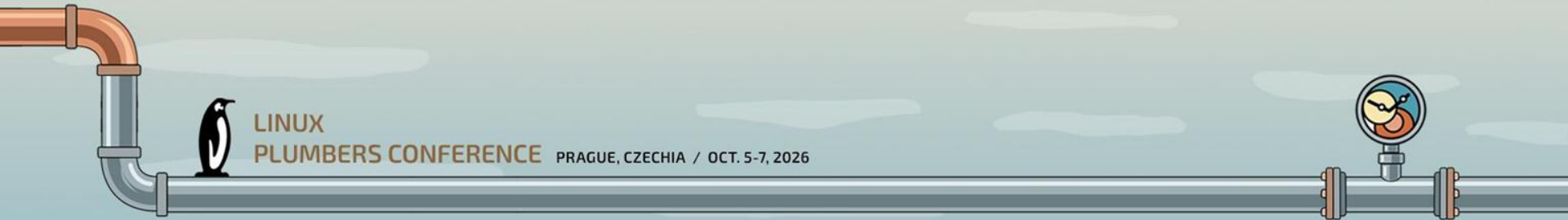
LINUX
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



Objectives: eBPF as an integrity object

- Use signing evidence to establish BPF program provenance.
- Appraise the BPF program, not just the loader's file.
- Keep identity and evidence with the resident object.
- Manage trust beyond admission.
 - Reappraise and revoke resident programs when trust changes.
 - Particularly important when handling vulnerable programs



Our proposal: IMA-BPF

- IMA is an established Linux integrity framework.
- We can reuse its infrastructure for eBPF.
- IMA selects objects for measurement, appraisal, etc.
- Apply IMA policy to BPF objects in the kernel load path.
 - `measure func=BPF_CHECK`
 - `appraise func=BPF_CHECK`
`ebpf_prog_type=BPF_PROG_TYPE_SCHED_CLS`



LINUX

PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



Measurements, keys, and hashes

- IMA records program digests for audit and attestation.
- Reuse Linux keyrings for BPF appraisal.
 - **.ima**: trusted signing authorities.
 - **.blacklist**: revoked keys and program digests.
- Able to denylist a vulnerable program's digest without revoking its signing key.
- Signers and hashes add finer-grained trust decisions, while the base rules provide the coarse ones



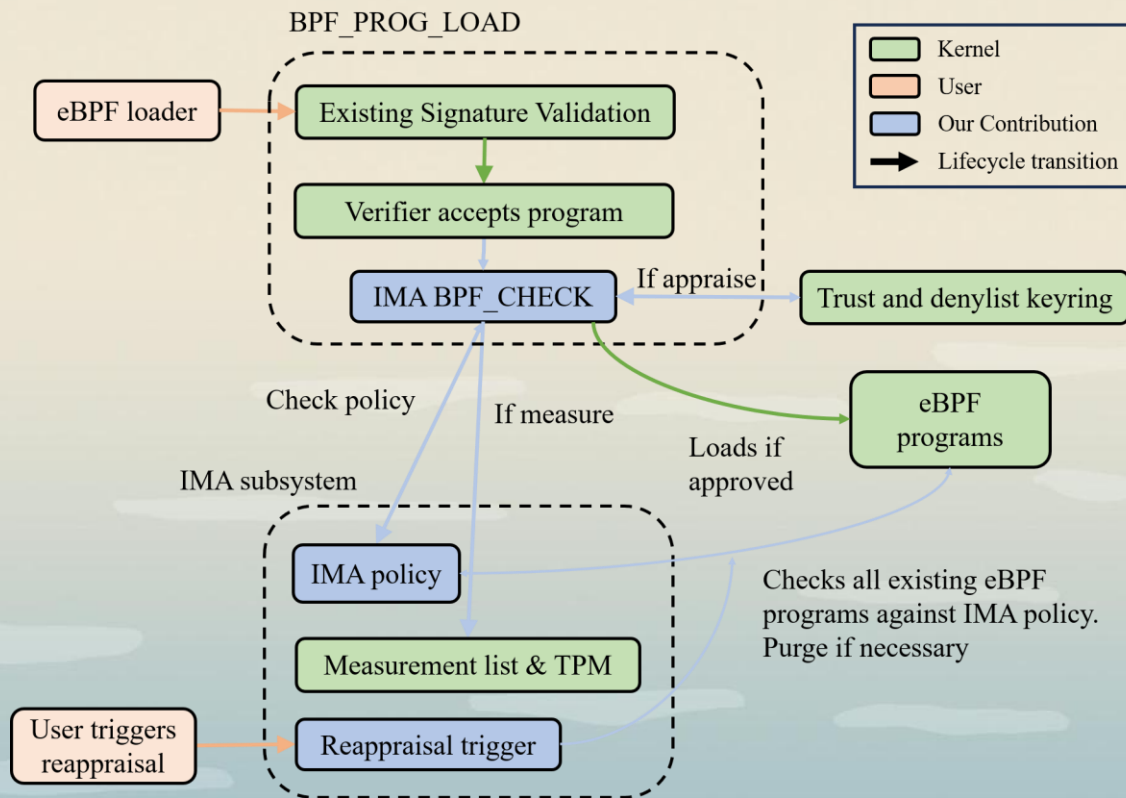
LINUX

PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



The IMA-BPF lifecycle



BPF loaders and provenance

- KP Singh, LPC 2025: signed BPF loaders.
 - Relocate and create child programs inside the kernel.
 - Child loads do not carry independent signatures.
- Our extension: propagate and retain the loader's signing evidence.
- Reappraise the child even after its loader exits.



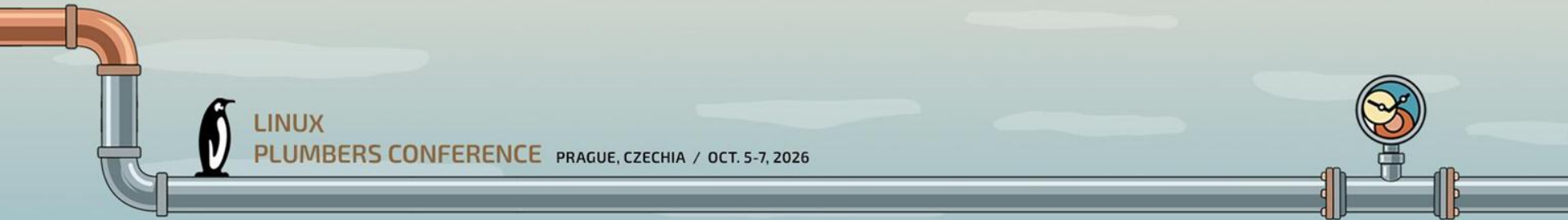
LINUX
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



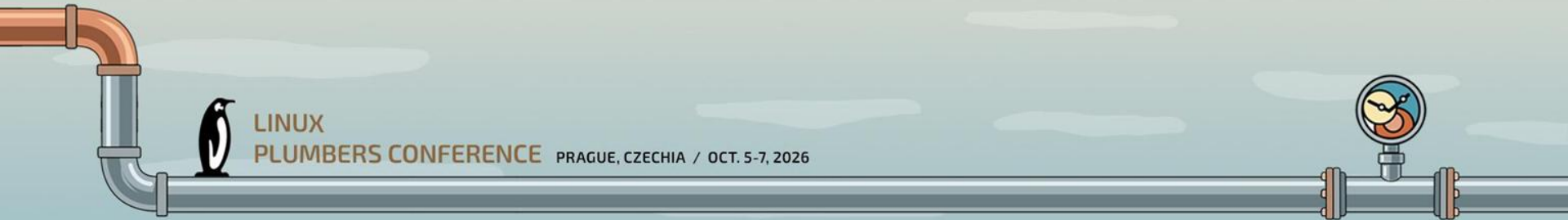
Reappraisal

- Redeployment or a vulnerability changes what we trust.
- Allow policy, trusted keys, and denied digests to evolve.
- Reappraise resident programs against updated trust on demand.
- Revoke noncompliant programs and admit replacements.



Purge

- Use BPF's existing reference-counted lifetime.
 - `bpf_link` refcount reaches 0: program detached from hotpath
 - Program refcount reaches 0: program reclaimed
- Block new references before releasing existing ones.
- Notify the application so it can release and replace.
- Does not yet work on all types of BPF programs



Use cases and adoption

- Block unauthorized or denylisted programs from loading.
- Revoke vulnerable code and coordinate patched redeployment.
- Example: our modified Cilium integration exercises the full lifecycle.
 - After denylisting and reappraisal, Cilium is notified.
 - Cilium reloads patched code without rebooting the node.



Adoption difficulties

- Pinning preserves the BPF object, not a process to notify.
 - The Cilium loader drops its handles after pinning.
- Our notification path needs a live manager with retained handles.
- Applications need a fast path to load compliant replacements.
 - Cilium can regenerate entire profile, with significant overhead
- Replacement must account for downtime and failure.



LINUX
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



Discussion

- What can break in real deployments when we revoke live BPF programs?
- Should critical services adopt signing first, and what are the main difficulties?
- Should loaders have distinct delegated certificates?

