01-19-2022 TSC Meeting Minutes

TSC Meeting Zoom link

Meeting Recording

Chat File

Attendees & Representation. Please add your name to the attendance table below.

Attendees	
Name	**Company**
Daniel Havey	Microsoft
VM (Vicky) Brasseur	Wipro
Eric Tice	Wipro
Dave Thaler	Microsoft
Kanthi P	Walmart
Satya Pradhan	Walmart
Rishabh Gupta	Walmart
Jason Niesz	Walmart
Karan Dalal	Walmart
Kiran Reddy	Walmart
Santhosh Fernandes	Walmart

LF Staff: LJ Illuzzi

Agenda

- LF Antitrust Policy
- Meeting note taker
- Welcome to new attendees
- LF ONE Summit recording to be added in l3af.io
- Last week TSC output
    - Standardize Interface for communication between BPF maps from user-space <-> eBPF kernel space
    - Gate keeping and security and concerns around public market place
    - How do we handle other programs that might load eBPF and conflict with l3AF. Do we block them?
    - How are we different from BumbleBee?
    - Boiler plate generator that generates all the bpf_tail call and manage the maps
    - Here's the notes from the BSC minutes, let Dave know if any corrections:
        - Who is authoritative for what programs run on the node? (answer: admin of machine, who does so via a JSON config file)
          Would you really trust a public program repository? (unclear, L3AF TSC is discussing same question, cases today use private ones with paths specified by machine admin)
          Any coordination with Kubernetes or OpenStack? (today is standalone, may integrate with other things in future)
          How do you config/customize ebpf programs? (today Walmart's programs used with L3AF factor config/customization to be done via maps)
          Any standard way of communicating with userspace, like protobufs etc? (not yet but L3AF would like to go in that direction)
          What about gatekeeping, i.e. reject "bad" tools that were rejected by kernel patch requests? Compare to bcc tool repo. Private

program repositories are safer, public repository is dangerous/risky if just accepted (could just be reference implementations, always use private repos from L3AFD?)
What happens if something other than L3AFD tries to deploy an ebpf program to the kernel? (today, can interfere, L3AF TSC should discuss) can you have SELinux like functionality or just use SELinux to control?
How can you disincent/prevent admins from pointing to a public repo and being unsafe?
How does it differ from the bumblebee project?

- PR to add l3af as one of the projects under ebpf.io
- Building eBPF programs (suggested by Luka from Sartura; they have a tool they find helpful)
- Status of PEN
- LFN Induction
  - Setup separate call. Propose weekly on Tuesday 09:30 to 10:30am ET/ 6:30amPT / 8pm IST
  - First meeting 02/01
- General Topics (cover as needed)
  - Use Cases
  - Roadmap
  - Project structure
    - Governance
    - Technical Steering Committee

Minutes/Updates

- Key takeaways BSC meeting
  - Standardized communication for maps from UM to KM
    - Good to have a map level spec around map and data types
    - Vicky: How useful outside of L3AF? Would be good to have a de-facto standard.
    - Agree, standards would be helpful.
    - Louis: Create sub-project to write documents?
    - Karan: Does this change for Windows?
    - Dave: Believe the answer is no.
      - Both use libBPF
  - Gatekeeping, security and public repository
    - Different types of signing.
    - Just because a program is in a public repo - it would be dangerous to install
    - Have a private repo and the only thing that pulls from a public repo is one that is trusted by admin
    - This isn't like an app store - this is kernel stuff
    - Karan: Could we allow only trusted sources to put stuff into a public repo
    - Dave: That would not change the security situation because trust is not transitive.
    - Karan: Could we have a hybrid model?
    - Dave: The BSC might go along with this, but I don't recommend it.
    - Vicky: Experience does not bear out trusting public repositories.
      - Incredibly rife with situations where security issues arise. We need the possibility for people to have private models
      - If we don't set up the pubilc repo then someone else will. We should do it so that we can at least have some safeguards for security.
      - At the same time users should do their own testing and evaluation.
    - Karan: Maybe it makes sense to include this in the KF writeup.
      - If we could set up L3AFd with a secure and an unsecure mode. Public repos would have more checks.
    - Vicky: Like the idea of secure and insecure mode. Default - always in secure mode. Will not pull from public repos.
      - If you want to pull from public then you have to go insecure
    - Dave: Why even have an insecure mode? Just don't let them do that.
    - Dave: What if there was identity verification to contrib to public repo?
    - Karan: How do we guardrail that?
      - Allow contribs only from trusted sources?
      - Some process to become a trusted source.
      - Even there if someone tries to pull from a public repo then have extra checks.
    - Vicky: Generally in favor of checks and balances for listing something in public repo.
      - The reality is that this is a huge burden. Who does this? Who is responsible for it?
    - Dave: If there are 100 eBPF programs submitted, who will decide which is trusted and which is not.
    - Dave: Is there liability involved? Pure reputation based repo manages itself. Will likely have more contribs.
    - Karan: Trusted source, but we recommend testing.
    - Vicky: Lawyers always say that there is risk. How do other repos handle legal risk?
      - Being under the LFN will give us access to legal assistance.
    - Karan: What do we agree on?
      - Get the PR updated with calling out the risks that we are aware of.
      - Take the first step into creating this eBPF repo.
    - Vicky: Should we form a WG around the marketplace?
      - Wary of work on the marketplace blocking work on L3AFd.
    - Karan: Thoughts on creating a WG?
      - Dave: Yes, I agree that the 2 topics are separable.
  - How do we handle other programs that might load eBPF and block L3AFd?
    - Do we block that?
    - Especially with XDP or TC. L3AF loads the root program and it loads the other programs in the desired sequence
      - Someone with privileges could load a new program and attach to XDP. Then the L3AF program would not work.
      - We don't currently block them.
    - Dave: Can you have SeLinux functionality to control things?
    - Karan: This is more of an eBPF foundation question than a L3AFp question.
    - Dave: Windows allows multiple programs to attach to XDP. Linux allows only one.

        - Kran: We could detect that the root program has been unloaded. We can't prevent it.
        - Jason: At least you would know.
        - Dave: Prevent would need a kernel change. Detection wouldn't.
        - Vicky: Bring this to the eBPF BSC? What i the process?
        - Dave: Is the eBPF foundation or the Linux kernel the right place to solve this problem.
        - Dave: Which is more appropriate?
            - Karan: add an API to BPFtool.
            - Each tool could write their own process.
        - Vicky: XDP and TC. In the XDP case this is not cross-plat.
        - Dave: Windows does not have tc yet. Windows would likely be able to attach multiple progs to tc (When we build tc)
        - Vicky: This is a Linux problem so it's a Linux problem
        - Dave: Must submit a PR and/or show up at office hours.
            - Process is actually easier than the BSC.
        - Vicky: 2 probs: Detection and prevention. Can be worked on in parallel.
    - How are we different from bumblebee?
        - Karan: Very different.
            - Bumblebee is for writing kprobes or tracepoints
            - They don't do BPF chaining.
            - They don't support XDP.
        - Dave: Bumblebee also submited an eBPF.io project
            - How do we make the 2 descriptions different?
        - Karan: I can write a few lines on how L3AF is different from Bumblebee.
        - Dave: Bumblebee is a command line system, L3AFd is an orchestration system.
            - Also workflow is different
        - Karan: If you just want kprobe or tracepoint you could already do it with 1 line of bash.
            - The complexity comes from running chaining progs, maps, ect.
        - Dave: Bumblebee must use an OCI package.
        - Dave: L3AF has an agent and orchestrator. Bumblebee does not. Add to description.
        - Jason: Bumblebee has no agent or control system so that you can shift the order of programs.

Action Items

- [x] LJ Illuzzihave draft working documents ready for LFN Induction meeting on 02/01, including proposed timeline

Future Agenda Items