01-12-2022 TSC Meeting Minutes

TSC Meeting Zoom link

Meeting Recording

Meeting Chat File

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

Attendees	
Name	**Company**
Daniel Havey Dave Thaler	Microsoft
Balachandra Kamat	Wipro
VM (Vicky) Brasseur	Wipro
Brian Merrell	Walmart
Satya Pradhan	Walmart
Steve Laughman	Microsoft
Jason Niesz	Walmart
Santhosh Fernandes	Walmart
Divya Reddy	Walmart
Karan Dalal	Wamart
Kanthi P	Walmart
Luka Perkov	Sartura
Christopher Liljenstolpe	

LF Staff: LJ Illuzzi (no audio today; attending Dev/Testing Forum)

Agenda

- LF Antitrust Policy
- Meeting note taker
- Welcome to new attendees
- General Topics (cover as needed)
    - Use Cases
    - Roadmap
    - Project structure
        - Governance
        - Technical Steering Committee

Minutes/Updates

- Karan: BSC Meeting - Jason and Brad will be doing the talk
    - What is the context?
    - Add L3AF as a project under eBPF.io
- Dave: Do not assume that everyone has gone through documentation ahead of talk

        - Is L3AF going to orchestrate other programs?
        - Get into program repo part, signing?
- Karan: Explain what L3AF does now, what we attend in future
    - Answer questions
- eBPF foundation is chartered to discuss things like this
- Dave: Meeting is open by invite only.
    - Essential L3AF people.
- Multiple companies represented.
- Karan: Talk about L3AF and L3AFd
    - eBPF program repository
    - critical for L3AFd and eBPF community
    - Talk about signing
- Dave: What it does now vs what it does in the future should have a clear delineation
- Dave: Use the term eBPF program rather than kernel function when talking to the BSC.
- Vicky: Criticality of signing to L3AF is important in this talk. Signing must be cross-platform.
- Jason: Two levels of signing: Package and kernel level
- Dave: Three levels. See document.
    - Sign package - checked by L3afd
    - Signature on eBPF byte code - checked by kernel. Already verified.
    - When using Windows on top of Hyper-V type 1 with HVCI
        - Only execute things checked by the type 1 Hyper-V - Signing on code.
    - Could have any one and up to all three of them.
- Brian: Should we use Karan's intro slide deck.
- Karan: Yes modified version. Add simplified chaining and eBPF repo, hyperlink the PRs.
- Dave: 60 minute meeting w/~30 mins for L3AF
- Discussion on L3AF Kernel Function Marketplace document: https://github.com/l3af-project/l3af-arch/pull/10
    - No longer going to use Kernel Function Marketplace eBPF Package Repository
    - TO Build or not to Build
        - Building would help L3AF get off the ground
        - Portability- building, what kernel versions are we compatible with?
        - Dave: Relocations done on static offsets - compile once run anywhere.
        - Only supported by Linux tooling
        - Jason: Windows would have to do something similar
            - Ship the bytecode and that bytecode could run across different Linux kernel versions
            - Maybe we don't want to go down this route.
        - Vicky: Not being cross-platform is a problem.
    - Dave: eBPF programs call various helpers, attach points etc. This will not be identical across all platforms eg., Linux specific code.
        - some structures in Linux are Linux only eg., Kprobes to functions that only exist on Linux. This program is only applicable to Linux.
        - In some cases it is possible to write code that is cross-plat. In other cases programs will be platform specific. Today core only applies to Linux platform. Even if it does become cross-plat there will be OS specific parts.
    - Dave: Windows has the same problem when you use private APIs. These same problems exist on other platforms such as Android, macOS, etc.
        - You could say: To submit on core you must use Linux.
    - Luka: All eBPF programs (MAriner) are written in core. Advise that every program has a way that it is unit testable.
    - Vicky: Meta data and requirements, etc. for a package are important here. Security: Alternative to Hosting Building Source Code. An actor with NPM caused a lot of problems because people were building programs remotely. Do we want to let people shoot themselves in the foot or do we want to guide them to do the right thing? What is the design philosophy?
    - Dave: Current App stores. Should be programming lang agnostic. Don't submit source. Will get more use.
    - Vicky: Should have a link to source code if under an OSI. Optional. Documentation required. Enforce best practices. This would help make L3AF packages user friendly.
    - Dave: Nuget.org does exactly this.
    - Brian: Verifying kernel function safety. If someone submits bytecode then verifying that it isn't doing bad things could be difficult just by looking at the bytecode.
    - Dave: This determination could be subjective. Should the customer deal with this?
    - Vicky: Should at least have a way to test these things and notify. Are these even things we can test for.
    - Dave: This problem may not be tractable.
    - Brian: Advantage that app stores have is that they are sandboxed by not providing any system level access. We have to be more security minded with eBPF programs with non-source code submissions.
    - Dave: Non-source code submissions may be better in private repos.
    - Brian: Want this doc to be useful for both public and private repos
    - Dave: Is the package repo only allowed to have things in it that are L3AF compatible?
    - Vicky: My vote is, yes for now. Start by supporting L3AFd, but there may not be any reason to limit it to L3AFd.
    - Brian: Currently L3AF requires special eBPF program chaining.
        - As we move to supporting simplified chaining we can move away from that.
    - Dave: Like that direction.
    - Karan: HAven't figured out signing mechanics yet. Once we figure out signing mechanics and chaining then we could support other than L3AF.
    - Dave: What if the package repo only worked for things that don't do tail calls. Does not enforce the custom chaining? Would that motivate people to do the right thing?

Action Items

Future Agenda Items

- Building eBPF programs (suggested by Luka from Sartura; they have a tool they find helpful)