IO Visor Project Overview

IO Visor: Introduction

Infrastructure Transformation

Cloud-based Applications
Accelerating and driving the IT industry to seek faster service delivery and higher efficiency

Data Center & Cloud Transformation
IO and networking must be open, flexible, distributed, secure, and easy to operate

Virtualization Growth
Changes requirement for IO and networking subsystems to support elastic and dynamic applications and services

IOVISOR PROJECT

Infrastructure Transformation

Infrastructure Needs

- Common way to develop and share new IO functions
- Programmable data planes abstractions & development tools
- Flexible and high performance technology

IO VISOR PROJECT

Introducing IO Visor Project

LINUX FOUNDATION
COLLABORATIVE PROJECTS

Evolution of Kernel BPF & eBPF
(Berkeley Packet Filter)

Led by initial contributions from PLUMgrid
(Upstreamed since Kernel 3.16)

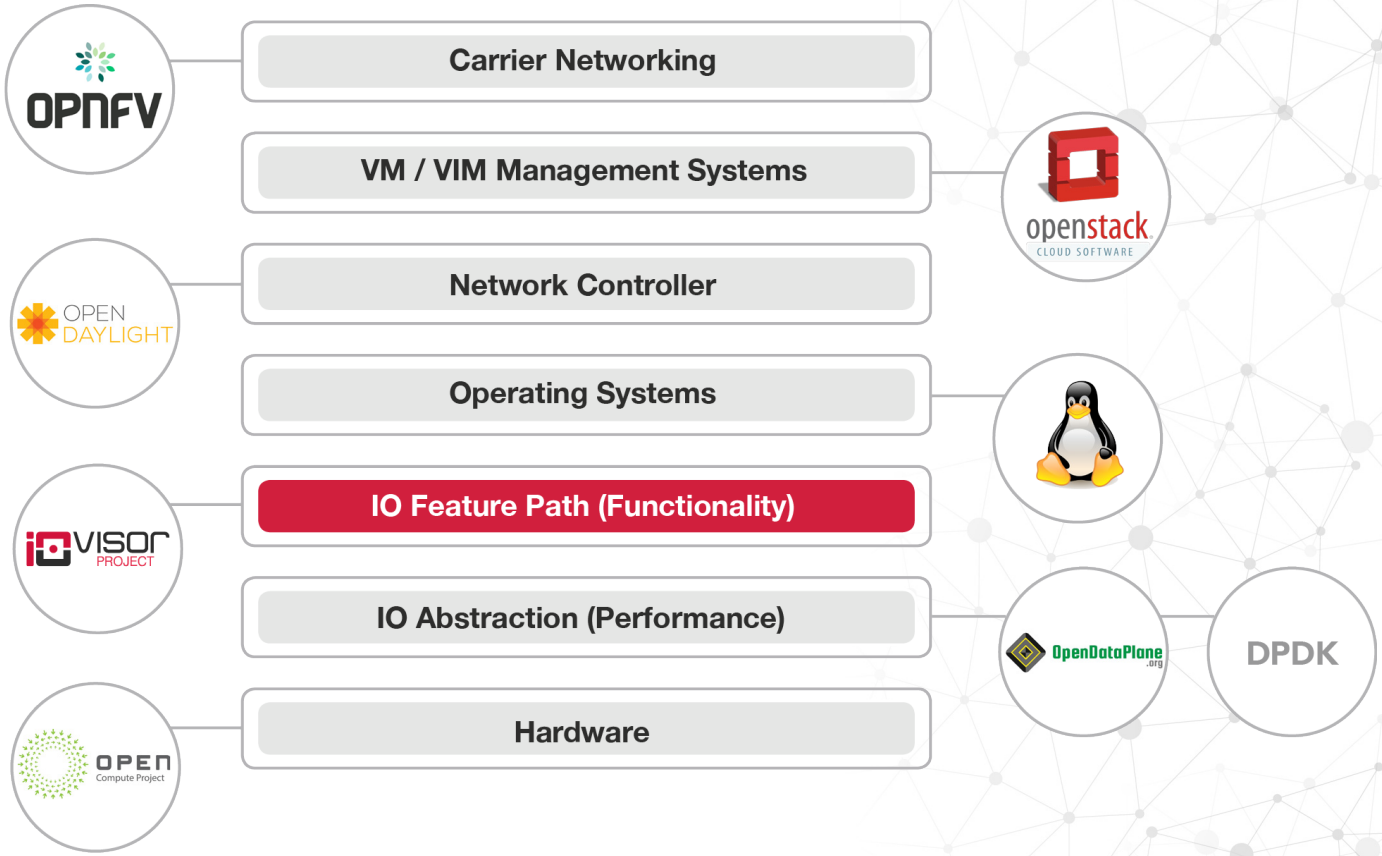
Future of Linux Kernel IO for software defined services

"IO Visor will work closely with the Linux kernel community to ***advance universal IO extensibility for Linux****. This collaboration is critically important as virtualization is putting more demands on flexibility, performance and security.*

Open source software and collaborative development are the ingredients for addressing massive change in any industry. ***IO Visor will provide the essential framework for this work on Linux virtualization and networking****."*

Jim Zemlin, Executive Director, The Linux Foundation.

Open Networking Ecosystem



Founding Members

eBPF: Overview

A little bit of history: BPF

- Introduced as Berkeley Packet Filters in kernel 2.1.75, in 1997
- BPF is now referred to as Classic BPF or cBPF
- Originally created as a way to analyze and filter network packets for network monitoring purposes

- BPF Goal: Accept packets you are interested in or discard them
- How: Userspace attaches a filter to a socket
- Example application: tcpdump/libpcap, wireshark, nmap, dhcp, arpd

A little bit of history: eBPF

- e(xtended)BPF
- Initial proposal was in 2013, by Alexei Starovoitov* and up streamed since version 3.16
- Referred to as the universal in-kernel virtual machine
- Designed to give ability to create any in-kernel IO modules

- eBPF Goal: Improve and extend existing BPF infrastructure
- How: Programs in C and translated into eBPF instructions, loaded in kernel and executed. In-kernel compiler: x86, ARM64, s390, powerpc*, MIPS*
- Example Application: networking, tracing, security …

*https://lkml.org/lkml/2013/12/2/1066

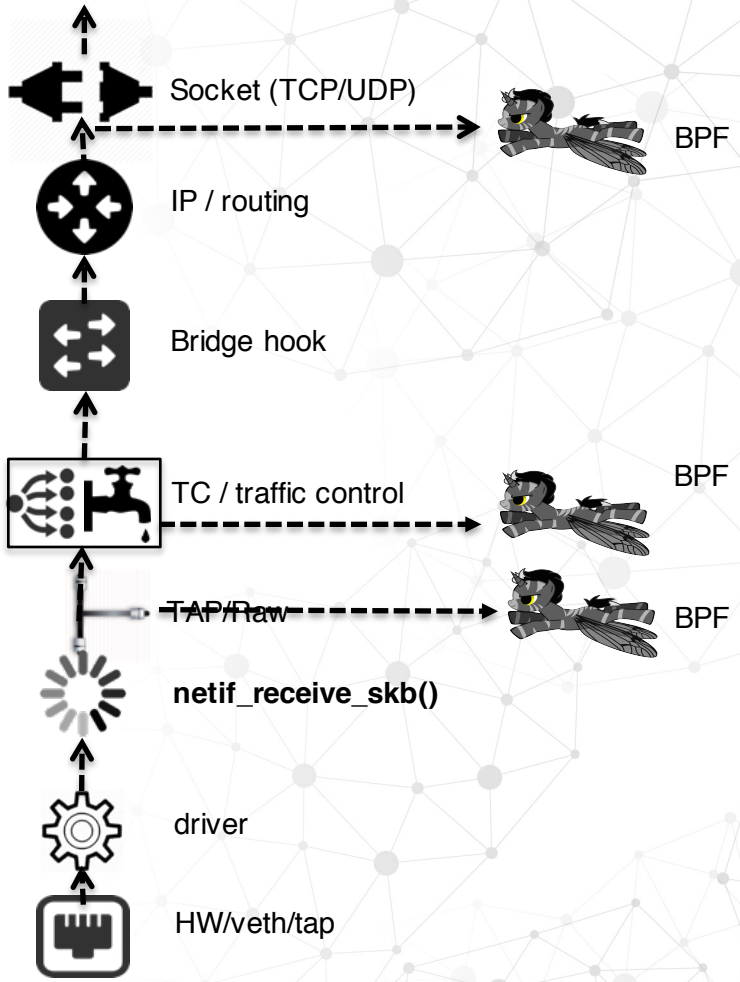
eBPF: Loading New Modules

BPF program written in C

Translated into eBPF instructions (LLVM)

Loaded in kernel and executed

Hooked at different levels of Linux Networking Stack



IO Visor: Overview

IO Visor Project: What?

1

Open Source & Community

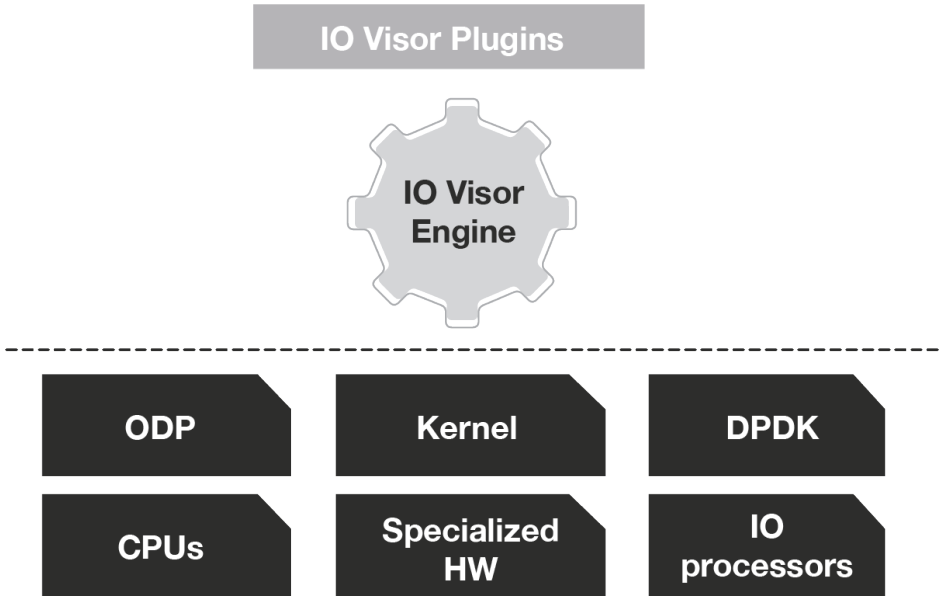
- An open source project and a community of developers
- Enables a new way to Innovate, Develop and Share IO and Networking functions

2

Programmable Data Plane

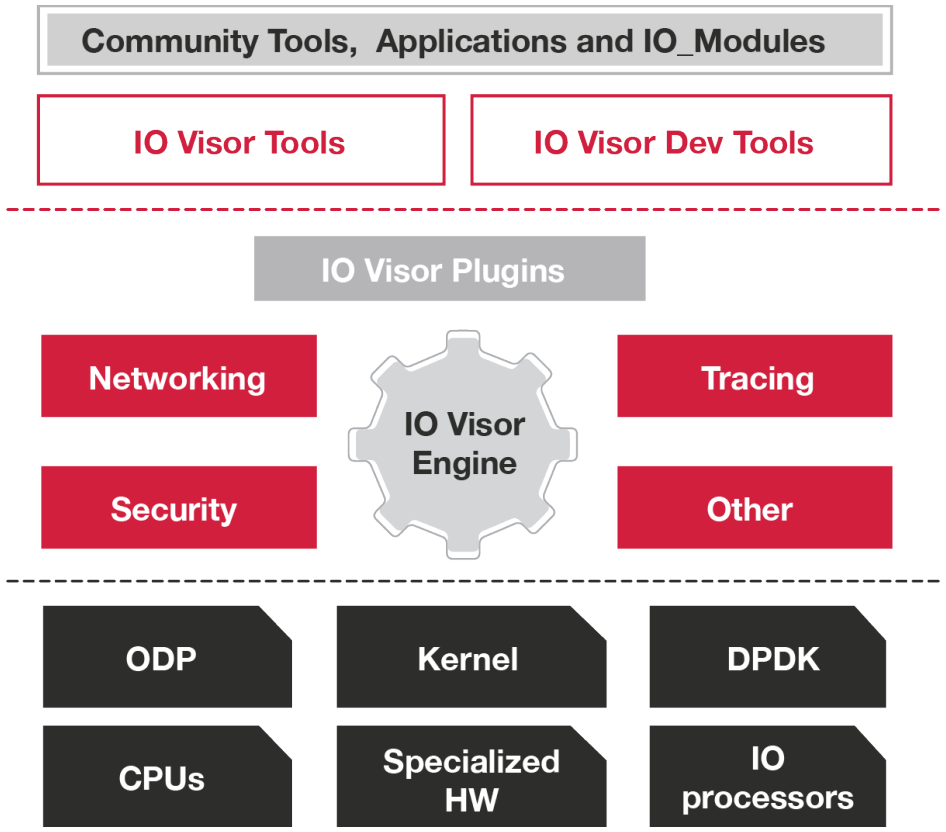
- A programmable data plane and development tools to simplify the creation and sharing of dynamic “IO Modules”

IO Visor Project, What is in it?



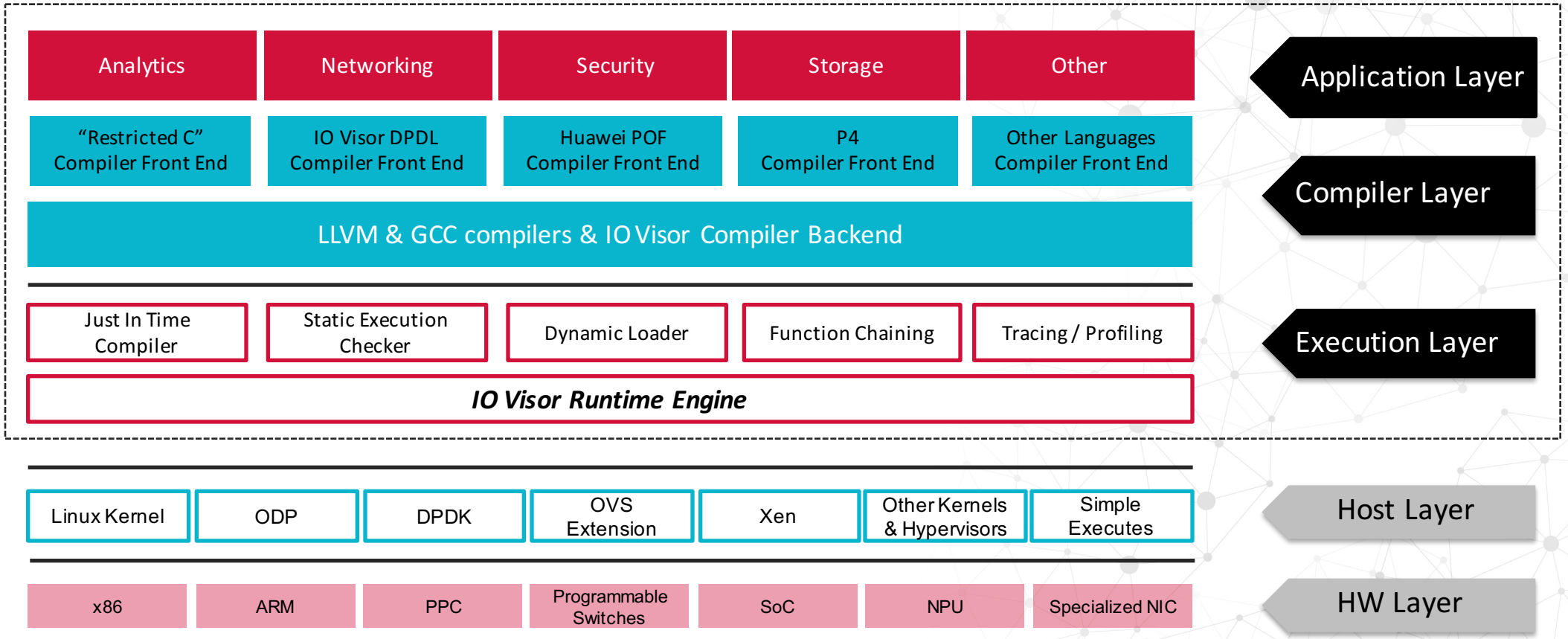
- **IO Visor** Project refers to a **collection** of open source components
- **IO Visor Engine** is an abstraction of an IO execution engine
- **Multiple** IO Visor Engines can exist, **Software** or **Hardware** based
- IO Visor Engine has a set of **IO Visor Plugins** to provide functionality to different areas

IO Visor Project, What is in it?



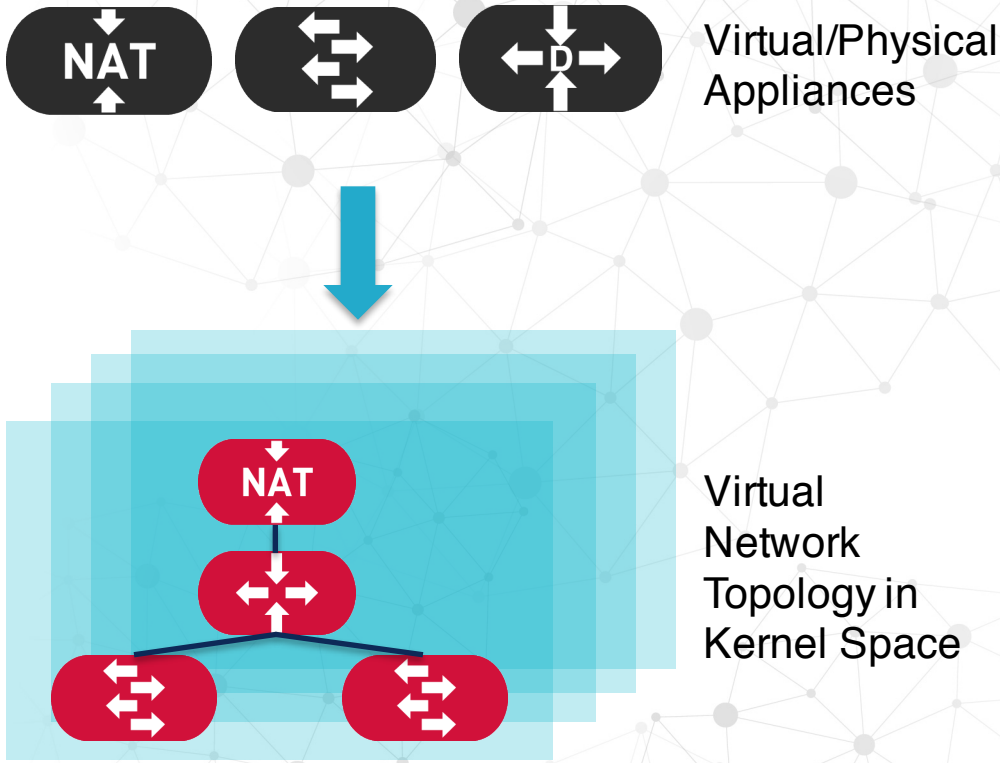
- A set of development tools, **IO Visor Dev Tools**
- A set of **IO Visor Tools** for management and operations of the IO Visor Engine
- A set of Applications, Tools and open **IO Modules** build on top of the IO Visor framework
- A set of possible use cases & applications like **Networking, Security, Tracing & others**

IO Visor Project – Enabling the Ecosystem



IO Visor Project Use Cases Example: Networking

- IO Visor is used to build a fully distributed virtual network across multiple compute nodes
- All data plane components are inserted dynamically in the kernel
- No usage of virtual/physical appliances needed
- Example here https://github.com/iovisor/bcc/tree/master/examples/distributed_bridge

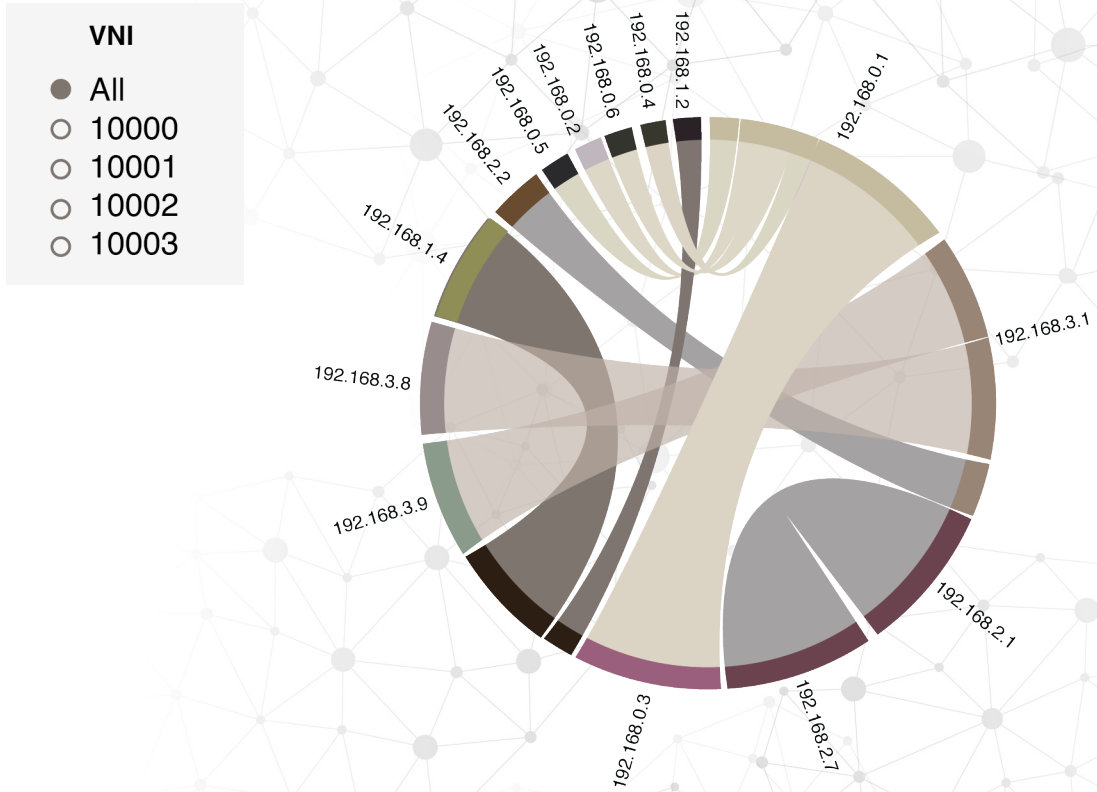


IO Visor Project Use Cases Example: Security

- IO Visor provides a powerful platform for **secure computing**
- BPF/eBPF can be used as the backend to enforce fencing of user space components (applications) in the kernel
- BPF program executed whenever an application is making a system call into the kernel
- Seccomp as an example

IO Visor Project Use Cases Example: Tracing

- IO Visor is used to build a **real-time, distributed analytics platform** that monitors the health of a VXLAN tunneling infrastructure
- Data plane component is inserted dynamically in the kernel and leveraged by the application to report information to the user
- Example here https://github.com/iovisor/bcc/tree/master/examples/tunnel_monitor



IO Visor: Community & Ecosystem

Governance/Releases

- Similar to other Linux Foundation Collaborative Projects
- Governing Board to drive business decisions and leadership
  - E.g. Marketing, Legal, Finance/Budgeting, Certification & Compliance programs
- Technical Steering Committee drives the technical development and engagement with other open source projects
  - E.g. oversees releases, coordinates sub-projects, sets standards/requirements for release participation

IO Visor Membership Levels

Membership Level	Annual Fee	Board Seat	TSC Seat	Marketing Committee	Notes
Platinum	Flat fee: $50K	Yes	Yes	Yes	Linux Foundation Membership Required at any level
Silver [1]	Tiered, based on org size[1]	1 per every 5 Silver members, at least 1, up to 3 total	No	Yes (non-voting)	Linux Foundation Membership Required at any level
Participating Academic Member	None	No	No	No	Requires Governing Board approval
Community Participant (non-Member)	None	No	No	No	Anyone can participate in technical development community and earn a TSC seat by becoming a maintainer

[1] Silver Annual Fee Scale
> 5,000 employees = $20K
1,000-4,999 employees = $15K
200 < 1,000 employees = $10K
< 200 employees = $5K

Developer Resources

- Code and documentation available at following links
  - main bpf man page: http://man7.org/linux/man-pages/man2/bpf.2.html
  - tc-bpf man page: http://man7.org/linux/man-pages/man8/tc-bpf.8.html
  - kernel code is available on kernel.org and you can find some examples in kernel samples/bpf/ directory.
  - llvm component is on llvm.org
  - user space bits: https://github.com/iovisor which includes 'examples' directory, readme, etc.

IO Visor Project Summary

FLEXIBILITY

- Programmable, extensible architecture
- Dynamic IO modules that can be loaded and unloaded in kernel at run time without recompilation
- Portable across any platform

PERFORMANCE

- High performance, in-kernel
- Distributed data plane and services without bottlenecks or hairpinning
- Scale-out forwarding without compromise on functionality

- Collaborative, open source project focused on IO and networking functions
- Code already up streamed to Linux kernel
- Hosted by the Linux Foundation with initial IP and code contribution by PLUMgrid
- Formed by industry leaders across systems, software, and silicon

Q&A

www.iovisor.org