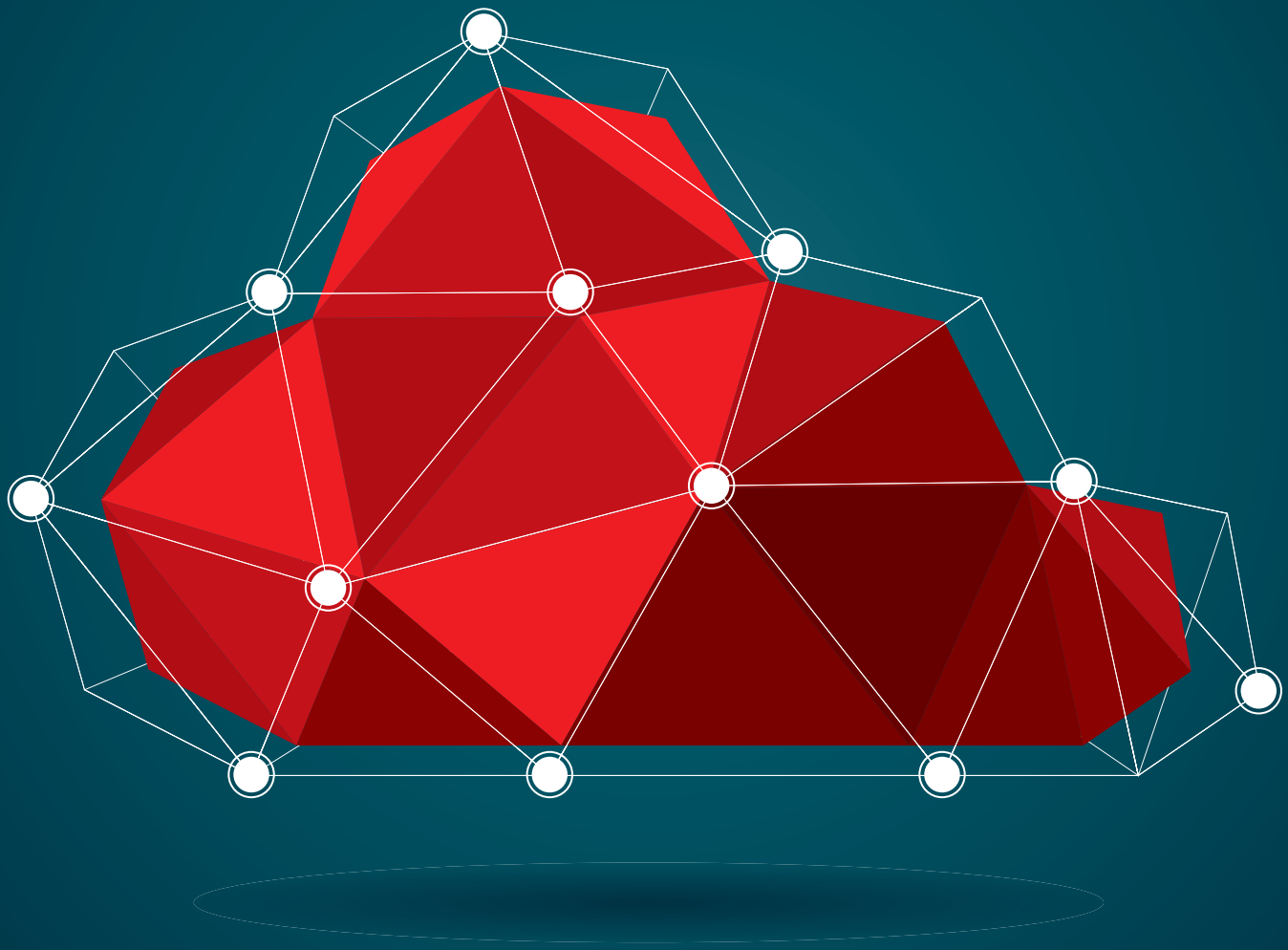
IO VISOR PROJECT

IO Visor: Programmable and Flexible Data Plane for Datacenter's I/O



LINUX FOUNDATION COLLABORATIVE PROJECTS

Introduction

As an industry, we have been building datacenter infrastructure for more than 20 years. If we have learned anything from that experience, it probably is that it doesn't matter how efficient, fast or highly available your new protocols are or how efficient your last design is. There will be new requirements on your infrastructure that will demand a change in your implementation. And those new requirements will appear earlier than you think. Instead of thinking of it as just a networking or security or monitoring challenge the same basic premise can be applied to any I/O problem leading to the need for a new fundamental connectivity paradigm.

Server virtualization proliferation, increased processor density, rapid adoption of solid-state disks and storage virtualization, improvements in application delivery controllers, software defined everything, advent of NFV and accelerated adoption of containers, all have had an influence on your datacenter's infrastructure and how it should be designed and operated . It also puts emphasis on how I/O has to adapt and be agile for unknown technology phenomenons which can hit your datacenter from networking, security, and tracing perspectives in the future.

Most of the above challenges and requirements seem to crystallize around the data plane component of your infrastructure. Whether it's the more traditional network hardware data plane, the new software-based in kernel or a specialized and dedicated one for monitoring, the data plane is the one component that constantly needs to change and adapt to satisfy ever changing requirements.

Why is a new IO approach needed?

In order to build an infrastructure that is ready to accommodate new seen and unseen challenges, you need a **flexible** data plane component that can adapt to new demands and services, and that is **programmable** so that the provisioning of these new services is driven by the application and not by manual intervention (which is prone to more mistakes) and that satisfies the extensibility requirements without performance tradeoffs.

Flexibility and Programmability are the two primary reasons why the network and security service provisioning layers are progressively evolving to be implemented entirely in software, inside the hypervisor. But in addition to these requirements, whether it's networking or security they still need to deliver throughput and features. The technology to future proof your networking and also provide benefits of throughput and features is the goal of IO Visor Project.

What is IO Visor Project?

The IO Visor Project is an open source project and a community of developers to accelerate the innovation, development, and sharing of new IO and networking functions. It brings Universal IO Extensibility to the Linux kernel and gives IO developers the ability to create applications, publish them, deploy them in live systems without having to recompile or reboot a full datacenter.

IO Visor Project provides a programmable data plane and development tools to simplify the creation and sharing of dynamic "IO Modules". At it's heart is IO Visor_Engine, which is a universal in-kernel IO Virtual Machine that provides run-time extensibility. IO Visor_Engine has a set of IO Visor_Plugins to provide functionality to different areas.

Under IO Visor Project, you'll get a set of development tools called IO Visor_Dev_Tools and a set of management and operations tools of the IO Visor_Engine called IO Visor_Tools. For more information how to use and build IO Modules using IO Visor toolkit, you can go to http://github.com/iovisor/bcc

With this framework and toolkit, you can build IO_Modules on top of the IO Visor framework for security, networking, tracing or any other area of your interest.

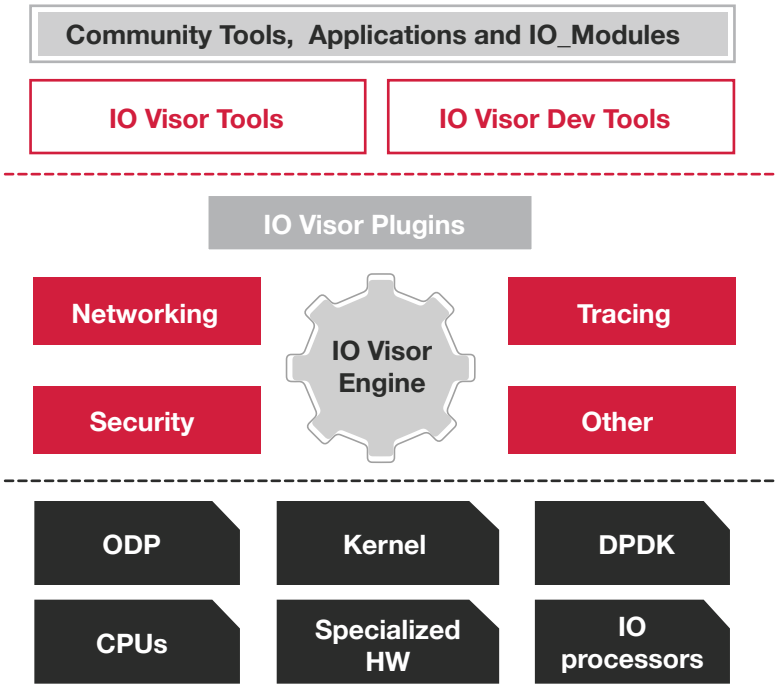


Figure 1: IO Visor

IO Visor project builds on the Linux community to bring open, flexible, distributed, secure and easy to operate technologies that enable any stack to run efficiently on any physical infrastructure.

IO Visor satisfies the needs of the infrastructure developer community helping build any IO for current and future datacenters:

1. **Common** ways to develop and share new IO and networking ideas to keep up with rapidly evolving requirements of SDx, Cloud, NFV, IoT.
2. **Programmable** data plane abstraction and developer tools that allows developers to innovate on top of different HW and SW engines.
3. **Open, flexible, distributed, secure,** and easy to operate IO and networking on any physical infrastructure.
4. **Flexible** technologies enabling any stack to run efficiently, such as in-kernel, load and unload IO modules in run time without recompilation.

What are the benefits of IO Visor?

The IO Visor provides the following benefits to the IO developer community:

- **Flexibility of programmable, extensible architecture** with dynamic IO modules that can be loaded and unloaded in kernel at run time without recompilation.
- **High performance and distributed**, scale-out forwarding without compromise on functionality.

With IO Visor, you can write in-kernel programs that implement atomic networking, security, tracing or any generic IO function. You can also attach these programs to sockets, so that they'll be executed as traffic arrives in the kernel. Let's go through few use cases for using IO Visor in the datacenter.

Networking based on IO Visor: You can have a software instance of your switch, your router, your load balancer, or your security appliance, which can be dynamically loaded and stitched together to form a full network topology, and rendered as a complete network dynamically inside the kernel of your compute node. Traffic can arrive there from your VM, container and traverse the entire network locally within the kernel, and leave the local compute node only when that's required to reach the destination. The biggest benefit of IO Visor is that you can program ANY network logic there (present or future) for ANY new version of your protocol, and then dynamically swap out the existing implementation for a new one.

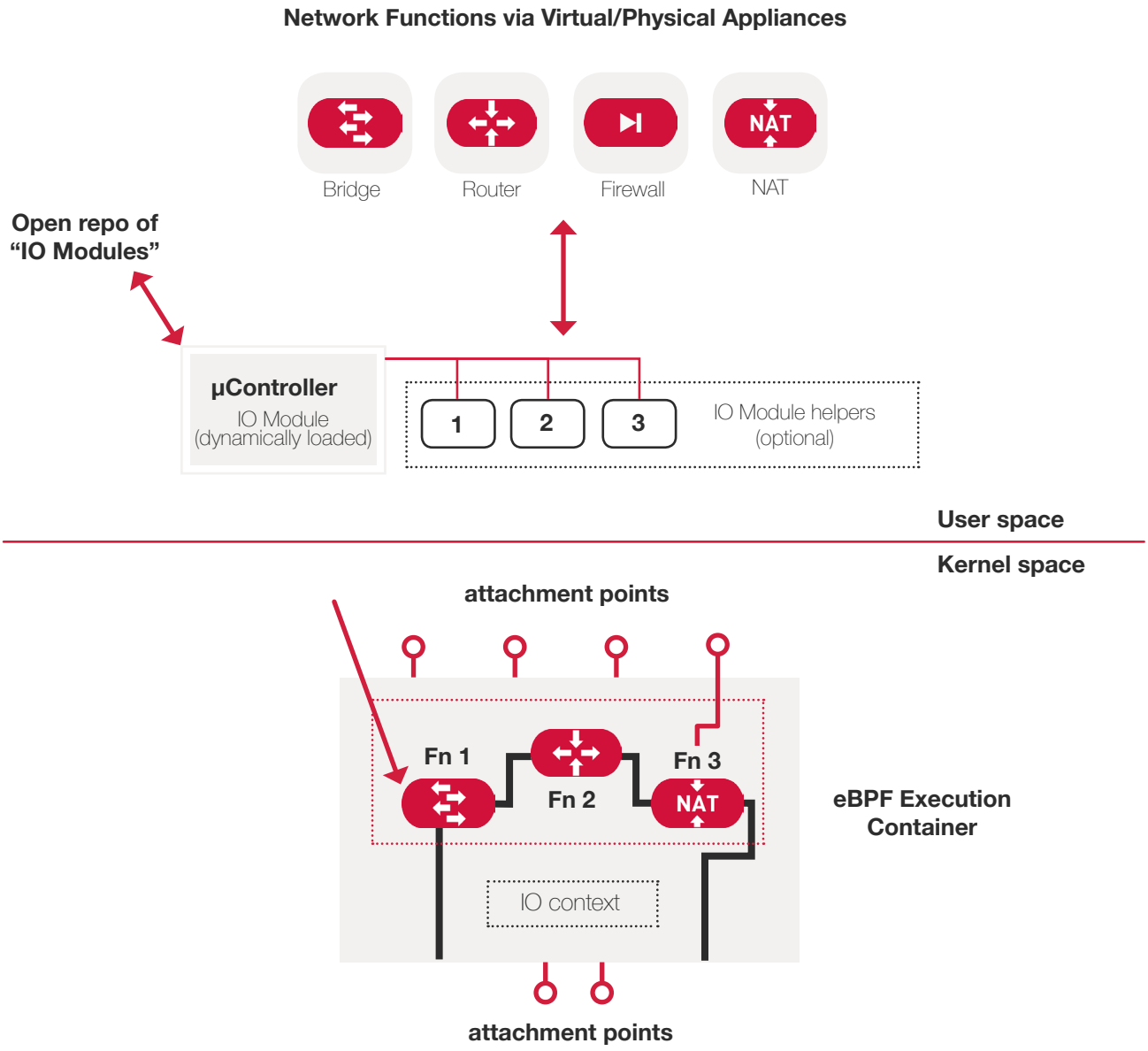


Figure 2: Building Virtual Network Infrastructure with IO Visor

Tracing with IO Visor: IO Module can be used to monitor physical interfaces in your multi-host vxlan environment. The IO Module will keep statistics on the inner and outer IP addresses traversing the interface without impacting live traffic and won't create performance bottlenecks. Developers can then build an application on top of this as shown in image below turning those traffic statistics into a graph showing the current health of the infrastructure.

Security with IO Visor: IO Visor Project enables advanced security functions (micro-segmentation, security groups and full-fledged firewalling) to be implemented in-kernel and distributed providing the optimal enforcement point for application traffic without the need to hairpin traffic through security appliances. An example of its appication to a security problem is seccomp https://en.wikipedia.org/wiki/Seccomp which implements fencing of an application leveraging eBPF as the backend.

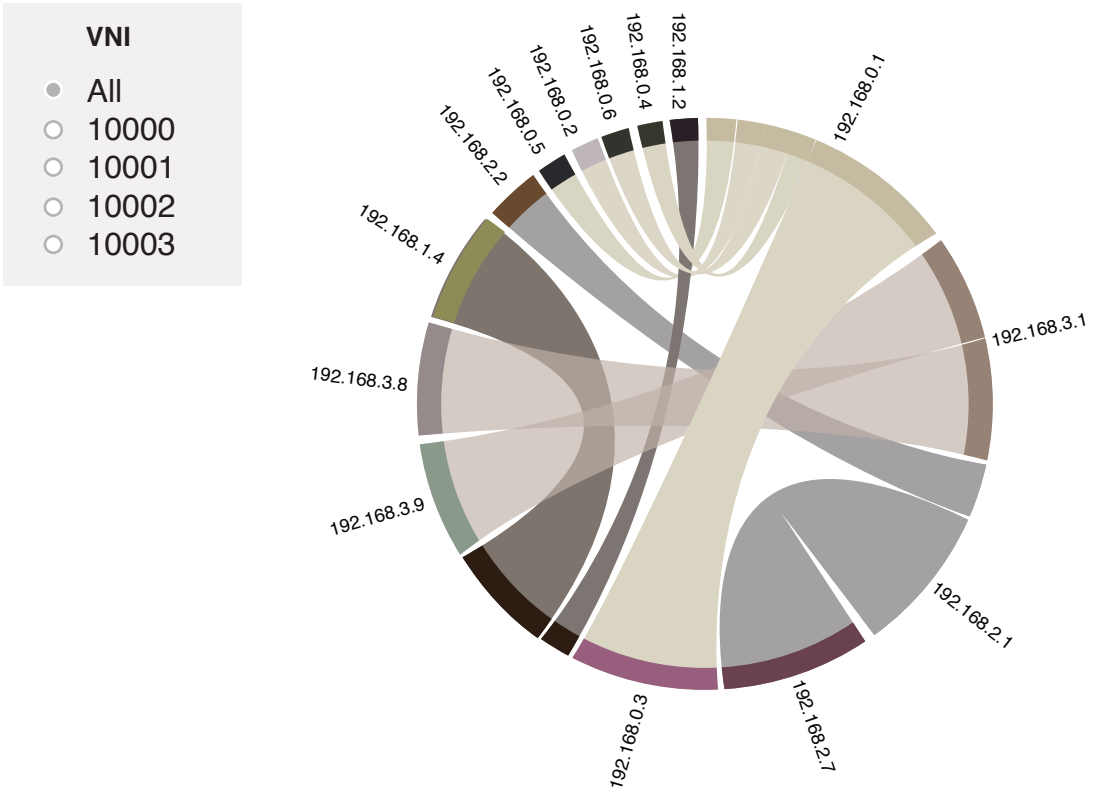
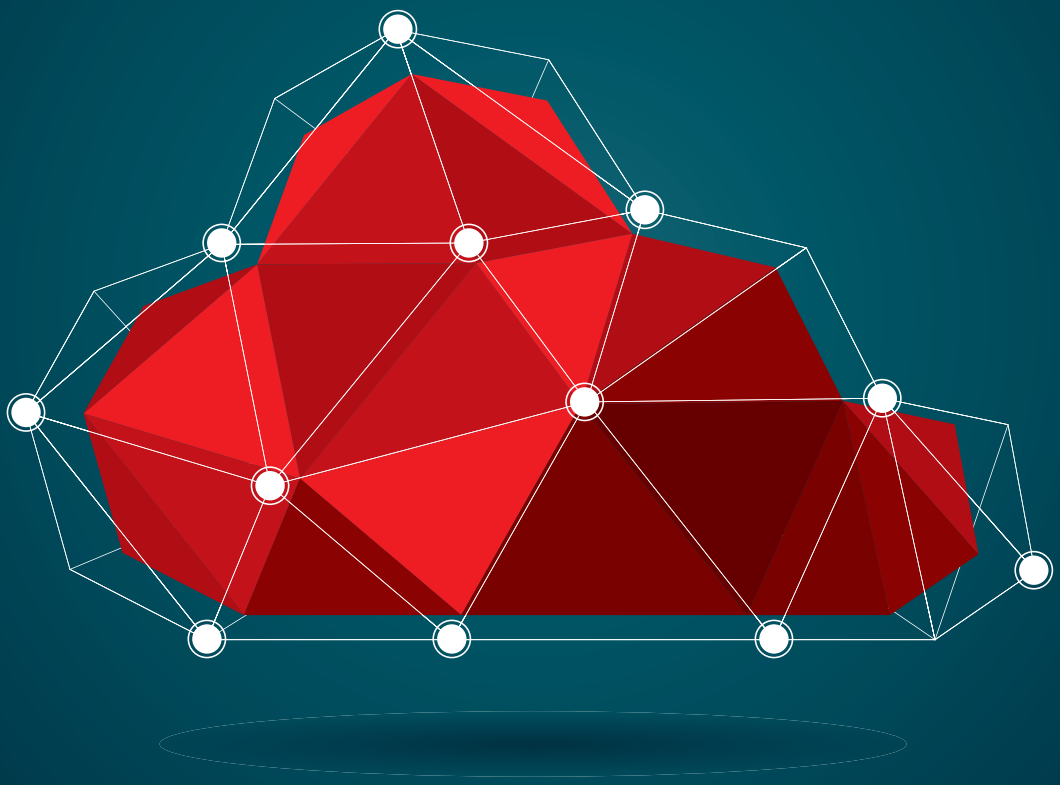


Figure 3: VXLAN Tracing with IO Visor

Conclusion

You can participate, leverage and contribute to IO Visor project to build IO flexibility and extensibility into your datacenter for security, networking, monitoring, tracing or any other applications you can think of. Using IO Visor to leverage the Linux kernel without adding any vendor-specific library or API framework prevents lock-in and allows you to dynamically change your datacenter according to your business needs. IO Visor provides infrastructure / IO developers the ability to create applications, publish them, deploy them in live systems without having to recompile or reboot a full datacenter.



Contact Us

IOVISOR PROJECT

contact-us@iovisor.org
events@iovisor.org
www.iovisor.org