
PLUMgrid
Virtual CPE Solutions
for Service Providers

Speaker Bio

Justin Moore

jmoore@plumgrid.com

- Principal Solutions Architect at PLUMgrid
- Background leading complex technology initiatives in the enterprise
- Work with customers to design, and deploy Cloud and SDN/NFV systems to meet business needs

Why vCPE?

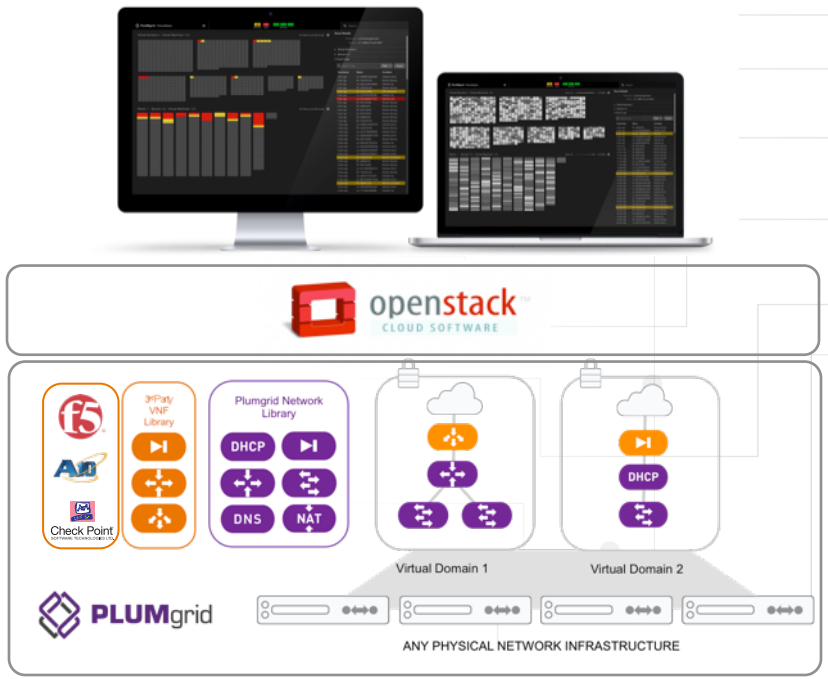
Enable Rich Services, Secure Multi-tenancy, and Automation

Services Rich for ARPU Growth allows operators to "stitch" in any 3rd party services through Service Insertion Architecture

Faster Time to Revenue with network on-demand provisioning, cloud-based services, automation, visibility, and analytics

Lower Capex & Opex with generic hardware, automation, visibility, analytics

Secure Multi-tenancy with built-in micro-segmentation for per tenant traffic isolation via Virtual Domains



Rapid Evolution of Service Delivery

Cloud-based Delivery

- ✓ Network on-demand
- ✓ Faster time to revenue
- ✓ Lower Capex and Opex
- ✓ Higher ARPU

Application Driven

- ✓ Network services as “apps”
- ✓ Download on-demand
- ✓ Chain & run VNFs
- ✓ No need for appliances

Instantaneous

- ✓ Cut provisioning from months to minutes
- ✓ Bring up & down VNFs or virtual networks
- ✓ Upgrade VNFs in run-time

"Customers are willing to pay more for unique, value-added services [Network on Demand] and are not expecting a discount,”

Randall Stephenson, AT&T chairman and CEO, October 2015

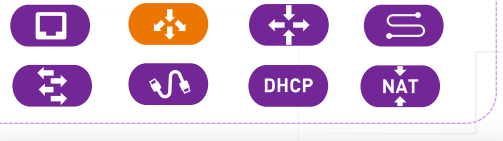
Source: AT&T Analyst Day 2015

PLUMgrid

Traditional to Software Defined Systems

- Custom hardware & OS
- Hardware defined systems
- Manual processes

SDN/NFV



- ✓ Generic hardware
- ✓ Software defined
- ✓ Full automation

PLUMgrid

What is vCPE?

A 3 part solution

Service Insertion / Chaining

NFV / VNF

Segmentation – distributed control - data plane

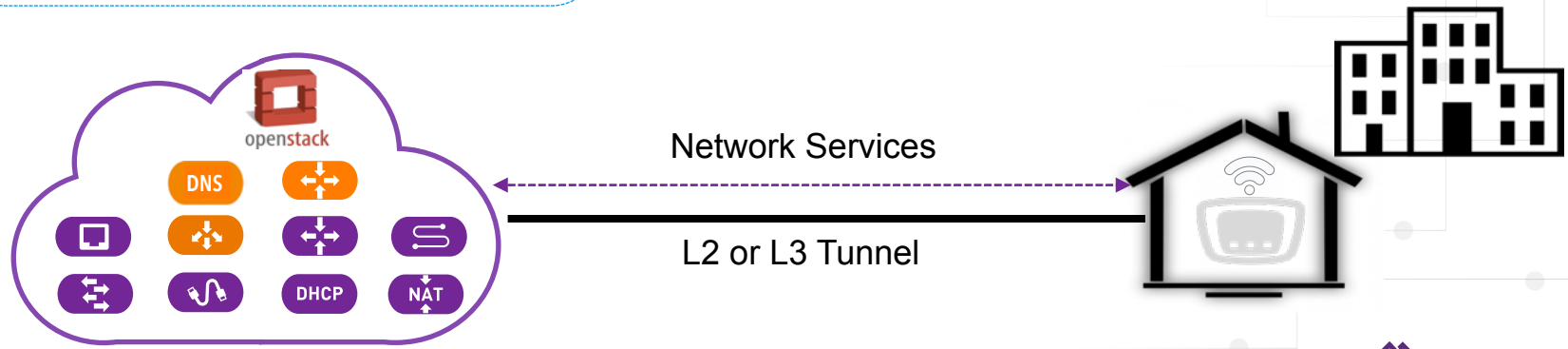
ETSI Virtual CPE Model

Virtual Network Functions in the Cloud

L3-L7 functions from services cloud

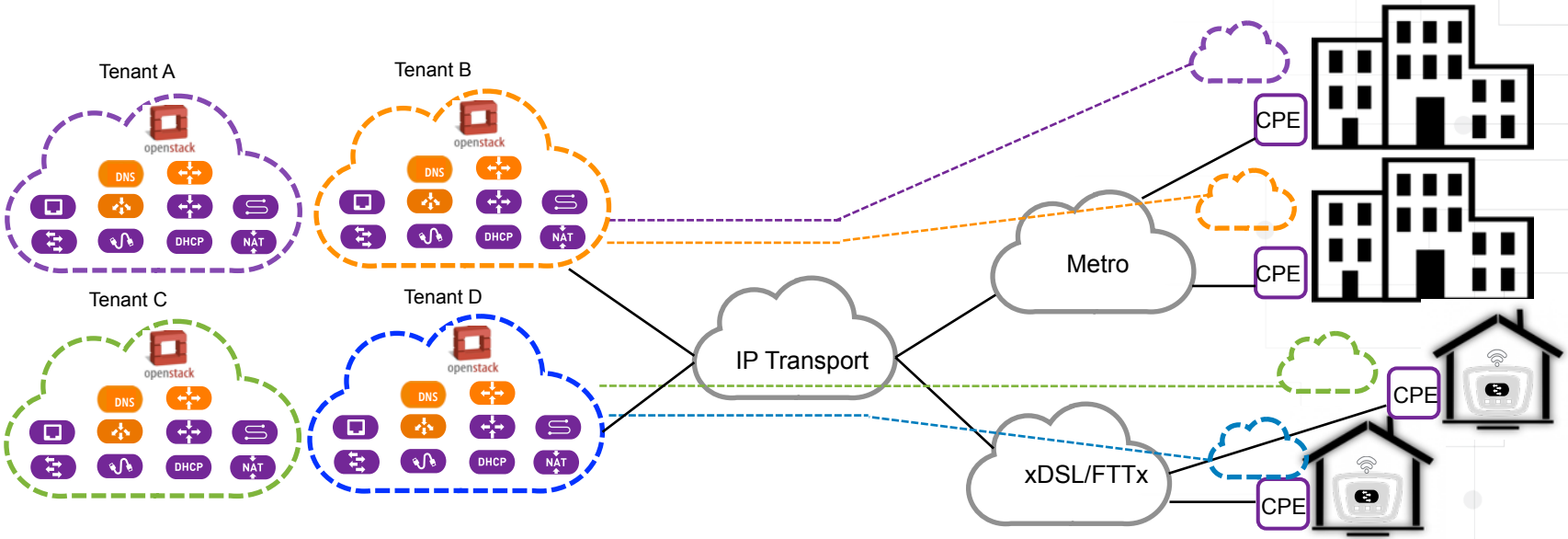
- Routing
- Switching
- Firewall
- NAT
- Load Balancer

Simplified CPE



PLUMgrid

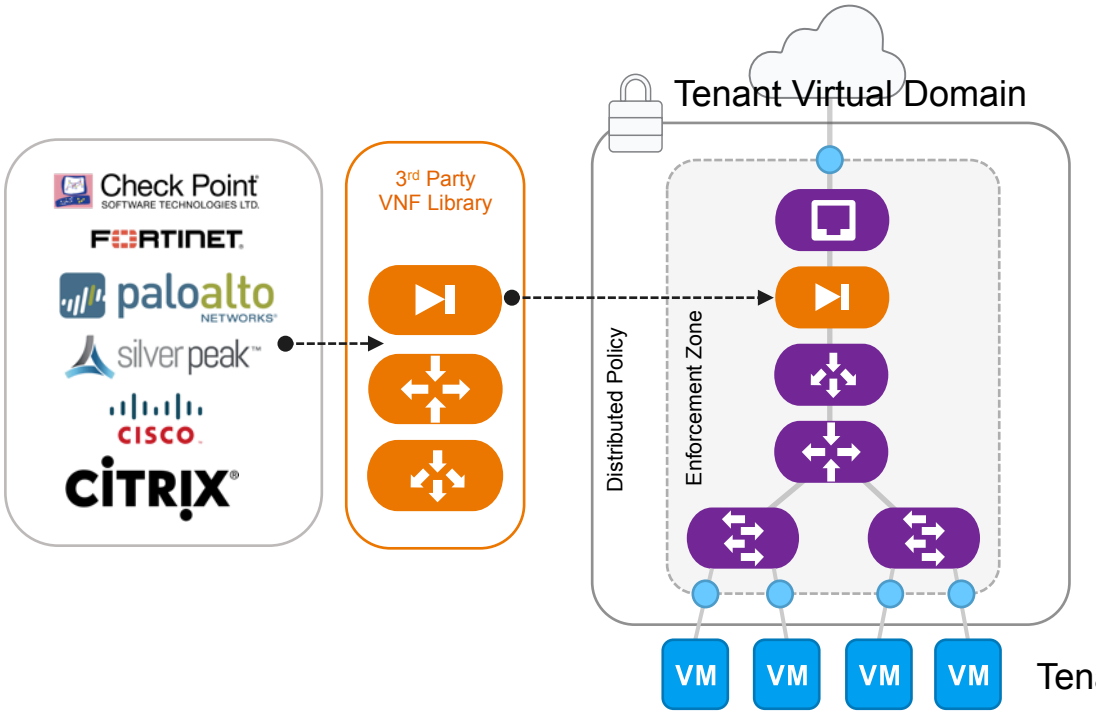
Secure Micro-Segmentation for Multi-Tenancy



- ✓ Micro-segmented virtual domain per tenant
- ✓ Complete separation of traffic, policies, and network functions
- ✓ Secure with each domain created on-demand, no hardware changes

PLUMgrid

Service Insertion Architecture & Chaining



- Seamless insertion of **3rd party services**
- Enables Service Function Chaining in automated + simplified way
- Accelerates service creation and delivery

PLUMgrid

SDN / NFV model

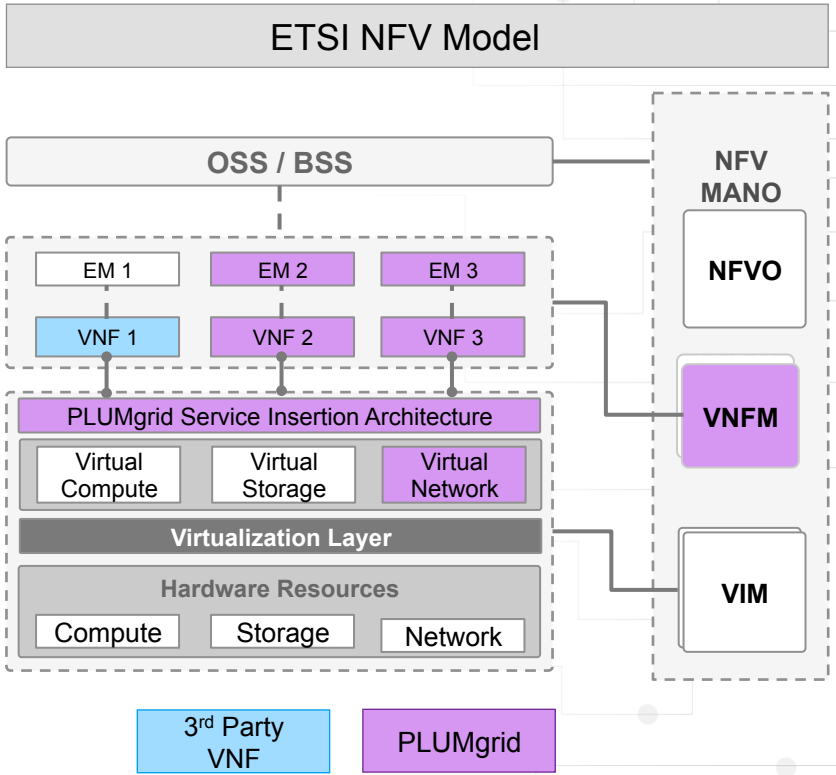
Separation of control and data planes

Control Plane

- Virtualized, runs on redundant controllers
- Provides “Remote control” of services
- Easy to provision, troubleshoot, patch or upgrade
- Runs on generic hardware

Data Plane

- Virtualized, runs on end user nodes/CPE
- Enables local forwarding and services
- Easy to provision, troubleshoot, patch or upgrade
- Runs on generic hardware



IO Visor Project

1 Open Source & Community

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

2 Programmable Data Plane

- Advancing in-kernel modular IO loadable at run-time without recompilation or reboot
- Programmable data plane and development tools to simplify the creation and sharing of dynamic “IO Modules”

PLUMgrid

Spearheading IO Visor Project

A Linux Foundation project started by PLUMgrid
IO Visor up streamed into the Linux Kernel since 3.16

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

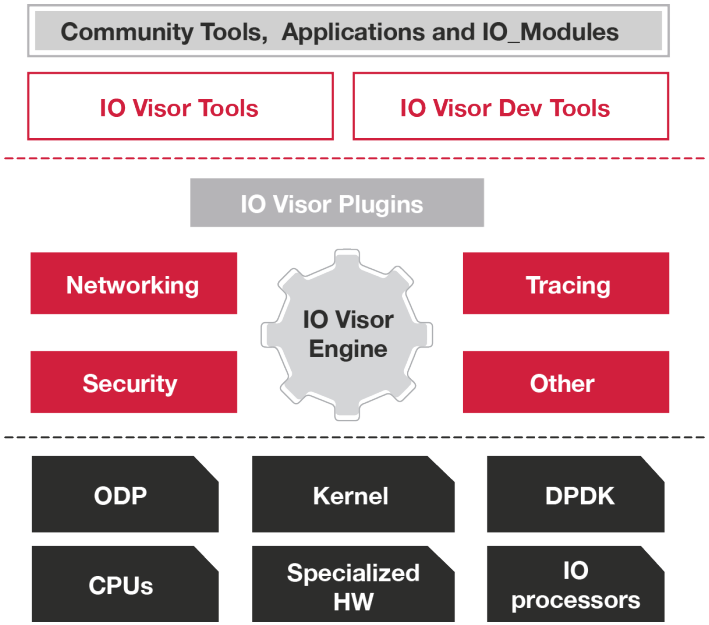
- High performance, in-kernel
- Distributed data plane and services without bottlenecks or hairpinning
- Scale-out forwarding without compromise on functionality

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

www.iovisor.org

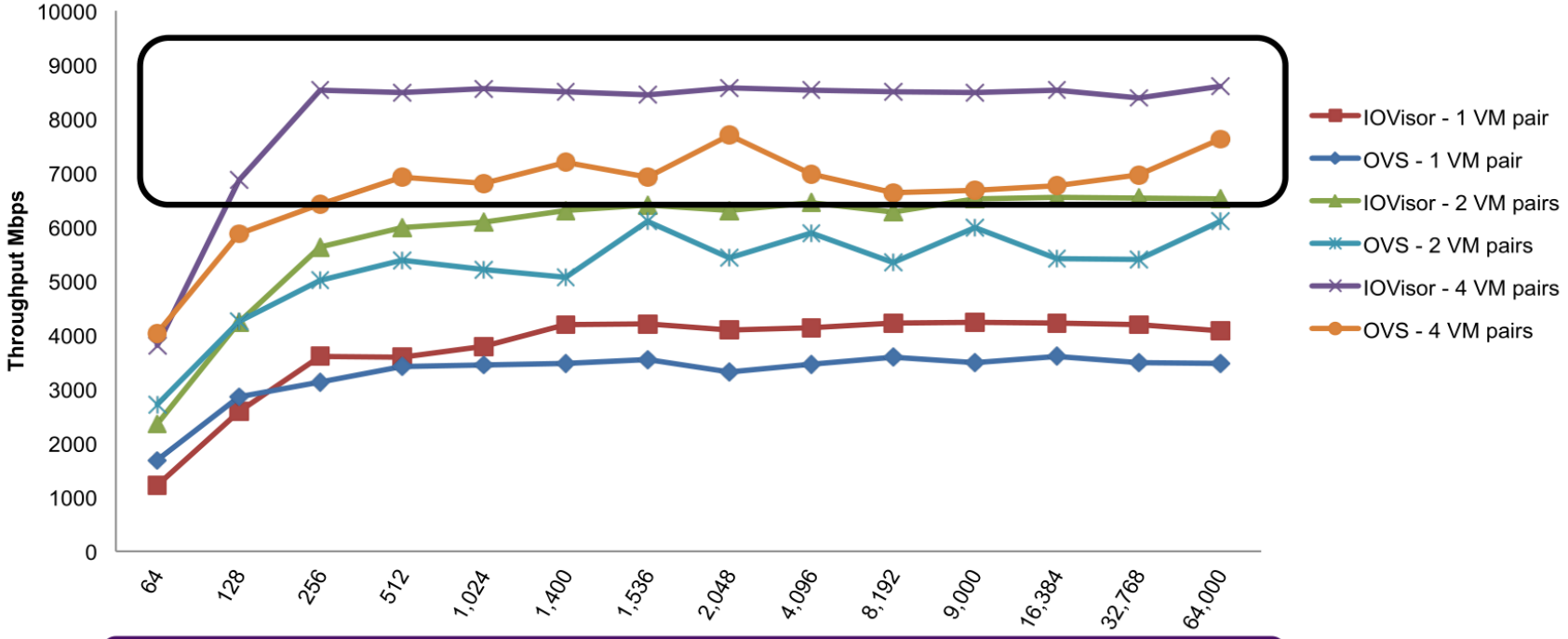
IO VISOR PROJECT

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**

Layer 2 Tenant Topology – Throughput Performance

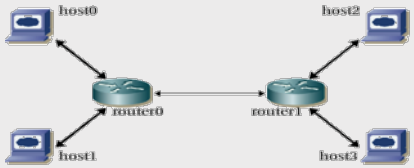


Outperforming average ~19% - ~28%

PLUMgrid

IO Visor Use Cases

Networking



✓ Fully distributed virtual networks across multiple compute nodes

Tracing

✓ Real-time distributed analytics platform for VXLAN infrastructure

Security

✓ Fencing of user space components in Kernel

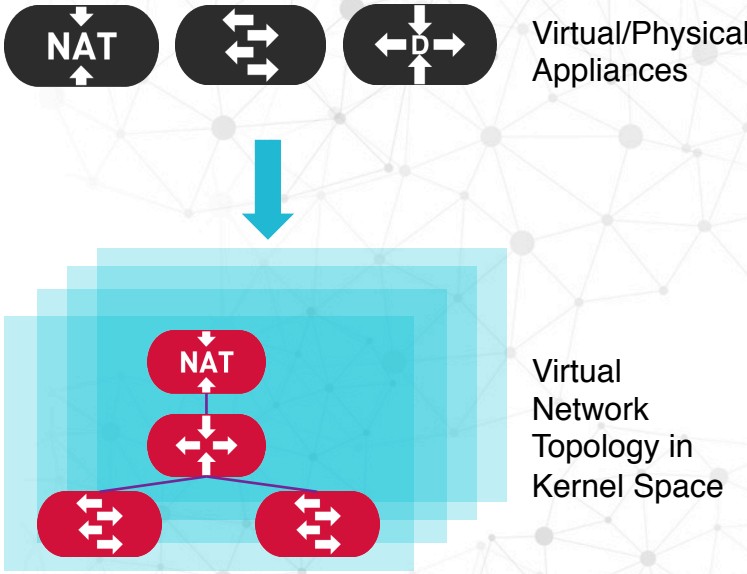
Storage Monitoring

✓ Storage System Monitoring in real-time

PLUMgrid

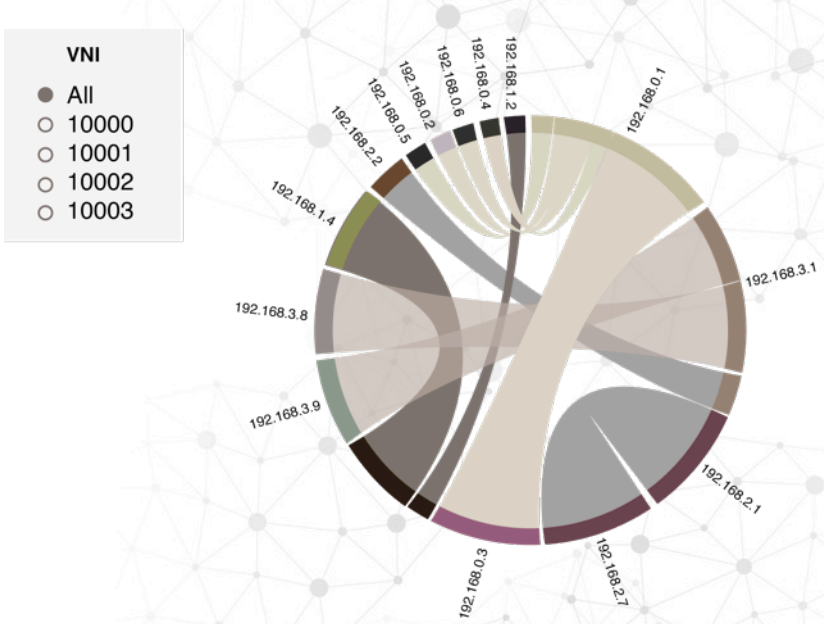
IO Visor Project Use Cases Example: Networking

- 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: 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 Project Use Cases Example: Storage Monitoring

- IO Visor is used to build a **real-time, distributed analytics platform** that monitors the health of the storage subssytem
- 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/tools

```
# ./biolatency
Tracing block device I/O... Hit Ctrl-C to end.
^C
     usecs               : count     distribution
       0 -> 1            : 0        |                                        |
       2 -> 3            : 0        |                                        |
       4 -> 7            : 0        |                                        |
       8 -> 15           : 0        |                                        |
      16 -> 31           : 0        |                                        |
      32 -> 63           : 0        |                                        |
      64 -> 127          : 1        |                                        |
     128 -> 255          : 12       |********                                |
     256 -> 511          : 15       |**********                              |
     512 -> 1023         : 43       |*******************************         |
    1024 -> 2047         : 52       |**************************************|
    2048 -> 4095         : 47       |**********************************      |
    4096 -> 8191         : 52       |**************************************|
    8192 -> 16383        : 36       |**************************              |
   16384 -> 32767        : 15       |**********                              |
   32768 -> 65535        : 2        |*                                       |
   65536 -> 131071       : 2        |*                                       |

The latency of the disk I/O is measured from the issue to the device to its
completion. A -Q option can be used to include time queued in the kernel.
```

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

THANK YOU!

Keep in Touch and Contact Us

(408) 800-7586

www.plumgrid.com

5155 Old Ironsides Dr.
Suite 200
Santa Clara, CA 95054

PLUMgrid