Software Refined Networking

The Path To Hell Is Paved With Good Abstraction...

Author: Christofer Hoff

...with thanks to @ioshints, @etherealmind and @networkstatic for the always-interesting discussions that prompted me to create this

PacketFilter
Masters Of the Flow

Software Refined Networking
The Path To Hell Is Paved With Good Abstraction

@Beaker

Tweet It If You Like!

$ cat ~beaker/TOC.txt > less

- What This Talk Is About
- What is SDN?
- How Does SDN Work?
- Why You Should Care

What This Talk Is About

Oh, look, it's this year's "Cloud"

Fundamentally, Networking and Security Are Undergoing a Disruptive Renaissance Of Sorts Thanks To:

- Virtualization
- Software Defined Data Center (SDDC)
- Software Defined Networking (SDN)
- (Hybrid) Cloud Computing
- Automation
- The DevOps Culture

2008

2009

2010

2010

2011

2012

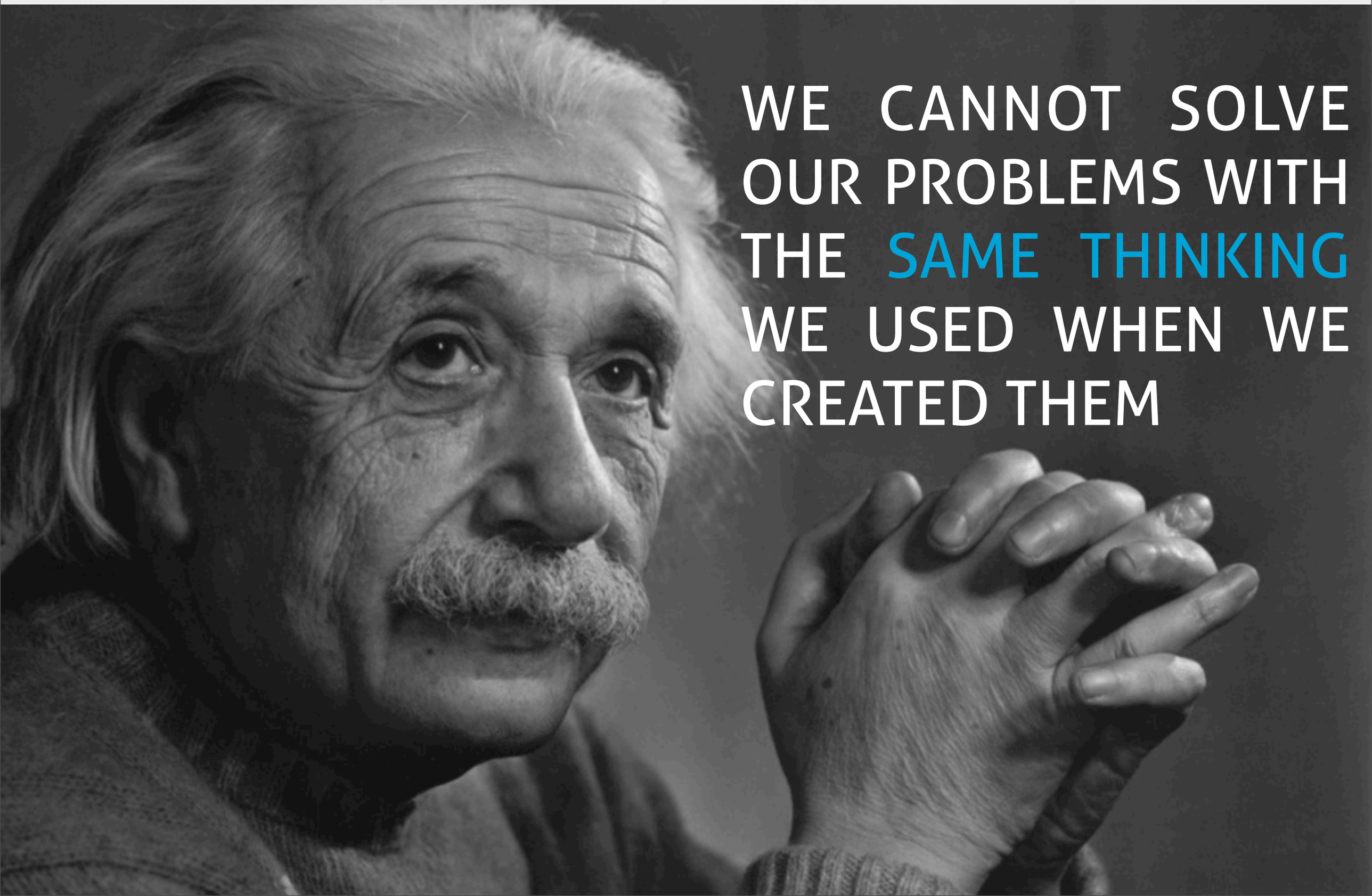
This Talk Touches On Much Of This, and What It Will Mean To You From a Security Perspective

What Is SDN?

Beyond The Hype

PacketFilter
Masters Of the Flow

Software Refined Networking
The Path To Hell Is Paved With Good Abstraction



So, What Is Software ~~Defined~~ Refined Networking?

- Can be thought of as software-enabled abstraction that segregates capabilities and functional "planes" in networking/security solutions
- An "API" or "Compiler" layer for abstracting networking & services
- Enables an operational model that focuses on *programming* a distributed network instead of *configuring* individual devices
- The networking version of DevOps
- Facilitates disambiguation of services by further decoupling software and services from hardware
- Provides for more agile application & service delivery ; allows leveraging automation to reduce TTP (time to provision)

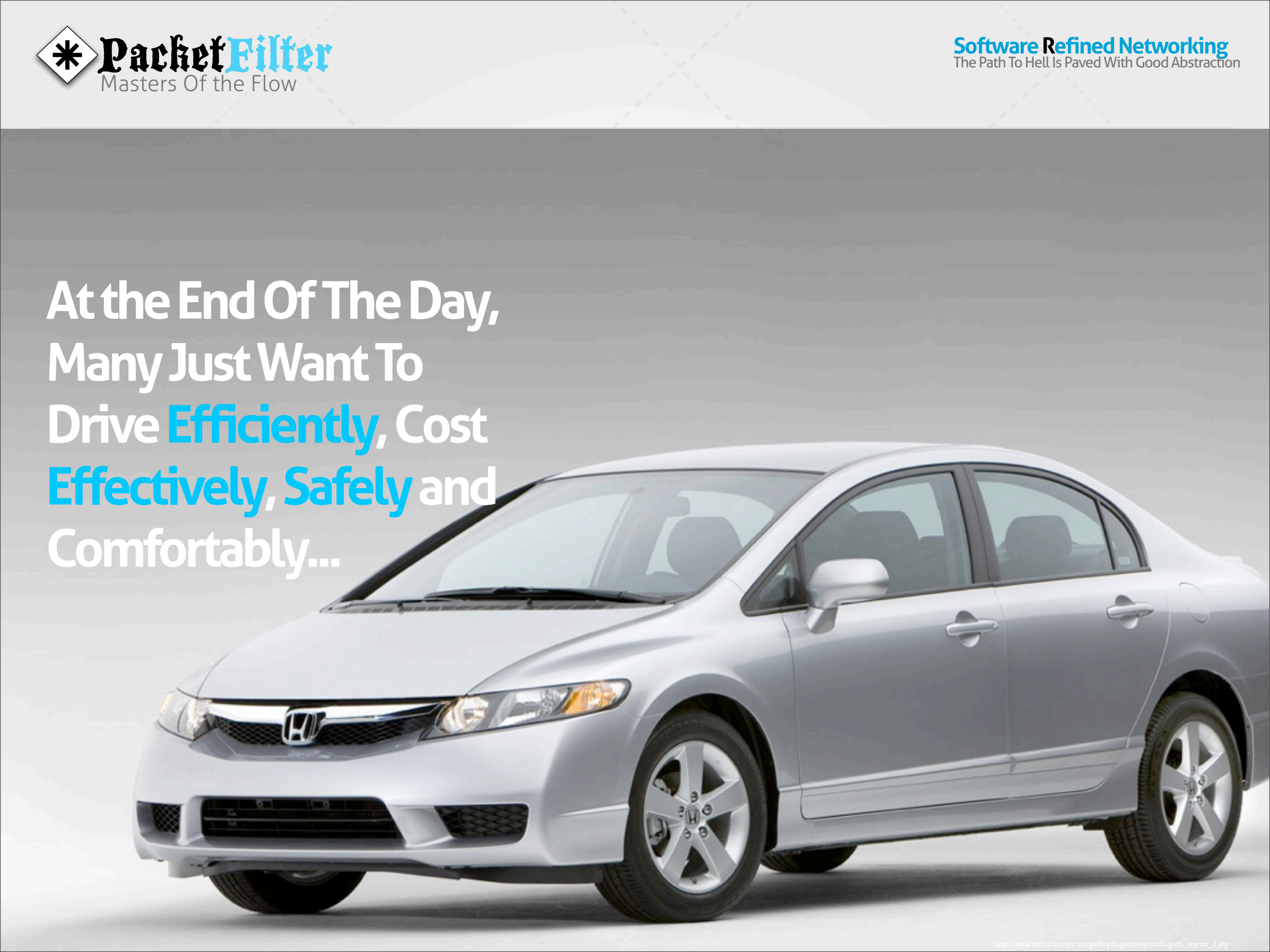
Any Gearheads?

I mean beyond Carbon Fiber Stick-Ons & Angry Swarm Of Bee Exhausts

PacketFilter
Masters Of the Flow

Software Refined Networking
The Path To Hell Is Paved With Good Abstraction

At the End Of The Day, Many Just Want To Drive Efficiently, Cost Effectively, Safely and Comfortably...



http://www.thecarsecrets.net/gallery/bugatti-veyron/bugatti_veyron_2.jpg

PacketFilter
Masters Of the Flow

Software Refined Networking
The Path To Hell Is Paved With Good Abstraction

...Some wish to transport large workloads effortlessly based on a powerful platform that is extensible and utilitarian...

http://www.thecarsecrets.net/gallery/bugatti-veyron/bugatti_veyron_2.jpg

PacketFilter
Masters Of the Flow

Software Refined Networking
The Path To Hell Is Paved With Good Abstraction

...While Some Want To Be Fast, More Agile and Responsive

http://www.thecarsecrets.net/gallery/bugatti-veyron/bugatti_veyron_2.jpg

...and some just care about convenience balanced with general availability without worrying about maintenance

zipcar®

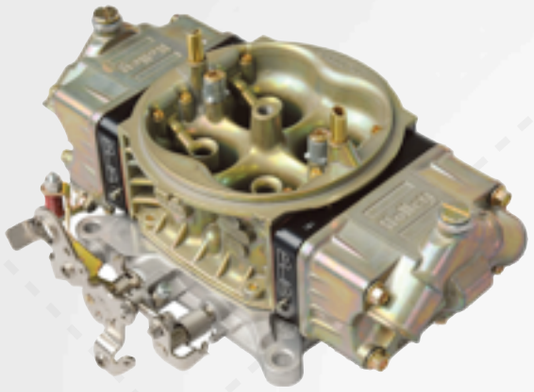
wheels when you want them

http://www.thecarsecrets.net/gallery/bugatti-veyron/bugatti_veyron_2.jpg

PacketFilter
Masters Of the Flow

Software Refined Networking
The Path To Hell Is Paved With Good Abstraction

Classical Car Tuning



Carbs

- Well understood basic designs, "common knowledge," choices: basic vs. high performance
- Cost effective initial purchase, tuning/maintenance costs higher over time
- "Kinetic" or "Mechanical" tuning; jets, floats, accelerator pumps, venturis etc...mostly open loop operation
- "Language" for setup/adjustment is coarse & imprecise
- Generally accessible, basic combustion theory. Pop the hood
- Reasonably easy to get into a basic state of tune & relatively easy troubleshooting but fine-tuning is a black art
- Repair and upgrades relatively straightforward, insensitive to other external modifications and "environmentals" but ultimately lots of tinkering/trial & error

Software Defined...

EFI/DFI

- Specialized knowledge, trend toward proprietary, few manufacturers but adaptable across use cases
- Expensive initial purchase, lower TCO over time
- Electronic "black box" with closed-loop & programmatic capability
- Interfaces like OBD enable standardized & explicit "language"
- More and more specialized and closed; explicit knowledge required but far more precise and adaptive
- Requires expertise, sophisticated equipment and in many cases owners are locked out (See also: DMCA)
- Debugging and RCA difficult without special tools, cascading system faults complex to isolate, feature upgrades possible and easier to address system changes, configuration and updates

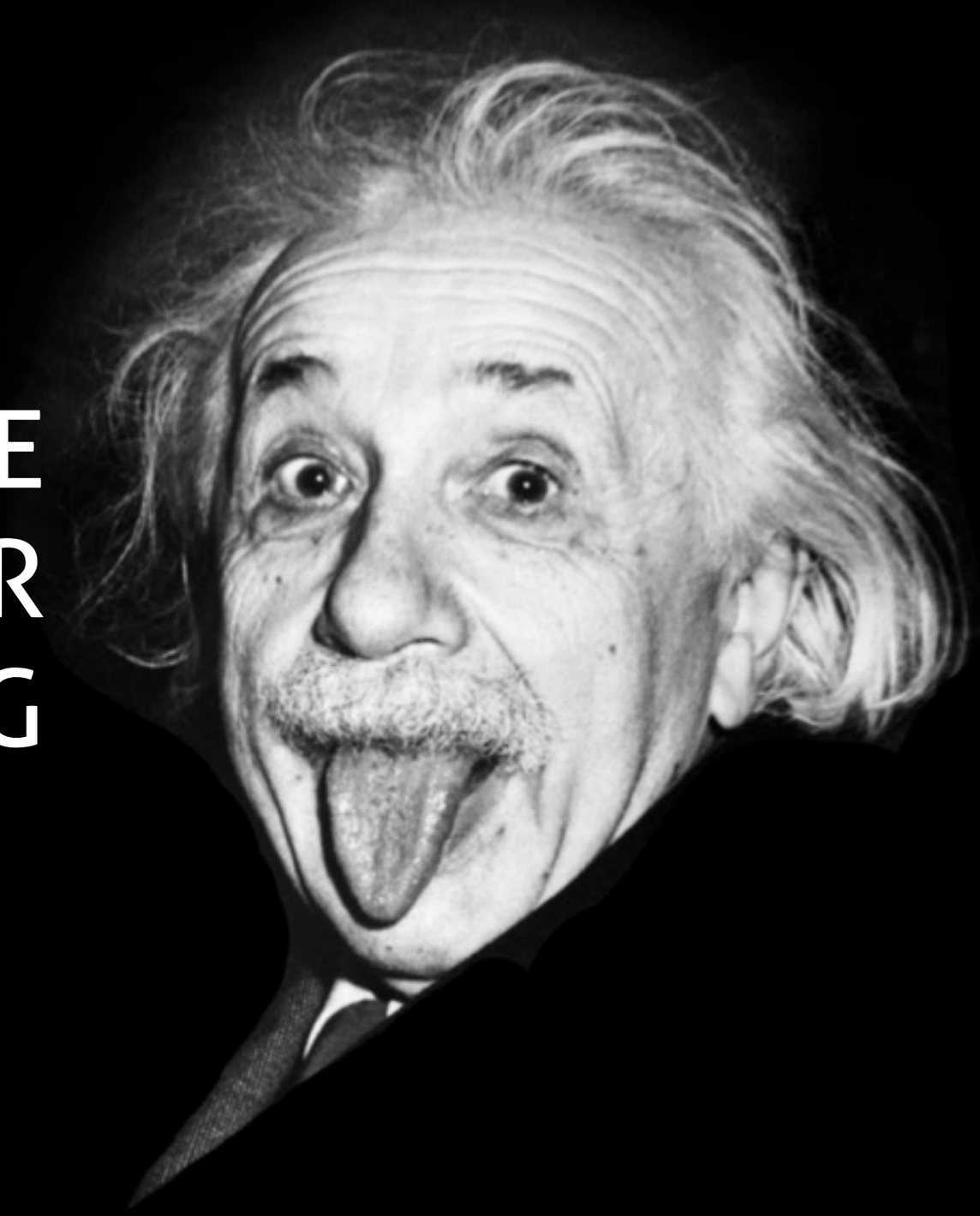
How Does SDN Work?

Making Sense Of the Magic

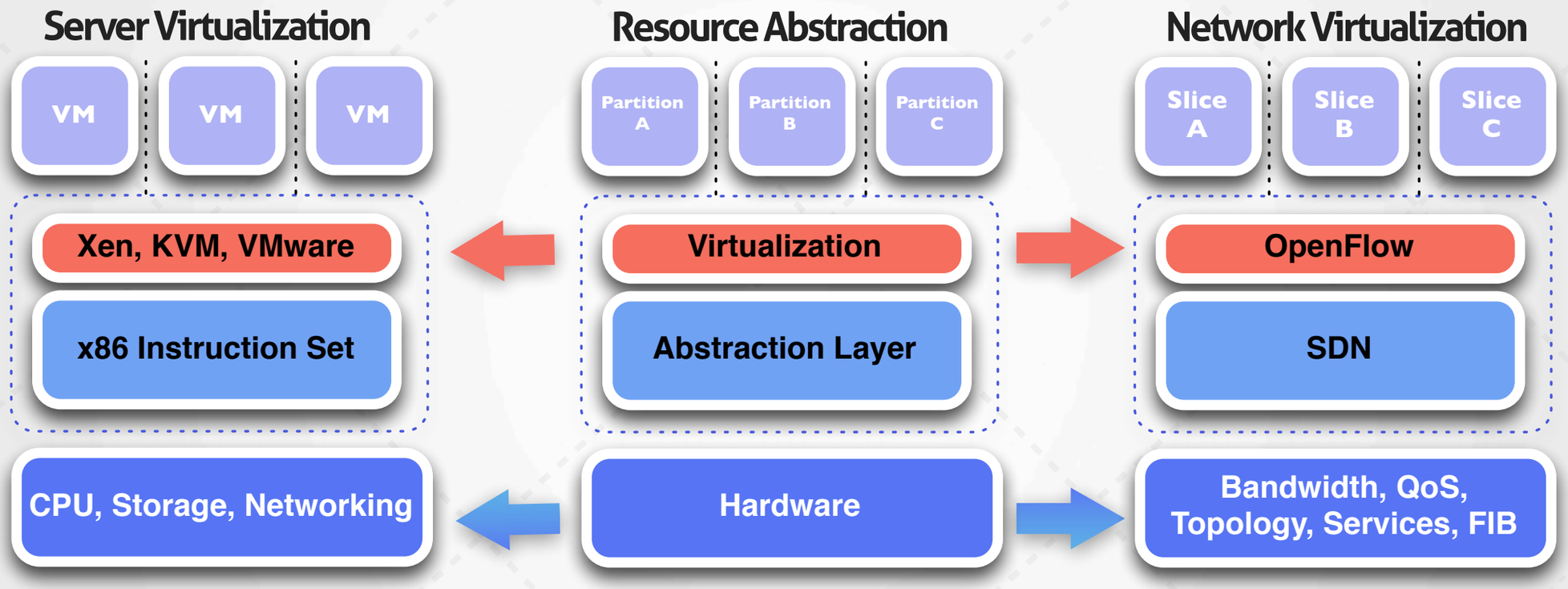
PacketFilter
Masters Of the Flow

Software Refined Networking
The Path To Hell Is Paved With Good Abstraction

INSANITY:
DOING THE SAME THING OVER & OVER & EXPECTING DIFFERENT RESULTS



Visualizing The Change...



Evolution: From FIB & RIB To SDN Ad Lib

Networking	Technology
Shared	
Bridged	
Switched	
Routed	
Virtualized	
Tunneled	

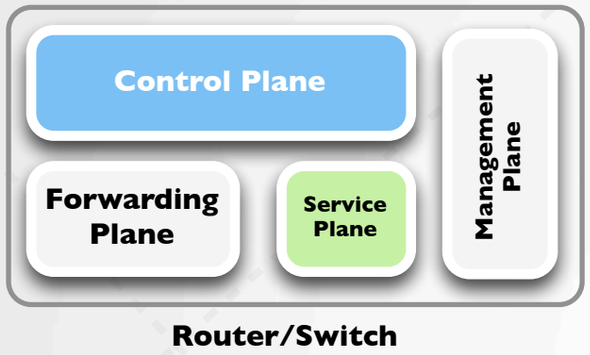
Networking Architecture

Flat
3-Tier
2-Tier
Flat
Virtualized
Overlays

PacketFilter
Masters Of the Flow

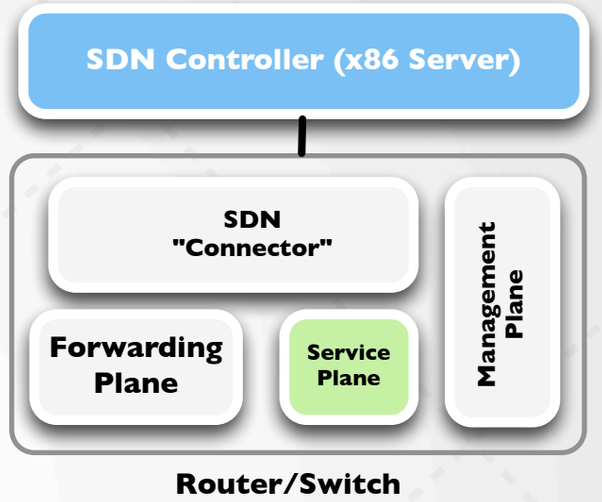
Software Refined Networking
The Path To Hell Is Paved With Good Abstraction

Classical Networking (Hardware-Centric)



- A Traditional switch or router features a control and forwarding plane co-located within the same physical device
- The Control Plane manages the control protocols and instantiates forwarding instructions in the Forwarding Plane.
- The Forwarding Plane generally utilizes merchant silicon, ASICs, or FPGAs and makes use of simple structured lookup tables or hardware such as Content Addressable Memory (CAM) or Ternary CAM (TCAM) upon which to decide where to forward frames/ packets

Software Defined...



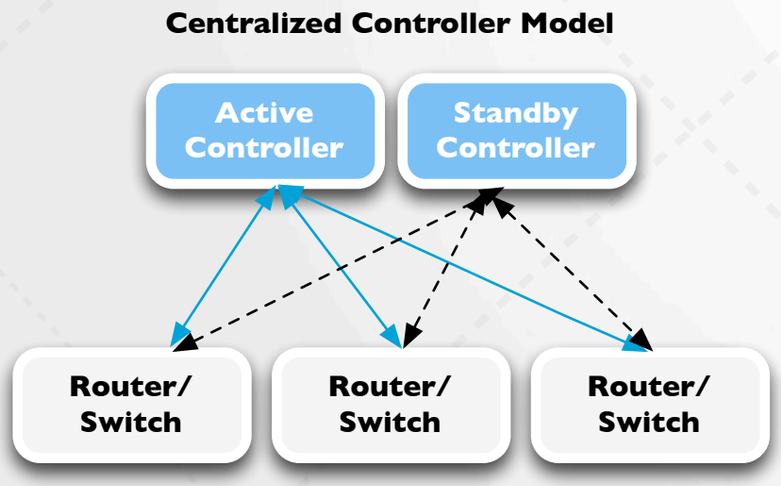
- SDN provides for the separation of the various "planes" in networking, including : **Control, Forwarding**, Management and Services
- In SDN, the Forwarding Plane utilizes software, merchant silicon, ASICs or FPGAs whilst the Control Plane runs as software on physically separate general-purpose x86 hardware
- The Control Plane (Controller) algorithmically calculates the topology of the network in software
- The Controller instantiates or updates Forwarding Plane instructions via API calls using an open or proprietary protocol

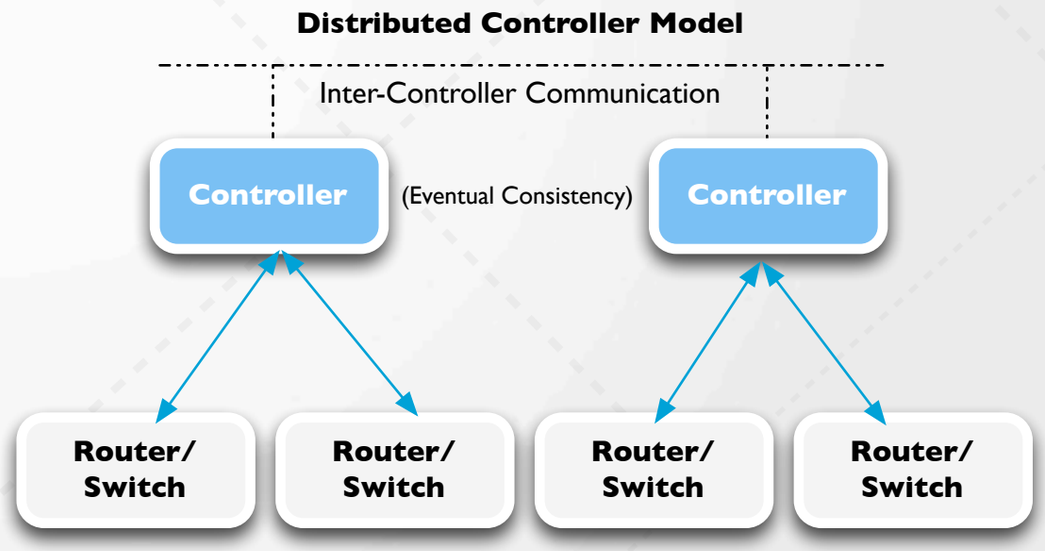
NFV?: Network Function Virtualization?

- Layer 0-3 technology and standards don't enjoy a high velocity of change
- Layer 4-7 Services, however, do
- Custom ASIC/FPGA hardware is great for L0-3, not so much for L4-7
- x86 is great for L4-7 services
- Service Providers (and to some extent Enterprises) must spend millions of dollars to deal with OSS/BSS (management/billing) system integration and additional hardware to deliver new services
- NFV is a methodology to enable the offload of L4-7 services from L0-3 devices to x86-based service delivery platforms to enable better cost control, limit platform lock-in, ease service integration, simplify management, scale better and improve feature velocity
- NFV is very much still hardware-bound; offload L4-7 network functions
- SDN is a superset of NFV

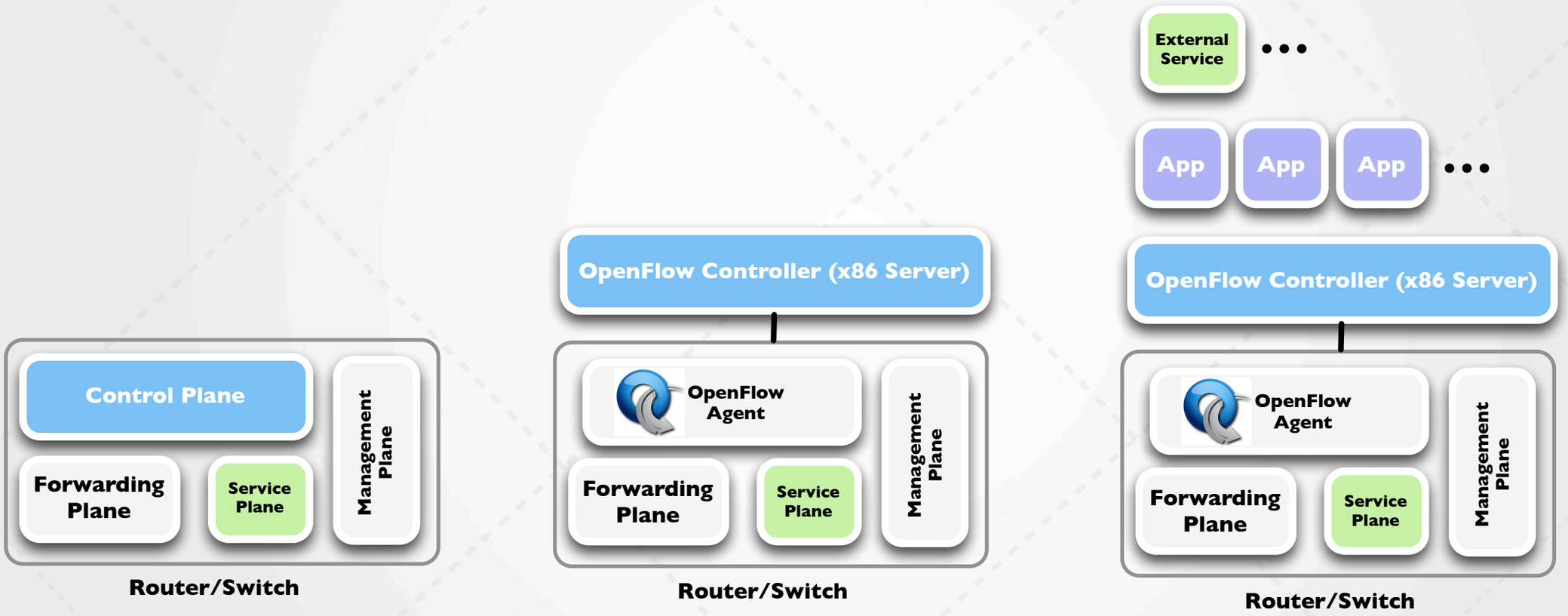
Centralized vs Distributed Control...

- Controller architecture can rely on centralized, distributed or a hybrid model -- and there will likely be many of them
- Communication between the controller(s) and devices utilize APIs/protocols (OpenFlow, XMPP, BGP, etc.) over a (hopefully) secure channel such as TLS
- Coherence, convergence and (eventual) consistency are challenging distributed networking computing problems





An Interesting & Popular Example Of SDN:
OpenFlow

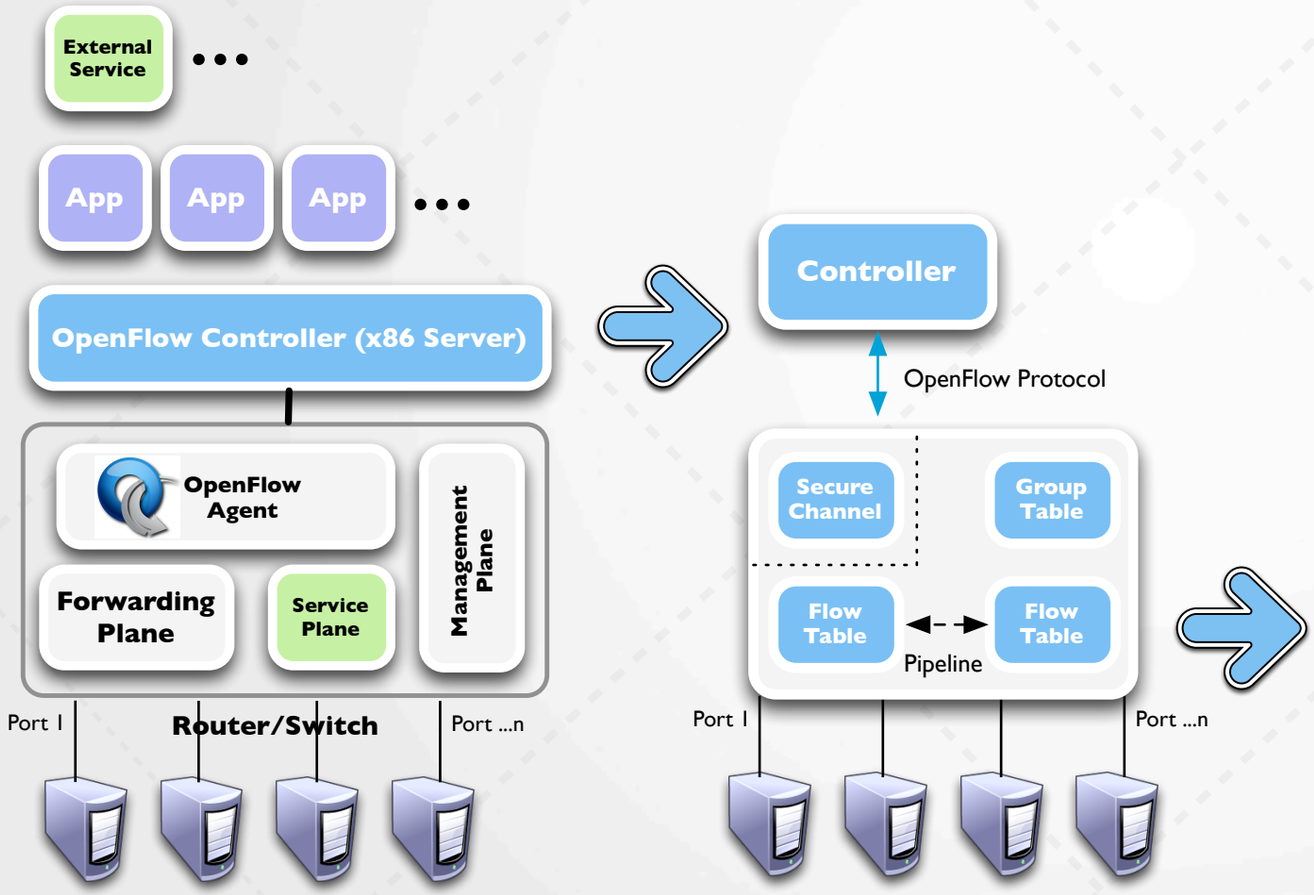


Please Note: OpenFlow is an example *protocol* implementation -- a form of SDN. SDN is **not** OpenFlow exclusively.

PacketFilter
Masters Of the Flow

Software Refined Networking
The Path To Hell Is Paved With Good Abstraction

A Grossly Understated Overview Of OpenFlow Technicalities



-VLAN Switching

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	00:1f..	*	vlan1	*	*	*	*	*	port6

-Routing

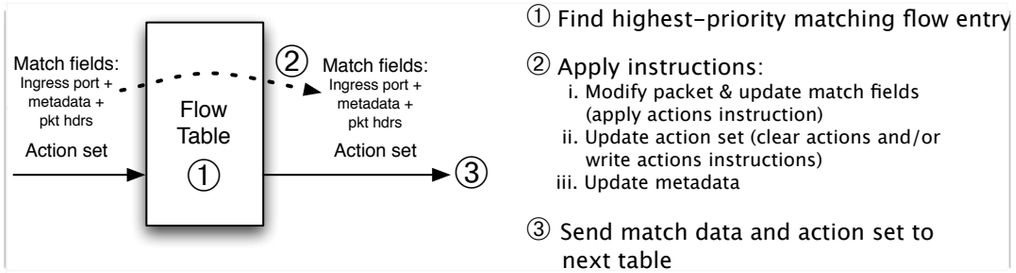
Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	*	*	*	*	5.6.7.8	*	*	*	port6

-Firewall

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	*	*	*	*	*	*	*	22	drop

-Flow Switching

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
port3	00:20.	00:1f..	0800	vlan1	1.2.3.4	5.6.7.8	4	17264	80	port6, port7,



① Find highest-priority matching flow entry

② Apply instructions:
- i. Modify packet & update match fields (apply actions instruction)
- ii. Update action set (clear actions and/or write actions instructions)
- iii. Update metadata

③ Send match data and action set to next table

See ONF OpenFlow v1.3 Spec: https://www.opennetworking.org/images/stories/downloads/specification/openflow-spec-v1.3.0.pdf

ANY PROBLEM IN COMPUTER SCIENCE CAN BE SOLVED WITH ANOTHER LAYER OF ~~INDIRECTION~~ ABSTRACTION

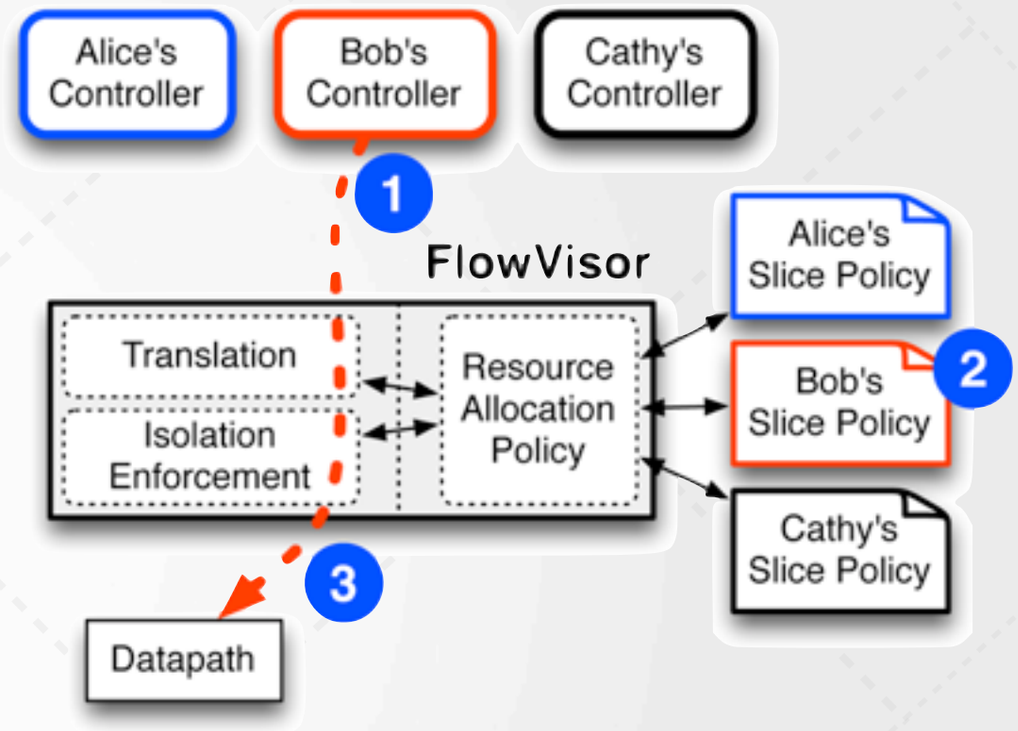
FlowVisor: OpenFlow...Abstracted

- FlowVisor is a special purpose OpenFlow controller that acts as a transparent proxy between OpenFlow switches and multiple OpenFlow controllers

- FlowVisor creates rich "slices" of network resources and delegates control of each slice to a different controller

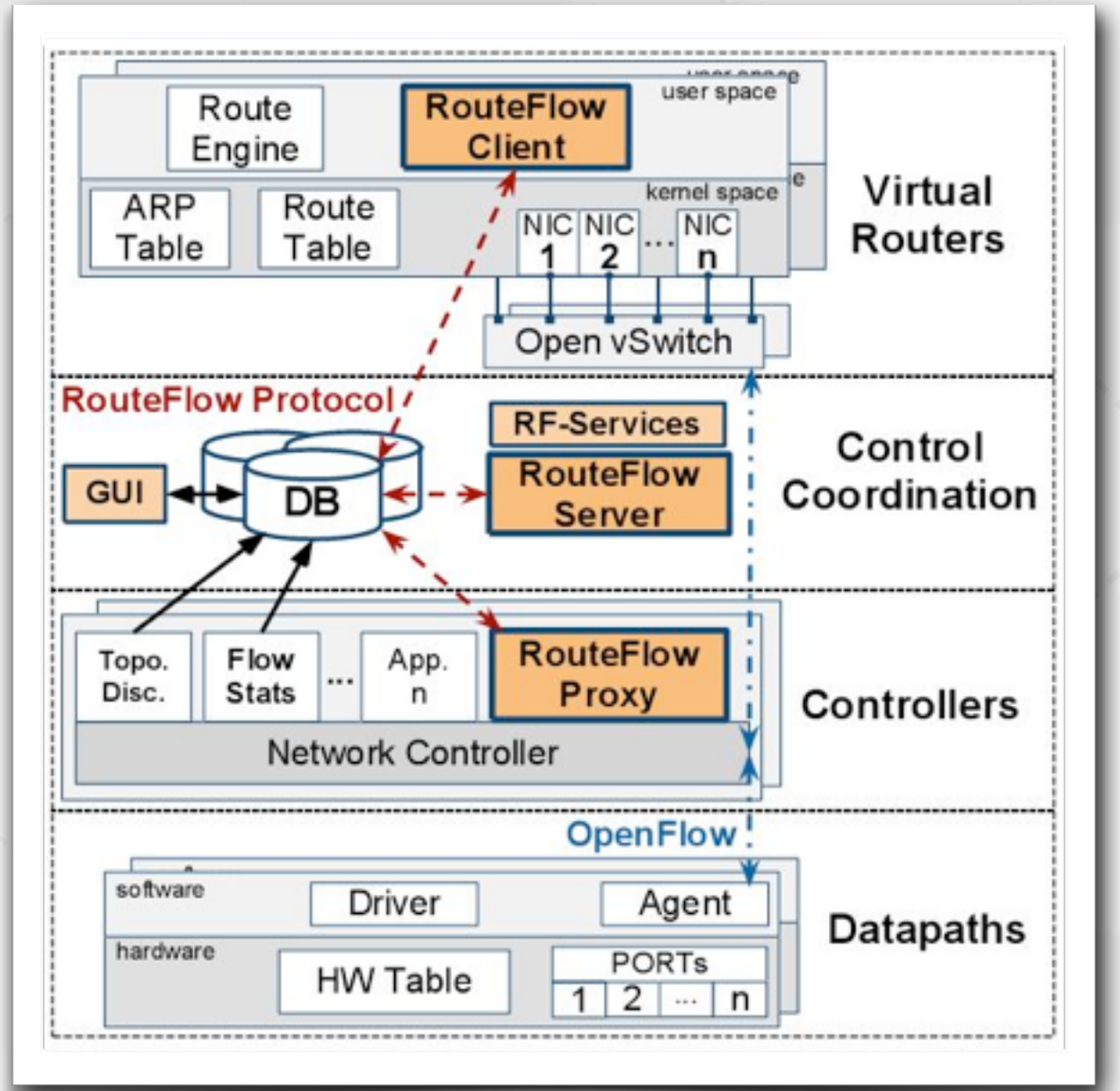
- Slices can be defined by any combination of switch ports (layer 1), src/dst ethernet address or type (layer 2), src/dst IP address or type (layer 3), and src/dst TCP/UDP port or ICMP code/type (layer 4).

- FlowVisor enforces isolation between each slice, i.e., one slice cannot control another's traffic



The FlowVisor intercepts OpenFlow messages from guest controllers (1) and, using the user's slicing policy (2), transparently rewrites (3) the message to control only a slice of the network. Messages from switches (4) are forwarded only to guests if it matches their slice policy

https://openflow.stanford.edu/display/DOCS/Flowvisor



https://sites.google.com/site/routeflow/

RouteFlow: Virtualized IP Routing

- RouteFlow, is an open source project to provide virtualized IP routing services over OpenFlow enabled hardware.

- RouteFlow is composed by an OpenFlow Controller application, an independent RouteFlow Server, and a virtual network environment that reproduces the connectivity of a physical infrastructure and runs IP routing engines (e.g. Quagga).

- The routing engines generate the forwarding information base (FIB) into the Linux IP tables according to the routing protocols configured (e.g., OSPF, BGP). In turn, the Linux IP and ARP tables are collected by RouteFlow Slave processes and then translated into OpenFlow tuples that are finally installed in the associated OpenFlow-enabled devices in the forwarding plane.

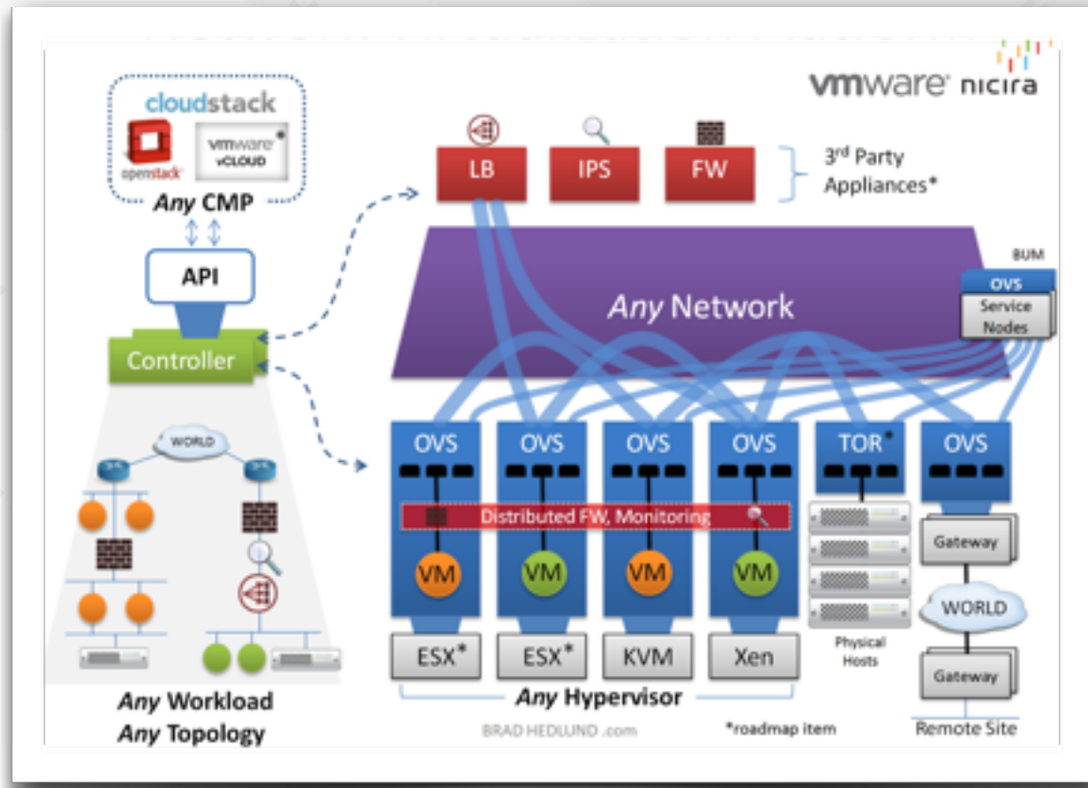
Nicira (VMware) NVP/NSX: Network Virtualization

- NVP is software that manages a network abstraction layer between end hosts and the physical network and **enables the creation of virtual networks that operate independent of the underlying physical network**

- The NVP Controller communicates directly with Open vSwitch (OVS), switch software deployed in server hypervisors. The hypervisor connects to the physical network and end hosts connect to the vswitch. **NVP does not talk directly to the physical network.**

- The NVP Controller Cluster dynamically updates the state of tunnel connections between OVS switches through the physical network. **These tunnels allow virtual networks to span across the data center, even between data centers**

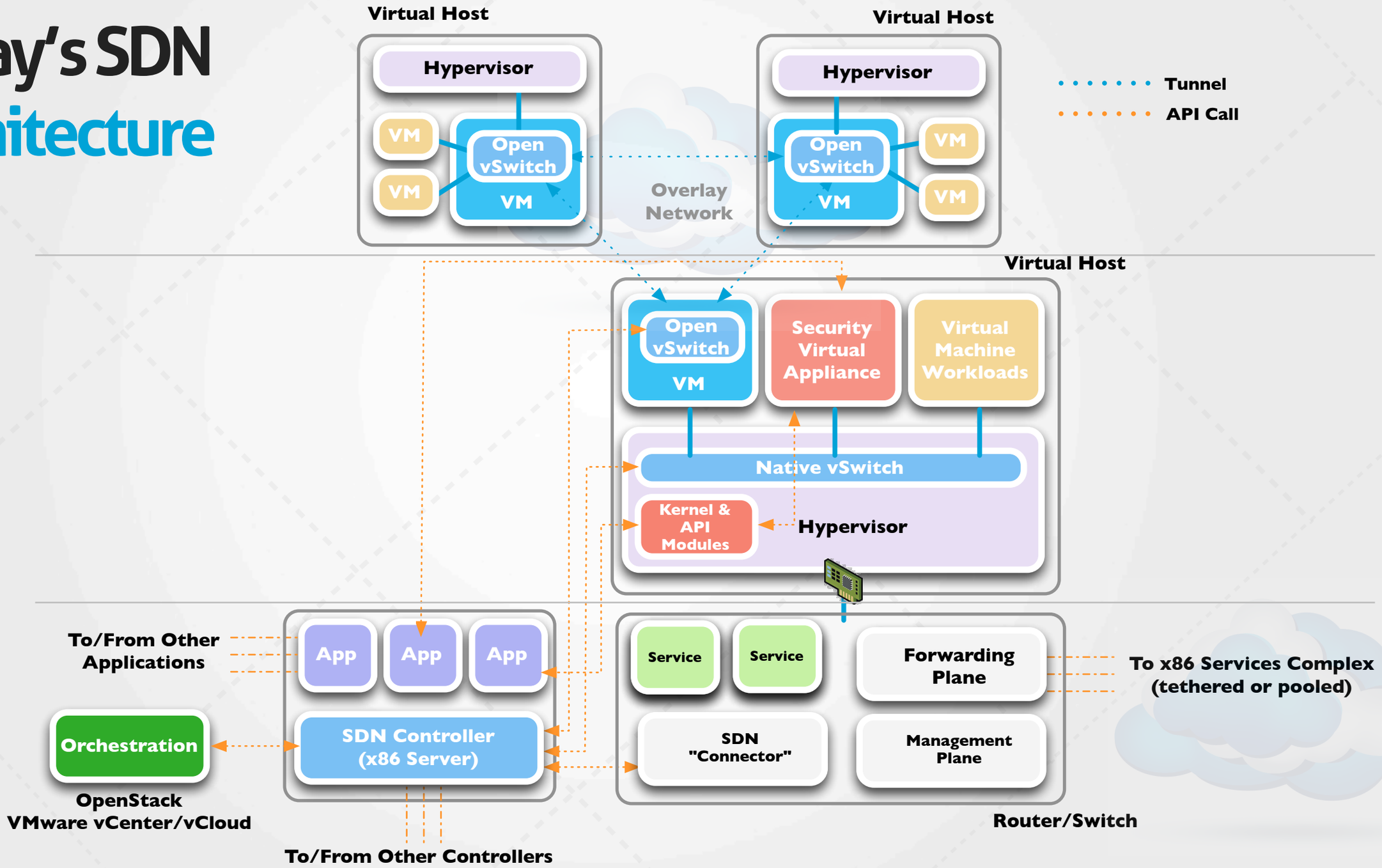
- **Data communications between workloads connected to virtual networks is encapsulated** and traverses the physical network, enabling VM mobility across subnet boundaries, while maintaining L2 adjacency



http://bradhedlund.com/2013/01/28/network-virtualization-a-next-generation-modular-platform-for-the-virtual-network/

- The operational state of the network is computed algorithmically in the NVP Controller Cluster, **avoiding any type of manual intervention,** such as scripts, whose function more closely resembles manual CLI replacement than computation.

Today's SDN Architecture

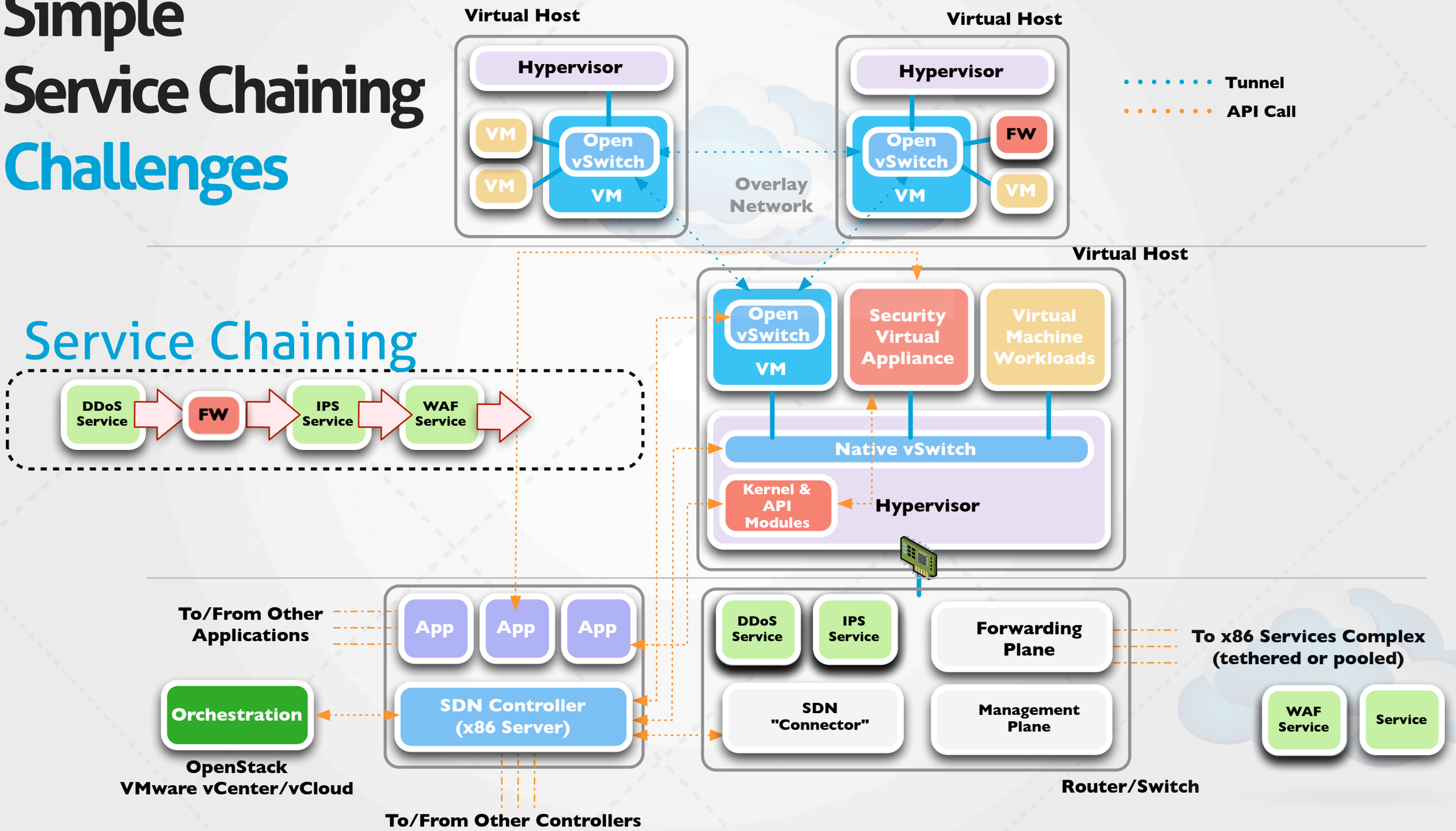


**Note: Tunnels may also be instantiated natively by Hypervisor and may not require VM-based vSwitch*

PacketFilter
Masters Of the Flow

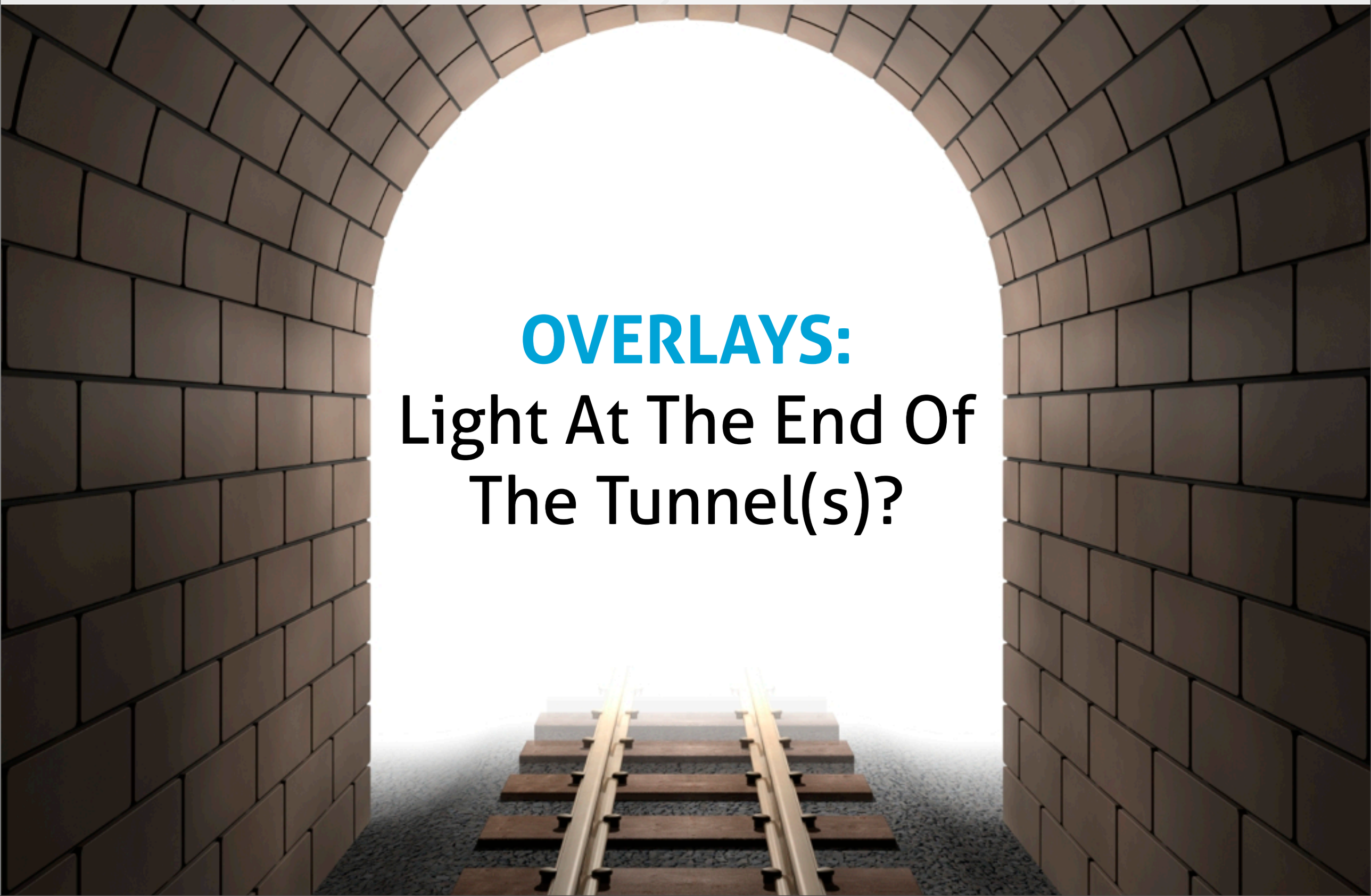
Software Refined Networking
The Path To Hell Is Paved With Good Abstraction

Simple Service Chaining Challenges



**Note: Tunnels may also be instantiated natively by Hypervisor and may not require VM-based vSwitch*

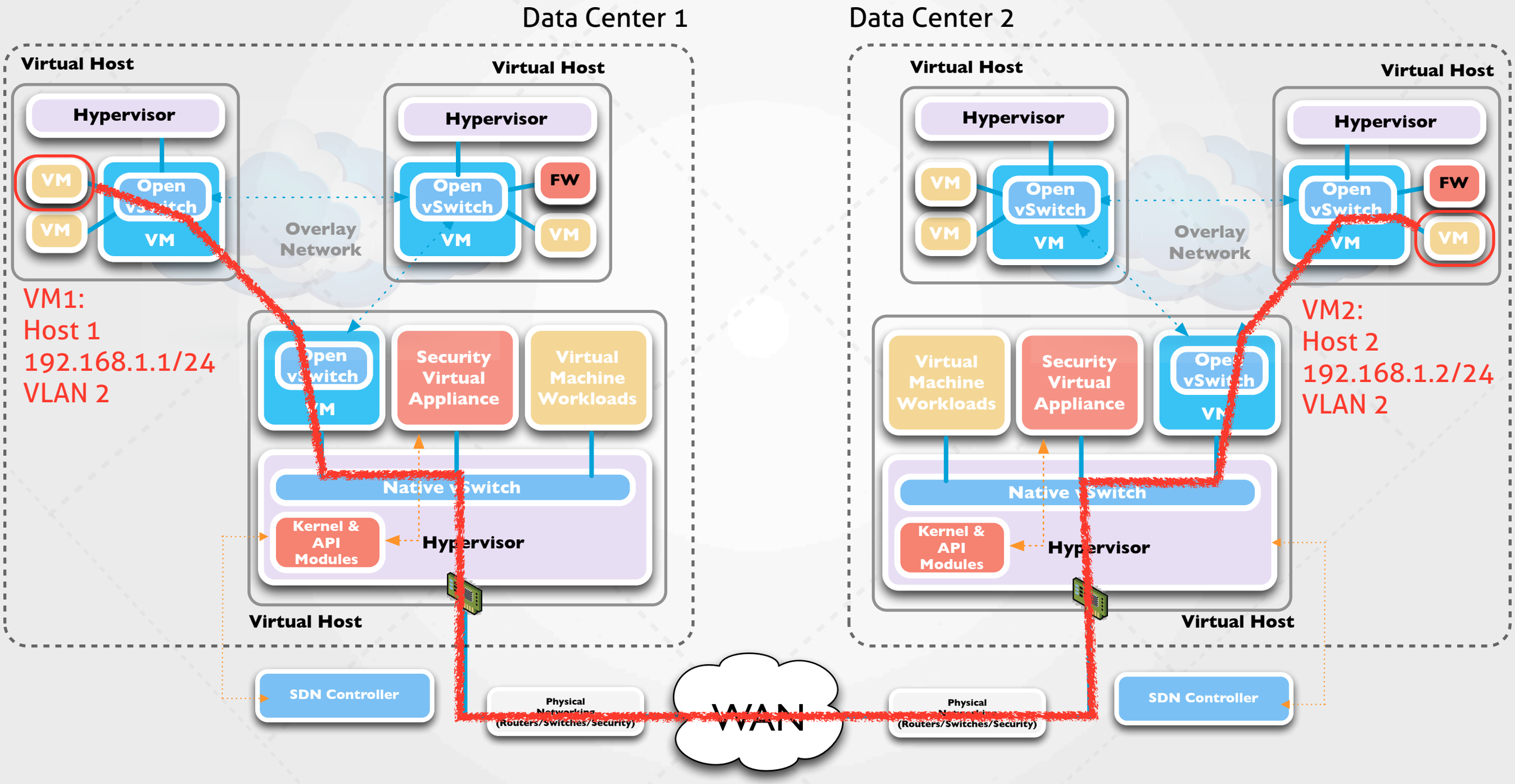
OVERLAYS:
Light At The End Of The Tunnel(s)?



PacketFilter
Masters Of the Flow

Software Refined Networking
The Path To Hell Is Paved With Good Abstraction

~~Turtles~~ Tunnels All The Way Down



**Note: Tunnels may also be instantiated natively by Hypervisor and may not require VM-based vSwitch*

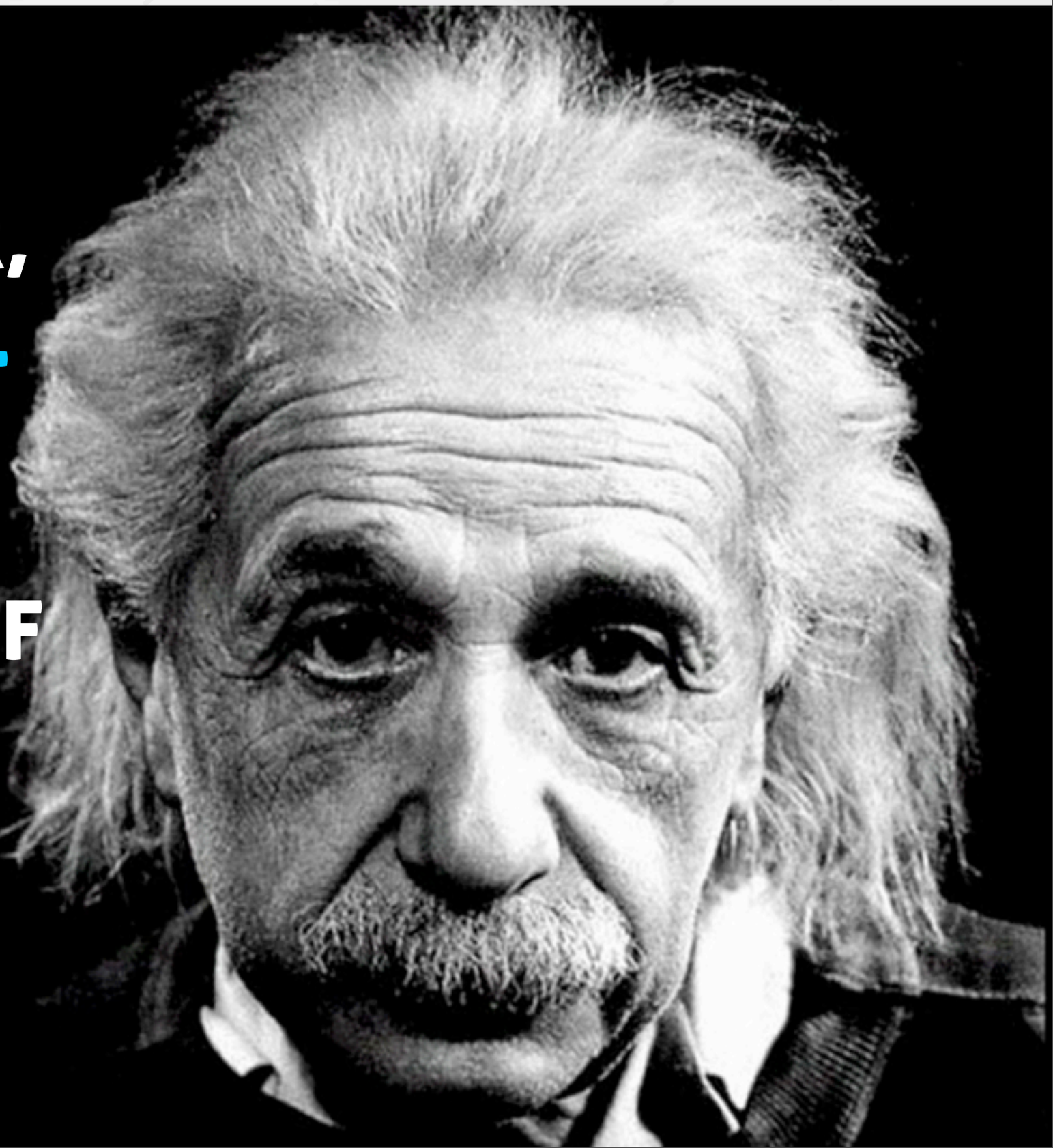
Why Should You Care?

It's All Fun & Games Until Someone Loses A Packet

PacketFilter
Masters Of the Flow

Software Refined Networking
The Path To Hell Is Paved With Good Abstraction

OUT OF CLUTTER, FIND SIMPLICITY. FROM DISCORD, FIND HARMONY. IN THE MIDDLE OF DIFFICULTY LIES OPPORTUNITY.



When You See "SDN" & "Security" Paired, People Are Generally Referring To These Functional Topics:

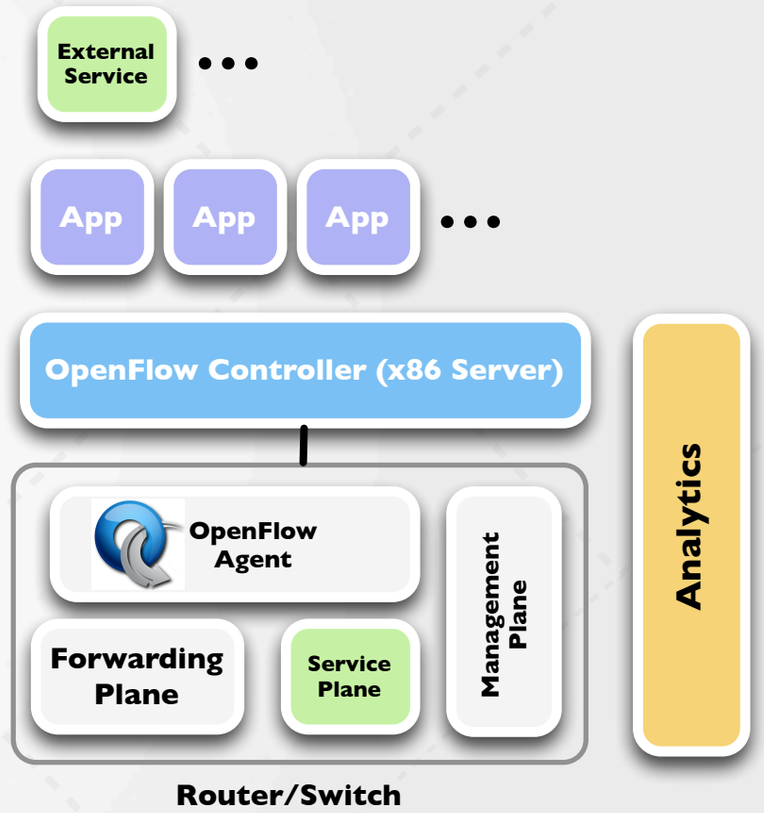
- Securing SDN -- Securing the SDN Infrastructure
- SDN-Enabled Security Services -- Using SDN to deliver security services

Let's Break These Down...

Securing SDN - Securing the SDN Infrastructure

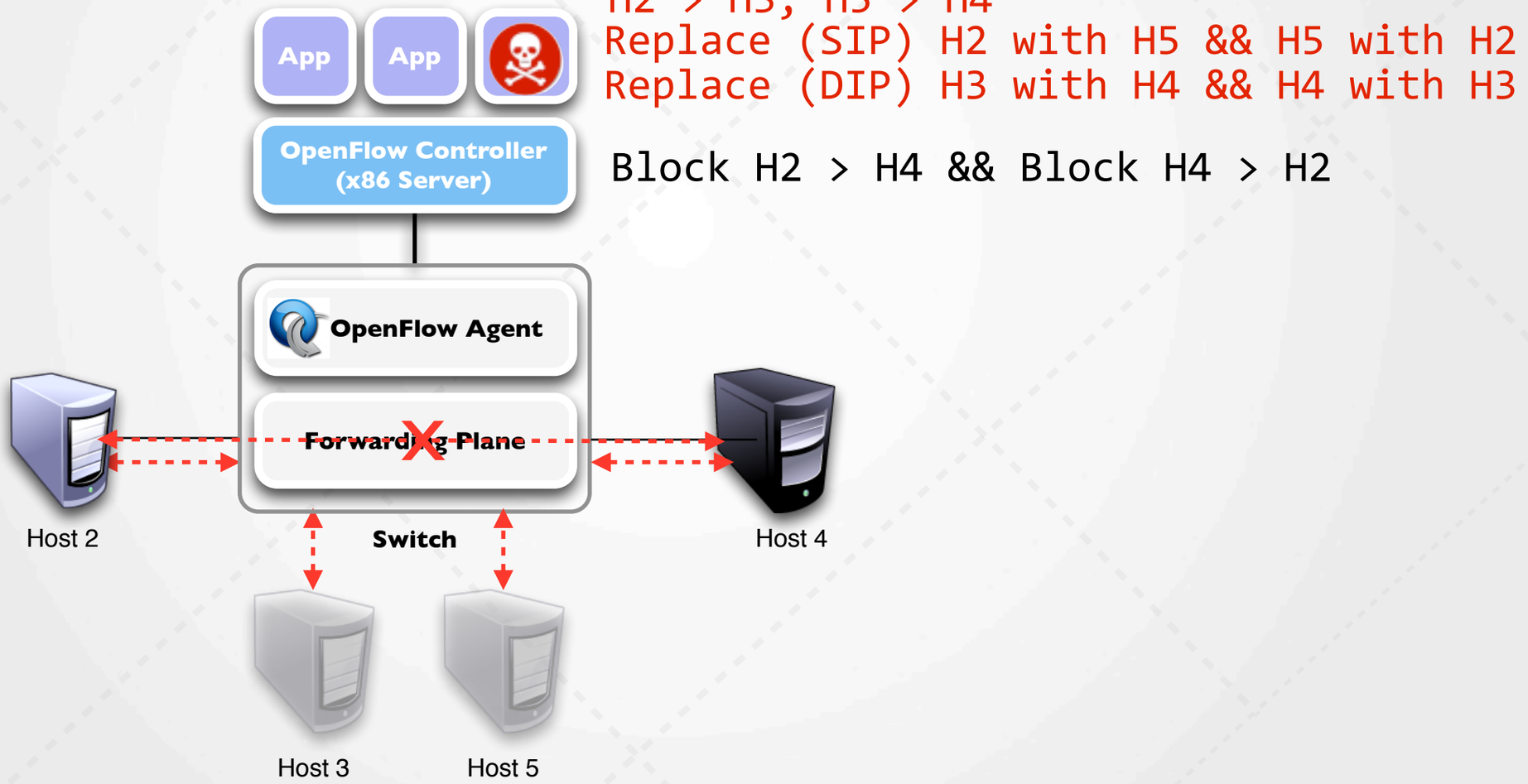
Some Examples Of Things We Must Pay Attention To:

- Secure the Control/Management Planes (SSL/TLS, Auth, etc.) & APIs (Security must **not** be optional)
- Secure the interfaces of AND limit capabilities of "Trusted" applications and plug-ins
- Enable Audit Trails across all layers
- Enable QoS/Connectivity/Bandwidth Between Devices & Controllers
- Reconcile Consistency between state machines, controllers & policy modes (proactive, reactive, hybrid)
- Reconcile Policy collisions & discern between automated versus interactive workflows



The Bypass Example*

Rogue Virtual Tunnels Are NOT Your Friend

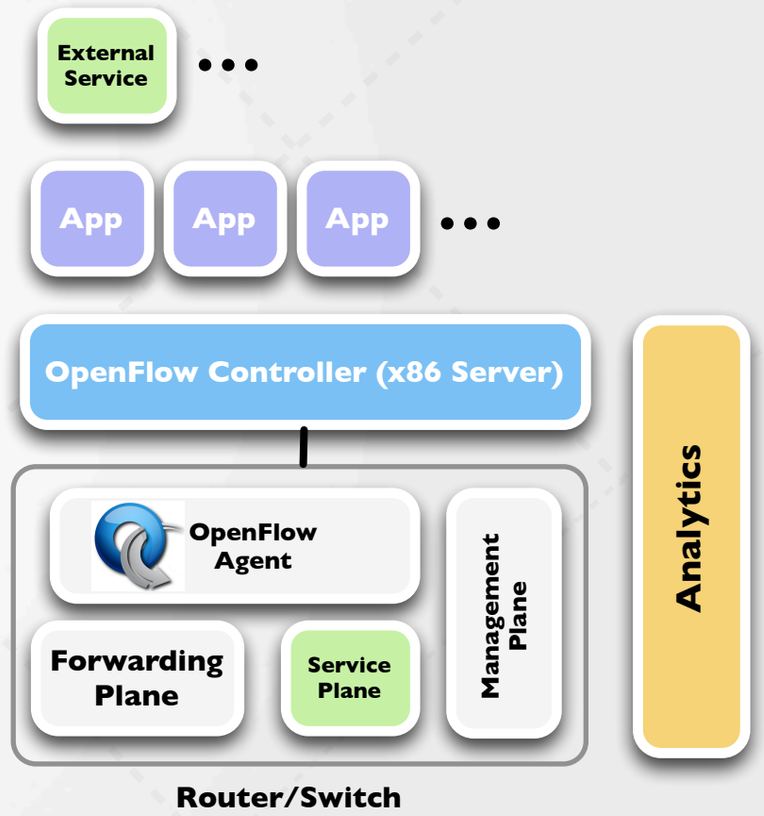


* Many thanks to Phil Porras or SRI for his gracious permission to re-use his FortNox example scenario

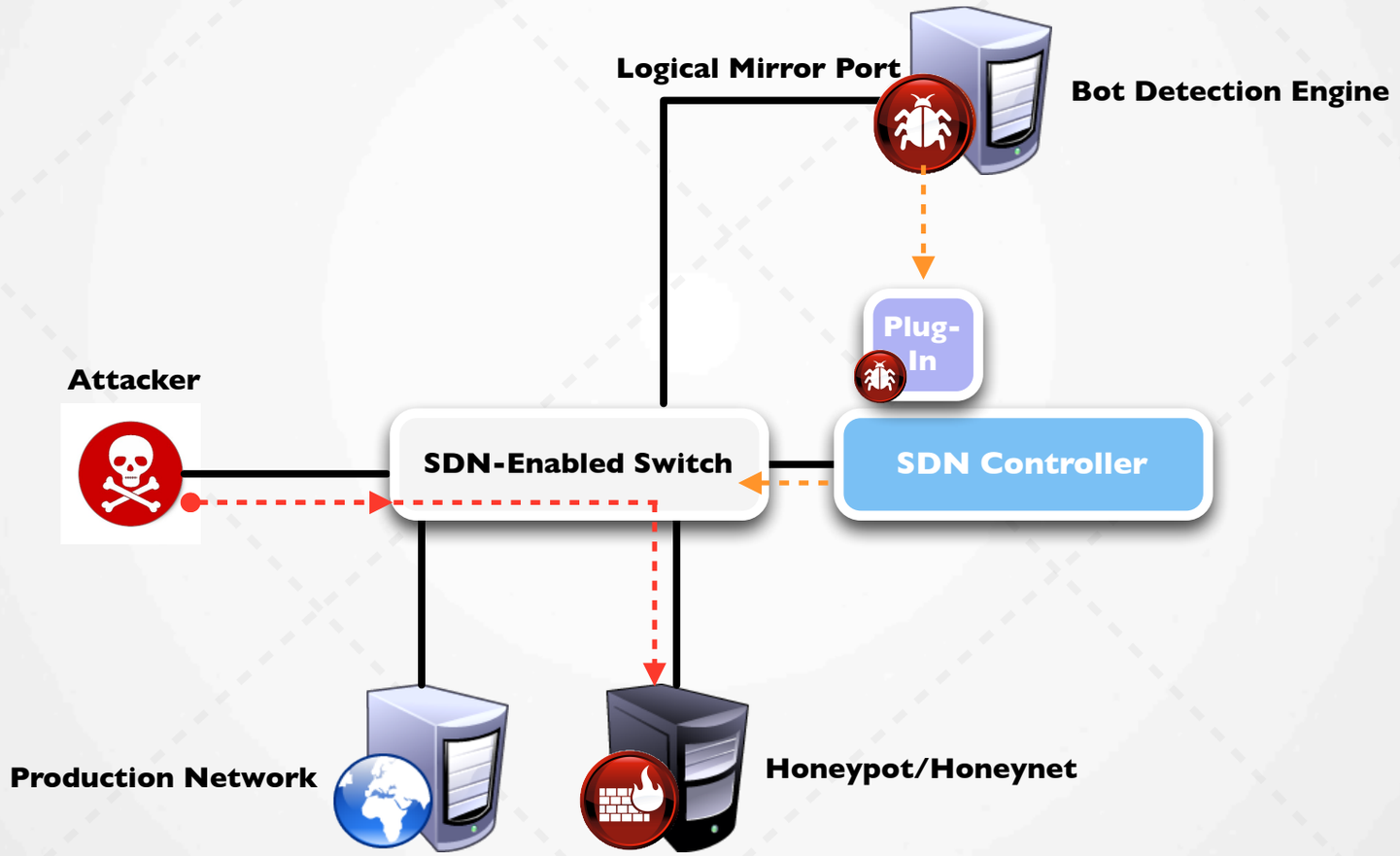
SDN Security Services - Using SDN to Deliver Security

Some Examples:

- Standardize abstracted Security Expressions
- Attach Policies Programmatically To Workloads
- Dynamic Security Service Instantiation
- Extend networks across DC's & Clouds
- Complex, Distributed Service Chaining...Simplified?
- Simplified Policy Management from L2++
- Higher Feature Velocity For L4-L7 Services
- "NAC" on Steroids
- Easier Monitoring At Scale
- Distributed Feedback Loops & Security Intelligence



Enabling Dynamic Feedback Loops With Security Intelligence



* Many thanks to Phil Porras or SRI for his gracious permission to re-use his FortNox example scenario

Software Defined Security

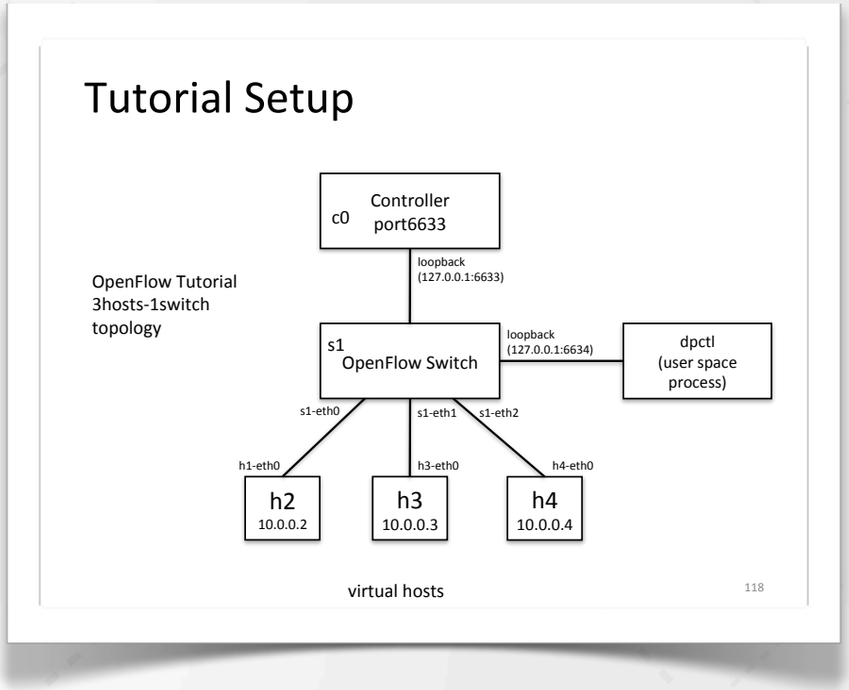
Because Sometimes It's About Moving The Ball Forward

A Free, OpenFlow Tutorial AND an OSS SDN System

- This tutorial is archived/out-of-date, but it's awesome anyway:

  http://archive.openflow.org/wk/index.php/OpenFlow_Tutorial

  Includes everything you need in a VM to build reasonably complex software-driven, OpenFlow-based networking environments with an OpenFlow Controller...you can add security as you see fit.



- Juniper Networks (*WARNING: $employer*) has open sourced Contrail, our kick-ass SDN/Network automation and virtualization "system" which you can download (for free,) use, and contribute to...

  http://www.opencontrail.com

Wrapping Things Up...

- SDN can fundamentally change how we will operationalize security...an abstraction that allows for commonality in policy definition evolving over time
- Provides reduction of human configuration errors, decrease in provisioning times, and elevation of programmatic networking and security for automated monitoring, isolation, multi-tenancy
- Like Virtualization and Cloud, this will disrupt & distract SecOps teams either way
- It's another point along the continuum of what we've already seen in terms of virtualized security
- Enables the off-loading of security capabilities to allow for more scale, better coverage in software with optimized hardware where needed
- The interaction of these tunnels, overlays and underlays are critical to security, availability and performance, visibility & transparency...
- Requires trust in automation and better definition of security policy and workflow
- Service Providers & Large Enterprises Are Deploying Now; this is the new backbone of software defined * Mobility and Cloud design patterns

Thanks!

I Really Value Your Input. Please Send Me Some...Positive Or Otherwise

Email: choff@packetfilter.com (not work)
choff@juniper.net (work)

Blog: http://www.rationalsurvivability.com
Twitter: @beaker

Other Presentations In The Series...Virtualization, Cloud, DevOps and (now) SDN...

2008

2009

2010

2010

2011

2012