LOOKING TO THE HORIZON: SDN

Software-defined networking is still coming together, but its promise is unmistakable.

READ ABOUT:

- Getting to know SDN technology, whether seeking deployment or not
- OpenFlow as an SDN first step
- SDNs: solving problems with existing networks
- What SDNs can teach about network abstractions
- SDNs offering a new take on data center configurations

Software-defined networking is one of the hottest buzzwords of 2014, but saying exactly what SDN is can be a challenge.

SDN has its roots in the early 1990s, when both network managers and service providers began to express frustration with typical network architectures that inhibited innovation and change, were plagued by complexity and rigidity, and yielded high investment and operations costs.

Many networking technicians have concluded that large networks can be built differently (and can deliver better results). A simple example of such a problem with current networks has a short name: IPv6. If networks had been built differently, the migration from IPv4 to IPv6 would not be the challenge that it is for many network managers.

PIECING TOGETHER SDN

Strategies for building networks differently have come under many different names, but three key ideas are common to what are now known as software-defined networks:

1) The control plane of the network is separated from the data forwarding plane and is accessible by some type of application programming interface (API). The control plane tells the forwarding plane what to do.

For example, in a typical router, the data-forwarding plane needs the answer to the question: If a packet comes in on this port, going to a particular IP address, what port and gateway should it be sent to next? The control plane provides the answer, then pushes it back down into the data-forwarding plane. The control plane runs routing protocols, as well as access controls and quality of service (QoS) management.

In a typical router, the control and data-forwarding planes are both in the same device, and the interface between the two is a private proprietary channel. In an SDN, the two are clearly detached, and this private channel is replaced by a software API. This creates the potential to run the control plane on a "controller" somewhere, which could be a single device or a distributed system.

2) The software-defined network has a networkwide operating system that has a global view of wired and wireless networks. Rather than an individual operating system on each switch, router and access point, the network has a single operating system that controls and manages all devices.

Having a network operating system allows for one of the key goals of SDNs: abstracting the network. Rather than looking at a network as a distributed system of independent devices, a software-defined network has some centralized control.

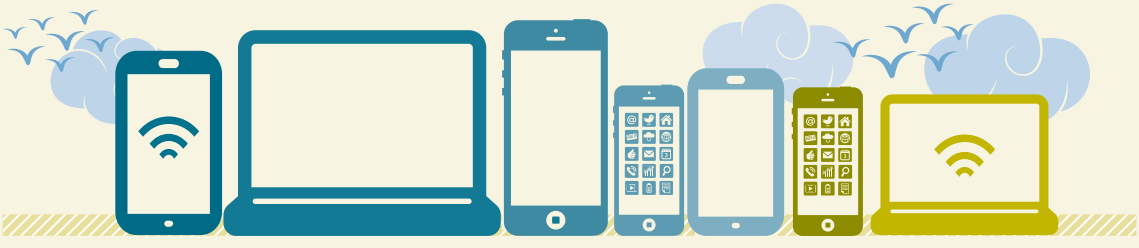
It's important to remember that "centralized control" doesn't mean a single controller, or that every flow in the network has to go through the centralized control. How SDNs implement their control is still a big topic for discussion and research.

3) Network features are part of the networkwide operating system, instead of running independently on individual devices. Applications that control the network, such as dynamic routing (Border Gateway Protocol, Open Shortest Path First, Routing Information Protocol and other protocols), access controls (such as stateful or stateless firewall rules), network virtualization (multiple virtual networks running on the same hardware and coexisting without conflict) and QoS management are all implemented inside the networkwide operating system.

The advantage of moving these features away from the device and into a software layer that sits over all devices is that the operating system functionality can be changed and updated very quickly. New features can be added at the software layer without the need to make changes to hardware, such as switches.

So what — if anything — should IT managers do about software-defined networking? Will they change the way networks are designed? Are SDNs critical for competitive advantage? Does SDN technology build better, more reliable networks? For most organizations, the answer to all of these questions — today — is no.

Although a number of vendors are selling SDN hardware and software, most IT managers should treat SDNs as an interesting technology worth testing and understanding — but they don't need to rush to deploy it right away.



76%

THE PERCENTAGE OF ORGANIZATIONS THAT HAVE IMPLEMENTED SDN OR ARE PLANNING AN SDN DEPLOYMENT THAT CITED "A MORE EFFICIENT AND FLEXIBLE NETWORK THAT SPEEDS SERVICE DELIVERY" AS A KEY BENEFIT.

SOURCE: *SDN Survey 2013: Growing Pains* (*InformationWeek*, August 2013)

THE STATE OF THE ART IN SDNs

For now, SDNs largely exist in labs in universities and the research departments of networking vendors. However, one early implementation of SDNs called OpenFlow is widely

>

available for commercial use.

OpenFlow has been called the Windows 3.0 of software-defined networking: a first attempt to put SDN concepts into practice, but certainly not the ultimate expression of what an SDN can and should be. OpenFlow gives both hardware and software manufacturers great experience in learning how SDNs might be implemented and has even been used in some very large private networks. Network managers interested in working with SDNs will run into OpenFlow almost immediately as an SDN candidate.

The most interesting thing about OpenFlow is that it works: a multivendor way of linking up switches and routers (as the data-forwarding plane) and software-based controllers (as the control plane) together into a single, integrated whole.

OpenFlow is being managed by the Open Networking Foundation, and its website lists more than 50 vendors that have released products or services that fit into its vision of OpenFlow. This includes switch and router hardware, virtual switches and routers, and several OpenFlow controllers.

Of course, OpenFlow isn't the only option, and vendors are releasing products that go a little further than OpenFlow as part of their own vision of what SDN should look like. However, for multivendor interoperability, OpenFlow really is the best place to start.

HOW TO PREPARE FOR SDNs

SDN isn't an improvement on existing networking; it's a different kind of networking altogether. For many network managers, SDN is of no interest at all. For them, nothing is fundamentally broken about the way that network devices are designed today. And for most organizations, traditional networking devices work well and have the right level of programmability and control.

However, many network managers object to the massive complexity required to configure reliable networks. In many cases, the complexity gets in the way, to the point where some network capabilities can't even be used properly.

SDNs are designed to solve problems with existing networks, such as:

1) Rigidity, lack of agility: Currently, networks are difficult to reprogram and reconfigure, and can't react quickly to changing requirements. New functions require huge upgrades, and updates may be blocked by older equipment.

2) Complexity, manual controls: Most networks operate as a set of independent elements, building into a distributed system. Even when an operations support system (OSS) is in place, devices are configured one at a time, and changes are prone to inconsistencies and human error.

3) Vendor dependency, no API: Most organizations looking to speed up network configuration changes would prefer a real API to control and manage networks. In data centers, for example, the process of rolling out new devices can be automated almost everywhere using orchestration tools – except at the networking layer.

Network managers who regularly deal with these problems will be most interested in adding SDN technology to their LAN and WAN deployments. Those for whom these problems are unfamiliar probably won't have much interest in deploying an SDN.

GETTING READY FOR SDNs

Regardless of whether SDN is in an enterprise's short-term plans, network managers can learn some interesting lessons from working on SDNs that can help simplify network configuration and pave

What Is OpenStack?

It's easy to confuse OpenStack with OpenFlow, but they are different projects with different goals. OpenStack seeks to develop an open-source cloud computing platform. As part of that platform, OpenStack includes three major components: OpenStack Compute, used to provision and manage large networks of virtual machines; OpenStack Storage, which provides file (Network File System) and block (iSCSI) storage for use with servers and applications; and OpenStack Networking.

OpenStack Networking handles network management functions at many layers, such as Internet Protocol and Virtual LAN management, middle-box management (intrusion detection system, virtual private network, firewall, load balancer) and network configuration management. If the network in a data center is built on an SDN standard, such as OpenFlow, then OpenStack Networking could communicate with the OpenFlow controller to configure and manage the network – one of many functions built into OpenStack Networking.

OpenStack doesn't require SDN, and tools such as OpenFlow are complementary to what OpenStack does in the same way that any manageable network is complementary. In the short term, most organizations building private clouds on OpenStack will probably be using more traditional networking equipment also supported by OpenStack, rather than SDN.

SDN ISN'T AN IMPROVEMENT ON EXISTING NETWORKING; **IT'S A DIFFERENT KIND OF NETWORKING ALTOGETHER.**

the way for future use of SDNs.

1) Getting Domain Name System (DNS) right: One of the most important lessons from SDNs is the need for abstractions in dealing with networking, and the mapping between names and addresses is one of the most critical abstractions.

Organizations that don't have a solid DNS system in place should be spending the time and money required to eliminate all dependency between IP addresses and applications, including system management. If IT staff members are still typing in IP addresses when logging into firewalls, switches and routers, then DNS is broken and desperately needs to be fixed.

A completely integrated IP Address Management solution that abstracts the network away from applications and names – yet makes it easy to translate IP addresses to names, and vice versa – is one of the best ways to simplify network management and reduce the amount of work needed when network reconfiguration occurs.

2) Becoming application-centric: SDN technology is most immediately applicable to data center devices although spans the entire network configuring switches, routers, controllers, access points and firewalls – eventually pushing out to the network edge from there. Configurations that previously were impossible to manage should become possible.

The key to making good use of SDNs will be an understanding of the architecture of applications and defining the flows both north-south (into and out of the application tiers) and east-west (between application servers in the same tier). Despite years of experience with enterprise systems such as Active Directory and Microsoft Exchange, many IT and security managers have only limited information about how clients and servers interact.

Enterprises should begin to build databases or management systems that look at applications as a series of layers and flows. Even in a non-SDN world, this type of information and mapping will be valuable in virtualization initiatives, where servers tend to multiply and the trend is to spin up another virtual machine rather than layer multiple applications on a single server.

3) Preparing for distributed hardware and centralized control: When SDNs centralize control architectures, they do it for a specific reason: Too many software and hardware components create too much complexity.

As organizations adopt SDNs, they will see a reduction in the number of large, centralized devices and much greater use of distributed hardware. Network managers who have not moved to top-of-rack topologies will be pushed in that direction as network vendors prepare for SDN as an option on newer hardware.

SDNs will be separate from other types of networks, such as edge Internet connections or WAN VPN concentrators. IT managers who have merged functionality in "Swiss Army knife" devices should slowly move to diverse devices, handling data centers separately from other networking functions, including user access, wireless, Internet and WAN connectivity. ■

Disclaimer

The terms and conditions of product sales are limited to those contained on CDW's website at CDW.com. Notice of objection to and rejection of any additional or different terms in any form delivered by customer is hereby given. For all products, services and offers, CDW® reserves the right to make adjustments due to changing market conditions, product/service discontinuation, manufacturer price changes, errors in advertisements and other extenuating circumstances. CDW®, CDW·G® and The Right Technology. Right Away.® are registered trademarks of CDW LLC. PEOPLE WHO GET IT™ is a trademark of CDW LLC. All other trademarks and registered trademarks are the sole property of their respective owners. CDW and the Circle of Service logo are registered trademarks of CDW LLC. Intel Trademark Acknowledgement: Ultrabook, Celeron, Celeron Inside, Core Inside, Intel, Intel Logo, Intel Atom, Intel Atom Inside, Intel Core, Intel Inside, Intel Inside Logo, Intel vPro, Itanium, Itanium Inside, Pentium, Pentium Inside, vPro Inside, Xeon, and Xeon Inside are trademarks of Intel Corporation in the U.S. and/or other countries. AMD Trademark Acknowledgement: AMD, the AMD Arrow, AMD Opteron, AMD Phenom, AMD Athlon, AMD Turion, AMD Sempron, AMD Geode, Cool 'n' Quiet and PowerNow! and combinations thereof are trademarks of Advanced Micro Devices, Inc. HP Smart Buy: HP Smart Buy savings reflected in advertised price. HP Smart Buy savings is based on a comparison of the HP Smart Buy price versus the standard list price of an identical product. Savings may vary based on channel and/or direct standard pricing. This document may not be reproduced or distributed for any reason. Federal law provides for severe and criminal penalties for the unauthorized reproduction and distribution of copyrighted materials. Criminal copyright infringement is investigated by the Federal Bureau of Investigation (FBI) and may constitute a felony with a maximum penalty of up to five (5) years in prison and/or a $250,000 fine. Title 17 U.S.C. Sections 501 and 506. This technology insights guide is designed to provide readers with information regarding networking, licensing and selection. CDW makes no warranty as to the accuracy or completeness of the information contained in this technology insights guide nor specific application by readers in making decisions regarding networking purchase or implementation. Furthermore, CDW assumes no liability for compensatory, consequential or other damages arising out of or related to the use of this publication. The content contained in this publication represents the views of the authors and not necessarily those of the publisher.

©2014 CDW LLC. All rights reserved.