NETEVENTS

CLOUD INNOVATION SUMMIT

FINAL

Keynote Presentation

The Next Horizon for Cloud Networking and Security

Guido Appenzeller

Chief Technology Strategy Officer for Networking & Security, VMware

Interviewer:

Manek Dubash Editorial Director, NetEvents

Good morning everyone. I'm Chief Technology Strategy Officer of the Networking Security Business Unit at VMware. So I want to talk about security, and I want to talk about cloud, but I want to first start by talking about networking. Networking today is a software industry. Now, for some of you this may be something obvious. I'm not sure if everybody has fully internalised this yet because there's a huge difference between a software industry and a hardware-driven industry. Right? I mean, we've seen this in many different industries over the last couple of years. For example, video delivery used to be a hardware-driven industry. You went to your local video rental store, rented a tape or later some kind of disc, like a DVD, and then this moved to an online distribution model where Netflix suddenly is the largest cable company in the United States. The same thing is fundamentally happening with networking right now. And let me provide you a little bit of background on that.

So a long time ago, I'm probably dating myself here, the way we purchased compute was the same as the way we purchased networking. So about 15 years ago you would buy -- or maybe 20 years ago even, you would buy a SPARC station from Sun Microsystems, a SPARC chip, a Solaris operating system – a monolithic, vertically integrated system.

Networking equipment back then was bought exactly the same way. Then something happened on the compute side, which is that this vertically integrated industry started layering and started coming apart. And the first thing that happened was we started seeing operating systems emerge. So suddenly I would buy a Gateway PC, with Windows on top of it and an Intel CPU. So suddenly we have multiple vendors that put together the system. Then Mendel Rosenblum at Stanford started server virtualisation and we started virtualising the compute side so suddenly we could run multiple operating systems on top of a hypervisor on top of the hardware. And, you know, more recently we added a consumption layer to that and now have something like OpenStack or VMware vRealize Automation running on top of it that allows me to, very dynamically, provision compute.

So on the compute side we've made huge progress over the last couple of years. There's a really interesting study by Gartner that shows that the efficiency of server administrators over the past 15 years has made huge gains. The number of servers a single server administrator can administer has gone up by about a factor of 10 or so.

On the networking side none of this has happened. The way we operate networks today is still very similar to the way we operated networks 15 years ago. You enable and configure the machine manually through a CLI. And the reason for this is that networking has remained this monolithic industry where today the way most networking equipment is purchased is as a monolithically-integrated vertical stack.

Now, this is starting to change. And so the moniker that's used for this change is software defined networking. And for me, personally, SDN started when in 2008 I left the start-up I had founded and came back to Stanford, became a professor there for two years, took over a little research project called OpenFlow, which then became a networking standard -- that's our crew here at Stanford, in the Stanford data centre [shows photo on screen] -- and there was a reporter from the MIT Technology Review actually visiting us and saying hey, this is amazing. What do you call this? And we were like, well it's called the OpenFlow protocol. He said, no, no, what do you call this whole new principle? We were like okay, first of all which principle, but we really don't have a name yet. And she was also writing about a software-defined radio so she looked at me and said like, well, it's a little bit the same thing but for networking. So in the article Kate called it software defined networking. And that name stuck right, and so here we are with software defined networking.

So over the past is it -- it's six years now -- right. So over the past six years we first saw initial SDN deployments in academia. We then started first products appearing around 2010; initial small group of concepts, pilot deployments, production deployments. And over this entire time, I've been pretty much fighting architectural battles. You would go to a large network organisation, initially it was in the academic side, later these were customers. You're trying to persuade them they should try out this thing called software defined networking; was this a good thing or a bad thing, how does it relate to hardware defined networking. And then in late 2014, late 2014 something very amazing and very personal to me happened, which is that the tone of the conversation around this really, really changed. I still remember last year I was in New York talking to a number of directors of networking -- in one case CIO level people -- at large banks. And suddenly in one particular meeting I was getting ready to show my SDN slide deck, and basically the CIO, the interim CIO there told me, look, you don't have to convince me that SDN is going to be in my future data centre;

I understand this. Tell me about the how. Tell me about what this means operationally. What does it mean for my team, what does it mean for my processes, you know. How does this change how I run my data centre? And so, for me, this was really sort of a pivotal moment because I think what this means is that software defined networking has crossed over into the mainstream. It's gotten out of the early adopters, it's really now mainstream technology that is often an integral part of modern data centre architecture.

Let me provide you some numbers to back this up. So at VMware we've had a phenomenal year in 2014. We've finished the year with more than 400 customers on NSX. There's more than 70 production deployments. And, more importantly, we're adding somewhere between 25 and 50 new product deployments per quarter. So we're growing very, very rapidly. But 80 percent of the top banks at this point have purchased NSX. We're seeing very wide adoption; very deep adoption in certain segments. We're currently doubling both the number of customers as well as the revenue, if you look historically, about every six months. So we're quadrupling every year, right? We're in a very, very fast growth. We have a fantastic set of customers. We have service providers, financial institutions across industries. This is really, from my perspective, a technology whose time has come and where we're now seeing this huge uptake.

So an interesting effect of this is the way that how people organise their IT teams is actually changing, right? It used to be, in IT you had these very clear silos; you had the server guys, you had the networking guys, you had the storage guys, you had your security folks. And with server virtualisation, network virtualisation and storage virtualisation and all these things coming together, we're starting to see the team structure change as well. It's suddenly, at the very least, these teams have to communicate much, much more tightly because at the consumption layer they're all tied together. In some case these teams are really getting merged together to large cloud teams and these individual silos disappearing.

So I personally think this is a great opportunity, if you're in network administration this is a fantastic time because now suddenly you have all the automation tools that the server folks had over the past couple of years, and you can really now take network administration to a whole new level right? This is network automation Nirvana that we're approaching today.

So, with SDN having matured, what's next? Where's this going? And there's a really interesting observation here I find which is, I think, really what we've created is a new thing which I want to call virtual infrastructure. And let me explain what I mean by this. So classically, if you think about how you run your data centre, you have two tiers. You have your endpoints, at the top, and you have your physical network underneath. And if you want to add a piece of functionality, say you want to increase your security, you can do it in either one or the other domains. So for example, I can have a host-based intrusion detection system and put in my end host, or I can add an additional firewall down my network.

What I realized about these two different tiers is that they are independent. If one gets compromised, that doesn't affect the other one. Somebody hacks into my endpoint, my firewall, my data loss prevention, my intrusion detection system that run at the network layer as appliances, they are still there. They are still -- they are uncompromised. They are not affected by this.

Now, we then replaced the servers as virtual machines, and then something interesting happened, we started effectively creating a new tier between these two. We had virtual machines, which has all the new endpoints, we had the physical network. When we started pulling pieces from the physical network into this virtual infrastructure layer, the first thing where we did this was the Layer 2 domains. So we took switches -- we need something to plug in the virtual machines -- basically created these virtual switches. This was long before the Nicira acquisition; VMware had this already. Then with NSX, we started taking routers and moving these virtual routers into the infrastructure layer. We did the load balancing and NSX now has a simple load balancer built in. And more recently we did a firewalling and micro-segmentation.

So basically we took a couple of constructs which before were physical appliances and started moving them up into the virtual domain where they can be provisioned basically as a service that just runs in the background. The infrastructure team can provision them, but there's no physical equivalent of these things any more. And for all these systems you can decide, do you want them in the virtual layer, do you want them in the physical layer, and in some cases you want them in both. And I don't think I know any customer who completely replaced all his physical firewalls with virtual ones. So they probably want a combination of both.

This sort of begs the question what else can we take from the physical world and move it into the virtual infrastructure layer? And if we take a step back, I think anything which was previously a hardware appliance in your data centre, you can probably now take and move into virtual infrastructure. Your VPNs, your intrusion detection, intrusion prevention systems, data loss prevention, application delivery controllers, all these things now can be virtualised.

I think what's probably more interesting is, it turns out you can actually take things from the endpoint as well and move it into the virtual infrastructure because one of the really nice things about this virtual infrastructure layer is that it still has the capability to look into the host OS. So at VMware we have a product where, for example, you can do virus scanning that comes from the hypervisor. So this is no longer something that's happening on the endpoint, but something that's now facilitated by the virtual infrastructure layer and basically transparent to the actual virtual machine.

So I want to talk a little bit about why customers are deploying this virtual infrastructure. And if you look at the use cases, I think you can put them roughly in three buckets. Those are security, automation and what we call application continuity. And let me start off with security. So security in 2014 was a terrible year. As somebody in Silicon Valley said, 2014 was the year where the black-eyed hackers achieved product market fit; they're now scaling their business. And we've seen many large compromises. I think for the first time ever we’ve seen the CEO of a company, of Target, getting fired because of a comprise of the data centre. That certainly has helped elevate the discussion in these organisations.

The pattern that we've seen in 2014 is that the attackers have gotten a lot more sophisticated. So typically how these attacks work is you have an attacker; they initially manage to break into one server behind the firewall. And often this initial server is not a very sensitive server; this may be a standby system or -- it does not necessarily have any sensitive data. But because in most data centres today the level of controls behind the firewall are fairly limited, you basically have freedom of movement there. They can, over time, start from that server, and then start infecting other hosts. It jumps basically from host to host. And the sophisticated attackers are often active behind the firewall for several months, waiting for opportunities, sniffing credentials, just looking for ways to get into additional servers. And as they do this, they deploy sleeper payloads, so even if they get detected and cleaned from some of their hosts, these sleeper payloads will wake up and give them a second bite at the apple. It sometimes takes months just to clean them up because it's very hard to find out where they are located.

Looking at this, you could ask the question, well, if unconstrained communication is the problem, couldn't we just put firewalls everywhere in the data centre? Well, it turns out this is really, really hard for two reasons. The first one is just sheer cost. I mean if you want to take all your east-west traffic in a data centre and run it through a firewall, you need the same firewalling capacity per rack as your top of rack switch. Doing this will be cost prohibitive. And also the management of the rules would be very, very difficult right? And it doesn't matter if you try to do it as physical firewalls or as virtual firewall appliances, in both cases, so doing this at this scale is very, very hard.

Now it turns out if you move your firewall into the virtual infrastructure, then actually this becomes possible. And we have this thing with VMware NSX called micro-segmentation which today is driving, I want to say, about 40 percent or so of our sales. So it's a major reason for customers to adopt software defined networking and NSX. So how this works is that if a virtual machine is deployed, and a virtual machine that has a firewall associated with it, we automatically push out the firewall rules to the hypervisor. And in the hypervisor switch there's a little firewall running, in pure software, that will perform the firewalling operations. So if this VM for example sends all the packet it's not supposed to send out, the firewall will intercept it and stop it; all done in software.

One big problem with firewalls is managing rules. With this distributed firewall, one of the big advantages is that if you move around virtual machines, the firewall rules always move with the virtual machine. And if you delete the virtual machine, the firewall rules go away as well. So basically we provide the lifecycle management of these firewall rules for the distributed firewall in a fully automated way. That makes it much, much easier to consume these things than a classical physical firewall where you have to do all of this manually.

So net result is you get a firewall for pennies to the dollar compared to a physical solution that scales out as you scale out your infrastructure and really allows you to get to this vision of alligning lots of internal controls in your data centre. This move will not necessarily help you with the initial infection, but it basically means that an attacker that has gotten in cannot spread further inside your data centre, which is a huge step forward in security.

Now a much, much newer thing is this idea of extending capabilities from the guest OS into the virtual infrastructure domain, basically leveraging hypervisors for security in the guest. This is something we have not released yet, but let me give you a very brief sketch of how this works. So the very basic idea is to say we take a piece of software, basically I think of it as a trusted module so for example from an application, and you can basically move this trusted module on what we call the protected domain. And once it sits there, two things happen. The first one is it's protected, meaning if the guest tries to alter the module it can no longer do that. The second thing is this trusted module can now have its own data and this data is only accessible to the trusted module.

It's a very simple idea. So we can do a couple of additional things. We can, for example, verify signature on the module to make sure this is legitimate. We can provision data externally into the module. And we can -- basically the module can communicate securely, for example, for logging or audit purposes. And it's amazing what you can do with a simple primitive. So just to give you a couple of ideas; so the first one is virus scanning. We can now create basically a virus scanner, an intrusion detection system which the host can no longer turn off. If you're breaking into a server, the first thing you do is turn off the intrusion detection system, right, modify the software. This is no longer possible; this is now protected. In fact we can scan the host without the host ever knowing that it's being scanned.

But you can go further. You can, for example, say we have credit card numbers on this server. Well, let's store the credit card numbers in this protected domain. They can only be accessed through the trusted module, and the trusted module is written in a way that, for example, that says, “never give out the credit card number in the clear, right?” What this means is, you have a server, the server is compromised by an attacker, the server has credit card information but the attacker can never get the credit card information because there's now an additional boundary here. We move the credit cards into this virtual infrastructure layer where they're protected. Very simple idea but I think very, very powerful for changing security architecture.

And putting these things together, you can then actually use this primitive of micro-segments to define policy. So if I run a multi-tier application, I put each tier into its own micro-segment, put a little firewall around it. But now, in addition to the firewall rules, I can now define rules such as where can certain information be stored, how does information have to be encrypted, where are certain certificates available, where are certain keys available. All these things can be provisioned through the infrastructure layer by an infrastructure team. And the application team can basically use these things, but no longer can make configuration mistakes that would expose potentially sensitive information. So that's security.

So the second big area is automation. The idea is very simple. And I'm sure you've seen this in some way or the other before. Like today, provisioning a new virtual machine in most modern data centres is quick and easy, right? I can go to my IT team, they go to the VMware console or one of our competitors, they start a new virtual machine and this is running within a couple of seconds. So I pick my image; it's running.

Configuring a network that goes with those virtual machines is much, much harder. I've seen organisations where basically getting a new virtual machine to run takes 30 seconds. Getting a new Layer 2 segment and firewall configured for these virtual machines takes two weeks because you have to wait one week for the maintenance window to put the firewall rules into test mode and then another week to put them into production mode. So basically with automation, software defined networking, what we can do is we can create templates. So the infrastructure team designs a template and says, this is what a multi-tier application looks like. The end user, or the IT team can then pick a template and basically the template will automatically instantiate the Layer 2 domains, the routers, the firewalls, the load balancers and eventually the virtual machines. Once you've moved all your network configuration into a software layer, it becomes much, much easier to automate this process.

And what's even better is you can not only automate your own network configuration, but we're working closely with a large number of partners to help automate their services in these networks as well. So you may think, well, now that we have a firewall, are we competing with firewall vendors? Turns out that's usually not the case. So Palo Alto Networks for example is a great partner of ours where we helped provision their firewalls in a highly dynamic way in the data centre. And so our functionality and their functionality nicely complement each other.

Automating IT, what I've seen with customers, falls into three large buckets. The first one is, automate networking for your own IT team. So you have an IT team, you want to make it more productive, so you give them more powerful tools. They can script, they can define templates; you're going to have workflows. Specifically if you're doing complex things like, for example, integrating two different organisations after an acquisition, this can be very, very beneficial. Some of our customers are going one step further and they're actually now automating this configuration to the end user. So for example, by creating a self-service developer cloud, well my dev team can just pick a template and automatically instantiate that. There's nohumans involved with manually configuring the network here anymore. This happens in a completely automated way. And last but not least you can go one step further and actually provide this for external parties, so build a real public cloud with these tools; as several of our customers are doing as well.

And so the third big use case for virtual infrastructure that I think is emerging is application continuity. And let me define what we mean by that. So one of the things that virtual infrastructure does is that basically it isolates your applications at the top from your physical hardware underneath. What type of switch is running here, what type of firewalls are running in my physical infrastructure, as an application I don't know any more and I don't care, right? I'm isolated from this. I'm [the application] seeing virtual switches, I'm seeing virtual routers, I'm seeing virtual components but the physical hardware that's powering that, I'm now independent of this. This is a nice thing because what it means is if I actually want to swap out the physical components underneath, this is now possible. So if you're doing things like a data centre refresh, want to migrate parts of your data centre to a different rack, then replace physical components and migrate them back, so then it becomes much, much easier because I'm independent of the hardware that it's running on.

That's even more powerful if I have multiple data centres. So one of the big topics, specifically with banks on the East Coast at the moment is disaster recovery. They've all learnt from September 11, they now all have a data centre in Manhattan and another one across the river in New Jersey or even a little more upstate New York and to basically make sure that if something bad happens in one location, they have a back-up location they can move to. Keeping exactly the same hardware in two data centres is really, really hard because often these facilities are leased. That actually may be out of your control what kind of equipment configuration is running there. Now once you've moved the definition of the infrastructure into a software layer, this becomes virtual infrastructure, you don't care about this anymore. The other data centre has maybe even a different network architecture. Your main data centre is mostly running L3, the other one still has L2 domains, that's fine, right. You have your virtual infrastructure layer that abstracts out all these details of the hardware underneath so the applications on top just see the virtual infrastructure. You can have duplicate IP addresses, you can have heterogeneous environments; all of these things become much, much easier once you're able to define the configuration of your network in pure software.

So that's virtual infrastructure. It's a huge trend in the current on-premise data centre. But there's one more thing which is happening and that's the cloud. You've all heard of the cloud. So I want to talk a little bit about how I think this will evolve for the cloud. This is something very new so bear with me here. So I think the first thing that's important to understand is that the cloud really is about two things. The first one is it's about consuming IT as a service, right, but the second one that often gets mixed up, it's also about writing the applications in a different way. Taking a couple of exchange servers and moving them on Amazon today isn't possible; I don't think that actually makes a whole lot of sense. Typically if you look at your typical San Francisco start-up somewhere across the Bay here, if they write an application for Amazon, they actually write it differently from how a classical enterprise writes software. And the term, I think that has emerged for these different ways of writing applications is third platform apps. The basic idea of a third platform app is that you build it in a different way; you build in redundancy. So you assume every process can get killed at any time and should [preserve it] against that. Two, you build it for scale out so it runs in a number of instances that you can define. It's often built in a stateless way, meaning the state, the important data is actually kept in a service. And it often -- you build on a higher level of abstraction so it looks a little bit more like Platform as a Service. It's often not quite yet Platform as a Service; it still runs on an operating system but it takes some of these elements.

So the infrastructure layer of choice or the provisioning layer of choice for these applications that's emerging is called containers. And now -- I'm sure you've heard about containers before. So the basic idea is in a classic, VM based deployment, having VMs, if VM has an operating system or my application platform sits inside the VM, the application sits inside the VM, I'm getting these very heavy VMs basically where I run a couple of them on a server. Containers try to do this in a more lightweight way because we're sharing the operating system, sharing most of the application logic and then having these very lightweight containers on top.

Now the first benefit that gives me is I can scale a lot better; I can have a lot of containers running on one host and that's good. The second thing though is this is also at the end in many cases a much better way of managing my applications. If I'm running a data centre, even in an enterprise, do I really want every single app to have its own operating system, own copy of an operating system, or rather have them share one joint operating system. So I think it's a very, very good way of packaging applications.

Now, I started talking to customers about how do you actually run containers and software defined networking? And you know, actually we have several production customers on NSX with containers today. And basically they all follow the model that's shown here. And my first reaction was like, well, you're clearly doing it wrong. You still have a hypervisor here. You're running your containers inside a VM. Isn't this redundant; do you need both of these things? And they came back and said, no it's not. We need to do it this way for a very simple reason and that is that the isolation provided by containers tends to be very weak compared to the isolation provided by virtual machines. Isolation for containers is roughly the same as the isolation provided by Linux kernel and that's a very different order of magnitude than the isolation from VMs.

So basically what they do is they take all the containers from one application or at least one security domain and put them into one virtual machine. And then they have two or sometimes even one, but usually a small number of virtual machines running on top of a hypervisor with a V-switch underneath. Actually I think this is the model that probably enterprises will continue to use for probably the next year or so because containers today often don't have their own IP addresses anyways, right. So with this you have only visibility into endpoints at the level of a security domain but I think that's perfectly fine. So the logical next step, after that, is to say, we want to provide visibility at the networking level into containers as endpoints. And what you do in order to do this is you basically move a switch into the virtual machines. And that's something which we'll support in the near future.

Now you may ask, well, do we really still need a hypervisor if we have containers? Couldn't we get rid of that altogether and just run the containers directly on the physical host, we can still have a vSwitch underneath, right, but just run the vSwitch in the same space as the container host. And it turns out that's actually a really bad idea because let's think through what this means. So if in this setting the container gets compromised, what happens? An attacker can usually fairly easily escalate kernel-level privileges, so they can take over the virtual switch, because that's just running in the kernel, and now they're on the network and they can compromise the network. And you know, once they've compromised the network, they can go back into the host and compromise additional hosts. So having a hypervisor here is actually a really good idea; even if you just have a single virtual machine per host, right, having a hypervisor here is a good idea because that basically allows you to have an additional layer of defence, right. If somebody compromises your container host, they still can't access the network directly. They're still confined into whatever sandbox we've put them in using the hypervisor or the virtual switch, or essentially all these constructs we define in the virtual infrastructure. So basically I think, in a container world, we still have hypervisors, but they take on a very different role. They become the anchor point for a variety of Layer 4 to 7 services, Layer 2 services, Layer 3 services, of security services that you need to run your data centre. Even for scheduling you may use containers instead.

So that's third generation apps. How does it relate to the cloud? So the cloud is amazing. Amazon just announced earnings; I'm not sure if everybody saw that, they're doing incredibly well. The basic promise of the cloud is to say, I can, at any given time, rent compute capacity, rent services, scale them up and down. If I don't like one cloud provider any more, I go to the next cloud provider, I go to the other cloud provider. I can get this internationally in lots of different locations. That's fantastic right? Then you talk to people who have actually started using this, you discover something interesting. So I talked to the head of application development of a very large retailer and asked him, on Amazon -- they're heavy on Amazon -- asked, him, which services on Amazon are you using. And he basically said only these three. I was like, well, that's not a lot; why is that? He's like, well, they have an explicit policy with a white list; if you're one of the developers in Amazon, which services can you use. I was like okay, that's interesting; why are you doing this? And he said, well, it's very simple. If I use all the different services that one cloud provider provides me with, I [lose] the ability to move to other clouds, because now I'm locked in. All these APIs, all these services are different. If I build heavily on them, I've essentially just created another silo. I'm trying to go to a different cloud provider, this doesn't quite fit; the API is different. I go to another cloud provider, it doesn't fit any more. I'm now stuck with my one cloud provider. And so, at the end of the day, what I've really done here, I've created new silos; I've created silos where I have now one team that understands one set of services for one cloud provider, builds apps there, and I'm no longer mobile. That completely defeats the purpose of cloud, right? That's not what he wants.

So what I think will happen -- and this is a really new idea, but what I think will happen is that for clouds we will see the same thing happen that happened for on-premise IT, which is we'll see the emergence of an isolation layer that basically isolates the application from the underlying services, provides a layer of abstraction that gives you this mobility across clouds, right. If you have a virtual infrastructure layer that sits on top of these services, on top of the different functionality of clouds, you again get the ability to take your apps and move them one cloud to another because you can basically now bring your own infrastructure. You have software modules that create this abstraction; they make all clouds look the same. And taking this one step further, I think you actually want to have the same abstraction layer for your local data centre and for your back up data centre and for your cloud; one set of virtual infrastructure that isolates you from the hardware underneath and that gives the portability of your complex second generation and third generation apps across all of these different locations. And with that vision, I want to leave you and open it up for questions. Thank you.

Keynote Interview and Audience Q&A

Manek Dubash

So thanks very much for that; interesting presentation. The first question I have for you is that what's interesting about that is this end-to-end virtualisation idea. It sounds like vendor lock in to me.

Guido Appenzeller

So I think it depends on the abstractions they're delivering. If you look at it, what abstractions does virtualisation give you at the end of the day? The abstraction of a server; we didn't invent the server at VMware. That was there before us, right, the abstraction of a Layer 2 network, the abstraction of a Layer 3 network. So I think we're taking abstractions that are very true and tried and standardised. I mean, everybody is using these. Every IT department, if you tell them, look, this thing looks like an x86 server, they know exactly how to deal with this thing. So I think -- I will disagree. I don't think there's lock-in. It depends what you do, right? If we had the super-proprietary VMware extension, yes. But the abstractions we deliver are very, very standard.

Manek Dubash

Okay. If you say so, we believe him, don't we? The other question that sprang to mind, I have to say, this idea of ignoring the [software] is very compelling, it's very attractive, it's -- but, at the same time, we now know that, for example, chips have been compromised. You can't ignore the hardware from a security perspective.

Guido Appenzeller

Yes.

Manek Dubash

How do you address that?

Guido Appenzeller

So I think you never ignore the hardware. This idea of virtual infrastructure is not to say physical infrastructure doesn't matter anymore. We still need switches to transport packets. We still need physical servers to do the work. I think there will be a lot of innovation, there will be a lot of great new products there, but I think the idea is one of isolation. We want to -- the application developer should not have to worry any more what hardware are they running on. You want to isolate, you want to abstract out; I think that is the key principle. It's not that it becomes unimportant.

Manek Dubash

So how do you deal with compromised chips?

Guido Appenzeller

Actually a silicon with a back door build-in -- I'm a software guy. That's well above my [inaudible] right here.

Manek Dubash

Not my problem.

Guido Appenzeller

Look, I mean, I see large IT customers getting a lot more sophisticated about their supply chain. They're suddenly asking suppliers where the chip is from, where was the motherboard assembled. You hear these stories about -- so, if you know, the perfect shrink-wrapped line card for a switch, coming with the right serial number and everything, and having a hardware back door on it. I think this is happening. But it's -- at the end of the day, this becomes a vendor trust issue and a supply chain issue.

Manek Dubash

Not a VMware issue; okay. The other thing that you mentioned was the organisational change, the way the IT teams are going to have to change.

Guido Appenzeller

Yes.

Manek Dubash

That's a really hard thing to do.

Guido Appenzeller

Yes.

Manek Dubash

I mean, how are you seeing that developing?

Guido Appenzeller

Slowly.

Manek Dubash

Yes.

Guido Appenzeller

I mean, for our customer, I think, this is a huge challenge, right, and in some cases a challenge purely because you need to find the right organisational structure. I think there's really no really well-established play book at this point. I think we're starting to see it a little bit, but you probably want a cloud team, you still need specialists. Right? I mean, the tightrope you have to walk here is you still need the specialists that have the in-depth knowledge about their respective areas. I mean the Rabin protocols haven't disappeared, protocol stacks haven't disappeared. So you still need networking specialists, but they have to become a little bit more like the Renaissance network administrator that, in addition to sculpturing, can also paint and build flying machines, right? They really have to take a broader look at all the different areas that you now have to cover. And I think the most important thing is actually communication, right, because often the different silos are now tied together at the consumption layer. These teams they have to talk to each other. They can't just live in separate parts of the buildings anymore.

Manek Dubash

Are training courses -- do training courses exist to actually develop people that -- who can do this now?

Guido Appenzeller

I think so. I mean, the -- if you are, for example, you are a network administrator, what I would suggest is just start learning a little bit of dev ops, right. Learn a little bit of Python, learn a little bit of REST APIs. It's honestly not that hard and, you know, you can have so much more fun once you have the basic tools to automate. This is actually -- I think this is a very positive thing at the end of the day.

Manek Dubash

Yes, even I can write Python. Okay, questions?

Guido Appenzeller

Yes.

Jeremiah Caron, Current Analysis

I'm Jerry Caron from Current Analysis. Thanks for your presentation. I really want to follow up on that last question; actually I'm glad Manek asked it. The -- you at some point said it's a really, really exciting, great time to be an IT administrator and, you know, that's sort of dressing up the pig a little bit, because it's actually a very difficult time to be an IT administrator. And I do believe my interpretation of your response to Manek's question was glossing over the fact that the training issue, the skills issue is actually massive. Organisation, I believe, is easy, okay? You can organise any way you want. AT&T for its NFV SDN has organised itself, but they're spending tons of money to re-train 133,000 people. And so I think that VMware and your competitors need to spend a lot more on training to make what you described a reality.

Guido Appenzeller

So I largely agree. I mean -- let me sort of parse this. May you live in interesting times; I consider that well wishes, but, you know, it's certainly -- if you're the classic CCIE who has survived on CLI alone for the past 20 years, that may not look so great, right? I mean, you really have to change and you have to seek training. Yes, training is an issue; I absolutely agree. On the VMware side actually we are currently hugely ramping up our training programme around network virtualisation and we're seeing fantastic uptake there, right? And, so, for large organisations, I agree. This is a real challenge. I talked to a CIO of a bank -- sorry, director of networking of a bank who basically told me he's thinking about moving his networking operations centre to a different city because he can't get the right talent in the city that he currently is in. That's -- before you move things across state borders, there's usually a sign that that you realise you're in trouble. So absolutely correct, right? I think if you're running the large organisations, you have a lot on your plate and a lot of work to do here. If you're an individual network administrator, I think there's a huge opportunity in front of you.

Manek Dubash

Well, if that's it for questions? Good. Well, in that case, I'd like to thank Guido Appenzeller.

[End]