Intel

Introducing the SRv6 Service Engine

CORPORATE PARTICIPANTS

Lilian Veras
Moderator

John DeMay
Intel – BXD Business Development

Sven Freudenfeld
Lanner – CTO Telecom ABU

Jesper Eriksson
NoviFlow – VP Product Management

PRESENTATION

Lilian Veras

Welcome, everyone, to the Intel Network Builders webinar program. Thank you for taking the time to join us today for our presentation titled “Introducing the SRv6 Service Engine”. Before we get started, I want to point out some of the features of the BrightTALK tool that may improve your experience.

There's the Questions tab below your viewer. I encourage our live audience to please ask questions at any time. Our presenters will hold answering them until the end of the presentation. Below your viewing screen, you will also find an Attachments tab with additional documentation and reference materials, including a number of websites and documents mentioned in this presentation.

Finally, at the end of the presentation, please take the time to provide feedback using the Rating tab. We value your thoughts and we'll use the information to improve our future webinars.

Intel Network Builders webinar series takes place live twice a month, so check the channel to see what's upcoming and access our growing library of recorded content. In addition to the resources you see here from our partners, we also offer a comprehensive NFV and SDN training program through Intel Network Builders University. You can find the link to this program in the Attachments tab, as well as the link to the Intel Network Builders newsletter.

Intel Network business partners have been working to accelerate network innovation by optimizing their solutions on Intel Technologies. These industry leaders are recognized in our Winners’ Circle program and Lanner is a Titanium partner. Learn more about our INB Winners’ Circle program by clicking on the link in the Attachments tab.

Today we are pleased to welcome John DeMay from Intel, Sven from Lanner, and Jesper Eriksson from NoviFlow.

John DeMay, Senior Director of Business Development for the Switch and Fabric Group at Intel. He joined Intel through the acquisition of Barefoot Networks by Intel in 2019. John was a member of Barefoot Networks since 2016, and was instrumental in the launch of three generations of the Tofino product line. Prior to Barefoot Networks, John was at Broadcom for almost 10 years, focused on networking solutions.

Sven is the Chief Technology Officer of Lanner’s Telecom Applications Business Unit. Sven possesses 30 years of experience in telecommunications, network computing, product management, and business development. Prior to assuming this position in Lanner, Sven had taken several managerial positions in global business developments for NoviFlow, LEI Technology, and Kontron, and had greatly contributed to the cloud, media, and telecom market segments. He was fundamental in accelerating the adoption of P4 and OpenFlow-based packet forwarding technology, virtual CPE/SD-WAN, and Advanced Telecommunication Computing Architecture technology in the service provider market.

Jesper has more than 30 years of experience working with service provider networking. He's one of the co-founders of NoviFlow. In his role as VP Product Management, he leads a team responsible for product management, sales, engineering, and prototyping new product concepts in collaboration with service provider customers and technology partners. Previously, Jesper held leadership positions at Ericsson and Nokia, leading large customer-facing teams serving service providers in North America. Jesper has a MSC from the Royal Institute of Technology in Stockholm, Sweden, and an MBA from the University of Virginia.

Welcome, John, Sven, and Jesper, and thank you again for joining us today. I will hand over to Jesper to start off. Thank you.

Jesper Eriksson

Thank you, Lilian. So welcome, everyone. So I'm going to lead off this presentation, and today we're going to talk about using SRv6 and programmable networks to reduce service costs at the network edge.

And I need to start the presentation by talking a little about what SRv6 is. It's an IETF standard with broad industry support from leading vendors and operators. And the way to look at it is as a way to program networks to deliver not only traditional services, but also new services. And the way it works is with SRv6, you create a domain, a network domain, with a set of predefined segments, and you should look at segments as an instruction for a particular node, how to process the packet. And these instructions or segments are easily extendable to allow new functionality being introduced.

An SRv6 domain is composed of three types of nodes. The headend node classifies the packet as it enters the network, and then it encapsulates the packet with an SRv6 header, and encodes an ordered list of segments or instructions into the packet using an SRv6 extension header. You can see that over on the right in the top drawing there. We have an outer IPv6 header, and then the SR extension header with an ordered list of segments, and this extension header is called the Segment Routing Header, and a list of segments is called a policy. So that's a policy for how the network will process the packet as it traverses the network.

The transit node is a node in the middle and helps and processes the packets based on the information in the SRH, and then the egress node removes the outer IPv6 header and the SRH, and forwards the packet using the packet's original protocol.

So what is SRv6 used for? Well, of course, you can do traditional services like traffic engineering, delivering L2/L3 VPN services, but you can also do service chaining, and more general service programming, and I will explain later what this is. And also, very important, SRv6 may be deployed as an overlay to an existing IPv6 network.

So what benefits does SRv6 provide? The first is reducing the cost of delivering network services, and this is exactly what the SRv6 Service Engine does that we're going to talk about today. We call this service programming, how to insert and chain network services in the network through a policy, rather than hardwired into specific locations in the network. It provides for better scaling of the network. Because SRv6 is based on IPv6, it has a much larger address space than say MPLS. And comparing 128-bit SIDs to a 20-bit MPLS tag.

It also simplifies the network. With SRv6, we can reduce the number of protocols needed to deliver network services. It's easy to deploy. It's supported by most router vendors today on existing platforms, and also it may be deployed as an overlay on an existing network. It also supports existing traditional IP and MPLS network functionality, like forwarding, traffic engineering, VPN services, high availability. But also, more importantly, it's extendable to new network functionality, and today we're going to talk about SRv6 proxy, service chaining, but there's also work being done implementing 5G user plane functions.

So I also want to talk about what other carriers and vendors say about SRv6, and starting with Daniel Voyer from Bell. "Segment Routing is fundamental for today's reality, which requires on-demand services as well as exponential bandwidth growth. Streamlining the IP protocol stack in order to provide a simplified service assurance support model for day-to-day is the key benefit." SoftBank, another leading operator with SRv6, "SoftBank keeps focusing on improving service quality and enhancing the reliability and agility of networks while reducing costs." And then lastly from Jonathan Davidson at Cisco. "SRv6 dramatically simplifies the routing architecture in comparison to legacy MPLS and VLAN designs, and makes it easier to automate and deploy services wherever you want to deploy them in the network."

So the SRv6 Service Engine has been deployed at Bell Canada, and the problems that Bell wanted to address with SRv6 Service Engine, and that we're solving, are the following. It's building for the future, introducing a new technology that can evolve the network to make it easier to deploy new network services, moving infrastructure to the network edge in order to be able to deliver low latency 5G applications. Also, at the same time, virtualizing the infrastructure, and flexibly inserting services where they are needed in an instance. And then efficient scaling. And here with NoviFlow and Tofino, some of the key services are embedded into the network fabric and we'll talk more about that later. And then lastly, application acceleration, accelerate these virtualized services deployment with various Intel Technologies like the Tofino, SRV, DPDK, et cetera.

So now we're going to talk about more in detail on the components of the SRv6 Service Engine. And starting with the NoviFlow software, you have the NoviFlow fabric, which performs a SRv6 service proxy function, as well as service chaining and load balancing of these virtualized network services. The control plane runs on one of the x86 compute blades, but the forwarding plane runs embedded into Tofino switch silicon.

We also have the NoviAnalytics and NoviDashboard, which provides visibility into the health and performance of the SRv6 Service Engine, including the hardware, operating system, virtualization layers, and the applications, and it allows for easy troubleshooting and predictive maintenance.

And then the most important part is the virtualized network services. Now, they are the payload of the platform, and the whole purpose of the platform is to insert and service chain these virtualized network services in the traffic flows of the network. These are things like firewalls, DDoS detection and mitigation, intrusion detection and prevention systems, TCP optimization, carrier-grade NAS, et cetera.

And lastly, the Lanner HTCA-6600 hardware platform, which combines compute, networking, and storage. The six compute blades have dual-socket Intel Xeon CPUs, each with an EA10 100-Gig connection to the backplane, and the two networking blades each have an Intel P4 programmable Tofino switch ASIC.

Now I'm going to hand over to John DeMay from Intel to talk in more detail about the Intel components of the platform.

John DeMay

Great. Thank you very much, Jesper. Hi, everyone. I'm John DeMay. I'm with the Intel Switch and Fabric Group, and I'm excited to be here today with my colleagues from NoviFlow and Lanner to discuss the SRv6 solution.

So at the heart of this SRv6 Service Engine is the Intel Intelligent Fabric, and as I go through the next few slides, there's a few or four takeaways that I want to hit upon here.

So first and foremost, networks are moving towards software and programmable networks, and why is this? This is due to lower costs, reduced complexity, and ease of deployment, and the programmable networks provide resiliency, optimization, and adaptability for ever-changing workload requirements.

Second, the Edge is transforming and automating every industry. So Edge Computing is transforming the way that data is being processed and delivered to billions of devices in the world, and with the explosive deployment of IoT, and the arrival of new technologies such as 5G, artificial intelligence, machine learning, et cetera, having that compute, storage, and analytics close to the end devices is gaining momentum.

Third, Intel Connectivity Group was purposefully created to address all these challenges. The group delivers end-to-end connectivity, and fully programmable solutions for all major enterprises, cloud, and telco customers.

And then fourth, Intel's technology will help accelerate the delivery of differentiated services, and provide flexibility to changing customer demands on the fly. Our vision is to partner with customers to optimize network performance and efficiency, better meet service level agreements, improve customer experience, and do this all with next-generation programmable networks.

So why is there such a huge growth? Why is there such a great demand? Well, we have some data here from Cisco, and what this graph is showing you, on the left-hand side there, there's about 3.9 billion Internet users in 2018, and that's expected to grow to about 5.3 billion users in 2023. So, that equates to a 6% CAGR over those five years, and what this means is that globally, devices and connections are growing faster than both the population and the number of Internet users, and so this trend is accelerating the increase in the average number of devices and connections per household and per capita.

So this is resulting in what we're calling a seismic shift. Legacy networks are complex, and they're difficult to manage. They involve many layers of hardware equipment and control software that is tightly integrated and proprietary, and these networks operate with vendor-specific protocols and technologies that increase their complexity and slow down innovation, while at the same time, because they're proprietary, costs are increasing. So enterprise and cloud service providers are looking for ways to innovate, to ease deployment, and reduce that total cost of ownership, and this has given rise to the programmable networks, where these network operators can not only contribute innovations they develop themselves, but it also lowers the barrier to entry for deployment, and reduces the complexity, while increasing performance and flexibility. So part of Intel’s strategy is to add intelligence at the Edge. By leveraging our software and our hardware, enterprises and cloud service providers can predetermine deployment scenarios, it provides ease of use for troubleshooting, and ultimately results in a significant return on investment, and the end goal is to enable fully automated and programmable infrastructure by leveraging open networking standards.

So what are some of the challenges that we're starting to see with 5G networking? Even more challenging than the hardware are the software loads. Many service providers are now on the move to cloud-native architecture and container-based processing, microservices, orchestration, and automation, and these changing workloads spawn the need for growing network optimizations, as these enterprises migrate to cloud-native architectures. Also, distributed storage and workloads in a scale-out world is changing the way that networks are being architected. In the distributed scale-out world, there's also a desire for putting control back into the hands of the operator, hence programmability.

The other thing that's needed is visibility. It's very important to have this increased visibility for root causing and remedying network slowdowns, as these networks become more and more complex, and of course, end-to-end security is important. It's always a major consideration. All of this needs to be done while keeping a finger on the pulse of CapEx and OpEx investment, ensuring that it doesn't go up over time. And thus, in addition to increasing bandwidth, the network also needs to get smarter.

Okay, so how does Intel envision this? Well, we have the Intel Intelligent Fabric. As you can see from the right-hand side of this slide, it's not only just the switching device, but it's a collaboration across all of the Intel portfolio, whether it's the IPU, Ethernet adapters, FPGAs, CPUs, and even optical conductivity. From both the hardware and software perspective, the goal is to provide innovations to data centers and networks.

Additionally, Intel has taken a leadership role in developing industry standards designed to reduce overhead and uplift networking performance. The Intel Intelligent Fabric provides the ability to take advantage of open-source programming tools and software, such as P4, the OCP SONiC project for network operating systems, and other open-source tools such as EBPF, DPDK, EBGP, et cetera. The end goals here are to, number one, increase the ease of use. Number two, to provide this massive bandwidth that's needed to meet these growing data needs. The third is to provide the ability to have an AI-driven self-monitoring, self-analyzing, self-healing network, all the while with number five, enhancing the end-to-end security, and number six, we do this while improving density, power, and lowering costs.

So what are the three key benefits from the Intelligent Fabric? Well, first, there's intelligence. I already talked about using this open-source P4 language to program a pipeline and customize as needed. But also to provide intelligent packet processing for accelerating workloads. Not only AI and ML, but other workloads as well, and we have the ability to leverage the FPGA family in our portfolio to expand the table and buffer sizes.

As far as performance goes, the Intelligent Fabric has a top-end throughput right now of 25.6 terabits per second. We have 112-Gig native SerDes on the devices that can operate at also 56 and 25-Gig. We couple that with high-speed Silicon Photonics. And then we have some power-optimized use cases using the Intelligent Fabric processors I'll talk about in a second.

And then lastly, visibility and control. This is a major request that we get from our customers, being able to use tools such as INT to enhance congestion control, identify delays or hotspots in the network, and then to be able to analyze this information in some fashion, and we have this Deep Insight Network Analytics Software that provides that functionality. And then also INT, or In-band Network Telemetry, is expanding beyond just the switches. Now we're able to provide this level of telemetry from the host, and basically from any programmable data plane product in the Intel portfolio.

Alright, so let's switch gears just a second here. We'll talk a little bit about the roadmap. So on the left-hand side, you can see the initial device that we brought to market (Tofino). This has been shipping for quite a few years now. It has a 25-Gig SerDes, so it supports I/O speeds of 10-Gig all the way up to 100-Gig. And the performance on the device is from 1.2 terabits per second all the way up to 6.4 terabits per second.

With Tofino 2, which we brought to market last year, we increased the bandwidth there. We doubled from 6.4 up to 12.8. And we built this on a 56-Gig SerDes as well. So, now, you can support all the same speeds as Tofino, but in addition, 200-Gig and 400-Gig. And on that device, there's a 400-Gig MAC as well.

And then Tofino 3, which is sampling now. We again doubled the performance from 12.8 terabits per second to 25.6 terabits per second. But now, with Tofino 3, we've added native 100-Gig SerDes. So, this will allow Tofino to talk directly to other devices in the network at 100-Gig. That SerDes is pretty versatile, it will also operate at 56-Gig. And this has a 400-Gig MAC on it as well.

At the bottom of the slide, you can see a reference here to a Tofino extended architecture. And this is basically where we use an Intel FPGA, as I mentioned in the previous slide, coupled with the Tofino Switch. And what that allows us to do is to enable anywhere from a 10 to 100x increase in the table and buffer capacity. And so, this has enabled us to deploy in use cases that are shown here on the lower right-hand side, so anything from gateways to network packet brokers, to carrier-grade security, and NFV, as well as many use cases with acceleration. So, that’s the Tofino roadmap.

Moving on. I think it’s important to talk a little bit about the performance and power efficiency gains that can happen at the same time. And so, what we've done here is that we – because the device is programmable, we provide an efficient pipeline that doesn’t require unused or unnecessary protocols. Rather, it provides the ability to only use the resources that are needed for the particular use case or even to implement new and customized protocols where needed.

And so, what does this result in? This gives you increased performance and power optimization for not only the hyperscaler use cases, but other types of networks as well.

Moving on. I know Jesper talked a little bit about IPv6 and SRv6, but just to kind of recap in terms of how we’re supporting it. So, first off, what is IPv6?

So, IPv6 stands for the Internet Protocol Version 6, and adoption is growing at a torrid pace. So, when you look at Google statistics, they indicate that 22% of users today reach Google over IPv6. And this is a twofold growth compared to just two years ago.

And why is this? Well, because things like IoT rely on IPv6 as it provides connections for existing devices and millions more in the future. IPv6 is also a key enabler for Segment Routing concept, as it provides the reachability to SRv6 capable nodes.

Segment Routing then is a new way of doing source routing where the source selects the path over a network, placing an ordered list of 128-bit IPv6 addresses into the header of an IPv6 packet. So, SRv6 and Segment Routing are next-generation IP bearer protocols that combine Segment Routing and IPv6. And we’re seeing this start to get deployed in large-scale commercial networks.

Some examples that are public are SoftBank, Iliad, China Telecom, China Unicom, China Bank, Bell Canada, and others.

And then just to kind of wrap up here, why SRv6? Well, it's an SDN solution for source routing. It provides this powerful programming ability that Jesper alluded to in his discussion. And it also provides a flexible solution for traffic engineering. And it's easy to deploy. It doesn't have the complex and difficult protocols such as MPLS, nor does it require hardware support for forwarding.

So, with that, I will turn back over to Jesper.

Jesper Eriksson

Thank you, John. And now, I'm going to have a couple of slides on where would you insert – where do you need to insert network services? And I'll start with a 5G network architecture and 5G networks.

So, in 5G networks, there are two places where you would need to insert different types of network services. The first one is on the N3 interface, the interface between the CU and the user plane function. And the reason you want to insert network services here is to protect the mobile core from traffic coming in from the access side. With the high uplink throughputs available in 5G, there is a need to protect your network also from the access side. And then the other place where you would want to put network services is on the N6 interface. And this is for traffic to and from the Internet, and also to and from other operator services, such as the 5G voice and messaging services.

More generally, the SRv6 Service Engine could be inserted in a number of locations of an operator's network. And starting on the left, at large enterprises on-prem, you can deploy an SRv6 Service Engine and provide some network services. The next is at the local center office where multiple enterprises and consumers aggregate. And the next step up would be at the local data center of the service provider. And then further into the network, you will have the global data center of the service provider.

With SRv6 service programming, these services in all these locations become addressable from anywhere in the network. And this is really one of the key benefits of the SRv6 Service Engine. You can basically address any service anywhere. And it's worth reminding everyone that the way this is done today is to essentially hardwire the often physical appliances into specific locations of the network, and only then be able to process the traffic that goes through those locations.

So, now, we're going to talk a little more about one of the key features of NoviFabric, one of the software components of the engine.

And one of the key features of NoviFabric is acting as an SRv6 service proxy for one or more of these network services like DDoS, firewall, or IDS. And these network services may be also service-chained by NoviFabric, and then a particular service chain of these services is called a Service Policy. And you program that into the NoviFabric through a northbound API.

And you can look at NoviFabric as an orchestrator or enforcement point of such a service policy. And the way this is implemented is that for each segment or SID that the NoviFabric matches on, the packet is sent through a specific ordered list of network services.

So, here is an end-to-end complete walkthrough of how a packet may be processed by an SRv6 network. So, the NoviFabric in this picture here is the NoviFlow software running on the Tofino switch in the SRv6 Service Engine. The different appliances here like firewall – or the services, firewall, DDoS, CGNAT are different services running on the compute planes of the SRv6 Service Engine.

So, let's start from the beginning, from the start, from the left. Say Host X wants to send a packet to Host Y. Well, so Host X would put together a packet – here in yellow and orange – where source address is the Host X and destination address is Host Y. And it sends that packet to Router A. Router A is the SRv6 headend router and will classify this packet based on some rules, and then apply a policy to the packet based on this classification.

So, coming out of the router, you will have this encapsulation IPv6 header and the SRH which defines what segments the network needs to process – run the packet through as it traverses the network.

So, the policy, in this case, is first send the packet to Router C and then to Router F, and then finally to Router H. Router B, E, D, and G are all SR-unaware. They're just traditional IPv6 routers. They just look at the destination address and forwards the packet to the next hop.

So, as the packet goes through the network, once it hits Router C, it will update the destination address to Router F. And then it gets to the NoviFabric, and it will match. And in this particular case, it might be just a single firewall, it will run the packet through the firewall, then update the header again and forward it to the next hop, Router G. And once the packet hits Router H, it's the egress router, it will remove the outer IPv6 header, the SRH, and deliver the pristine packet, original packet to Host Y.

So, when we at NoviFlow were developing this SRv6 proxy function, we needed to understand what types of network services there are out there. In particular, we need to classify their behavior in relation to SRv6. So, we came up with three types of network services. The first type is if the – is if the network service is SR-aware and, basically, understands SRv6 and accesses segment in an SRv6 domain, and is able to process the SRv6 header information. This would have been the easiest solution. And if this was available, a service provider could simply insert the service in the SRv6 policy at the headend router or ingress router. However, we are not aware of any firewalls or DDoS services that are SR-aware yet. So, this is not really available yet.

The next type of SR services that we came up with was an SR-passthrough. And the idea with the SR-passthrough is the network service ignores the SR encapsulation and basically processes the inner packet without touching the outer packet. And this is actually our first solution and was put into production at one of our customers. It requires some software changes in the network service and that network service had to be updated and treat SRv6 encapsulation, so to speak, as just another tunneling protocol and only process the inner packet.

And then the third, Type III, this is probably the most prevalent. Here, the SRv6 is called SR-unaware. And what happens here is that if the Type III network service receives an SRv6 packet, most likely it will fail to process the packet. It will probably drop the packet. And the way to solve this is you basically have to remove the SRv6 header and then send the packet to the network service, and then when the packet comes back, reattach the SRv6 outer header and SRH, update them and then forward it to the next hop in the network.

And this was the second solution that NoviFlow implemented, and is available on the Tofino.

I will now hand off to Sven who will take a deeper look at the SRv6 Service Engine hardware platform.

Sven Freudenfeld

Thank you, Jesper, and welcome everyone. My name is Sven. I'm part of the Lanner team of the CTO office focusing on some of the solutions for software-defined networking in the telco space, but also in the enterprise space.

So, let's take a look at the underlying elements to make this actually happen and to make it work.

So, we have created a platform, which is a true carrier-grade telco platform, which is aimed for Edge Computing. And most of the traditional terms of Edge Computing is really focusing on computing. So, most solutions in the market today, they have separated networking and computing as independent platforms, and we have seen the need of having it fully integrated where we are combining computing, networking, acceleration, and management into a single device, a single platform. Because the Edge locations are very tight in terms of space, power, but also in terms of rack space to manage that.

The concept is nothing new. A couple of years ago, there was an initiative focusing on standard hardware which was Advanced Telecommunications Computing Architecture. Fundamentally, the framework is very similar and identical, but we made a platform which is completely built by us for Edge Computing, where we have a bladed architecture.

So, we have multiple elements here. It's a middle plane approach where we have up to six compute nodes as a modular approach. We have two redundant switch fabrics which are based on the Tofino currently with a self-contained control process, as well the option of supporting IEEE 1588 for some of the other applications, for example, virtual RAN and so forth. And then we have additional options for storage, but also for the NIC modules to add in aggregation for some of the use cases, for an example, which are BNG use cases and ours.

But the key takeaway on this platform is it's scalable, so there is no need for a forklift update. The current platform is leveraging – the majority of the platform is using and leveraging Intel technology, in particular the Intel Xeon Scalable CPUs, and the Intel Barefoot Tofino, plus the latest Intel NIC modules throughout the backplane connectivity with a scale of multiple 100-Gig in the platform.

And to wrap that all into a management layer, we have implemented a visual analytics tool, which helps to manage, monitor, and maintain the platform in remote locations where we can reduce the overhead of truck rolls to do physical interaction with the platform, and we manage all elements and resources through that tool, which is based on an open-source framework, Cabana. But it's really optimized to manage all the sensoring data and everything to have a full picture of the status of the different elements in the platform and to be able to scale.

So, when we kicked off this project, we were looking at Edge Computing overall. And as I mentioned earlier, everybody talks computing at the Edge, but nobody talks about networking. But the reality is networking is the most dynamic framework at the Edge location, at the Edge point of presence, rather than with data center, top-of-rack spine, or leaf switches. Because on the Edge location, you have different types of traffic coming in, it might be IoT traffic, it might be voice traffic, it might be encrypted traffic, so there's different types of traffic hitting the network right at the Edge location. And it's very dynamic and, therefore, programmability is a big aspect on that. And using the P4 framework available from the Tofino with the SDN framework of using P4 runtime to manage the pipelines in combination with compute is the next logical thing to really make a scalable solution at the Edge.

And the other aspect also is you don't want to have all traffic packets going to the entire network for additional processing. You would like to – or you need to have processing capabilities available at the Edge and, therefore, we have the resources available right at the Edge where we can scale based on the different VNFs we are going to be deploying in the network.

The tool from Tofino with the – and what John and Jesper highlighted briefly, the In-Band Network Telemetry is the greatest tool at the Edge location, because the telemetry packets will be able to identify the traffic route, or the shortest path, or the best available path within the network. When traffic is hitting the network, it's typically coming from the Edge. And having that capability on the Tofino architecture is a great enhancement and a cost reduction for some of the workloads where we can pass on critical applications, the local compute resources, or redirect traffic into different directions within the network.

And then the platform is built to scale. So, today, we are on Intel Scalable, also known as – the codename was previously Cascade Lake, and it's moving towards the next-generation platform. The elements here in the compute nodes are the 100-Gig NICs going through the backplane. So, we really eliminate the need for dedicated top-of-rack switches, cabling, fibring, wiring, and so forth, and combining it all into a single platform approach.

There is a follow-on platform, which we also are leveraging the same concept, fundamentally. But even with a much greater density on this. We're leveraging also with Tofino Switch Fabric. These are all carrier-grade redundant switches, and we have compute modules, which can host also the combination of acceleration with an FPGA. For example, we can use the same framework for the SRv6 Service Engine for multiple applications. So, not just only for, let's say, DDoS or virtual firewall and so forth, we can actually also bring it into a network topology where we can address radio access network elements into the platform and leverage the combination of Intel Tofino architecture, but also the acceleration in the platform.

So, that deployment concept is really to be able to use the SRv6 Service Engine anywhere in the network. In the service provider we mentioned earlier, the intention is to have that simplified network topology to introduce Edge Computing into multiple use cases, and reuse the implementation for different areas in the network.

Since I talked about the Intel technology, of course, the big helper in this deployment model here is that we are using dual-socket compute nodes or server blades. And the way we're implementing it – so, we're leveraging the Intel 100-Gig NIC elements, as well the technology for acceleration with QuickAssist Technology. And the way we are distributing the workload is we are splitting it in two NUMA nodes or two compute sockets operating independent to reduce latency, and also to have greater performance for the 100-Gig towards the backplane.

One other thing to mention is that the Tofino ASIC also is acting as a load balancer, so there is no need for a dedicated load balancing function in the platform. And therefore, we can reduce the overall footprint by leveraging the technology from the Tofino for some of the network functions we are deploying in combination with SRv6. And we specifically give it the name SRv6 Service Engine, because we are really mapping the applications into an SRv6 framework. And therefore, you need a combination of compute and networking at the Edge. And we are greatly reducing the overhead cost at the Edge location.

So, just in summary, it's about close to 90% cost reduction from a previous purpose-built appliance model where we use pizza box servers, top-of-rack switches, purpose-built load balancers, and dedicated fibres, QSFPs, or transceivers just to deploy this platform. But the cost also is – the cost reduction for the truck roll. As I said, the biggest burden today in Edge Computing is truck roll, going onsite, doing configuration changes, doing – fibring up the systems, and so forth.

So, since it's fully integrated, we are managing the platform as a true software-defined architecture. So, we eliminate the cost burden on the actual truck roll to go onsite. And then with that, there comes a cost reduction for space but also for power.

So, that's some of the key elements on why this platform for Edge Computing, in combination with the Intel Tofino, but also with NoviFlow the SRv6 implementation is the right solutions for Edge Computing.

One of the use cases, since we talked a bit of multiple use cases. Our newest addition to that with the same type of architecture will introduce the SRv6 Service Engine also to provide great flexibility for the radio access network where we can distribute the workload over multiple – five servers, in combination with an accelerator card. So, we can consolidate some of the workloads into a single platform in a short-depth framework with telco carrier-grade characteristics.

And the use case that we have in deployment today, which is carrying live traffic in a network in Canada is really a combination of the service-chaining for multiple security engines. So, we are basically – have a proxy for the service provider in the hyperscaler. And the reason why this is an interesting approach because, from a security aspect, most of the service providers would like to maintain the control over security at the Edge locations. Because as I said, this is the most critical one and it requires scalability at the Edge location to scale up and scale down depending on the traffic flow.

So, to summarize what we are presenting here in the solutions that we have built together is really aiming towards the Edge location, introducing SRv6. So, why SRv6 Service Engine now? There are a few elements on there. I'm just highlighting. I'll let Jesper and John pick up the other ones.

So, the programmable data plane, the open programmable data plane really highlights the capability of changing traffic flows dynamically. And that will reduce the time into production, it reduces the time of reconfiguring network topologies. So, it's really an enabler for the Edge location side.

So, Jesper, I'll let you maybe highlight a few things on the SRv6 side.

Jesper Eriksson

Yes, so I think to answer that question, why SRv6 Service Engine now? I would say SRv6 networking technology is taking off right now worldwide. The Intel Tofino allows us to very quickly prototype and develop fairly sophisticated SRv6 functionality using P4, and then run it at terabit speeds. And the speed by which we can prototype new functionality is just amazing.

As an example, we originally had a full SID implementation, 128-bit SIDs, and we prototyped a micro-SID solution in about a week.

And then the third thing I want to highlight here is that service programming is an important issue for all network operators. How do I insert security services into the traffic flow, in my network, in an affordable way? And how do I service-chain these different network services?

So, I'm going to hand off to John for a last word here.

John DeMay

I think, really, in addition to the comments you made on programmability and the SRv6 – the benefits of SRv6, we have a great partner ecosystem such as Intel, Lanner, and NoviFlow.

And I would just end with this slide here just to remind people the future of networks is now. With P4 and with network intelligence, customers can rapidly customize their flows, add new protocols, et cetera, so they can innovate, adapt, and be ready for whatever comes in the future as these workloads rapidly change.

So, with that, maybe turn it over for questions.

Lilian Veras

Awesome. Thank you all so much for the great insightful presentation. We do have a few questions that have come in while you were presenting, so let's get started on those.

Before we jump into questions, just a quick glance through our references and Intel notices and disclaimers. This deck will also be available for download after the webinar ends.

So, question number one we have here. "How does the Lanner platform differentiate itself from other ODM vendors?"

Sven Freudenfeld

Let me take this first up. So, the differentiator here is that we are integrating computing, networking, and acceleration all into a single platform. And we manage all the different components by using a tool to be able to help the service provider to predict maintenance, as well to manage all the different resources.

In addition to that, in order to deploy this in that fashion of a programmable networking infrastructure, we are leveraging the Intel technology, including SR-IOV, DPDK on the compute side. We are leveraging the 100-Gig NIC infrastructure. So, it's all fully integrated. So, we are leveraging most of the available tools to optimize it for an SDN framework.

So, that's the differentiator from a platform perspective compared to other solutions in the market, which are addressing that still slightly differently on its traditional way with top-of-rack switches. They might not be programmable, and then dedicated compute or pizza box compute nodes, which are not really fully integrated and needs additional management layers on top of that.

Lilian Veras

Awesome. Thank you, Sven. Another question we have here. A member of the audience says he's seen lots of announcements where carriers are working with hyperscalers for the Edge. What are the advantages of the service provider for adopting this solution?

Jesper Eriksson

Yes, I'll take that one. So, this solution that we presented today is for the service providers that want to run their infrastructure on their own internal cloud. It provides a platform for the service providers to run their own network services on their own infrastructure. And it allows them to protect that infrastructure by easily inserting different network services such as firewalls and DDoS. And it also helps the carrier to push the infrastructure – their infrastructure to the Edge.

With 5G, part of the infrastructure is being pushed to the Edge, but you also need to push the security services that protects that infrastructure at the Edge, and that's what this platform does. So, it really allows the service provider to manage the Edge.

Lilian Veras

Awesome. Thank you. Question number three. "What are the enterprise use cases for the Edge?"

Sven Freudenfeld

So, I can kick it off. So, some of the use cases, I already highlighted it, it's more on a larger scale, but enterprise use cases could be a mix of private 5G, for an example, where we can use it in a deployment model. And these edge platforms are aimed towards that. Private 5G is an easier adoption because we don't deal with their legacy infrastructure. We can easily adopt it very quickly.

But the majority on that enterprise-scale is around the massive scalability on network security. So, network security and to deploy multiple virtual network functions from different vendors. It's an open platform. We can deploy virtual DDoS, virtual user plane function, virtual BNG, for an example, and other virtual network services right at the Edge. One concept was also to use it as a gateway within the multi-dwelling units, for an example, where it's sitting somewhere in a telco closet. And we can aggregate traffic and we can mitigate traffic right at the Edge location.

Lilian Veras

That's great, thank you. We are running out of time. So, we have one last question. "What are the Intel technologies that are being used in this platform that provide differentiated architecture solution?"

John DeMay

Well, I think, first and foremost, is the Intel Intelligent Fabric processor, the Tofino family. But I believe they also are incorporating the FPGA product line as well as our Xeon processors. Any other components that I missed there, Sven?

Sven Freudenfeld

You are missing the largest, but they are plastic. So, pretty much 95% of the platform is built on Intel technology.

Lilian Veras

Awesome. Well, thank you all for joining us today. I will ask our members, our audience to please do not forget to give our team a rating for the live recording so that we may continuously improve the quality of our webinars.

Thank you all. And this concludes our webcast. Bye for now.

John DeMay

Thank you.

Sven Freudenfeld

Thank you.