

Data Center Technologists

Blog

Search

Article Options



BGP in the Data Center: Why you need to deploy it now!

by [Doug Hanks \(JNPRdhanks\)](#) ★ 01-30-2014 01:19 PM - edited

Overlay networks in the data center are here and are here to stay. It's now easier than ever to programmatically provision new networks with a click of the mouse than ever before. No need to worry about VLAN IDs, integrated bridging and routing, MC-LAG, and spanning tree. Overlay networks use data plane encapsulations such as VXLAN or GRE to transport both Layer 2 and 3 between virtual machines and physical servers. One of the key requirements in an overlay architecture in the data center is to have a rock solid IP Fabric; simply Layer 3 connectivity between every host in the network that participates in the overlay network.

Does that sound familiar? Maybe a little bit like MPLS? You're right. A MPLS architecture requires a stable Layer 3 transport in order to provide IP services across Layer 2 and Layer 3 VPNs. Although similar, there are a few key control plane differences in a MPLS and data center overlay architecture. Let's walk through them.

MPLS has a hierarchy of control plane protocols that make up the network. It's common to see IS-IS or OSPF provide reachability between all nodes and provide a traffic engineering database. The next step is that each provider edge runs MP-BGP; it's the 18-wheeler in networking. MP-BGP carries all sorts of data from MAC addresses to identifying which VPNs should be installed into each provider edge. Finally there is LDP and RSVP which are responsible for label distribution and traffic engineering across the network.

As of today the control plane in a data center overlay network is fairly simple. The first option is to simply not use a control plane protocol. In this scenario we can use multicast to flood traffic control traffic to all hosts in the network. The next option is to use either OVSDB or DMI as the control plane protocol. These options prevent unnecessary flooding throughout the network and allow for a more efficient utilization of resources.

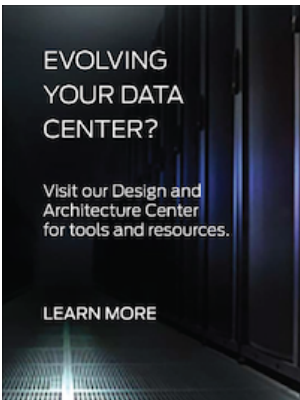
The biggest question is when you build an IP Fabric, what control plane protocol do you use? The options are the usual suspects: OSPF and IS-IS. But what about BGP? But isn't BGP a WAN control plane protocol? Not necessarily.

When creating an IP Fabric there are a few services that we need: prefix distribution, prefix filtering, traffic engineering, traffic tagging, and multi-vendor stability. Perhaps the most surprising requirements are traffic engineering and multi-vendor stability. When creating a large IP Fabric, it's desirable to be able to shift traffic across different links and perhaps steer traffic around a particular switch that's in maintenance mode. Creating an IP Fabric is an incremental process; not many people build out the entire network to the maximum scale from day one. Depending on politics, budgets, and feature sets companies may source switches from different vendors over a long period of time. It's critical that the IP Fabric architecture not change over time and the protocols used are stable across a set of different vendors.

Let's map the requirements of an IP Fabric and map them to the options in the control plane: OSPF, IS-IS, and BGP.

Requirement	OSPF	IS-IS	BGP
Prefix distribution	Yes	Yes	Yes
Prefix filtering	Limited	Limited	Extensive
Traffic Engineering	Limited	Limited	Extensive
Traffic Tagging	Basic	Basic	Extensive
Multi-vendor stability	Yes	Yes	Even more so (think about the Internet)

Announcements



Top Tags

[Virtual Chassis Fabric](#)

[Automation](#)
[CAE](#)

[Network Director](#)

[analytics](#)

[Disaggregated Junos](#)
[ND](#)

[openstack](#)
[QFX5100](#)

[VCF](#)
[vmware](#)
[BGP](#)

[View All](#)

Community Resources

[Event calendar](#)

[Technical webcast recordings](#)

[Products](#)

[Training](#)

[Terms of Service](#)

Labels

[VCF \(3\)](#)

[Virtual Chassis Fabric \(2\)](#)

[100GE \(1\)](#)

[Adaptive Flowlet Splicing \(1\)](#)

What is interesting is that BGP pulls ahead as the best protocol choice in creating an IP Fabric. It excels in prefix filtering, traffic engineering, and traffic tagging. BGP is able to match on any attribute or prefix and prune prefixes both outbound and inbound between switches. Traffic engineering is accomplished through standard BGP attributes of Local Preference, MED, AS padding, and other techniques. BGP has extensive traffic tagging abilities with extended communities; each prefix can be associated with multiple communities to convey any sort of technical or business information. The best use case in the world for multi-vendor stability is the Internet; the backbone of the Internet is BGP.

BGP in the data center makes the most sense in the data center when building out an IP Fabric. Maybe it isn't so crazy after all. The benefits include prefix filtering, traffic engineering, tagging, and stability across a set of various vendors.

The biggest decision you need to make when designing an IP Fabric with BGP is to you use eBGP or iBGP. Again, each option has its benefits and drawbacks. One of the key factors is ECMP. It's critical that each leaf support full ECMP going northbound to each spine. The best way to scale a 3-stage Clos network is to increase the number of spines in order to support additional leaves. With the addition of each spine further increases the ECMP requirements of each leaf. The second factor is how many peering sessions do you want to manage in the IP Fabric.

Requirement	iBGP	eBGP
ECMP	Requires BGP AddPath	Requires Multi-AS Pathing
Peering	Requires Route Reflector to mitigate full-mesh	BGP session only between each spine and leaf
Traffic Engineering	Not supported	Extensive

Let's take a closer look at BGP peering. In an iBGP network, each switches is required to have a BGP session to every other switch in the network. This means that every leaf in the network must peer with each other, in addition to each spine. This gets pretty wasteful very fast. The answer is to use a BGP route reflector in the spine of the network. This allows each leaf to become a route reflector client and only have to peer with each spine / BGP route reflector. The downside of a BGP route reflector is that it only reflects the best route. What if there are multiple? Tough luck, you only get one. The answer to support full ECMP with BGP route reflectors is to use another BGP feature called AddPath; this allows each client to receive multiple paths instead of only the best.

From the point of view of a 3-stage Clos or spine and leaf network, eBGP makes the most sense. It supports traffic engineering and doesn't require you configure and maintain a route reflector and AddPath. However this decision becomes a bit more involved in a 5-stage Clos design, but that's a subject for another blog post.

Now that we have decided to use eBGP in our 3-stage Clos, what other things do we need before we can create a final blueprint of what the network will look like? Let's walk through them one by one:

1. BGP autonomous system number assignments
2. IP address base prefix
3. Subnet masks to be used between point-to-point interfaces
4. IP address assignments
5. Loopback assignments

The first step is to assign a BGP ASN per switch; this is a 1:1 ratio of ASNs to switches. Now each spine is able to peer with each leaf via eBGP. The next step is to consider what IP address base prefix to use across the entire network. The answer isn't so simple and it depends on the number of switches, number of links, and the network mask used on the point-to-point links. Let's walk through the options.

Let's assume we have a simple 3-stage Clos network with four spines and 16 leaves; this creates a total of 20 switches. Assuming that each leaf has a full mesh of links to each spine, this creates a total of 64 links. 16 switches times four links (one for each spine) equals 64 point-to-point links. The next step is to think about what network mask to use between each point-to-point link. The most common options are 30-bit and 31-bit. A 3-bit network mask has four IP addresses per subnet. The 31-bit mask has two IP addresses per subnet. With the assumption that each point-to-point link only requires two IP addresses (one per switch), we can conclude that a 31-bit network mask is the most efficient use of IP space. Juniper switches support both the 30-bit and 31-bit network mask, but some other vendors may only support a 30-bit mask. The result is that using a 30-bit mask requires twice the IP space when compared to a 31-bit mask. Generally an IP base prefix of 192.168.0.0/16 is enough in most cases, unless you're building a very large IP Fabric.

The last task is to assign a 32-bit loopback address to each switch in the network. This allows us to quickly test routing connectivity through ping, traceroute, and other tools. BGP must be configured to advertise the loopback address to all of its peers. If a switch is able to communicate to another switching only using loopback addresses, we know that BGP is configured correctly and has reachability.

At this point you have enough information to build the transport mechanism of the IP Fabric, but the last component that's missing is the Layer 3 gateway services that the hosts and other end-points will use. Simply put, each server needs a default gateway address. The good news is that we can limit the gateway services to each leaf. There's no need to span the same Layer 3 gateway address across a set of leaves. This means that Layer 2 is limited to each leaf as well, thus removing any requirements for MC-LAG, STP, or any other Layer 2 protocols to span a bridge domain across a set of switches.

The easiest way to enable Layer 3 gateway services is to create a 26-bit IRB interfaces per leaf. The 26-bits would allow for a maximum of 62 hosts (reserved one for the gateway and the other for broadcast) per switch; the assumption is that each switch has 48 ports, so we have 14 IP addresses left over per leaf. Not bad.

Now that each switch has a unique 26-bit IRB interface, the next step is to advertise these prefixes to the rest of the network. Just like with the loopback addresses, each IRB prefix must also be flooded

About the Author



Amit is a Software Engineer in the Campus and Data Center Business Unit. Off late, he has been working on visualizing distributed systems and automatic anomaly detection.



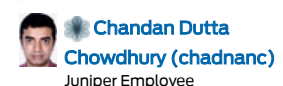
Anil is a Sr. Cloud Technology Architect with Juniper Networks and focuses on Data center/cloud technologies such as VMware, SDN, Analytics and OpenStack etc



Anil Lohiya is a Principal Engineer in the Campus and Data Center Business unit in Juniper Networks. In his current role, he is leading some of the SDN and Network Virtualization initiatives.



Apoorva is a Software Engineer in the Campus and Datacenter Business Unit. In his current role, he is working on the development of SDN and network virtualization features on all Juniper platforms.



throughout the entire network. This ensures that each server in the IP Fabric has full Layer 3 reachability to every other host. The BGP export policy must be configured to advertise the IRB prefix to each BGP neighbor.

A good step to ensure the stability of the IP Fabric is to configure a set of BGP import policies. The policy should only accept loopback addresses and IRB prefixes. There's really no need to accept any other prefixes as they aren't critical for the operation of the IP Fabric. This keeps the table sizes small and allows for faster convergence and updates.

One of the least talked about requirements of an IP Fabric is high availability and convergence. By itself BGP can only support a 7 second interval (per the RFC) and would cause traffic to drop during this window. To speed up convergence during a failure, a faster mechanism is required. A really good tool is Bidirectional Forwarding Detection (BFD). It's a protocol that was specifically designed to be light-weight and detect forwarding errors in the network. Depending on the hardware and software support BFD can be configured as low as 10ms or 20ms. Data center switches typically don't have hardware support for such fast intervals and a more reasonable timer is around 100ms; this still achieves sub-second convergence during a failure.

The other aspect is network maintenance. How do you avoid traffic loss during a software upgrade of the switch? There are two options: traffic steering and in-service software upgrade (ISSU). The first method simply evacuates all traffic from the switch so that a software upgrade doesn't impact production traffic. The drawback is that other switches have to take the burden and responsibility of transporting the traffic until the upgrade is complete. This may or may not be possible depending on the amount of traffic in the network. The next option is a feature called ISSU which allows a switch to transport traffic at the same time it upgrades the software. If ISSU is to be used, it's important that this feature is supported across the entire IP Fabric and not limited to leaves for example.

A really great platform for building IP Fabrics is the Juniper QFX5100 series. It comes in various configurations supporting 40GE and 10GE. Let's check them out:

- QFX5100-24Q: supports 32x40GE interfaces
- QFX5100-48S: supports 48x10GE and 6x40GE interfaces
- QFX5100-96S: supports 96x10GE and 8x40GE interfaces

As you can imagine the QFX5100-24Q makes a great spine switch. It has enough port density to build some very large IP Fabrics. The QFX5100-48S and QFX5100-96S make great leaf switches; they offer create 10GE density and enough 40GE uplinks for 2:1 or 3:1 over-subscription.

For example using (8) QFX5100-24Q switches in the spine and (32) QFX5100-96S switches as a leaf, the total number of ports in the IP Fabric is 3,072x10GE. Not bad for 40 switches, 72U of rack space, and 3W per port.

The Juniper QFX5100 also supports both options for iBGP and eBGP. It's no problem enables BGP route reflection and AddPath in the spine to support ECMP in an iBGP environment. Running eBGP has less requirements and is also no problem for the QFX5100.

In terms of high availability, the QFX5100 supports BFD and provides sub-second convergence times. Most surprisingly the QFX5100 also supports ISSU. You can upgrade the network software while it continues to pass traffic through the IP Fabric. The QFX5100 accomplishes this through virtualization of the control plane. It uses Linux KVM to create virtual machines for Junos. As the ISSU takes place there are two copies of Junos running. The master will continue to operate the control plane while the backup is being upgraded. Once the backup upgrade is complete, the routing engines will switchover and the old backup becomes the new master. Now the other routing engine is upgraded while the new master continues to operate the control plane. Once the process is done, both routing engines will be upgraded without traffic loss.

The QFX5100 takes advantage from all of the control plane features and maturity that comes from the M, T, and MX series over the past 16 years. The BGP implementation in Junos is carrier-class and provides robust traffic engineering, tagging, and policy filtering features that make it a perfect choice for building a rock solid IP Fabric.

Automating the creation and maintenance of an IP Fabric is very malleable in the hands of the QFX5100. The platform supports the execution of Python scripts and has an extensive API that allows you to provision changes and read data from the switch with ease.

In summary the use of BGP in the data center supports and exceeds the requirements of an overlay network in the data center. It easily scales in large environments with 1000s of switches; extensive traffic tagging and engineering capabilities; and is very stable in the face of switches from different vendors. When BGP is implemented with Junos and the QFX5100, the result is an IP Fabric that's carrier-class and is a pleasure to use.

Go implement BGP with the QFX5100 in your next IP Fabric.



Douglas Richard Hanks Jr. is a Director of Product Management with Juniper Networks and focuses on solution architecture. He is certified with Juniper Networks as JNCIE-ENT #213 and JNCIE-SP #875. Douglas' interests are network engineering and architecture for enterprise and service provider technologies. He is the author of the Juniper MX Series book by O'Reilly Media and several Day One books published by Juniper Networks Books. Douglas is also the co-founder of the Bay Area Juniper Users Group (BAJUG). Douglas can be reached on Twitter @douglashanksjr.



Eric Zhaohui Ji (ericji)
Juniper Employee



Harish Pandey (Harish Pandey)
Juniper Employee

I am an Engineer with expertise in Data Packet Forwarding, Software Design & Programming with major domain expertise in QoS (Quality of Services). I have worked across the domains in Data communications field. I love water and am a good swimmer too.



Jai Kumar (Jai Kumar)
Juniper Employee

Jai Kumar is a DE with Juniper Networks. He is one of the key architects of QFabric. He is also an author and architect of OpenFlow support on MX platforms, Open Convergence Framework (OCF) for converged wireless and wired networks, MPLS in data centers and Juniper Cloud Analytics Engine (an Open Analytics Platform). He holds 18 patents on various technologies.



Jonathan Davidson (djonathan)
Juniper Employee

Jonathan Davidson is executive vice president and general manager, Juniper Development and Innovation (JDI). In this role, he is responsible for driving strategy, development, and business growth for Juniper's entire portfolio including routing, switching, and security, as well as for the ongoing evolution of silicon technology and the Junos

[9 Comments \(9 New\)](#) [Permalink](#)



1 Kudo

[« Back to Blog](#) [« Newer Article](#) [Older Article »](#)

Comments

by Matt Hite
on 01-31-2014 05:35 PM

[Options](#)

Great article, Doug! Only thing I think worth mentioning that is missing is discussion of next-hop-self vs. running an IGP for resolution of BGP protocol next-hops in the fabric. It seems to me that you are potentially advocating just running BGP and nothing else, so next-hop-self is relevant to touch upon.

Thanks again for the great topic and article!

[Permalink](#)

0 Kudos

by **Doug Hanks (JNPRdhanks)** ★
on 01-31-2014 11:51 PM

[Options](#)

Hi Matt, Good point. I need to make it obvious that BGP is the new IGP :-). You're absolutely right; another consideration is that when using eBGP you'll need a next-hop self export policy.

[Permalink](#)

0 Kudos

by **scottdware**
on 02-04-2014 02:45 PM

[Options](#)

Great article, Doug!

[Permalink](#)

0 Kudos

by Laurent Vanbever
on 02-05-2014 08:56 AM

[Options](#)

Hi Doug,

Thanks for the article. It uncovered a bit of the mystery behind running BGP in the DC. But I still have a few questions/doubts:

- What do you mean by "Extensive Support" of TE? Can you be a bit more specific on what aspects of TE you support? Are you restricting Traffic Engineering to ECMP only? I guess you have to in a sense since BGP is destination-based, a router will never care about where the traffic is coming. TE policy such as: if the traffic is coming from this pod send on the left, otherwise on the right, then cannot be implemented? I guess you can use policy-based routing then, but it is not quite plain BGP anymore... Also, you can start to have problem such as: routers announce NH X in BGP, but in practice uses X,Y and Z (assuming no add-path on eBGP).

- How would you deal with non-equal load-balancing on multiple paths?

- Like a Route Reflector, eBGP routers picks a single best path for each prefix and advertise that route to their peer. How do you prevent your spine routers from advertising you the same path to reach one prefix (e.g., the one with the lowest peer-id if they all have the same AS-PATH length and LocPref). Since fast convergence in BGP assumes that routers have learned backup paths to fall back on after the failure. How

operating system. Prior to his current position, Davidson was senior vice president and general manager for Juniper's Security, Switching and Solutions Business Unit (S3BU). In this role, he was responsible for leading innovation, growth and product development in data center, campus, branch, and cloud. Davidson joined Juniper in 2010 as vice president, Product Line Management for the Edge and Aggregation Business Unit where he was responsible for the product lifecycle management, strategy, implementation, solutions and go-to-market activity for a range of leading edge routing product families, such as the E, M and MX Series. Before joining Juniper, Davidson had a 15-year career in various leadership positions at Cisco.



Ken Briley (kbriley)
Juniper Employee

Ken Briley is Data Center TME at Juniper Networks focused on Juniper switching product lines. Prior to Juniper Networks, Ken worked at Cumulus Networks as a TME supporting the disaggregation movement and before that he spent 15 years at Cisco Systems working in various roles: Technical Support, Technical Marketing Engineer, Network Consulting Engineer and Product Management. Ken has an MS in Electrical Engineering and is CCIE # 9754.



Krishnalah Gogineni (gogineni)
Juniper Employee



Lakshmi Namboori (Lakshmi Namboori)
Juniper Employee

Lakshmi Namboori is a Senior Product Line Manager with Juniper Networks and focuses on datacenter switching portfolio and fabric architectures. Lead product manager for optical solutions and strategy and Enterprise solutions. She is certified in switching and routing technologies. She is CCIE # 15656. She held various roles in Cisco for 9 years before moving to Juniper. She is passionate about networking industry and her work.



Madhusudan HV (madhuhv)

do you guarantee diversity?

- Unlike MPLS, BGP does not enable you to do TE based on other fields than the destination (e.g., the destination port). Isn't it a problem in some DC that want to perform application-specific TE (e.g., route delay sensitive application on some dedicated paths)?


- You mentioned only positive points, do you see any drawbacks in using BGP in the DC? BGP is famous for the possibility of routing oscillations, deflections (paths advertised is not the path really used), etc. Are you taking any measures to prevent these from happening?

Thanks for the clarification!

-- Laurent

[Permalink](#)

0 Kudos

by  **Doug Hanks (JNPRdhanks)** ★
on 02-07-2014 01:08 PM

[Options](#)

Laurent,

Thanks for the feedback.

When you compare BGP to alternatives such as OSPF and IS-IS, there is much more support for traffic engineering capabilities that are natively part of BGP. For example you can use Local Preference, MED, AS padding, and also change the next-hop. Obviously TE in BGP has its limitations and isn't as extensive as RSVP with EROs, but offers more flexibility than OSPF or IS-IS.

Juniper does support unequal load balancing using the bandwidth extended BGP community.

The only downside is that you really have to pay attention to the BGP import and export policies in the data center. In my next blog post I would like to share a reference architecture with configurations as well. One of the great things about Junos is that we can implement a commit script that double checks critical configuration pieces such as the existence of BGP import filters and specific terms. If the switch detects something is missing because of user error, the new configuration would trigger an error and not commit until the missing configuration is present.

[Permalink](#)

0 Kudos

by Laurent Vanbever
on 02-26-2014 06:27 AM

[Options](#)

Hi Doug,

Looking forward to your next blog post then ;-)! From experience, checking any (non-trivial) property of a BGP configuration is extremely hard (read "NP-hard" as something for which no polynomial time algorithm has been found yet) So, I'd be very interested in knowing how sophisticated your commit scripts are!

Best,
-- Laurent

[Permalink](#)

0 Kudos

by **GuillermoC**
on 11-03-2014 05:37 AM

[Options](#)

Hi Doug,

Great article! It's good to see other options besides those proprietary L2 fabrics, which don't really scale. I'm considering a similar setup for a customer DC and I got here by searching for L3 Leaf-Spine case studies.



Juniper Employee

Madhusudan HV is a Staff Engineer in Juniper Networks. He is passionate about Networking, Virtualization and Cloud technologies. At Juniper he is working on various integrations between Juniper Networks products and cloud solutions like VMware and OpenStack. He is a VMware Certified Professional (VCP).



mlkep ★★ ★
Juniper Employee

Michael Pergament, JNCIE-SP #510, JNCIE-ENT #23, JNCIP-SEC



Pradeep H Krishnamurthy (hkp)
Juniper Employee



Rajesh Patil (Rajeshpatil)
Juniper Employee

Raj is a Sr. Cloud Technology Architect with Juniper Networks and focuses on technologies such as VMware, SDN, and OpenStack etc.



Rakesh Dubey (rdubey)
Juniper Employee

Rakesh Dubey is the engineering head for Campus and Data Center business unit at Juniper Networks. He has been with Juniper for past six years leading multiple switching products.



Rakesh Kumar (rkkumar)
Juniper Employee



Renuke Mendis (Renuke Mendis)
Juniper Employee



rshekhar
Juniper Employee



Sachin Natu (snatu)
Juniper Employee



Sachin Vasudeva (sachin vasudeva)
Juniper Employee



Salman Zahid (szahid)
Juniper Employee

Salman Zahid is Data Center Architect at Juniper Networks .

I just wanted to point out two things:

1 - This architecture works great for enterprise/single-tenant DC. If you need to support several customers and provide L2/L3 isolation, you will need to deploy some sort of overlay technology such as MPLS VPNs.

2 - Server dual homing is almost always required in enterprise DCs. Trying to stay away from stackable/virtual chassis solutions, what would be your approach? MC-LAG or STP with a FHRP?

Thanks!
Guillermo

[Permalink](#)

0 Kudos

by [Felix Li](#)
on 08-18-2015 08:07 PM

[Options](#)

Hi Doug,

In regarding of iBGP and eBGP, it is somehow contrastive in DC EVPN implementation? I think the BGP EVPN solution is using MP-iBGP.

[Permalink](#)

0 Kudos

by [Felix Li](#)
on 08-18-2015 10:40 PM

[Options](#)

Hi Doug,

The MP-iBGP i mentioned actually is only used for EVPN signalling on the overlay network, it is completely irrelevant to your post speaking of underlay network.

[Permalink](#)

0 Kudos

[« Back to Blog](#) [« Newer Article](#) [Older Article »](#)

You must be a registered user to add a comment. If you've already registered, sign in. Otherwise, register and sign in.

[Post a Comment](#)

Salman has been with Juniper for 4 years in this role and been part of the team that has driven double digit growth for switching product lines across all verticals. Prior to Juniper Networks, Salman spent 9 years in Cisco Systems in various roles ranging from Technical Marketing Engineer to Solution Architect for data centers. Salman has an MS in Electrical Engineering and is CCIE # 16406



[Sameer Nanajkar](#)
(sameern)
Juniper Employee



[Sarath Chandra](#)
[Mekala \(Sarath Chandra Mekala\)](#)
Juniper Employee



[Sriram Subramanian](#)
(srirams)
Juniper Employee

Sriram is a Sr. Manager in the Campus and Datacenter Business Unit. He is part of the Network Director team and focuses on technologies such as VMware integration, OpenStack etc.



[Suresh Palguna](#)
[Krishnan \(spkrishnan\)](#)
Juniper Employee



[Yafan An \(yafan\)](#)
Juniper Employee



[Yang Yang \(Yang Yang\)](#)
Juniper Employee

Latest Articles

[The Force is with the Juniper QFX10002 Switch](#)

[QFX10002 Performance and Scalability for the Data ...](#)

[Juniper Content Pack for VMware vRealize Log Insig...](#)

Latest Comments

[anthonyw](#) on: [Juniper Content Pack for VMware vRealize Log Insig...](#)


[speedxs_git](#) on: [SMF: A New Era for Data Centers?](#)

[Anupam Barua \(abarua\)](#) on: [25GbE: There's a New Type of Server Access in the...](#)


[Rajesh Patil \(Rajeshpatil\)](#) on:


Why Physical Network Topology
API is relevant for ...

Silvia on: Network-aware
placement of OpenStack
workloads

 **Sarath Chandra Mekala**
(**Sarath Chandra Mekala**) on:
SRX and OpenStack: Neutron
Firewall Plugin

MikeJ on: Junos Fusion:
Simplicity & Agility at Scale for
Da...

 **bshelton** on: Juniper
QFX10002 Technical Overview

 **tcohick702** on: Automate
using Juniper's True Zero Touch
Provision...

StuckInActive on: Juniper
OCX1100-48SX Technical Deep
Dive