- Home
- About
- Search

Jason Nash's Blog

Just one guy's opinion and hopefully some helpful information.

Feeds:
Posts
Comments

« Follow-up to Microsoft's Linux Kernel Contributions
New Whitepapers by VMware »

vPC (Virtual Port-Channel) and the Nexus Platform

August 10, 2009 by nashwj

We've been working with the Nexus 5Ks a good bit lately and they are just cool, no two ways about it. One really nice feature that the Nexus 5000 and 7000 support is vPC, or virtual port-channels. Take note that support for vPC is new on the 5K and you'll need at least version 4.1(3)N1(1) of NX-OS. If you are below that first grab the latest version from Cisco and update your systems.

Why We Bundle Connections

I'm sure most people reading this are familiar with port-channels, but if not…read on. A port-channel is a way to group two or more interfaces together. Some people call it bundling, Etherchanneling (a specific type of port-channel..but commonly used), and other names. The idea is that you are making multiple connections look like one. There are considerations around using that additional bandwidth that I won't get in to here, but the point is that it's a single bundled connection.

Why do we do this? A couple of reasons. The first is redundancy. This way a single connection going down doesn't cause a disruption. If I port-channel four Gb connections from a server to a switch and lose a single NIC in the system I'm still up and running. We also do it for speed. You can take advantage of higher throughput by bundling connections together. Again, there are special considerations there, but it's an option. Finally, we also do it to reduce complexity of the network. Go a little deeper here. If you have two switches and connect them with two connections not bundled what do you have? You have a loop. That's bad. If a host on one of those switches sends out a broadcast it will just circle those switches. Add in a few more switches and another loop or two and you'll get a multiplier effect on each broadcast which will cause a broadcast storm.

We normally use a protocol called Spanning Tree to find loops in a switched network and get rid of them. Spanning Tree does this by blocking ports so in the example it would block one of those connections between the switches. Now if the other connection failed it would open that blocked connection but you would never get more than a single connection's worth of bandwidth between the switches. That sucks! So we bundle connections. This way we get redundancy and bandwidth but to the switches (and Spanning Tree) they appear as one connection, so no loop and no blocked ports.

What vPC Gives Us

So we know why we bundle connections together and one of those reasons was redundancy. You have a VMware ESX server with four NICs on it. You decide to bundle those four NICs (not always the case..may do 2 and 2, but in this example all 4). One restriction you'll find is that the switch has to know you're doing this. It has to know that all four of those connections terminate to the same place on the other end. That means that you have to terminate those four NICs from the server to the same switch, or at least the same switch stack (since members of a stack talk and act as one). But doesn't that defeat the purpose of what we're doing? The switch failing will take us down. So really we've only protected against a NIC or single port failure.

That's where vPC comes in. The goal behind a Virtual Port Channel was the ability to terminate the connection on separate switches without making everyone support a new protocol or standard, because that would never happen. What good would this new, though cool, protocol be if no one else supported it? vPC allows you to connect to redundant switches without the host (or other switch) knowing they are separate. That host (or switch) thinks that its port-channel is going to a single switch, but it isn't. This way you can take something like a VMware ESX host or non-Nexus switch and connect it to a Nexus 5K or 7K with diverged connections and those end devices will be configured just like a normal port-channel.

Configuring the vPC

Configuration on the Nexus switch for a vPC is actually very simple. In this example we'll be configuring two Nexus 5020 switches to support vPC connections.

The first step is to enable vPC functionality on the switch. If you haven't used a Nexus yet it's a bit different than other IOS based systems. Many features are not enabled by default to save memory, performance, and cut down on running processes. To enable vPC you type the following on both switches:

```
N5K1(config)# feature vpc
```

The next option is to create a vPC domain. A domain is just a number assigned to the switches that will share vPC information. In this example since these two switches will both be terminating the same connections they'll both get the same vPC domain:

```
N5K1(config)# vpc domain 1
```

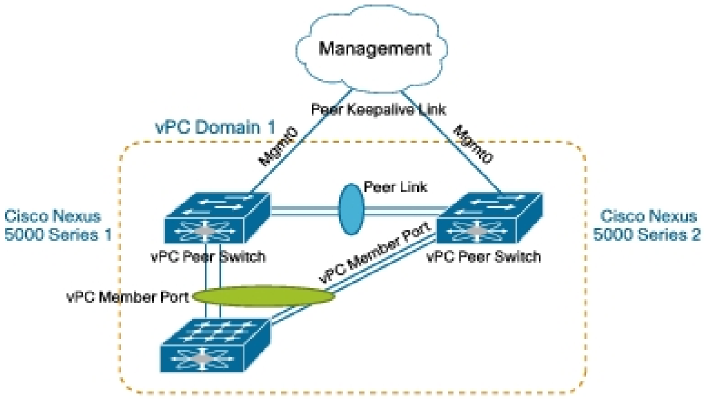
You will then be in the configuration for that vPC domain. There are several options you can set here such as switch priority, but the only required option is the destination peer. This is the IP address of the other switch so the two can talk and exchange information. In this example we'll be using the IPs of mgmt0. On the first switch, N5K1:

```
N5K1(config-vpc-domain)# peer-keepalive destination 10.180.0.103
Note:
 --------:: Management VRF will be used as the default VRF ::--------
```

And on the second switch, N5K3:

```
N5K3(config-vpc-domain)# peer-keepalive destination 10.180.0.101
Note:
 --------:: Management VRF will be used as the default VRF ::--------
```

The note below each command just says that the Management VRF will be used since we're using the management IP address. The next step is to create a "peer link" between the two switches. This is used for high-speed communication and will carry data from one switch to another if the links on one side fail. Example diagram below:



The idea of the peer link is that if the connections to one switch, say the one on the left, go down and traffic gets sent to the switch on the right, that switch can then forward traffic to the left switch. The destination for the hosts traffic may be off of either switch so you need a way to route traffic in the event of a downed connection. For this reason the peer link is usually fast and made up of two or more physical connections, for throughput and redundancy. In our example here we'll be connecting them with two 10Gb connections. This is done by creating a standard port-channel of two interfaces between the switches.

On the first switch:

```
N5K1(config)# interface e2/3-4
N5K1(config-if-range)# switch mode tru
N5K1(config-if-range)# channel-group 100 mode active
N5K1(config-if-range)# interface po100
N5K1(config-if)# vpc peer-link
Please note that spanning tree port type is changed to "network" port type on vPC peer-link.
This will enable spanning tree Bridge Assurance on vPC peer-link provided the STP Bridge Assurance
(which is enabled by default) is not disabled.
```

And on the second switch we'd do the same thing. At this point your vPC domain should be up and operational between the two switches. You can check this with the "show vpc" command:

```
N5K1(config-if)# show vpc
Legend:
 (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id                    : 1
Peer status                      : peer adjacency formed ok
vPC keep-alive status            : peer is alive
Configuration consistency status: success
vPC role                         : primary

vPC Peer-link status
---------------------------------------------------------------------
id   Port   Status Active vlans
--   ----   ------ --------------------------------------------------
1    Po100  up     1,101,400-404,999
```

Now you are ready to put interfaces in to other port-channels spread across the two switches. It's the same as any other port-channel with one exception. Notice the "vpc 10″ configuration below:

```
N5K1(config)# int e1/40
N5K1(config-if)# channel-group 20
N5K1(config-if)# interface po20
N5K1(config-if)# vpc 10
```

The "vpc 10″ line tells the switch that this port-channel is part of a vPC. You configure it the same way on the second switch as well and the switches match up the port-channels in vPC 10. So if you connected two connections to each switch from the ESX box you would put the two connections in channel-group 20 on each side and then the switches would know that both of those are in vPC 10 and therefore bundle all 4 connections together across both switches.

Your ESX server now has redundant connections to redundant switches. To check the status on these connections you use the "show vpc" command again. Below is the output from our example but note that my four interfaces are not yet connected so it shows as down.

```
vPC status
----------------------------------------------------------------------------
id     Port        Status Consistency Reason                     Active vlans
------ ----------- ------ ----------- -------------------------- -----------
10     Po20        down*  failed      Consistency Check Not      -
 Performed
```

From this status screen you can make sure no links have failed. If they have you'll see that it is forwarding traffic for that vPC via the peer link between switches.

Conclusion

There you have it. vPCs are really easy to configure and use. They let you have redundant connections to separate switches from other hosts or downstream switches. Once you do the basic vPC domain and peer-link configuration the rest is not much more than a generic port-channel that we use all the time.

Share this: Share

Like this: Like Be the first to like this post.

Posted in cisco | Tagged 1000v, 10gb, 5000, center, cisco, cna, consolidated, data, nexus, virtualization | 14 Comments

14 Responses

1. *on August 20, 2009 at 7:28 pm* | *Reply* *Jason Spruance*

   Aswesome post. I wasn't aware that the 5Ks supported vPC yet. I'm getting ready to implement this on a pair of 7Ks and just wanted your thought on one thing, the keepalive link. I see you used your management for the keepalive. I assume that you ran them through another switch vs. a private link with a crossover? If so, what are your concerns? Do you worry about STP events or one of the switches going down. I was planning on a single port on each switch and with a crossover between the two and a private/non routed network. I'd put these ports in their own VRF domain. The price per port is so expensive though I really don't want to waste another.

   - *on August 24, 2009 at 1:58 am* | *Reply* *nashwj*

     The 5Ks support vPC as of the latest NX-OS release..about a month ago. My keepalive link is normally through another switch, but keep in mind that it's a L3 connection. I normally do mine through the mgmt0 interface and have yet to see a problem.

2. *on August 24, 2009 at 1:56 am* | *Reply* *A small bug with the Nexus 1000v…. « Jason Nash's Blog*

   […] Nexus and vPC (Virtual Port Channels) […]

3. *on December 8, 2009 at 1:58 am* | *Reply* *marcus*

   Hi Jason,

   For the peer-keepalive, it is using the management vrf, which is showing a single connection into the management cloud.

   What happens if these management connections fail? Do we lose vPC capability at this point?

   Thanks,

   Marcus

4. *on December 25, 2009 at 3:46 am* | *Reply* *MohAmed*

   Hi Jason,
   Great post. Pls. Keep up the good work.
   I'm new to NX & N5K.
   Got 2 Qs:

1) How do you save the config on N5K & N7K…just: wr mem like IOS??
2) If I need to upgrade the image, is it the same way like IOS..download to TFTP & then copy it to the box? or there are special commands?
Thx.

5. *on [January 19, 2010 at 6:30 pm](#) | [Reply](#) Chase Lincoln*

   We currently have a VFC bound to our ports. How can we create a VPC on these ports?

   We are running VMware 4.0; each host has a dual port CNA. Each port is connected to a 5020. We would like to create redundancy of Ethernet and fiber channel.

6. *on [February 9, 2010 at 3:59 pm](#) | [Reply](#) Scott K*

   Juniper already does this with RTG – redundant trunk groups.

   - *on [July 12, 2011 at 6:29 pm](#) | [Reply](#) Ashish*

     RTG is Juniper's equivalent of the Cisco Flex link configuration. vPC is an etherchannel to different chassis or in other words a multi chassis etherchannel

7. *on [September 4, 2010 at 2:52 pm](#) | [Reply](#) [First dive into Storage Networking « Network Janitor](#)*

   […] http://jasonnash.wordpress.com/2009/08/10/vpc-virtual-port-channel-and-the-nexus-platform/ […]

8. *on [September 22, 2010 at 2:40 pm](#) | [Reply](#) Michael*

   Jason, awesome post…quick question…

   Do you know of any reason a vPC cannot be formed to a non-Cisco device? For example…assume you had dual 5020′s Top of Rack…and wanted to form a vPC to a non-Cisco switch in the back of a blade chassis. Would this work?

   - *on [September 27, 2010 at 1:56 am](#) | [Reply](#) nashwj*

     If the blade chassis is uplinked to two switches that support vPC, such as Nexus 5000s then it should work fine. The end device doesn't know it's talking to two different switches. But you can't establish a vPC partnership between a Cisco switch and a non-Cisco switch…the two "up level" switches must be Cisco and support vPC. That make sense?

9. *on [February 7, 2011 at 1:40 pm](#) | [Reply](#) Michoco*

   If we use the management interface for the vPC keepalive-link, how will we manage the two switches? Will they be reachable through one virtual IP address or each device will be reachable through his own management IP idenpendantly?

10. *on June 3, 2011 at 9:47 pm* | *Reply* *Rafael Bianco*

    Great post Jason! Easy reading and awsome content! A+++

    Cheers!

11. *on August 25, 2011 at 12:15 am* | *Reply* *Kollieboi Johnson*

    i am configuring a nexus 7010 with a 4500 switch, and what i do not understand is that, do i have to allow all my vlans on the port channel that is configured for the VPC peek link??? right now the 4500 is waiting for the nexus interface to come up… do i have to add the other vlans on the vpc peer link interface before the link can come up??

    can you allowed/add vlans on a vpc peer link port channel interface while it is active with out is going down?

Comments RSS

Leave a Reply

Enter your comment here...

Fill in your details below or click an icon to log in:

Email (required) (Not published)

Name (required)

Website

□ Notify me of follow-up comments via email.

Post Comment

- RSS 2.0
- Subscribe to Jason Nash's Blog by Email
- twitter Do you follow me?
- I'M A VARROW BLOGGER Click here to see posts from all Varrow Bloggers.

- **Twitter Feed for @nash_j**