Resilient RoCEv4: The Experiment Continues

Comparing RDMA over Ethernet Alternatives Configuration

Overview

Modern datacenter applications demand high throughput of 40Gbps and above with ultra-low latency of < 10 microsecond per hop from the network, with low CPU overhead. Remote Direct Memory Access (RDMA) can meet these needs on Ethernet. There are 2 competing RDMA technologies over Ethernet; iWARP (RDMA over TCP/IP) and RoCEv2 (RDMA over UDP/IP). iWARP uses TCP/IP/Ethernet to provide a plug and play deployment for all solutions of rack, datacenter, cloud, LAN, MAN, WAN, wired and wireless solutions. RoCEv1, which started as IBoE, tried to provide a solution for rack and LAN. Then it was revised to be Routable RoCE (this version was not numbered and lived for about a year), then came along RoCEv2 with more layers being added for IBoUDP/IP/Ethernet to accommodate datacenters. It is notable that none of these versions were interoperable. Now yet another solution is being provided called Resilient RoCE (let's call that RoCEv4 since it's the 4th incarnation) to try to add congestion notification via ECN, but **ECN won't create a lossless network, and performance will suffer greatly when packets are lost**. **So, while it is true that RoCEv2 can operate in presence of loss, it will do so at an extreme penalty such as to be irrelevant as a datacenter or cloud communication protocol**.

On IP-routed datacenter networks, RDMA deployed using RoCEv2 protocol, relies on Priority-based Flow Control (PFC) to try to enable a drop-free network. However, **PFC can lead to poor application performance due to problems like head-of-line blocking and unfairness**. For example, a flow can come into a switch on a higher speed link than the one it goes out, or several flows can come in over two or more links that total more than an output link's bandwidth. These will eventually exhaust any amount of buffering in the switch. However, blocking the sending link will cause all flows over that link to be delayed, even those that are not causing any congestion. This situation is a case of head-of-line blocking (HOL), and can happen more often in core network switches due to the large numbers of flows generally being aggregated.

One RoCEv2 deployment uses unbiased QCN to try to avoid saturation trees that can happen with PFC. Standard QCN puts congestion points in the output buffers of switches which leads to intrinsic unfairness of QCN under typical fan-in scenarios. This is alleviated by installing congestion points at the input buffers of switches. QCN at input buffers cannot always discriminate between culprit and victim flows. To overcome this limitation, a marking scheme of occupancy sampling was proposed. In another effort to try to alleviate PFC problems, DCQCN was introduced. This is an attempt to try to patch together Ethernet and DCTCP into a sideband congestion control scheme (DCQCN). In order to try to optimize DCQCN performance, guidelines are given for tuning switch buffer thresholds, and other protocol parameters. **This is some heavy lifting to try to achieve high performance RDMA over Ethernet when a plug and play solution already exists with iWARP RDMA**.

Yet another RoCEv2 version was introduced to try to make deployments of RoCEv2 easier with Resilient RoCE (RoCEv4), which is RoCEv2 with ECN. Typically, ECN works on a TCP/IP network and

not on a UDP/IP network that RoCEv2 uses. ECN is 2 bits in the IP header that are set by the endpoints after ECN capability has been established and then set by the switch/router during congestion on a queue. Upon receiving a congested marked packet, the TCP receiver endpoint echoes back this congestion using the ECE bit in the TCP header. The sender endpoint receives a TCP segment with ECE bit set and reduces the congestion window as it would for a dropped packet. Then the sender endpoint acknowledges the congestion indication by sending a segment with CWR bit set to the receiver endpoint. Resilient RoCEv4 tries to do this same ECN scheme, but over UDP/IP instead of TCP/IP and therefore has to send the echo and congestion reduction message over UDP/IP instead of TCP/IP. This in itself is an issue as the congestion environment could drop these messages as they are on UDP and not TCP for Resilient RoCEv4. **Yet another RoCE experiment.**

With DCB/PFC, ECN, QCN the network topology, storage nodes, high traffic servers, hot points, etc. have to be mapped out to determine what values should be used for PFC, ETS, QCN & ECN buffer thresholds, etc. Adding more storage, high traffic server or moving/creating hot points will require the user to tune/change those values for the adjusted traffic pattern. Traffic patterns can be very dynamic, depending on the data being accessed, so **the only true plug and play protocol is TCP/IP**.

Ethernet packet loss can be caused by a lossy network (poor hardware) or a highly congested network due to network devices losing packets in between leading to lost data and timeouts. Typical Ethernet non-blocking networks are done by implementing a Clos or Fat Tree.

Clos network - leaf (access), spine (distribution), core: every lower-tier switch is connected to each of the top-tier switches in a full-mesh topology.

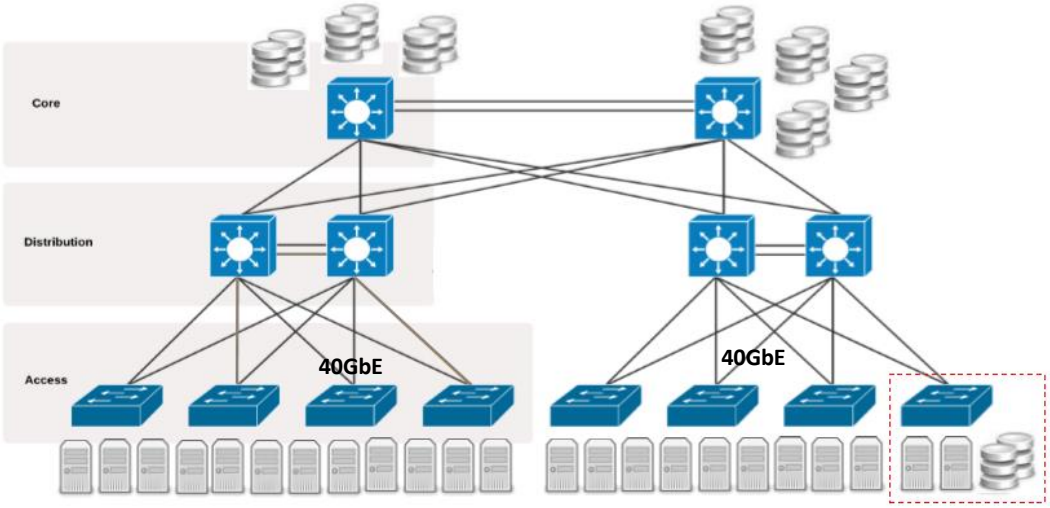

Figure 1 - Clos Network Topology

Fat Tree: branches nearer the top of the hierarchy are "fatter" (thicker) than branches further down the hierarchy. This network is also called CBB (Constant Bi-sectional Bandwidth) network.

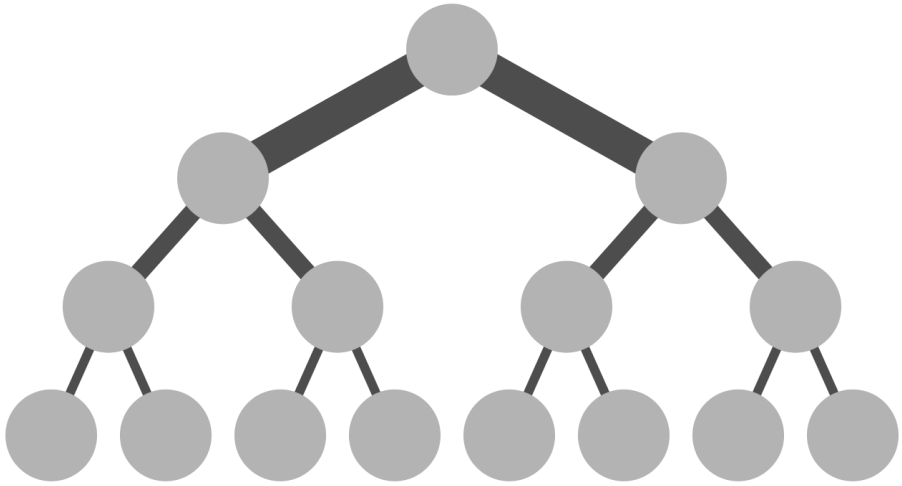

Figure 2 – Fat Tree Network Topology

Yet there are still more network topologies, wired or wireless, below that deal with different packet loss and congestion characteristics. TCP/IP works with all of these in a plug and play fashion.

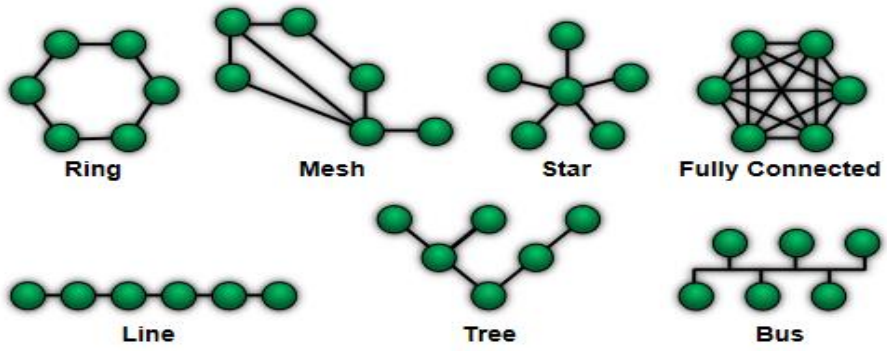

Figure 3 – Network Topologies

Line Point to point: The simplest topology is a permanent link between two endpoints. Switched point-to-point topologies are the basic model of conventional telephony. Each computer or server is connected to the single cable.

Bus: In local area networks where bus topology is used, each machine is connected to a single cable. Each computer or server is connected to the single bus cable through some kind of connector.

Star: In local area networks with a star topology, each network computer is connected to a central hub with a point-to-point connection. All traffic on the network passes through the central hub.

Ring: A network topology that is set up in a circular fashion in which data travels around the ring in one direction and each device on the right acts as a repeater to keep the signal strong as it travels. Each computer incorporates a receiver for the incoming signal and a transmitter to send the data on to the next computer in the ring. The network is dependent on the ability of the signal to travel around the ring.

Mesh: Certain nodes are connected to exactly one other node; but some nodes are connected to two or more other nodes with a point-to-point link. This makes it possible to make use of some of the redundancy of mesh topology that is physically fully connected, without the expense and complexity required for a connection between every node in the network.

Fully Connected Mesh: Every computer has a connection to every other computer on the network. The value of fully meshed networks is proportional to the exponent of the number of computers, assuming that communicating groups of any two endpoints, up to and including all the endpoints.

Tree: Every branch has the same thickness, regardless of their place in the hierarchy—they are all "skinny" (skinny in this context means low-bandwidth).

There is no shortcut to providing end to end reliable, error detection, flow control, congestion control and congestion avoidance protocol, by trying to use DCB/PFC/ETS/QCN/ECN to accomplish what TCP/IP does for you. There is a reason why the major Internet applications such as the World Wide Web, email, remote administration and file transfer all rely on TCP/IP. **It just works**.

RoCEv2 RDMA does not work error free out of the box, so you do have to install the DCB feature on **every host** and configure it together with **every single switch that may be in the path of the traffic**. DCB/PFC is a requirement for RoCEv2 and **the busier the network gets the faster the performance will drop**. You will lose performance right when you actually need performance. Demos may not show RoCEv2 issues, as in a test scenario that has two servers for a total of 4 RoCEv2 ports on the network consisting of a 48 port 10Gbps switch won't show negative results. DCB (Data Center Bridging) and Priority Flow Control are considered a requirement for any kind of RoCEv2 deployment. RDMA with RoCEv2 operates at the Ethernet layer. RoCEv2 is left on its own to deal with Ethernet-level collisions and errors. For that it needs **DCB/PFC otherwise you'll run into performance issues from packet loss due to a ton of retries at the higher network layers**. Other vendors' claim that RoCEv2 is higher performance than iWARP since RoCEv2 does not have the TCP/IP overhead that iWARP does, but this is false as all the TCP/IP processing is offloaded by the iWARP RDMA adapter and all of the TCP/IP and iWARP headers are processed at the same time to ensure the highest performance. What you do get is the robustness of TCP/IP that allows it to run over wired, wireless, any switch, switches that may chop up packets and covers any networking corner cases to run over any network deployment.

In addition:

- RoCEv4 is proprietary and it will take a herculean effort to make it interoperate with adapters from anyone else. Just look at the switch link auto-negotiation standard. It takes at least a year and a full product cycle to get this simple very well-documented feature working among switch and adapter vendors. It will be nearly impossible to get this resilient feature working among adapter vendors in a timeframe that matters.
- RoCEv4 evolution is not done and the science experiment will continue.
- RoCEv4 was introduced July 2016. TCP/IP is 35 years old. iWARP is 9 years old. It took a long time to get them working perfectly and there is really no need to reinvent the wheel. TCP/IP and iWARP just work.
- RoCEv4 is not done by a standards body but by a trade association. Evolutions will be exposed only after products are available from one vendor. It is a vehicle to drive Ethernet-RDMA towards an essentially sole-sourced solution similar to IB.
- RoCEv4 has not been thoroughly tested and RoCEv2 only succeeded to some level only with Microsoft Azure. Even there, a new congestion mechanism DCQCN had to be introduced after months of work, testing and support. It has largely failed in general deployments where time for deployment and support budgets are limited.

RoCEv2 Configuration

Below are some of the steps that you would have to take to get a 1000 node RoCEv2 network up and going. You don't have to do any of this with iWARP, as it is a plug and play RDMA over Ethernet technology.

For each RoCEv2 card, you have to perform the following steps for every card in the network. For a 1,000 node setup, you have to perform these steps 1,000 times. 15 steps for each card, 15,000 steps for 1,000 node setup.

Configuring Priority Flow Control (PFC)

In order to function reliably, RoCEv2 needs the configuration of PFC (Priority Flow Control) on all nodes and also all switches in the flow path.

Configuring PFC on Windows

To configure PFC on the Windows Servers, you need to perform the following steps:

- Clear previous configurations, if applicable
- Enable the Data Center Bridging (DCB) feature on both client and server
- Create a Quality of Service (QoS) policy to tag RoCEv2 traffic on both client and server
- Enable Priority Flow Control (PFC) on a specific priority (the example below uses priority 3)
- Plumb down the DCB settings to the NICs (the example below assumes the NIC is called "Ethernet 4")
- Optionally, you can limit the bandwidth used by the SMB traffic (the example below limits that to 60%)

Here are the cmdlets to perform all the steps above using PowerShell:

```
# Clear previous configurations
Remove-NetQosTrafficClass
Remove-NetQosPolicy -Confirm:$False

# Enable DCB
Install-WindowsFeature Data-Center-Bridging

# Disable the DCBx setting:
Set-NetQosDcbxSetting -Willing 0

# Create QoS policies and tag each type of traffic with the relevant priority
New-NetQosPolicy "SMB" -NetDirectPortMatchCondition 445 -PriorityValue8021Action
3
New-NetQosPolicy "DEFAULT" -Default -PriorityValue8021Action 3
New-NetQosPolicy "TCP" -IPProtocolMatchCondition TCP -PriorityValue8021Action 1
New-NetQosPolicy "UDP" -IPProtocolMatchCondition UDP -PriorityValue8021Action 1

# If VLANs are used, mark the egress traffic with the relevant VlanID:
Set-NetAdapterAdvancedProperty -Name <Network Adapter Name> -RegistryKeyword
"VlanID" -RegistryValue <ID>
Set-NetAdapterAdvancedProperty -Name <Network Adapter Name> -RegistryKeyword
"VlanID" -RegistryValue <ID>

# Enable Priority Flow Control (PFC) on a specific priority. Disable for others
Enable-NetQosFlowControl -Priority 3
Disable-NetQosFlowControl 0,1,2,4,5,6,7

# Enable QoS on the relevant interface
Enable-NetAdapterQos -InterfaceAlias "Ethernet 4"
Enable-NetAdapterQos -InterfaceAlias "Ethernet 5"

# Optionally, limit the bandwidth used by the SMB traffic to 60%
New-NetQoSTrafficClass "SMB" -Priority 3 -Bandwidth 60 -Algorithm ETS
```

For each switch port that the RoCEv2 card ports are plugged in to, you have to perform the following steps. For a 1,000 node setup with each card having 2 ports, you have to perform these for 2,000 ports: 2,000 ports / 48 ports per switch = ~40 switches x 90 steps = 3,600 steps, 2,000 ports / 24 ports per switch = ~80 switches x 90 steps per switch = 7,200 steps and 2 reboots per switch.

Dell switch configuration

These steps show you how to configure two S4810 switches with a Link Aggregation Group (LAG). The switches are interconnected using two of the 40 GbE Quad Small Form-factor Pluggable (QSFP) uplink ports, and the LAG is configured for Dynamic Link Aggregation Control Protocol (LACP).

1.1. Hardware configuration

i. Power on the two switches.
ii. Connect a serial cable to the serial port of the first switch.

iii. Using Putty or another terminal utility, open a serial connection session to the switch.
iv. Open your terminal emulator and configure it to use the serial port (usually COM1 but this may vary depending on your system). Configure serial communications for 9600,N,8,1 and no flow control.
v. Connect the (QSFP) LAG cables between the switches, by connecting port 48 on switch 1 to port 48 on switch2 and port 52 on switch 1 to port 52 on switch 2.

1.2. Delete startup configuration

Note: The following commands will delete all configuration settings and switch will reboot.

```
FTOS>enable
FTOS#delete startup-config
Proceed to delete startup-config [confirm yes/no]yes
FTOS#reload
System configuration has been modified. Save? [yes/no]no
Proceed with reload [confirm yes/no]yes
```

1.3. Configure out of band (OOB) management port

```
FTOS>enable After the startup configuration is deleted, the factory default
Enable mode password is force10.
FTOS>#config
FTOS(conf)#interface ManagementEthernet 0/0
FTOS(conf-if-ma-0/0)#no shutdown
FTOS(conf-if-ma-0/0)#ip address ipaddress mask
FTOS(conf-if-ma-0/0)#exit
```

1.4. Configure route for OOB management port (optional)

```
FTOS(conf)#management route X.Y.Z.0 /24 A.B.C.1
```

Note: X.Y.Z.0 is the network your management system is connecting from and A.B.C.1 is the gateway for the switch. If your management system is on the same subnet as the switch, the previous step may be omitted. The example above assumes a class C subnet mask.

1.5. Configure login credentials

```
FTOS(conf)#username admin privilege 15 password 0 yourpassword
FTOS(conf)#enable password level 15 0 yourpassword
```

1.6. Enable switch ports

Option 1: You can enable ports individually by entering the port number.

```
FTOS#configure
FTOS(conf)#interface tengigabitethernet 0/0
FTOS(conf-if-te-0/0)#switchport
FTOS(conf-if-te-0/0)#no shutdown
```

```
FTOS(conf-if-te-0/0)#exit
FTOS(conf)#exit
```

Option 2: You can enable multiple ports at once using the 'range' parameter.

```
FTOS#configure
FTOS(conf)#interface range tengigabitethernet 0/0 - 47
FTOS(conf -if-range-te-0/0-47)#switchport
FTOS(conf -if-range-te-0/0-47)#no shutdown
FTOS(conf -if-range-te-0/0-47)#exit
FTOS(conf)#exit
```

1.7. Enable Jumbo Frames (if you want to enable jumbo frames)

```
FTOS#configure
FTOS(conf)# interface range tengigabitethernet 0/0 - 47
FTOS(conf -if-range-te-0/0-47)#mtu 12000
```

1.8. Configure flow control

```
FTOS(conf -if-range-te-0/0-47)#flowcontrol rx on tx off
```

1.9. Configure spanning tree on edge ports

```
FTOS(conf-if-range-te-0/0-47)#spanning-tree rstp edge-port
FTOS(conf-if-range-te-0/0-47)#exit
FTOS(conf)#protocol spanning-tree rstp
FTOS(conf-rstp)#no disable
FTOS(conf-rstp)#exit
```

1.10. Configure port channel for LAG

These commands configure the switch interconnect as a LAG.

```
FTOS(conf)#interface Port-channel 1
FTOS(conf-if-po-1)#mtu 12000
FTOS(conf-if-po-1)#switchport
FTOS(conf-if-po-1)#no shutdown
FTOS(conf-if-po-1)#exit
```

1.11. Configure QSFP ports for LAG

These commands assign 40Gb QSFP ports to the Port Channel.

```
FTOS(conf)#interface range fortyGigE 0/48 , fortyGigE 0/52
FTOS(conf-if-range-fo-0/48,fo-0/52)#no ip address
FTOS(conf-if-range-fo-0/48,fo-0/52)#mtu 12000
FTOS(conf-if-range-te-0/48,fo-0/52)#no shutdown
FTOS(conf-if-range-fo-0/48,fo-0/52)#flowcontrol rx on tx off
FTOS(conf-if-range-fo-0/48,fo-0/52)#port-channel-protocol lacp
FTOS(conf-if-range-fo-0/48,fo-0/52-lacp)#port-channel 1 mode active
FTOS(conf-if-range-fo-0/48,fo-0/52-lacp)#exit
FTOS(conf-if-range-fo-0/48,fo-0/52)#exit
FTOS(conf)#exit
```

1.12. Save configuration

```
FTOS#copy running-config startup-config
```

1.13. Configure additional switches

Repeat the commands from above section to configure more switches.

Note: The preceding procedure places all switch ports in the default VLAN. If you prefer to place ports in a non-default VLAN, refer to the documentation for your switch.

Configure Data Center Bridging (DCB)

To enable DCB mode on the switch, use the following commands.

1.1. Disable 802.3x flowcontrol on SFP+ ports

```
FTOS#configure
FTOS(conf)#interface range tengigabitethernet 0/0 - 47
FTOS(conf-if-range-te-0/0-47)#no flowcontrol rx on tx off
FTOS(conf-if-range-te-0/0-47)#exit
```

1.2. Disable 802.3x flowcontrol on QSFP ports

```
FTOS(conf)# interface range fortyGigE 0/48 , fortyGigE 0/52
FTOS(conf-if-range-fo-0/48-52)#no flowcontrol rx on tx off
FTOS(conf-if-range-fo-0/48-52)#exit
```

1.3. Enable DCB and reload

```
FTOS(conf)#dcb enable
FTOS(conf)#exit
FTOS#copy running-config startup-config
FTOS#reload
```

Note: The switch will reboot.

1.4. Create tagged VLAN for all ports and port-channels

```
FTOS#configure
FTOS(conf)#interface vlan vlan-id
```

Note: You must supply a VLAN id. The valid range is 2-4093.

```
FTOS (conf-if-vl-vlan-id*)#no shutdown
FTOS (conf-if-vl-vlan-id*)#tagged tengigabitethernet 0/0-47
FTOS (conf-if-vl-vlan-id*)#tagged port-channel 1
FTOS (conf-if-vl-vlan-id*)#exit
```

1.5. Configure DCB policies

```
FTOS(conf)#dcb-map profile-name
FTOS(conf-dcbmap-profile-name*)#priority-group 0 bandwidth 50 pfc off
FTOS(conf-dcbmap-profile-name*)#priority-group 1 bandwidth 50 pfc on
```

Note: The sum of the bandwidth-percentages must be equal to 100.

```
FTOS(conf-dcbmap-profile-name*)#priority-pgid 0 0 0 0 1 0 0 0
FTOS(conf-dcb-profile-name*)#exit
```

1.6. Apply policies to switch ports

```
FTOS(conf)#interface range ten 0/0 - 47
FTOS(conf-if-range-te-0/0-47)# dcb-map profile-name
FTOS(conf-if-range-te-0/0-47)#exit
FTOS(conf)#interface range fortyGigE 0/48 , fortyGigE 0/52
FTOS(conf-if-range-fo-0/48,fo-0/52)# dcb-map profile-name
FTOS(conf-if-range-fo-0/48,fo-0/52)#exit
FTOS(conf)#exit
```

Note: The sum of the bandwidth-percentages must be equal to 100.

1.7. Save configuration

```
FTOS#copy running-config startup-config
```

Switch specific configuration steps

Each switch brand will have different steps that you have to perform. In this section, there are links to multiple switch-specific commands to correctly configure DCB for the RoCEv2 network.

- **Configuration for switches with DCB support**

The following switches feature DCB with the requirements necessary to support RoCEv2 networks.

- **Force10 S4810**

  Refer to the Dell Force10 S4810 Switch Configuration Guide for instructions related to DCB or non-DCB configuration at http://en.community.dell.com/dell-groups/dtcmedia/m/mediagallery/20220824/download.aspx

- **Force10 S4820T**

  Refer to the Dell Force10 S4820TSwitch Configuration Guide for instructions related to DCB or non-DCB configuration at http://en.community.dell.com/dell-groups/dtcmedia/m/mediagallery/20293376/download.aspx

- **Force10 MXL**

  Refer to the Dell Force10 MXL 10/40 GbE Blade Switch Configuration Guide for instructions related to DCB or non-DCB configuration at

http://en.community.dell.com/dell-groups/dtcmedia/m/mediagallery/20279157/download.aspx

- **PowerConnect 81XX Series**

Refer to the Dell PowerConnect 8100Series Switch Configuration Guide for instructions related to DCB or non-DCB configuration at http://en.community.dell.com/dell-groups/dtcmedia/m/mediagallery/20308559/download.aspx

- **Cisco Nexus 5548UP**

Refer to the Cisco Nexus 5548-UP Switch Configuration Guide for instructions related to DCB or non-DCB configuration at http://en.community.dell.com/dell-groups/dtcmedia/m/mediagallery/20293377/download.aspx

- **Brocade B8000**

Refer to the Brocade B8000 Switch Configuration Guide for instructions related to DCB or non-DCB configuration at http://en.community.dell.com/techcenter/extras/m/white_papers/20439106/download.aspx

- **Configuration for switches that do not support DCB**

Mixing of RoCEv2, FCoE and iSCSI on the same converged fabric is not recommended or supported.

- **PowerConnect 8024, 8024F, and M8024**

The first step in configuring this line of switches is to first verify the firmware version installed is 5.0.0.4 or higher. Upgrade instructions can be found under the corresponding product firmware download pages at http://support.dell.com

Note: If an earlier firmware version is installed, it will not be possible to fully disable DCB on the switch and correct the invalid flow control condition on the arrays and hosts.

To disable DCB, refer to the following guide:
http://en.community.dell.com/techcenter/extras/m/white_papers/20437136/download.aspx

- **Cisco Nexus 5010/5020 and Nexus 7000**

The Cisco Nexus 5010/5020 and Nexus 7000 family enables DCB by default, but it does not support the iSCSI TLV. Furthermore, Cisco does not provide a method for fully disabling DCB on the switch.

Note that if you change the topology, or add or subtract from it, then will have to repeat all steps again. If you choose to simplify the problem by doing a rip and replace and buy only a single kind of switch, you should make sure you always purchase that same sole sourced switch brand.

Summary

Chelsio iWARP:

- Enables incremental, non-disruptive server installs
    - Support the ability to work with any legacy (non-DCB) switch infrastructure.
    - Enable a decoupled server and switch upgrade cycle and a brownfield strategy to enable high performance, low cost enablement.
- Is easy to use and install
    - Have equivalent network switch configuration requirements "as non-RDMA NICs[1]".
    - Is cheaper to deploy ➔ end user can purchase more compute servers for the same investment amount.
    - Does not require gateways or routers to connect to the TCP/IP world.
    - Saves significant CPU cycles
        - Enables cheaper CPU's for equivalent performance.
        - Enables significantly lower datacenter and utilities.
- Utilizes very robust and stable protocols
    - iWARP has been an IETF standard (RFC 5040) for 9 years, TCP/IP has been an IETF standard (RFC 793, 791) for 35 years.
        - No surprises; no fine print; plug and play.
    - Has multi-vendor support.
- Is scalable to wherever the datacenter can scale to
    - Inherits the loss resilience and congestion management from underlying TCP/IP.
- Is very high performance
    - Extremely low latency, high bandwidth and high message rate.

By compiling available real world experience and evidence from the field, it shows RoCE as a difficult to deploy, difficult to operate and difficult to debug technology with gaping holes in reliability and scalability. There is no reason to travel down the rocky road of RoCE, with unpleasant discoveries at every corner, and a specification that is known to be incomplete and still undergoing structural changes. There simply is no reason to incur all the costs and aggravation.

[1] https://blogs.technet.microsoft.com/clausjor/2015/11/23/hardware-options-for-evaluating-storage-spaces-direct-in-technical-preview-4/

Related Links

[Competitive Analysis]
[iWARP RDMA for Microsoft Storage Spaces Direct]
[iWARP Targets Data Center and Cloud Applications]
[RoCE Exposed]
[RoCE Fails to Scale]
[RoCE – Plug and Debug]
[The Case Against iWARP]
[RoCE is Dead, Long Live RoIP?]
[RoCE at a Crossroads]
[RoCE: The Missing Fine Print]
[RoCE: Autopsy of an Experiment]
[A Rocky Road for RoCE]
[RoCE vs iWARP]