



[Home](#) [Archives](#) [Profile](#) [Subscribe](#)

« [Partner Mac Address: 00:00:00:00:00:00](#) | [Main](#)

02/13/2012

FC and FCoE versus iSCSI - “Network-centric” versus “End-Node-centric” provisioning

Which is easier to use: FC, FCoE, or iSCSI?

This is the question we wanted to answer as we set out few months ago to test a series of nine configurations in the lab. Our approach was to determine the number of provisioning steps required when using each protocol to connect to storage and then compare them. We felt if one of the protocols was significantly more labor intensive than another, then perhaps this would shed some light on some of the recent trends we had been noticing. Actually to put a sharper point on it, we wanted to determine why iSCSI seems to be more attractive than FC or FCoE in low end server virtualization environments. A summary of the data we captured as well as the testing process used have been included below, but they were summed up perfectly by Mark Lippitt, one of EMC’s Distinguished Engineers, when he said something along the lines of:

“FC and FCoE provisioning are *Network-centric* and iSCSI provisioning is *End-Node-centric*.”

Before I explain what he meant by this statement, let me start by stating (yet again), we don’t *really* care what protocol you use between your host and storage; we’ll support them all! We do want to make sure that whatever protocol you have chosen to use is the best fit for your environment and it provides you with easy and error free operation. Getting to the bottom of why iSCSI is successful in certain circumstances is important to us because it represents a change.

The testing results are shown the following table and I believe clearly show:

1. Native FC requires the least number of configuration steps when compared with both iSCSI and FCoE.
2. Configuring FCoE requires many more network configuration steps and requires fewer configuration steps on each individual host when compared with iSCSI.



[Erik Smith](#)

[Follow](#)

[ARCHIVES](#)

[September 2022](#)

[August 2020](#)

[April 2019](#)

[December 2017](#)

[November 2017](#)

[July 2017](#)

[April 2017](#)

[September 2016](#)

[July 2016](#)

[February 2016](#)

SEARCH

[CATEGORIES](#)

[Subscribe to this blog's feed](#)

DISCLAIMER

This is not an official EMC blog. The opinions expressed here are my personal opinions. Content published here is not read or approved in advance by EMC and does not necessarily reflect the views and opinions of EMC nor does it constitute any official communication of EMC.

3. Configuring iSCSI requires much more work on each individual host and requires fewer configuration steps in the network when compared with FCoE.

I'll describe why this matters in a bit.

FC vs. FCoE vs. iSCSI provisioning steps...							
OS	Protocol	...on host	...on network		...on storage	Total	
			no TDZ	with TDZ		no TDZ	with TDZ
Windows	FC	2	5	0	7	14	9
	FCoE	2	37	32	7	46	41
	iSCSI	19	7	7	7	33	33
Linux	FC	1	5	0	7	13	8
	FCoE	1	37	32	7	45	40
	iSCSI	14	7	7	7	28	28
VMware	FC	1	5	0	7	13	8
	FCoE	1	37	32	7	45	40
	iSCSI	23	7	7	7	37	37

A few notes about the numbers in the table:

- The steps that were counted and entered into each cell have been included for your reference at the end of this post. See “Configuration steps”
- Since we had to deal with the comparison of “mouse clicks” and “CLI commands”, we tried to break them down into logical steps. With the CLI this was basically one step for every CLI command. For the GUI, we tried to count one step per dialog box unless I had to type in a value and then we counted each value entered as one step. Perfect? No! But we tried to be fair..
- I have not counted steps that are common to all protocols such as attaching cables.
- I've also included information that shows how [TDZ](#) would reduce the total number of provisioning steps.

Getting to the point

As you may have noticed, in terms of the number of provisioning steps, the difference between FC, FCoE and iSCSI is not that great. In fact, based on the numerous “FC is harder to use than iSCSI” complaints I've heard over the past couple of years, I was very surprised to see that FC requires 1/3 the number of configuration steps as iSCSI and I was also surprised to discover how many configuration steps are required to configure FCoE. If you were just to look at these numbers you may come to the conclusion that FC is the clear winner and it will continue to dominate in the SAN, but then how do we explain iSCSI's growth in the server virtualization space. Obviously there are multiple factors at play but again Mark had a hypothesis about this that resonates as true to me “**reducing the number of user interfaces is more important than reducing the total number of configuration steps**”. And this is where the concepts of “Network-centric” and “End-Node-centric” environments come into play.

Network-centric

FC and FCoE are Network-centric. Both protocols rely on fact that the network will control what each end device has access to. This control is centrally managed and requires a special set of FC skills to administer properly. Since these are fairly specialized skills, I think you are more likely to find people that possess them in large organizations where controlling access to a pool of shared storage is a necessity. In addition, since control is centrally managed, the end devices have evolved so that they will try and utilize every device that they can

discover in the SAN. This evolved behavior is one of the things that made the T11 FC-SCM technical report so difficult to write and unfortunately so unimplementable. The bottom line is a Network-centric approach is probably better suited for large organizations that need centralized control of access to storage resources.

End-Node-centric

iSCSI is End-Node-centric. iSCSI relies on the fact that the network will allow communication between the iSCSI Initiator and whatever iSCSI Target the Server Admin points the Initiator at. iSCSI requires no special set of FC SAN Administration skills, so it provides a very low hurdle for entry into the SAN space. I suspect this low hurdle is the reason for uptick in iSCSI's adoption in the low end of the server virtualization space. In addition, since control is managed at each individual end point, the end devices have evolved so that they will only discover what they are told to discover. Maybe this partially explains why the iSCSI iSNS has been adopted by so few. The bottom line is an End-Node-centric approach is probably better suited to smaller organizations that do not need centralized control of access to storage resources.

LUN Masking

In both the Network-centric and End-Node-centric provisioning models, there is typically some kind of LUN Masking implemented on the storage array. I suppose that this could be viewed as a centralized control of storage resource but since it is common across both iSCSI and FC/FCoE, I'm assuming that it would not influence the decision to use one protocol over another.

Conclusion

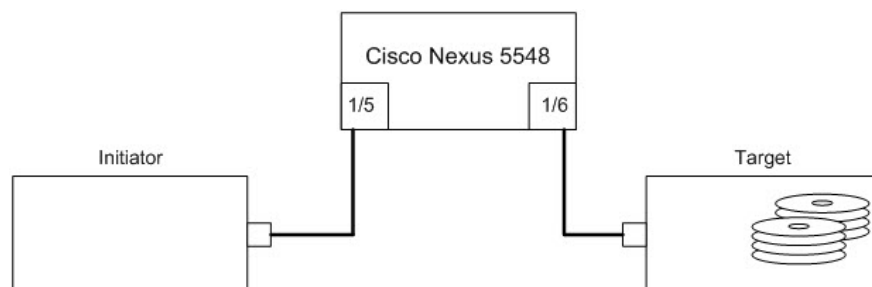
So which one is right for you?

I don't know! Figure it out for yourself!!! :-)

Seriously, it depends on what you need:

- if you require a separate high performance (up to 16GB/s) and fault tolerate SAN, then choose FC or FCoE.
- If you are just entering into the SAN space and want to connect to your external storage without having to learn about FC/FCoE, then iSCSI is probably the best fit for you.
- If you are somewhere in the middle, then it's mostly a matter of personal preference and you should choose what works best in your environment.

Topology tested



Configuration steps

Note: Please keep in mind that the steps listed below are just a description of each task and do not necessarily represent all of the details required to complete that particular task. An example is “Install HBA”, obviously there is a reboot implied in this step. I didn’t include all of the detail because the document that contains all of details was 50+ pages long..

Hosts

Windows:

FC:

1. Install HBA
2. Install Driver

FCoE:

1. Install CNA
2. Install Driver

iSCSI:

1. Locate the adapter that will be used for iSCSI in the “Network Connections” dialog
2. Right click the adapter, select properties, then select “TCP/IPv4” and click properties
3. Select “Use the following IP Address”
4. Enter an IP address
5. Enter a subnet mask
6. Enter a default gateway
7. Click OK and then close.
8. Install the Microsoft iSCSI Initiator software (can be downloaded from microsoft.com if necessary)
9. Open the Microsoft iSCSI Initiator properties dialog by clicking Start / Administrative Tools / iSCSI Initiator.
10. Add an iSCSI Target by clicking the Discovery Tab and then Add Portal...
11. Enter the IP Address of both iSCSI Targets into the “IP address or DNS name:” field as shown below:
12. Click Advanced and the Advanced Settings dialog is displayed
13. In the Local Adapter field, choose Microsoft iSCSI Initiator from the pull down list.
14. In the Source IP field, choose the IP Address of the Adapter that will be used to access this target.
15. Click OK and then OK again.
16. Click on the Targets tab and the iSCSI Targets should be displayed
17. Select the correct target in the list and then click Log on..., the Log On to Target dialog is displayed
18. Ensure that the Automatically restore this connection when the computer starts checkbox is selected.
19. Click OK.

Linux:

FC (FC drivers supported by EMC are inbox):

1. Install HBA

FCoE (FCoE drivers supported by EMC are inbox):

1. Install CNA

iSCSI:

1. Download the appropriate driver
2. Unzip the driver
3. Install the driver and interact with installation script
4. Start the iSCSI driver
5. Verify the iSCSI driver was started
6. Edit the `/etc/iscsi/iscsid.conf` file and verify:
 7. `node.session.iscsi.InitialR2T` is set to yes;
 8. `node.session.iscsi.ImmediateData` is set to No; and
 9. `node.session.timeo.replacement_timeout` is set to 60.
10. Set the run levels of the iSCSI daemon to automatically start at boot and to shut down when the server is brought down using `chkconfig -level 345 iscsid`
11. Assign an IP Address to the NIC that is to be used as the iSCSI initiator.
12. Use service network restart to allow the IP Address change to take effect.
13. Discover the Symmetrix iSCSI target using `iscsiadm -m discovery -t st -p 10.246.54.109`
14. Log into the storage using `iscsiadm -m node -L all`

VMware:

FC:

1. Install HBA

FCoE:

1. Install CNA

iSCSI:

1. launch the vSphere client and login
2. Select a host from the inventory panel.
3. Click the Configuration tab and click Networking.
4. In the Virtual Switch view, click Add Networking.
5. Select VMkernel and click Next.
6. Select Create a virtual switch to create a new vSwitch.
7. Select a NIC to use for iSCSI traffic.
8. Click Next.
9. Enter a network label - A network label is a friendly name that identifies the VMkernel adapter that you are creating, for example, iSCSI.
10. Click Next.
11. Specify the IP settings and click Next.

12. Review the information and click Finish. Verify the vSwitch that was created is shown
13. Select the host from the inventory panel of the vSphere client
14. Click on the configuration tab and then storage adapters.
15. Click on “add” above the storage adapters field and select Software iSCSI Adapter. The iSCSI Software Adapter will be shown in the Storage Adapters section.
16. Select the iSCSI adapter and click properties in the details text area and the iSCSI Initiator Properties dialog is displayed.
17. Click on the Network Configuration tab and then Add.
18. Select the appropriate VMkernel Adapter and then OK.
19. Repeat for the other VMkernel Adapters. When done, the iSCSI Initiator Properties dialog will appear as follows.
20. Click the Static Discovery tab and then click Add
21. Enter the IP Address of the iSCSI Target and its IQN.
22. Click OK.
23. Click OK and a rescan will be performed. Once LUNs have been made available from the storage, perform a re-scan and devices should be visible from ESX.

Network - Not operating system specific:

FC:

Zoning:

1. Create Zone
2. Add WWPN member 1
3. Add WWPN member 2
4. Add zone to configuration
5. Activate configuration

FCoE:

1. Enable FCoE Feature (one time)
2. system qos (The following QOS settings are done once)
3. service-policy type qos input
4. fcoe-default-in-policy
5. service-policy type queuing input
6. fcoe-default-in-policy
7. service-policy type queuing output
8. fcoe-default-out-policy
9. service-policy type network-qos
10. fcoe-default-nq-policy
11. vsan 600 (one time)
12. vlan 600 (one time)
13. fcoe vsan 600 (one time)
14. int vfc 5
15. bind int e1/5

16. switchport trunk allowed vsan 600
17. no shut
18. int vfc 6
19. bind int e1/6
20. switchport trunk allowed vsan 600
21. no shut
22. int e1/5
23. switchport mode trunk
24. switchport trunk allowed vlan 1, 600
25. spanning-tree port type edge trunk
26. int e1/6
27. switchport mode trunk
28. switchport trunk allowed vlan 1, 600
29. spanning-tree port type edge trunk
30. vsan database
31. vsan 600 interface vfc5
32. vsan 600 interface vfc6
33. Create Zone
34. Add WWPN of host
35. Add WWPN of target
36. Add zone to configuration
37. Activate configuration

iSCSI:

1. VLAN 700 (one time)
2. int e1/5
3. switchport access vlan 700
4. no shut
5. int e1/6
6. switchport access vlan 700
7. no shut

Storage - Not protocol specific:

1. Create the storage group
2. Add storage devices to the storage group
3. Create the port group
4. Add Symmetrix interfaces to the port group
5. Create the initiator group
6. Add the initiator to the initiator group
7. Create a masking view

Thanks for reading!

Posted at 09:25 PM | [Permalink](#)



Comments

 You can follow this conversation by subscribing to the [comment feed](#) for this post.



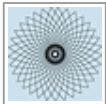
Two about things about iSCSI that might be worth considering.

1. Many more people understand TCP/IP, or think they do, than FibreChannel. FibreChannel is a complex technology to learn and be confident about when compared to a technology you use everyday.
2. The storage industry originally embraced iSCSI, only to turn to FC and have been acting like snobs about iSCSI ever since. It's hard to have a rational discussion about FC over provisioning, unnecessary costs, and pointless over specification.

When you point out that iSCSI has many steps it sort of fails to calculate many of the complexities that went into establishing the FC network. Whereas an Ethernet network is much simpler to build and use, and costs less than an FC network.

But, nice article. Does highlight some of the advantages of FC especially for more certain applications where simplicity is important for less errors on the configuration side.

Posted by: [Etherealmind](#) | [02/14/2012 at 05:12 PM](#)



Hi Greg, thanks for the comment! I agree with your point about the perceived complexity of FC / simplicity of Ethernet. This is one of the reasons that I've been pushing TDZ.

In regards to your second point, I'd place myself firmly in the "recovering FC snob" category of users... I think my prejudice against iSCSI was due mainly to:

1. Customer adoption rates,
2. the bandwidth constraints of 1GbE; and
3. the lack of iSCSI management tools.

However, since:

1. customers are now voting with their pocketbooks in favor of iSCSI,
2. 10GbE is available (if required by the application); and
3. the environments that are using iSCSI don't need the same class of management tools as customers managing larger FC SANs.

clearly I needed to revisit my bias or fall out of touch with reality...

Besides, unless someone defines how to use DCB in an SDN environment, iSCSI is clearly going to be the only choice when there's a need to perform block I/O in these environments.

Regards, Erik

Posted by: [Erik Smith](#) | [02/15/2012 at 08:09 AM](#)



"if you require a separate high performance (up to 16GB/s) and fault tolerate SAN, then choose FC or FCoE"

Sorry but this is simply not true... You can easily build iSCSI SAN using MPIO and 10 GbE connections (as many as you wish). Resulting solution would run circles around any FC SAN implementation @ fraction of cost. You may wish to study NetApp and VMware FS Vs. iSCSI results like the one provided below:

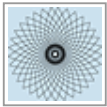
<https://communities.netapp.com/community/netapp-blogs/virtualstorageguy/blog/2010/01/06/new-vmware-and-netapp-protocol-performance-report>

Good luck!

Anton

StarWind Software

Posted by: [Anton Kolomyeytsev](#) | [02/17/2012 at 07:31 AM](#)



Anton, thanks for taking the time to comment, but I disagree. Your assertion of "...Resulting solution would run circles around any FC SAN implementation @ fraction of cost.", has a few problems:

1. I specifically mentioned 16Gb/s FC, can you explain how 10GbE is going to run circles around 16G FC?
2. FC has no overhead from TCP. While this adds a non-trivial requirement to the network (i.e., must be lossless), the end result to the host is more CPU cycles available for doing real work. I agree that modern CPUs have sufficient capacity to do the extra TCP related work and so achieving line rate transfer with TCP based protocols is not a problem. But they are not the same and FC comes out looking slightly better here. The point is I believe most "real world" performance measurements that compare FC/FCoE and TCP based protocols will show that there's practically very little difference between them today..
3. In regards to cost, I specifically said "if you require a separate high performance (up to 16GB/s) and fault tolerate SAN". The two key pieces of information that you need to re-consider are "Separate" and "Fault Tolerant". For those environments that require these attributes, there would be no cost savings since the networking components are, by definition, separate. In addition Fault Tolerant indicates that you'll have at least two of these separate networks and they'll most likely be constructed from "Director Class products". With these requirements in mind, how are you proposing to provide an iSCSI solution at a (significant) fraction of the cost.

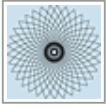
Regards, Erik

Posted by: [Erik Smith](#) | [02/17/2012 at 08:45 AM](#)



Interesting article, the FC set up seems a little over simplified, but its still my clear favourite. I have always found iscsi quite simplistic as being ip based no understanding of fabrics is required. Interesting comments around 10Gb networking! Was there any testing around vmw and NFS?

Posted by: [Stuart](#) | [02/17/2012 at 04:35 PM](#)



Hi Stuart, thanks! In regards to FC, did I miss any configuration steps?

Regards, Erik

Posted by: [Erik Smith](#) | [02/17/2012 at 05:36 PM](#)



This is REALLY good.

I mean I'm sure each group would pick on little things.

If I had to pick on something, I would pick on the lack of NFS (which others already did =)

but overall the two ideas:

1.) Look at each protocol from a "how many physical "things" from start to finish to get it working.

2.) Network Centric vs Node Centric

Are REALLY good items to cover and very interesting topics.

I also think item #2 begs for another look from a different angle.

How much time is really spent on the CNA/HCA/HBA-

>TOR/EOR& | Director vs how much time is spent managing the storage targets themselves.

I think you are on to something here. I really more analysis is needed on "care & feeding" and then a look at how much impact that really has on buying/usage decisions.

Again, nice work guys.

Posted by: [The socialist](#) | [02/29/2012 at 07:04 PM](#)



Thanks Jon, I appreciate the feedback.

The post was intended to compare the provisioning steps for block I/O protocols and as a result NAS wasn't considered.

In addition, I've only started REALLY learning about NAS in the past few weeks. Putting aside the multipathing, performance and security considerations for a moment, I'll say that I've spent some time in the lab working with NAS over the past week or so and I really like it. I was able to configure an NFS server and 4 clients in less than an hour and I started the process without having a clue.

I'm not sure I'll revisit the topic because I still don't feel a file to block provisioning comparison makes sense in the general (non-VMware) case.

The CNA/HBA and Network provisioning steps were all that were required in my topology, but your point about more complex topologies requiring additional work is a good one.

Erik

Posted by: [Erik Smith](#) | [02/29/2012 at 08:09 PM](#)



I figured as much on the NFS, I was just looking for something to pick on.

I wasn't clear, I didn't mean complexity due to topology.

I meant that when I was a SAN Admin I found that I spend VERY little time dealing with hbas/switches/directors. The majority of my time was spent managing the arrays. Symms and Lightnings took most of my time. Managing, upgrading, building luns, optimization, replication, backups etc.

As so for me I found that adding NFS or iSCSI access to my arrays didn't really change my job much. Just meant more customers were using my storage.

And in some ways the ethernet based access was easier. For the FC stuff I was on task all the way down to the HBA. For NFS, I got to send the pissed off user to the Network team =)

Posted by: [The socialist](#) | [03/01/2012 at 05:28 AM](#)



Good article. I'm not sure it is an apples to apples configuration. For example, you have VSANs in FCoE but your FC example assumes no VSANS. If you are using VSANS, you have some additional steps in FC of configuring VSAN membership for ports.

Also I would argue you probably should be creating 2 separate zones in 2 different zonesets (assuming you have separate fabrics).

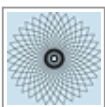
Also, any modern Linux variant and Windows 2008 have the iscsi initiator built-in, so that cuts off like 3 steps in Windows and Linux. Even so, you mention that the install guide for HBA/CNA is a 50 page document so maybe that should be in bold.

Having used all three, I personally think iSCSI is a lot less complex operationally. You eliminate the zoning aspect and thus a whole separate tool to learn and use.

I've also experienced more network issues with iSCSI as compared to FC, but they are generally much easier to troubleshoot with existing tools (like ping, traceroute, etc) versus having to break out one or more FC analyzers and pray you catch the issues. So FC breaks a lot less but when it does it's a lot worse.

I believe the move to iSCSI is probably driven by cost than anything else at this point.

Posted by: [Chris Greer](#) | [03/14/2012 at 09:21 PM](#)



Hi Eric,

I applaud your article and underlines what I've always preached.

FC comes out of the IPI enterprise space with leverage from the things of FDDI. It has been build from the ground up to be able to transfer channel protocols like SCSI, HIPPI, ESCON etc. The fact it also could transport TCP/IP was a bonus but not really spectacular. The IP FCP4 mapping was written within a week. Besides that the

different classes of service provides a very adaptable way of transporting upper layer protocols like SCSI, SBB, IP etc. (I think FC is the only protocol which provides a multicast service with guaranteed delivery.)

There is a huge difference between network centric and channel centric transport.

In my presentations and blogs I also pointed out that network people have a more horizontal view of the world whereas storage people have a vertical view. What I mean by this is the network people in general don't really care about IO profiles, block sizes, packet drops, in-order delivery etc but these are extremely important factors to storage people. That's the reason FC is the only protocol which is able to reliably steer these massive amounts of data back and forth. It is sheer impossible to achieve the same amount of data with the same accuracy and reliability with iSCSI as you can with FC.

iSCSI was invented for server folks who needed a simple means of externally attached storage with minor or no investment to overcome the limitations of DAS.

As for FCoE I think my standpoint is pretty clear. I don't really see the benefits above native FC beside the fact you need some fewer adapters and cables BUT from a support and troubleshooting perspective bolting one protocol on top of another is asking for more troubles and a significant longer recovery time especially when two "sorts" of people are involved (networking and storage).

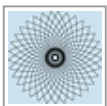
I've seen many examples in my daily job when FCIP is involved where IP networks are used for remote replication. I predict similar issues will happen with FCoE.

Again, Eric, I applaud this post and it proves that from a daily operations and a reliability perspective FC (even more so with TDZ) is a preferred way of managing storage.

Also let me make clear that I'm not saying one protocol or technology is better than the other but they do serve different purposes. Lets keep it that way.

Regards,
Erwin van Londen

Posted by: [Erwin van Londen](#) | [03/15/2012 at 01:01 AM](#)



Hi Chris, thanks for taking the time to provide the detailed feedback.

I tried to keep the configurations as simple as possible and only included steps that are required. Since VSANs are not even available on FC switch implementations other than Cisco, I didn't feel this fell into the required category and left those steps out. Same type of response in regards to the zoning configuration I used.

In regards to iSCSI with Linux and Windows, these steps were captured a while ago and it's possible that additional support for the

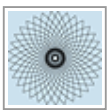
adapters I used has been included since then. However, with Linux in particular, I installed SLES 11 SP1 a few weeks ago and noted that I still had to manually select the iSCSI components in order for them to be installed.

In regards to ease of use, iSCSI definitely has the single management interface advantage, this is a big part of why I've been pushing TDZ on FC/FCoE.

Finally, in regards to troubleshooting IP versus FC/FCoE, I have exactly the opposite opinion! I suspect this has much more to do with background/experience/knowledge rather than anything inherent in the protocol, so I'm willing to agree to disagree on that point.

Regards, Erik

Posted by: [Erik Smith](#) | [03/15/2012 at 08:06 AM](#)



Hi Erwin, I agree with you on the protocol evolution aspects of your comments. Actually, I'll have to take your word on some of those details because they were before my time.

I think it's fair to say that people who work with block I/O tend to be more concerned with "...IO profiles, block sizes, packet drops, in-order delivery etc..." but there are big pockets of interest in this area in the traditional networking space. One such area is WAN optimization but I'm sure someone like Greg Ferro (see comment above) would be able to articulate it much better than I.

Finally, in regards to your concerns about troubleshooting in an FCoE environment. I'll agree that it can be more difficult at first, but once you get a handle on the basics, it's like anything else..

Posted by: [Erik Smith](#) | [03/15/2012 at 08:26 AM](#)



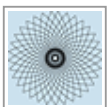
These kind of post are always inspiring and I prefer to read quality content so I happy to find many good point here in the post, writing is simply great, thank you for the post.

Posted by: [WAN optimization](#) | [03/26/2012 at 06:39 AM](#)



Need to add "install & configure multipathing software" to the FC host deployment list

Posted by: [Skywalker](#) | [05/28/2015 at 06:39 PM](#)



Hi Luke?

In any case, with regards to multipathing... Although it's not very realistic, all three of the protocols can work without multipathing, so I didn't include those steps for any of the protocols.

Posted by: [Erik Smith](#) | [05/28/2015 at 06:59 PM](#)

The comments to this entry are closed.

[Brass Tacks](#) [Powered by TypePad](#)