



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

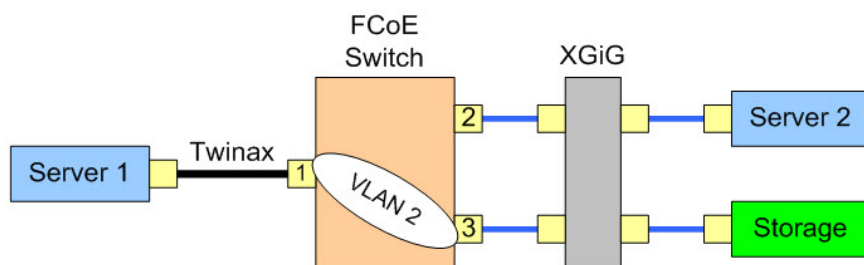
[Main](#)

09/05/2012

A problem with I/O Consolidation... and Network Virtualization?

Some number of weeks ago I helped troubleshoot an FCoE problem involving a flaky twinax cable that was causing excessive bit errors. The symptom being observed was very strange at first glance; CRC errors were being logged on interfaces that should have been idle. At first we thought the switch was simply incrementing the wrong CRC counter, but as we would soon discover, the root cause was much more interesting. In addition, after reviewing the NVGRE, VXLAN and STT encapsulation proposals, I believe a similar sort of problem may exist with all three proposals. If this turns out to be the case, I think now would be the time to address it.

The problematic FCoE topology was similar to the following:



As shown in the above diagram, the defective twinax cable was used to connect Server 1 to interface 1 on the FCoE switch. Since we were performing FCoE I/O between Server 1 and the Storage, and we knew the twinax cable was causing bit errors, we expected to see:

1. the “receive CRC error” counter increment on FCoE switch interface 1,
2. the “transmit CRC error” counter increment on FCoE switch interface 3; and
3. the “receive CRC error” counter increment on the storage array.

Unfortunately, this wasn’t the case and instead we observed:



[Erik Smith](#)

[Follow](#)

[ARCHIVES](#)

SEARCH

[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.

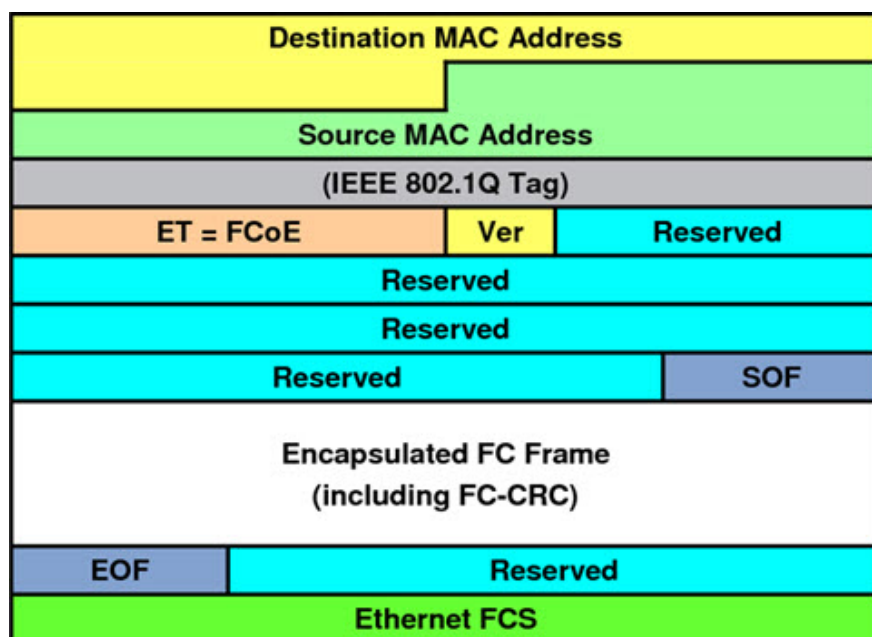
1. the expected behavior from above; AND
2. the “transmit CRC error” counter occasionally increment on FCoE switch interface 2! This was completely unexpected because only Server 1 was generating any kind of significant I/O and it was all FCoE based. Therefore this traffic should only have been visible to the storage array and Server 2 should not have been receiving any frames.

Upon reviewing an xgig trace between FCoE switch interface 3 and the storage port, I noticed not only a few CRC errors, but I also noticed a number of ABTS and bad SCSI status frames. As I expected, these were related to an exchange or I/O that was impacted by a corrupted frame. However, in at least one of the cases, the bad SCSI status was being returned by the Storage port when it detected a missing frame. The significance of this wouldn't be clear until later.

Finally we inserted the analyzer between between Server 2 and FCoE Switch interface 2. We set it to trigger on bad CRC/FCS and when it triggered I was very surprised to see an FCoE data frame from Server 1 that was addressed to D_ID of the storage array! I mean think about this for a second, a frame containing data sent from a host to the storage array was being transmitted to another host on a completely different VLAN!

The reason this happened has to do with one of the basic mechanisms of Ethernet switching known as the unicast flood. In a previous post, I described the process of forwarding in an FCoE environment, so rather than rehash that here, I'll direct those of you who want more detail to [that post](#). In this case, the FCoE frame was being flooded because the corruption occurred in Ethertype field.

For additional detail, refer to the diagram of a typical FCoE Frame below. Note that with FCoE, the frame typically starts with the Ethernet DA and SA and then is followed by the 802.1Q VLAN Tag.



In general, when an Ethernet frame is received with a VLAN tag, if the DA is known, the frame will be forwarded based upon a MAC Address table that reflects the switches understanding of the correct route to the destination address. If the DA is unknown, the frame will be flooded on that VLAN and

eventually the MAC Address table will be updated to allow future frames to be forwarded to the appropriate destination. Taking this a step further, if the frame is a FIP or FCoE frame, additional forwarding rules will be applied that are (hopefully) consistent with FC-BB-5 annex C and D. These additional rules are described in another post that has to do with [FIP Snooping Bridges](#).

Three additional items that I have to mention before getting to the point:

1. the 802.1Q tag is actually an Ethertype (0x8100) that allows the switch to interpret the rest of the frame (e.g., VLAN ID) and then actual Ethertype of the frame (e.g., FCoE or 0x8906).
2. The FCoE capable switches that I am aware of do not learn Fabric Provided MAC Addresses (FPMAs) in the same way that normal MAC Addresses are learned. Part of the reason has to do with the special handling requirements outlined above.
3. Due to performance concerns, the FCoE capable switches that I am aware of all support cut-through routing of FCoE frames, this means that the frame header can be transmitted before the FCS is received. As a result, frames with bad FCS are forwarded..

Getting back to the trace between FCoE Switch interface 2 and server 2; in the trace I captured, I noticed that the frame with a bad FCS had a VLAN Tag that was incorrect (let's say 8101, to be honest I can't remember the exact value). As a result, instead of the switch detecting an Ethertype of 0x8100 and handling it appropriately, it detected a different Ethertype (0x8101) and did what any Ethernet switch would do when a frame is received on the default VLAN (remember, no VLAN ID since the tag was corrupted) and the DA is unknown (due to the FPMA learning restriction) and this is to flood it on the default VLAN.

To make matters more interesting, in order for FIP to work properly, a default VLAN must be defined and typically all FCoE devices will use the same default VLAN for FIP VLAN discovery. Note, this isn't a requirement of the protocol, but it is easiest to configure it this way and it works perfectly fine as long as there are no bit errors.

The bottom line is this; if you are using FCoE and you are experiencing bit errors, there is a chance (albeit a small one) that an FCoE data frame may be flooded on the default VLAN and this VLAN will probably be used by other FCoE capable devices. As a result, ensuring that there are no bit errors logging in your FCoE environment is a VERY good idea.

Does this mean you should avoid FCoE? Of course not! The same problem can occur in any Ethernet network (as long as cut-through routing is used). I'll go out on a limb and say that this problem could occur any time protocol based isolation is used to partition a physical link. Note that I explicitly said PROTOCOL based, I am not asserting that this could happen with DWDM or TDM based approaches.

The implications to Network Virtualization

Network Virtualization supports multi-tenancy and the current approaches use a protocol based partitioning method. Essentially, all of the proposed tunneling encapsulations (e.g., VXLAN, NVGRE and STT) make use of a "Tenant ID" to signal to the tunnel end points which tenant the frame belongs to. Although I have not had the opportunity to force a similar type of problem to occur with any of the proposed NV approaches, I think it may be possible for a bit in the

Tenant ID field to get flipped and for a data frame to be delivered to the wrong Tenant. In fairness, this problem would seem to be much less likely to happen with NV and could probably be prevented from occurring by the NV Control Plane. As a result, I'll avoid speculating much more on what could happen until I have more data. That having been said, I'm raising the issue now so that it can be investigated by those of us who are interested and a solution can be built in to the protocols (if needed).

Thanks for reading!

Posted at 01:33 PM | [Permalink](#)



Comments

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



I would love to know which switch you were using. In the good old days a switch (OK, maybe it was called a bridge then) would drop a frame with a CRC error, not forward it.

There should be no transmit CRC errors in a store-and-forward switch. Cut-through switches are a different story. They were a bad idea in 1990's and they still are.

Posted by: [Ivan Pepelnjak](#) | [09/05/2012 at 02:43 PM](#)



Hi Eric,

Cut-through switching is extremely uncommon in data networking, or at least it was until recently. Now that you mentioned it, I'm very interested in checking with the modern Ethernet "kings of low latency" to see if they are using cut-through.

IMO, it is bad idea; I'm very much with Ivan on this one.

Posted by: [Dmitri Kalintsev](#) | [09/05/2012 at 11:20 PM](#)



Great post! It's a very interesting trouble and a good opportunity to remember what happened "under the hood".

Posted by: [Waldemar Pera](#) | [09/06/2012 at 07:01 AM](#)



Hi Ivan, in this case I'm hoping to avoid naming a specific FCoE switch vendor or product. That having been said, I believe that Brocade, Cisco and Juniper FCoE switches will all exhibit this symptom. I'm only naming these three because they are the only FCoE switches that I have direct experience with.

In regards to NV and whether or not the switches that will be used in the underlay will use cut-through or store and forward. I've noticed that most of the 10GbE ToR switches I've been looking at recently are all claiming latencies of less than 1 usec. To me this indicates

they must be using cut-through because it will take at least 1.2 usec to receive 1500 Bytes when using 10GbE (12375 encoded bits / signaling rate of 10.3125 Gbps)

Posted by: [Erik](#) | [09/06/2012 at 08:31 AM](#)



Hi Dkalintsev, I think it's becoming much more common especially since switch vendors appear to be using the latency metric as a way to differentiate themselves from their competition.

Posted by: [Erik](#) | [09/06/2012 at 08:34 AM](#)



Thanks Waldemar!

Posted by: [Erik](#) | [09/06/2012 at 08:35 AM](#)



Erik,

It seems that part of the reason for the bad behavior is because Ethernet relies on frame flooding to determine an unknown address.

In the case of Fiber Channel, since devices login and register with a name service, finding an unknown address does not rely on frame flooding.

As you point out, cut through switching becomes mandatory as link rates go up so the original store and forward architecture isn't feasible and the CRC can only be acted upon by the edge device at the destination.

A tentative conclusion open for debate follows:

1. Finding an unknown destination should rely on a name service and device registration when cut and forward switching is prevalent.

BTW, very useful post with great detail.

Posted by: Brook Reams | [09/06/2012 at 10:25 AM](#)



Hi Brook, thanks!

Not using cut-through with OVNs would potentially solve the problem but could create others.

Another idea would be to add a checksum to the "TNID" field and specify that the Tunnel End Points validate and then discard if the validation fails. Not sure how realistic this is though..

Posted by: [Erik](#) | [09/06/2012 at 03:07 PM](#)



Hi Eric,

I followed up with one of my vendor friends, who confirmed that their low latency data-only gear also operates in cut-through mode. His recommendation was that the Layer 1 errors are to be closely and pro-actively monitored, and identified bad talkers acted upon as soon as possible. Some network devices can trigger port actions (such as shut) based on error thresholds, based for e.g. on CRCs, but

whether to use this or not would depend on the particular environment, of course.

I agree with the recommendation to monitor and quick act on errors; and it should work equally well with NV.

Cheers,

-- Dmitri

Posted by: [Dmitri Kalintsev](#) | [09/06/2012 at 06:39 PM](#)



This is common problem with any cut thru switching. Nexus 5500 introduces a feature called stomping in which switch does bit flapping in a CRC portion of the ethernet frame. 9000 byte frame with bit flaps/ bad CRC arrives to the switch port 1. After first 64 bytes received switch decides to send to port 2 and port starts receiving the full frame and putting on wire.

Now last 4 bytes of frame received on ingress port and switch finds that oh this is bad frame but you cannot pull bits from wire. So switch does bit flapping on the CRC and continues to forward the frame however, it accounts on interface counter RX CRC on ingress port and output error on egress port.

If next switch is stored and forward this bad CRC frame will be dropped on ingress port.

If next switch is cut thru that switch does the same process.

All above does not prevent switch from making wrong switching decision because of bit flap in DST_MAC or ether type. But bad CRC frames are identified by next switch.

Posted by: [Krunal](#) | [09/06/2012 at 10:44 PM](#)



Hi Krunal, I agree with what you are saying and have observed this behavior in the lab as well.

What do you think about adding a checksum that will cover the VNID field?

Posted by: [Erik](#) | [09/07/2012 at 08:46 AM](#)



Dmitri, thanks for the confirmation. Perhaps the answer is as simple as monitor for bit errors or set an aggressive policy that would force an interface offline after exceeding a certain bit error rate..

That having been said, I was hoping for something a bit more automated and was thinking that by adding some kind of checksum to the encaps header, that the networking gear could take care these situations automatically (like store and forward architectures do).

Posted by: [Erik](#) | [09/07/2012 at 08:53 AM](#)



I dnt know but as you might be already be aware checksum also cannot detect full error. We have checksum filed for each IP header

and Not all routers calculates checksum for all IP packets passed thru it. This is one reason IPv6 header is not protected by a checksum.

Posted by: Krunal | [09/07/2012 at 09:20 AM](#)



Eric,

I guess the "correct" answer is, as always, "it depends". In some environments it may be better to drop the defective frames and try to carry on, while in others it could make much better sense to cut off the offender as soon as possible.

My "gut feel" preference is clearly on the side of isolating and fixing the root cause as soon as possible (because even if you carry on, you still know that you'll need an outage to fix the problematic hardware), but on the other hand if bad frames were dropped (meaning their impact is isolated to the directly affected system/s), and their volume wasn't very high, there would be more operational flexibility in regards to when to do the repairs in a more planned manner.

As I said, "it depends". :)

Regarding the idea of introducing an additional CRC for headers - I'm quite sceptical, as in addition to changes to networking standards it would almost certainly require changes to the networking hardware.

Cheers,

-- Dmitri

Posted by: [Dmitri Kalintsev](#) | [09/07/2012 at 07:45 PM](#)



I was able to find good information from your articles.

Posted by: [dort](#) | [09/19/2012 at 01:02 AM](#)



Thanks Dort.

Posted by: [Erik](#) | [09/25/2012 at 10:31 AM](#)



Erik,

This was a great post and I learned a lot. Thanks a lot!

Posted by: Will Hogan | [11/08/2012 at 08:33 AM](#)



Hi Will, I'm glad you found it useful! Thanks for taking the time to comment.

Erik

Posted by: [Erik](#) | [11/09/2012 at 08:06 AM](#)

Hi Erik,



Good post and great find. It's always a PITA to find these kind of issues without an analyzer.

In FC we don't have "VLAN/VSAN flooding". If an in-frame error is encountered whereby the CRC is incorrect and cut-through switching is used (as all FC switches do these days) we simply terminate the frame with EOFti where we invalidate the frame. We do not drop the frame on the next ingress port but we let the recipient handle this. This does prevent additional overhead on the all the ASIC's and is the fastest way of handling these kind of errors.

As for discussing whether cut-through routing/switching is better or worse then store&forward is up for grabs. There are pros&cons to both.

Cheers,
Erwin

Posted by: [Erwin van Londen](#) | [11/19/2012 at 12:56 AM](#)



Hi Erwin, thanks! I agree with your description of how this is handled with FC.

Posted by: [Erik Smith](#) | [11/24/2012 at 06:45 PM](#)



Your post was very informative thank you for that

Posted by: [Bill Hubert](#) | [03/06/2013 at 01:20 AM](#)



Thanks Bill!

Posted by: [Erik](#) | [03/29/2013 at 08:17 AM](#)

The comments to this entry are closed.

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