



FCoE: Fabric Crosstalk Update

David L. Black

December 2007

Background

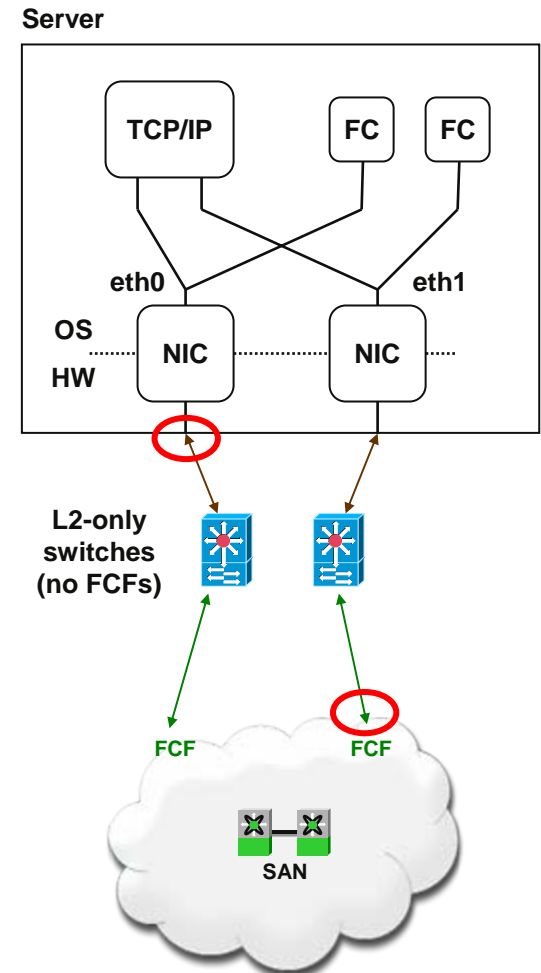
- Action item from July meeting (Colorado Springs)
 - AI-2: EMC/HP/IBM group to provide concrete forwarding loop configuration examples
- FCoE and Layer 2 (L2) Ethernet-only switches (bridges)
 - Native FC links are always private (2 ends)
 - Ethernet-only L2 switches: FCoE links can be shared (3+ ends)
- October (07-558v1): Shared link issues (three scenarios).
 1. Cross Connect: Traffic swap between two VN-VF port sessions
 2. Association: VN port logs into wrong VF port
 3. Forwarding Loop: Frames live forever
- Also – Scenario 4: Rogue Host (New - see 07-546v1)

Crosstalk Scenario Status

1. Cross Connect: Traffic swap between two VN_Port-VF_Port sessions
 - Risk: Data Corruption
 - Solution Approach: ACLs, agreed to in principle
 - ACL structures differ between Mapped Addresses and Server Provided Addresses
2. Association: VN_Port logs into wrong VF_Port
 - Risk: Topology error, (hidden) loss of multipath redundancy
 - Solution Approach: Pre-FLOGI discovery protocol, agreed to in principle
3. Forwarding Loop: Frames live forever
 - Risk: Network and/or fabric collapse
 - Solution Approach: VN_Port MAC address range prefix (Mapped only)
 - Mapped Address Status: Ok in principle
 - Server Provided Addresses: Important progress, still Incomplete
4. Rogue Host
 - New scenario

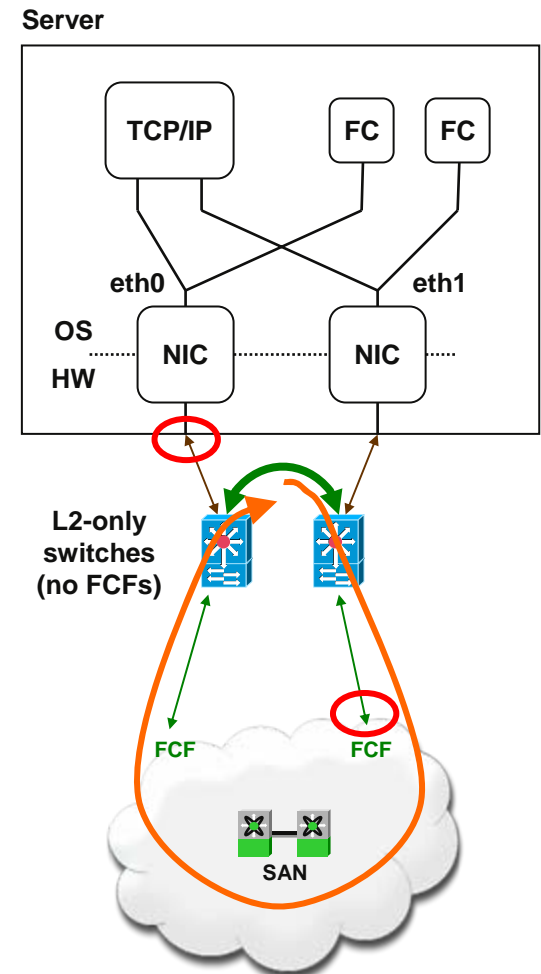
Reminder: Fabric Crosstalk Problem Setup – Scenario 3

- Edge switches: L2, Ethernet-only
 - No L3 (IP), no FCF
 - FCFs are at edges of FC cloud
- One FC fabric (cloud) this time
- VLAN 1 (left and right instances) for FC
- Get the FCFs involved
 - Suppose left NIC and right FCF have same FCoE MAC
 - Left FCF believes right FCF's MAC is a VN_Port



Reminder: Fabric Crosstalk Problem – Scenario 3

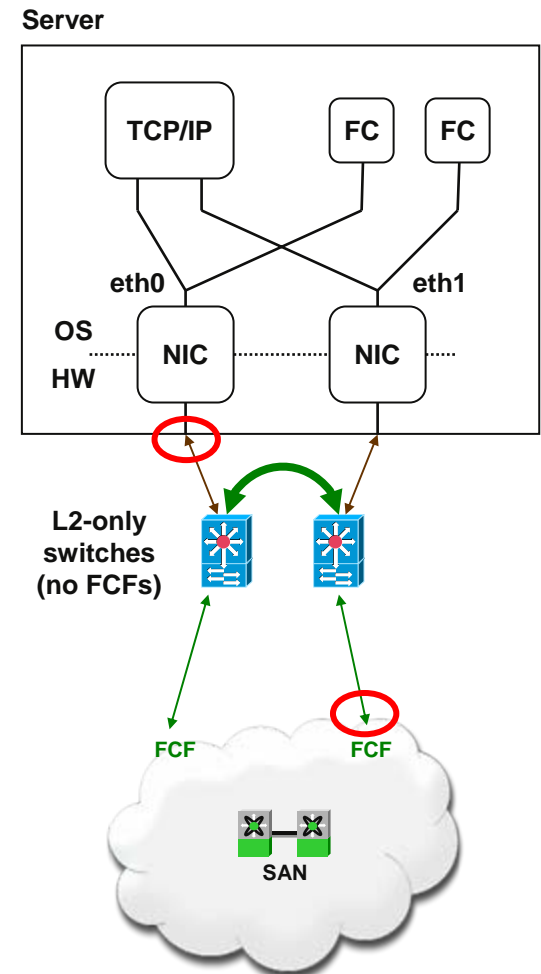
- **Similar Mistake: Cross connect edge switches with VLAN 1 link**
- Send FC frames to left NIC's FCID
 - Left FCF thinks the right FCF's MAC is for a VN_Port
- Nothing breaks immediately, but
 - Suppose left switch forgets where that MAC is located
 - Right FCF may send frame that "helps" that switch forget
- Left NIC frames now loop forever:
 - Sent to left FCF for NIC VN_Port
 - Left FCF uses VN_Port's MAC
 - Frame arrives at right FCF
 - Destination FCID hasn't changed
 - Fabric forwards to left FCF
 - Lather, rinse, repeat ...



Scenario 3: Good News (John Hufferd, 07-630v0)

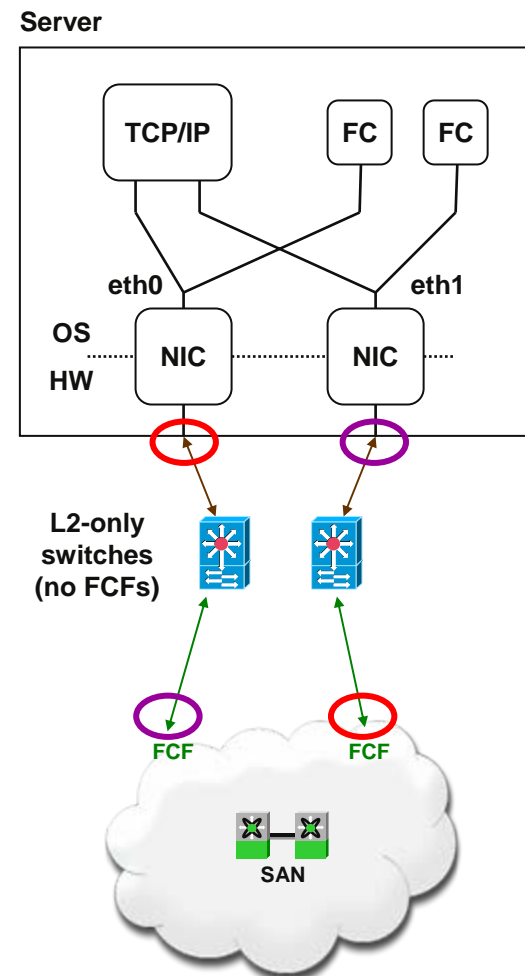
- **Similar Mistake: Cross connect edge switches with VLAN 1 link**
- Send FC frames to left NIC's FCID
 - Left FCF thinks the right FCF's MAC is for a VN_Port
- Nothing breaks immediately, but
 - Suppose left switch forgets where that MAC is located
 - Right FCF may send frame that "helps" that switch forget

- Left NIC frames **can be stopped:**
 - Sent to left FCF for NIC VN_Port
 - Left FCF uses VN_Port's MAC
 - Frame arrives at right FCF
 - Destination FCID hasn't changed
 - **Source MAC is wrong (Check It!!)**
 - **FCF discards frame**



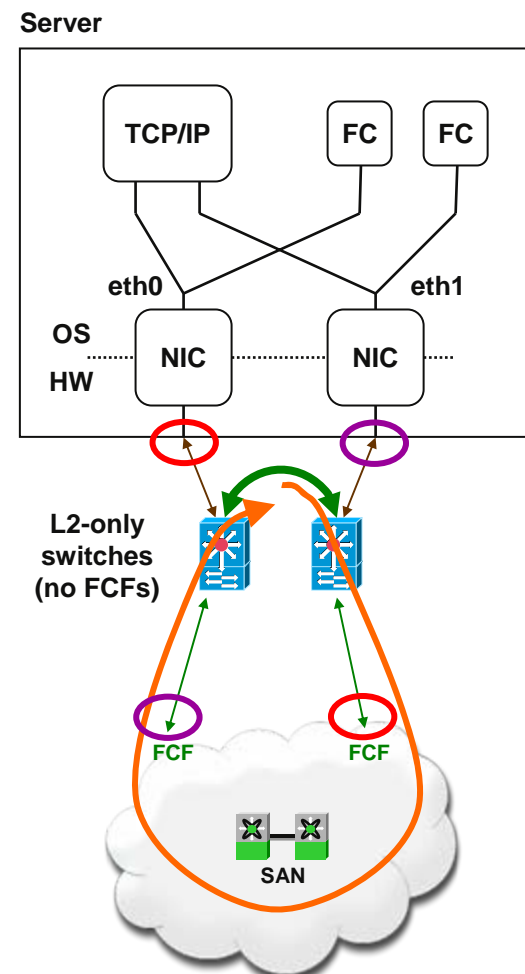
New: Scenario 3a Setup

- Edge switches: L2, Ethernet-only
 - No L3 (IP), no FCF
 - FCFs are at edges of FC cloud
- One FC fabric (cloud) this time
- VLAN 1 (left and right instances) for FC
- Get the FCFs involved
 - Suppose left NIC and right FCF have same FCoE MAC
 - Left FCF believes right FCF's MAC is a VN_Port
 - **And the same for the other two MACs (left FCF, right NIC)**



Scenario 3a – Another Forwarding Loop

- **Similar Mistake: Cross connect edge switches with VLAN 1 link**
- Send FC frames to left NIC's FCID
 - Left FCF thinks the right FCF's MAC is for a VN_Port
- Nothing breaks immediately, but
 - Suppose left switch forgets where that MAC is located
 - Right FCF may send frame that "helps" that switch forget
- Left NIC frames now loop forever:
 - Sent to left FCF for NIC VN_Port
 - Left FCF uses VN_Port's MAC
 - Frame arrives at right FCF
 - Destination FCID hasn't changed
 - **Source MAC matches right NIC**
 - **Fabric forwards to left FCF**
 - Lather, rinse, repeat ...



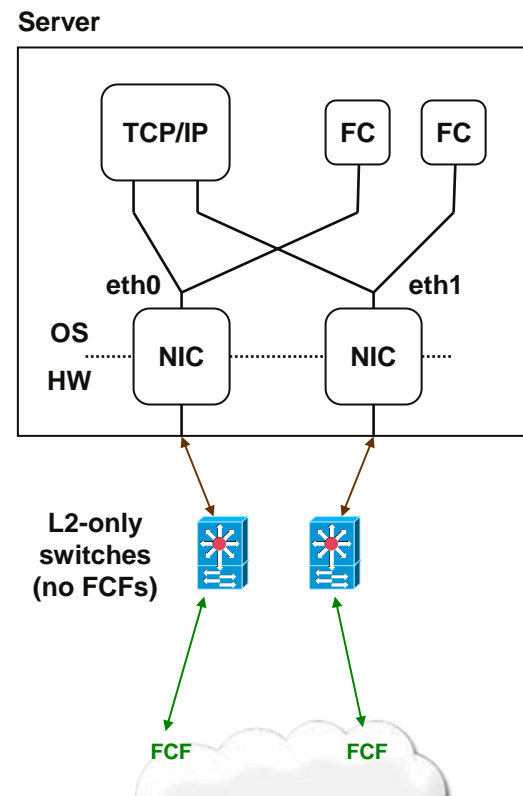
Scenario 3a Considerations

- Requires two FCF-ENode (switch-NIC) address duplications
 - The FCF has to check the source MAC (Also see 07-630v0)
 - Duplications must span FCF and ENode (switch and NIC)
 - Duplications must offset each other across cabling mistake
- Simple Idea: Have FCFs always use burnt-in MACs
 - Then FCF addresses won't ever be duplicated
- **Unfortunate FACT: T11 has limited control over MAC usage**
 - Existing hypervisor MAC usage must be dealt with **(installed base)**
 - Software implementation of FCoE is coming **(fact of life)**
 - Virtual switches: may see software-based FCoE switch MACs
 - Hypervisors are proliferating
 - FCF (switch) as hypervisor guest? Definitely possible.
 - Same vendor can provide FCFs and ENodes

Crosstalk Scenario 3/3a and Mapped MAC Addresses

- Edge switches: L2, Ethernet-only
 - No L3 (IP), no FCF
 - FCFs are at edges of FC cloud
- One FC fabric (cloud) this time
- VLAN 1 (left and right instances) for FC
- ~~Get the FCFs involved~~
 - ~~Suppose left NIC and right FCF have same FCoE MAC~~
 - ~~Left FCF believes right FCF's MAC is a VN_Port~~

Mapped MAC Addresses: FCFs are prohibited from using a MAC that has the VN_Port prefix



Still need to see specification of VN_Port prefix format and management details

Reminder: Scenario 3: Diagnosis and Solution Approach

3c will probably be found here, and there's also FC-IFR ...

Server Provided
MAC Addresses

- Shared Ethernet links (L2 switches) strike again:
 - FCoE frame transmission path that's impossible in native FC
 - FCoE can't isolate the logical FC link as native FC would
 - VE_Ports may make this worse
- Goal: Robust solution to make FCoE behave like native FC
 - Drop frames received from wrong type of port, and complain (e.g., log)
- Solution Approach: FCoE frame indicates FC port type of FCoE port that is intended to receive the frame
 - Example: VN_Port frame sent to fabric is received by VF_Port
 - Could use combination of FCoE header and native FC header
 - Example: Class F traffic must be received by a VE_Port
 - Also identifies VN_Port to VN_Port direct traffic (not via VF_Port)
- Alternative (poor): Forbid use of dynamic MACs for VF_Ports
 - Hard to enforce - which OUIs are banned and why?
 - Prohibits software FCFs in hypervisors and virtualized OS guests
 - FCF configuration mistake (left FCF) can still cause forwarding loop

Put bits in
FCoE
Header

This is 3b:
One MAC
Duplication

3a, 3b, 3c ... – Want to Play Whac-A-Mole™?

Server Provided
MAC Addresses

- I can keep generating scenarios
 - Eventually I'm going to get bored
- What should we tell FCoE users?
 - “These forwarding loops can't happen, here's a simple proof.”



~~– “David Black can't find any of these loops and he's really smart.”~~

Put bits in
FCoE
Header



Scenario 3: Any Questions?

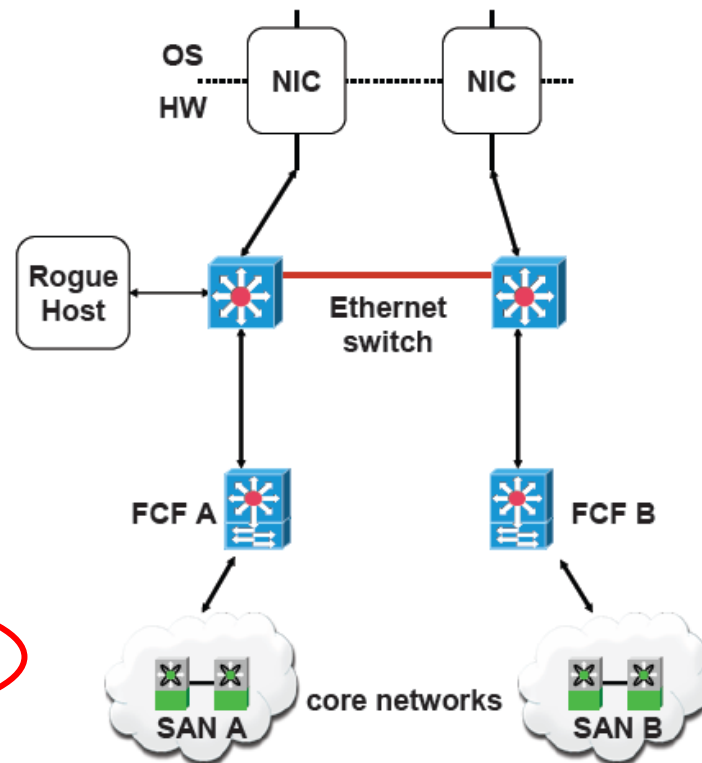
Scenario 4 is Next

Reminder: Rogue Host (from 07-546v1)

Scenario #4: Rogue Host

- Theorem #1
 - **Uniqueness in the presence of a Rogue Host is an oxymoron**
- Proof
 - The system administrator has no control on the MAC address used by a rogue host
- Corollary
 - A Rogue Host is typically caused by erroneous configuration in a Virtualized environment, but it may also be a deliberate attacker

Focus on configuration mistakes



Data Center LANs: Limited defenses against malicious attacks

Rogue Host vs. Ethernet bridge (switch) learning

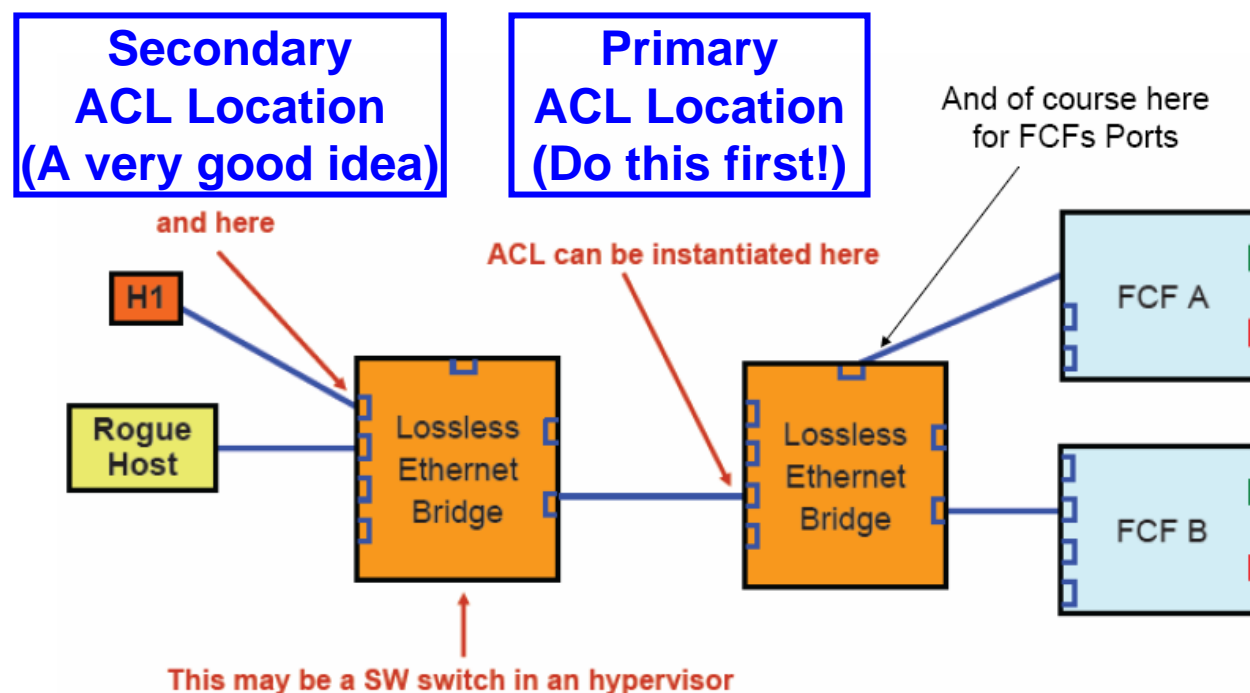
- Every few minutes the rogue host sends either a discovery or an FLOGI frame mimicking the attacked ENode [07-546v1, p.23]
 - SA=MAC of the attacked ENode
 - DA may be multicast, broadcast, or unicast
- **Malicious intent not required!!**
 - EMC has seen defective HBAs that repeatedly FLOGI
 - And do other unwelcome things repeatedly, without human “help”
 - To damage FCoE, problematic frames need not be FCoE
 - E.g., DHCP or IPv6 address configuration from OS in reboot loop
- This is **NOT** “Rogue Host”, it’s “Fault & Error Containment”
 - Non-malicious attack on learning is realistic
 - Need to provide mechanisms that can deal with this

Fault & Error Containment: Protecting Ethernet Learning

- Primary containment mechanism: ACLs (already agreed)
- ACL Must block all traffic types to protect bridge learning
 - Not a problem if there's only FCoE-related traffic on blocked MAC
 - 07-546v1 recommends dedicated MACs for FCoE
- Dedicated MACs for FCoE
 - Prevents TCP/IP interference with FCoE
 - Simplifies test, qualification, incident analysis, etc.
- Simple requirement that prevents peculiar failures/errors
 - Should be a strong recommendation or a requirement
 - NB: Need to hear from OS and Hypervisor vendors on this

Protecting Learning: ACL Locations (07-656v0 diagram)

- 07-558v1 crosstalk scenarios cross-connect two switches
- Switch ingress ACL is the primary line of defense



Protecting Learning: Default Switch Ingress ACL

- Switch Ingress ACL is primary containment mechanism
 - Must support both host and switch connection
 - Unlike FC, Ethernet doesn't distinguish host vs. switch
- Host connection: Can't block FCoE discovery by default
 - Consequence: "block everything" won't work as default ACL
- Switch connection: Don't trust ACL config on other switch
 - Motivation: Defense in depth (good robustness principle)
 - Consequence: Block traffic that other switch should have blocked
 - Helps with incremental deployment
 - Consequence: Don't allow all FCoE traffic by default
 - Verify ACL config on other switch before allowing all FCoE traffic

Default Switch Ingress ACLs and Addressing approaches

- Mapped Addresses (07-546v1, p.33)
 - Source Address: Block all VN Port MACs, Block each FCF MAC
 - 1 entry (block VN_Port site prefix) + 1 entry per FCF (block its MAC)
 - Destination Address: Allow discovery (multicast and FCF unicast)
 - 1 entry (multicast) + 1 entry per FCF + 1 entry (block all other FCoE traffic)
 - Total: 3 entries + 2 entries for each FCF: $3 + 2 * F$
 - Probably ok, based on assumption that F is a small number
- Server Provided Addresses (07-656v0, p.12)
 - Switch has to know every allowed VN Port MAC
 - For each source MAC, need 1 entry for discovery + 1 entry per FCF: $S * (1 + F)$
 - S must be a small number for this to work well
 - Hypervisors: Count individual MACs, not hypervisor MAC prefixes
 - Allowing entire hypervisor MAC prefixes open access too widely.
 - **Upshot: Management headache for switch-to-switch ACL**
 - FCF must automatically deploy this ACL (2-3 entries per source)
 - Additional admin operation to deploy new server (or guest VM) is problematic

Conclusions

- Both addressing formats can be made to work
 - No potentially fatal design flaw visible in either format
- Mapped Addresses: Need to spell out MAC prefix details
- Server Provided Addresses: Need FCoE header bits
 - Whac-A-Mole design approach is not robust
- ACLs are still a major problem for both address formats
 - **If we've made a major architectural mistake, this is it!!**
 - MAC duplication causes and prevention need additional attention
 - Automatic switch ACL deployment & update is a requirement

EMC Perspective on FCoE MAC Addressing

- Both MAC addressing formats can be made to work
 - No potentially fatal design flaw visible in either format
- **Strongly Prefer:** One MAC address format
 - The “right” technical answer
- **Unacceptable:** One MAC address format + standards war
 - If/when one format is chosen, decision must become final, quickly!!
- Bottom line: Decision on single format now is premature
 - Too much risk of an Unacceptable outcome
 - Do it right or do it over ...

FCoE information (as of Monday)

Date	Date	Title	File
▼ FC-BB-5			
T11/07-591v2	12/03/2007	FCoE and server driven MAC addresses	PDF
T11/07-694v0	12/01/2007	FCOE aware Ethernet switches	
T11/07-693v0	12/01/2007	FCoE: Virtualization	
T11/07-547v2	12/01/2007	FCoE: Addressing & High Availability	
T11/07-692v0	11/30/2007	FCoE: Mechanism for handling duplicate MAC addresses	
T11/07-691v0	11/30/2007	FCoE: Considerations on Data Integrity	
T11/07-690v0	11/30/2007	FCoE: Use of ACLs in a virtual machine environment	
T11/07-689v0	11/30/2007	FCoE: Considerations on Duplicate MAC Addresses	
T11/07-688v0	11/30/2007	FCoE: Checking rules prevent corruption	
T11/07-685v0	11/29/2007	FCoE: ENode MAC Address Implementations	
T11/07-683v0	11/28/2007	FCoE: Fabric Crosstalk Update	
T11/07-682v0	11/28/2007	Xgig Trace Viewer Support on FCoE	
T11/07-680v0	11/27/2007	FCoE Address Considerations	
T11/07-670v0	11/14/2007	dpANS - Fibre Channel - Backbone - 5	PDF

FCoE information (as of Tuesday)

Date	Date	Title	File
▼ FC-BB-5			
T11/07-689v0	12/04/2007	FCoE: Considerations on Duplicate MAC Addresses	PDF
T11/07-682v0	12/03/2007	Xgig Trace Viewer Support on FCoE	
T11/07-715v0	12/03/2007	FCoE: Comparing ACLs to Oranges	
T11/07-714v0	12/03/2007	FCoE: Mapped Addresses in Review	
T11/07-683v0	12/03/2007	FCoE: Fabric Crosstalk Update	PDF
T11/07-591v2	12/03/2007	FCoE and server driven MAC addresses	PDF
T11/07-694v0	12/01/2007	FCOE aware Ethernet switches	
T11/07-693v0	12/01/2007	FCoE: Virtualization	
T11/07-547v2	12/01/2007	FCoE: Addressing & High Availability	
T11/07-692v0	11/30/2007	FCoE: Mechanism for handling duplicate MAC addresses	
T11/07-691v0	11/30/2007	FCoE: Considerations on Data Integrity	
T11/07-690v0	11/30/2007	FCoE: Use of ACLs in a virtual machine environment	
T11/07-688v0	11/30/2007	FCoE: Checking rules prevent corruption	
T11/07-685v0	11/29/2007	FCoE: ENode MAC Address Implementations	
T11/07-680v0	11/27/2007	FCoE Address Considerations	
T11/07-670v0	11/14/2007	dpANS - Fibre Channel - Backbone - 5	PDF

EMC²[®]

where information lives[®]