VN2VN: A new framework for L2 DAS

Prafulla Deuskar
Mark Wunderlich
Intel
L2 DAS Vision

- Provide a simple extensible L2 block storage network
- From host SCSI layer devices discovered as Direct Attached Storage
  - Block devices are “owned” by the host
  - Assume close physical device locality or proximity
- Storage traffic is isolated on a VLAN
- Framework for discovery/authentication
- DCB critical to provide “no drop” behavior
Usage Models

- Storage in a Rack
  - Self contained PODs with compute, network and storage which today might use iSCSI
  - L2 DAS, especially with microservers as compute density doesn’t allow space for local storage

- Caching Appliance in a Rack
  - Flash based storage appliance that is used to introduce a caching tier in the server
What is VN2VN?

- VN2VN is a new protocol for FCoE that allows two ENodes to communicate with each other without the need for FCF.

- Two variants for VN2VN
  - Pt-to-Pt Mode
    - Similar in concept to Fibre Channel Pt-to-Pt
  - Multi-point Mode
    - Similar in concept to Fibre Channel Arbitrated loop
Why use VN2VN?

- VN2VN decouples FCoE from FCF
  - Needs only a DCB aware Ethernet switch
  - Reduces the total cost of solution
  - Reduces the management burden

- Expands the usage models where FCoE can play
  - Not all usage models require a traditional SAN
FCoE: Initial Network Deployment Model

Server with Open-FCoE stack over Intel x520 CNA

Intel 10GbE Switch Silicon

Ethernet L2 Segment

FCoE Switch (FCF)

Storage Devices
FCoE: New Network Deployment model using VN2VN

- Server with Open-FCoE stack over Intel x520 CNA
- Ethernet L2 Segment
- Intel 10GbE Switch Silicon
- FCoE

No need for FCoE switch with VN2VN

VN2VN Protocol End-to-end

Storage Devices
ENode Initialization

- Fabric Mode and VN2VN mode are independent modes
- ENode MAC can run Fabric and VN2VN protocols in parallel
- ENode MAC can also be configured to operate only in VN2VN mode
VN2VN Address assignment

- Unlike FCF, in VN2VN – there is no central entity to assign FC_IDS to ENodes
- FC_IDS assigned by VN2VN are in range of 000001h to 00FFFFFFh
- ENode MAC needs to run a protocol to assign Locally Unique N_Port_IDs (ie FC_IDS) and ensure that there is no collision for that address in the network
LUID Assignment Protocol

- **Probing State**
  - Wait random time between 0 and PROBE_WAIT, select tentative LUID and Probe the network for collisions
  - After two Probe Requests and there are no replies, Goto Claiming state
  - If any Probe Reply received, select new LUID and try again

- **Claiming State**
  - Send Claim Notification to claim LUID and wait up to ANNOUNCE_WAIT to collect Claim Responses and build Neighbor state
  - If you receive Claim Notification or Beacon for same LUID, resolve conflict by checking port name. Goto Probing state in case local port name is less than remote port name
  - Goto Operational state

- **Operational State**
  - Send periodic Beacon messages
  - Perform FIP FLOGI and FCoE PLOGI
LUID Exchange

Initiator

1. Select LUID (Recorded or Random Generation)
2. Probe Request (#1) (Multicast: All_VN2VN_Enode_MACs)
3. Probe Request (#2)
4. Claim Notification (Multicast: All_VN2VN_Enode_MACs, inc. FC-4 attributes)
   - Look for target FC-4 zone information, add to neighbor table
5. Beacon (Multicast: Send every Beacon_Period)
   - Virtual links not required while maintaining LUID

Target

1. All nodes see Probe requests
2. First opportunity to indicate ID conflict. Claim Notification or Beacon can also be used to indicate conflict. Node with larger WWPN win conflict*.
3. Not used to indicate ID conflict. Opportunity to share FC-4 and add to local neighbor list
4. Completes LUID Exchange – Enter Operational State (Perform FIP FLOGI/PLOGI)
Virtual Link Establishment
WWPN Control

Initiator

LUID Exchange including FC-4 attributes

Beacon (Multicast)

FIP FLOGI

Accept

Ex. I-WWPN < T-WWPN
WWPN Precedence Control
Larger WWPN is to perform PLOGI

Accept

PLOGI

Accept 'Initiator Mode'

Target

Indicates remote node is ready for virtual link establishment

Allow/Expect Both sides of virtual link may attempt to FLOGI. Not required from both sides for Pt-to-Pt.

Largest area of variation to date.

Must resolve to respect WWPN control
Windows FCoE stack with VN2VN
Open-FCoE target stack (supports VN2VN)

- tcm_fc – acts as transport specific service provider
  - On load registers itself as FC4 provider with libfc
- libfc invokes tcm_fc entry points on receiving plogi/prli/read/write from initiator if lport is acting as target. Default role of lport is initiator. (lport gets created when fcoe is enabled on a device)
- Supports X520 offload features in target mode. (FC CRC, DDP, LSO)
- All fcoe target components are upstream (Linux 3.0.0)
- RHEL 6.2 and SuSE SLES11 SP2 have integrated open-fcoe target stack
Open-FCoE Linux with VN2VN
3 Million IOP SAN Target POC
Based On Storage Networking Fast Path Technology

- Storage Networking Fast Path takes a platform perspective on optimizing SAN performance to achieve 3M IOPS
  - Fast Storage + Fast Networking ≠ Fast Storage Networking

- Intel harmonized the Open FCoE networking stack with the NVM Express storage stack while exploiting Intel Data Direct I/O on the Intel® Xeon® E5 processor

- This technology demonstration illustrates the impact NVM will have on future network performance requirements (10GbE → 40GbE → 100GbE)

- World’s first demo of end-to-end FCoE (VN2VN) using standard 10GbE DCB-enabled switches
NVMe SAN Storage Networking Fast Path

- 3M Random Read (4K) IOPS from fio workload with 12 initiators
  - 120 Gbps of FCoE traffic
  - 12.7 GBps of NVMe SSD throughput
  - Intel® DDIO reduced memory bandwidth by 5X

- Six prototype PCIe 32GB SSDs connected via PCIe expansion chassis
- Dual Intel® Xeon® E5 processor Linux storage target supporting Open-FCoE VN2VN target and NVM Express storage
- Twelve ports of Intel 10GbE Converged Network Adapters
- Intel 24-port DCB-enabled Ethernet switch
Conclusion

- VN2VN is a compelling entry level option to provide scale and performance of FC at the cost and simplicity of Ethernet

- Target Vendor – call to action
  - Consider adding VN2VN support to your FCoE solutions
References

- T11/10-156v0 - VN2VN Multi-Point and Point-to-Point, DeSanti et. al
- T11/12-280v1 – VN2VN VLAN discovery protocol, Deuskar et. al
- T11/12-007v1 – FC-BB-6 draft
What is Open-FCoE?

- Goals of the project
  - Develop native FCoE initiators for different OS.
    - Not all OS initiators are open-source.
  - Develop open source FCoE target implementation.

- Hosted at www.open-fcoe.org
  - Git repository
  - Developers mailing list