FCoE Boot Configuration Setup on Intel using Lifecycle Controller

Zhan Liu
Roger Faulkner
FCoE Boot Configuration

Contents
Introduction .................................................................................................................. 4
FCoE boot workflow........................................................................................................ 4
  1) FQDD selection ........................................................................................................ 4
  2) Device enablement .................................................................................................... 4
  3) FQDD check .............................................................................................................. 5
  4) Link status check ...................................................................................................... 5
  5) FCoE boot enablement check .................................................................................. 5
  6) Boot order check ...................................................................................................... 6
  7) Configuration .......................................................................................................... 6
Multi-port boot for resiliency ........................................................................................... 7
References ..................................................................................................................... 7
Glossary .......................................................................................................................... 7
**Introduction**

The ability to programmatically configure a network controller card for FCoE boot is a powerful feature of Dell’s Lifecycle Controller. FCoE boot configuration allows you to set up a network controller with FCoE boot support to boot into a remote FCoE target. Lifecycle Controller also provides the ability to override some of the I/O identity attributes of the card (i.e. WWPN) with its Virtual Attributes. This feature in turn provides more flexibility in deployments that need rapid re-configuration of system workloads to another system. The configuration workflow varies by vendor (Intel, Qlogic, etc.). This paper will focus on the Intel Mezzanine card implementation.

This document describes the FCoE boot workflow steps using the remote API exposed by the LifeCycle Controller 2 (LC2) capability of Dell PowerEdge 12th generation servers. The goal of this paper is to provide clear steps to set up FCoE boot on Intel’s network controllers.

Additional FCoE information with accompanying PYTHON scripts, Qlogic and Intel cards inclusive, can be found in the *Best Practice Guide*.

**FCoE boot workflow**

First, delete all pending jobs and pending values as they may prevent further configuration changes.

1) **FQDD selection**

Select the network device that is connected to the FCoE boot target by the Fully Qualified Device Descriptor (FQDD). The FQDD (for example, NIC.Mezzanine.2B-1) of the network device uniquely identifies the device. This information can be acquired by enumerating the `DCIM_NICView` class. The `InstanceID` field is the FQDD of the device.

If the FQDD is not visible on the target server, one of the following two circumstances is possible:

- The server does not contain such a device
- The device is disabled in the BIOS and needs to be enabled prior to configuration

2) **Device enablement**

To use the NIC card, the card must be enabled, and to enable a disabled device, we must identify the type of network device, i.e. whether it is on-board, a daughter card, or an add-in. The FQDD contains this information: for example NIC.Mezzanine.2B-1 is an add-in Mezzanine card.

For Dell PowerEdge 12th generation servers, NDC and add-in controllers can be disabled in the BIOS. If the target device is disabled and not visible in the `DCIM_NICView` enumeration, that device must be enabled in the BIOS prior to configuring FCoE.

To enable these devices in the BIOS, the following BIOS attributes must be manipulated:

- For NDC on Dell PowerEdge 12th generation servers
  - `IntegratedNetwork1 = Enabled` for 1-1, 1-2, 1-1-1, 1-1-4, etc
  - `IntegratedNetwork2 = Enabled` for 2-1, 2-2, 2-1-1, etc
  - ...
- Add-ins on Dell PowerEdge 12th generation servers
  - `Set Slot1 = Enabled` for Slot.1-1 or Mezzanine.1-1
  - `Set Slot2 = Enabled` for Slot.2-1 or Mezzanine.2-1
  - ...


Using the `SetAttributes()` method on the `DCIM_BIOSService` class to set up these attributes as required.

Dell also recommended that you ensure `BootMode = Bios` at this point. The current value of the `BootMode` attribute can be obtained by enumerating the `DCIM_BIOSEnumeration` class.

Since a BIOS job is being scheduled, Dell recommends disabling all boot sources on the system. This will ensure that the system will not boot into another source until FCoE boot configuration is completed. Disabling boot sources can be done with the `ChangeBootSourceState()` method on the `DCIM_BIOSService` class on every source.

All boot sources on the system can be listed by enumerating the `DCIM_BootSourceSetting` class.

Next, using the `CreateTargetedConfigJob()` method on the `DCIM_BIOSService` class, create a BIOS job in order for the changes to be committed.

Since boot source changes only occur after reboot, they get detected only after CSIOR completes and the iDRAC database (configDB) is refreshed. This information can be polled by invoking the `GetRemoteServicesAPIStatus()` method from the `DCIM_LCService` class.

3) FQDD check

After a disabled NIC is enabled and the BIOS job is completed, re enumerate the `DCIM_NICView` class to ensure that the target FQDD is now present.

If the FQDD is still not returned by the enumeration, the device FQDD is invalid.

4) Link status check

To ensure a successful boot it is a good practice to check the link status prior to reconfiguration. This can be achieved by performing a `get` on the `DCIM_NICEnumeration` class for the `LinkStatus` attribute for the targeted FQDD. The `instanceID` for the `get` should be `<FQDD>:LinkStatus` (example: `NIC.Mezzanine.2B-1:LinkStatus`). The value of this attribute is either ‘Connected’ or ‘Disconnected’. Make sure the value of this attribute is ‘Connected’ before trying to boot to FCoE target.

5) FCoE boot enablement check

Since the NIC is enabled, it is visible now. The next step is to ensure that the network device is enabled for FCoE boot. This can be done on the Intel card we are dealing with here by performing a `get` on a particular `instanceID` (For example: `NIC.Mezzanine.2B-1`) in the `DCIM_NICEnumeration` class. Look for `AttributeName = LegacyBootProto`, and check if the `AttributeValue = FCoE`.

If `AttributeValue != FCoE`, a `SetAttributes()` call on the `DCIM_NICService` class can be used to set `LegacyBootProto = FCoE`.

In this way FCoE boot is enabled. When it is enabled, there will be a corresponding entry in the boot sources list for the NIC. As mentioned earlier, all sources on the system can be listed by enumerating the `DCIM_BootSourceSetting` class.

Since a NIC job is being performed on the NIC, Dell recommends configuring all FCoE settings within the same job. The list of attributes to set is enumerated in Step 7).
FCoE Boot Configuration

Next, a NIC job needs to be created in order for the changes to be committed. This can be done using the `CreateTargetedConfigJob()` method on the `DCIM_NICService` class.

If not already done in Step 2, Dell recommends that you set `BootMode = Bios` and disable all boot sources on the system to assure that until FCoE boot configuration is completed, the system will not boot into another source.

Once again, since boot source changes only occur after reboot, they get detected only after CSIOR completes and the iDRAC database is refreshed.

If `LegacyBootProto = FCoE` already, then the boot sources need to be checked for an entry for the device FQDD. The condition to look for is:

- InstanceID = `IPL`
- source contains `IPL` and device FQDD

6) Boot order check

In this step, ensure that the network device is the first boot source in the IPL boot list. In previous steps, all other boot devices have been disabled and the network device is readied for FCoE boot enablement.

Ensure that the network device is the primary bootable device in the boot order:

- Verify if the device FQDD will boot
  i. Enumerate `DCIM_BootSourceSetting` class
  ii. Get `EnabledState` of IPL entry for device FQDD
  iii. If `EnabledState = 0`, device needs to be enabled
  iv. Invoke `ChangeBootSourceState()` method on `DCIM_BIOSService` to enable source

- Verify if the device FQDD will boot *first*
  i. Get `CurrentAssignedSequence` from enumeration above
  ii. If not first, need to ensure all preceding sources are disabled
  iii. If other sources will boot before device FQDD, need to fix boot order
  iv. Invoke `ChangeBootOrderByInstanceID()` method on `DCIM_BIOSService` to reorder source

If the above methods are invoked to fix the boot order, create a BIOS job to commit the changes. This can be done using the `CreateTargetedConfigJob()` method in the `DCIM_BIOSService` class.

7) Configuration

The following attributes must be set in order for successful FCoE boot:

**NOTE:** The values (c,d,e) depend on your SAN FCoE target configuration.

a. LegacyBootProto = `FCoE`
b. ConnectFirstFCoETarget = `Enabled`
d. FirstFCoEEFCFVLANID = `xxxx` (For example: 1002)
e. VirtWWPN = 12:34:56:78:90:ab:cd:ef (For example: 20:00:00:1B:21:CE:8B:09)
**Multi-port boot for resiliency**

It is possible to configure both ports of the Intel card for FCoE boot for resiliency. This allows the system to try the second port if the first port fails to boot. To accomplish this, repeat workflow steps 1-7 with the FQDD of the second port. For Step 6, the boot order of the second port should follow the first port.

**References**

http://www.delltechcenter.com/page/Lifecycle+Controller

[2] Lifecycle Controller Best Practice Specification  
http://www.delltechcenter.com/page/Lifecycle+Controller


**Glossary**

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSIOR</td>
<td>Collect System Inventory on Restart</td>
</tr>
<tr>
<td>BIOS</td>
<td>Basic Input / Output System</td>
</tr>
<tr>
<td>NIC</td>
<td>Network Interface Controller</td>
</tr>
<tr>
<td>NDC</td>
<td>Network Daughter Card</td>
</tr>
<tr>
<td>iDRAC</td>
<td>Integrated DELL Remote Access Controller</td>
</tr>
</tbody>
</table>