2012

Troubleshoot SAN Boot with lunlist and freinds

Using native Palo/Serrano adapter CLI and UCSM tools for Troubleshooting and Configuration

Karl Vietmeier
Cisco Systems
2/19/2012

CONTENTS

Introduction

Cisco is often asked why we don't have a BIOS utility available for configuring the Palo/Serano (Cisco VIC) interface, especially when someone is struggling with a SAN Boot configuration that isn't working. This is a fair question, and easily answered. There isn't a "legacy" BIOS utility because we don't need one, the UCS has better, more efficient ways to access and configure BIOS parameters. When using the UCS, an end user should never have to enter the BIOS for any reason, all management and configuration of objects within the system is accomplished through the UCSM interfaces (GUI, CLI, XML), and several subshells. *A thorough discussion of UCSM is outside the scope of this document, the focus will be on specific tools and interfaces related to the VIC and booting blades*.

The purpose of this whitepaper is to document these tools and provide some example use cases to demonstrate their power and usefulness

Configuration of the VIC is done via the following UCSM Policies and a key Parameter:

- **SAN Boot Policy**

  It is the SAN Boot Policy that holds the key to properly configuring a UCS Service Profile for SAN Booting. This policy configures the BIOS Boot order and the Option ROM, directing the VIC where to look for int13h devices, i.e. which FC target to connect to and which LUN the MBR has been written to. The BIOS Boot Policy is explicitly defining the device path where the bootstrap process expects to find the MBR. To successfully boot a server needs 3 things correctly defined:

  1. A "Controller", this is the vHBA
  2. A Target – the storage array port
  3. A LUN

  If there are errors in this policy – typos, wrong LUN, wrong target, etc. the blade will not boot. What causes the most confusion here is that while you can't boot, you are often able to successfully install on a LUN. *It is very important to understand the difference between Option ROM int13h boot devices and a running OS kernel with drivers loaded*. So, before looking for any other configuration problems like bad zones or incorrect LUN masking, make triply sure you have set this up correctly, and then check once more. A *large* percentage of "mysterious" Boot from SAN (BFS) issues are directly attributable to an error in the Boot Policy.

- **Adapter Policies**

  Adapter policies set common parameters like queue depths and IO timeouts, the defaults have been tested and certified with the storage arrays in the HW compatibility matrix and should not need to be altered.

- **VSAN**
  While not a policy, failure to configure this correctly will prevent any connectivity.

Intended Audience:

This document is focused on the operational aspects of the Cisco VIC, UCSM, NXOS, and Storage Arrays and is intended for use by experienced field personnel. This is not a tutorial on Fibre Channel or Storage Array concepts and the author assumes a degree of knowledge and experience in these areas, as such - the following concepts should be familiar to the reader:

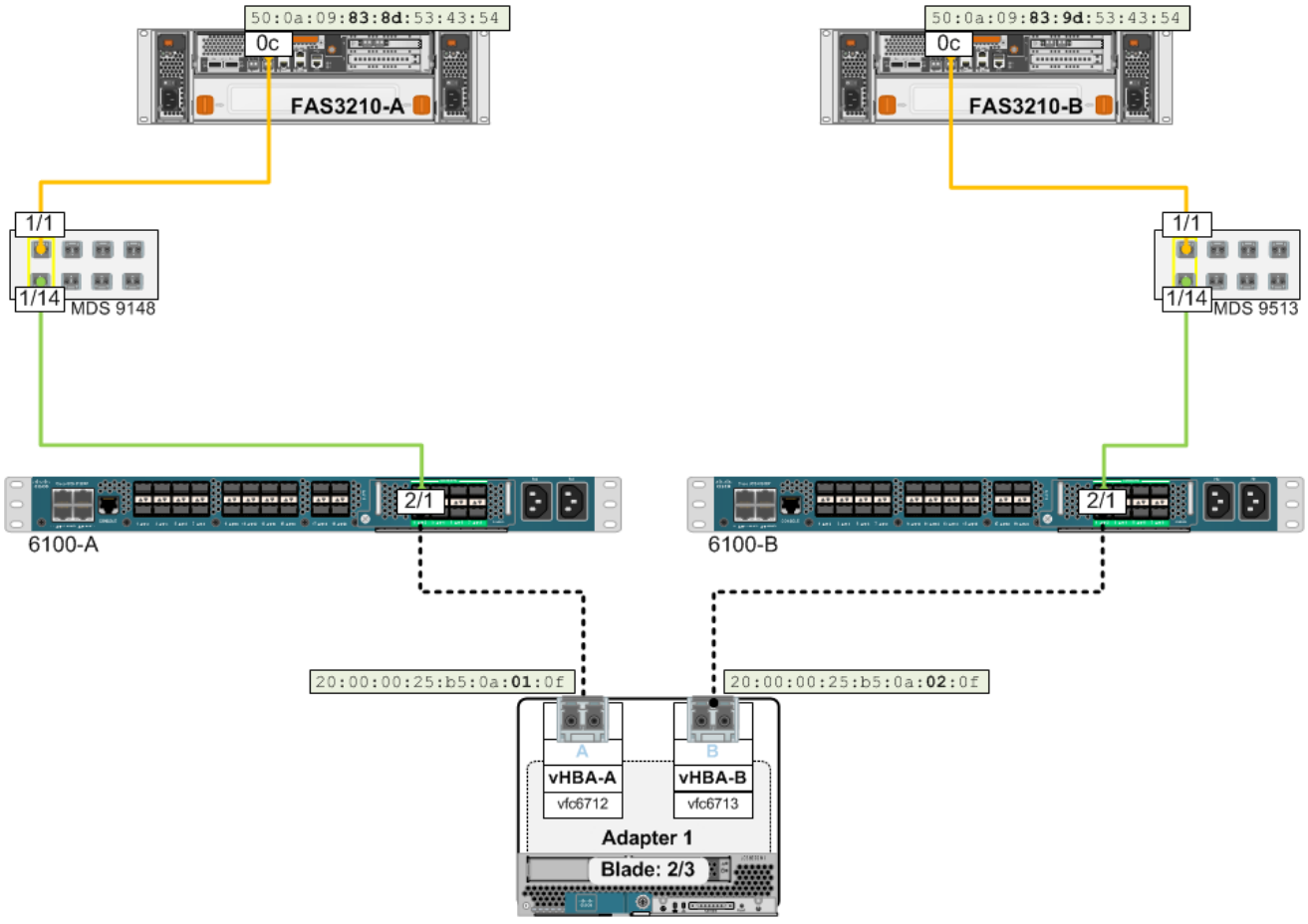
FLOGI, NPIV, LUN Masking, Fibre Channel Zoning, Option ROM, Int13h Devices.

1 Lab Configuration

The Equipment used in this set of examples consists of:

- Cisco MDS
    - 9148
    - 9513
- Cisco UCS running 2.0.1t Firmware
    - 6120
    - 2104
    - B200 Blades with M81KR Adapter
- NetApp 3210

1.1 Physical and Logical Connections



1.2 Example SAN Object Identities

Target NetApp 3210:

```
    Fabric A:  50:0a:09:83:8d:53:43:54 - port 0c
    Fabric B:  50:0a:09:83:9d:53:43:54 - port 0c
```

Initiator M81KR:

```
    Fabric A (vHBA-A): 20:00:00:25:b5:0a:01:0f
    Fabric B (vHBA-B): 20:00:00:25:b5:0a:02:0f
```

2 Troubleshooting Entry Points/Tools

There are 3 primary points of instrumentation for troubleshooting a SAN Boot problem, this paper focusses on UCS but we will use examples from the others when appropriate. Below is a quick summary of the key commands you can use to verify configuration and connectivity. You can see from the list that one need not necessarily be an expert on these platforms to get useful information; the list of relevant commands is fairly small. If you don't have access to the storage or SAN switches, you can use this list to let the appropriate Admins know what information you need. Each of these tools will be used in examples later in the document; this is simply a "Quick Reference/Summary".

2.1 UCS

2.1.1 Fabric Interconnect

- **GUI**
  - Currently configured BIOS Boot Order:
    - ⇒ Equipment/Blade#/"Boot Order Details" (also on the Server Details tab of the Service Profile)
  - Link status and Identity of vHBAs and vNICs instantiated on the blade
    - ⇒ Org/ServiceProfile/"VIF Paths"
- **UCSM CLI**
  - Get a list of associated Service Profiles - find the one you are interested in

    ```
    # sh service-profile assoc | grep –b 1 Assoc
    ```
  - Display VNIC/VHBA logical names mapped to veth/vfc and status of pinning (same as VIF paths)

    ```
    # sh service-profile circuit name <sp name>
    # sh service-profile circuit server #/#
    ```
  - What is the Boot Policy?

    ```
    # sh server boot-order #/# detail
    ```
  - What is the actual boot order currently configured in the BIOS?

    ```
    # sh server actual-boot-order #/#
    ```
- **NXOS** <`connect nxos [FI]`> (Read Only - but good information):
  - What initiators and targets are known by the Fabric Services?

    ```
    # sh fcns database
    # sh flogi database
    ```
  - Check FC interface pinning and VSAN Trunking config (only works in FC EHM)

    ```
    # sh npv status
    # sh npv flogi-table
    ```

- What VSANs are configured?

```
#  sh vsan
```

- What VSANs are the physical ports configured on and what veths are in which VSAN.

```
#  sh vsan membership
```

2.1.2 VIC shell interface

- List vnics configured on an adapter – need the output for later commands:

```
#  vnic
```

- Query FC port database

```
#  lunlist <vnic#>
```

- Show PLOGI information

```
#  login <vnic#>
```

- Get PWWN and FCID of initiator

```
#  lif <lif#>
```

2.2 MDS

- Show devices present in the Fabric: FCID, Type, Vendor, PWWN

```
#  sh fcns database
```

- Show devices present in the Fabric: Physical Port, VSAN, FCID, PWWN/WWNN

```
#  sh flogi database
```

- Zones that are currently active and device status in the zone

```
#  sh zoneset active
```

2.2.1 Storage (In this example NetApp)

- What are the WWPNs of the target ports (verify with Boot Policy)

```
#  fcp show adapters
```

- What initiators are currently FLOGI/FDISC'd into the ports

```
#  fcp show initiators
```

- What igroups are those initiators mapped to:

```
#  igroup show
```

- What LUNs are currently created and then mapped to an igroup and the LUN ID:

```
#  lun show
#  lun show -m
```

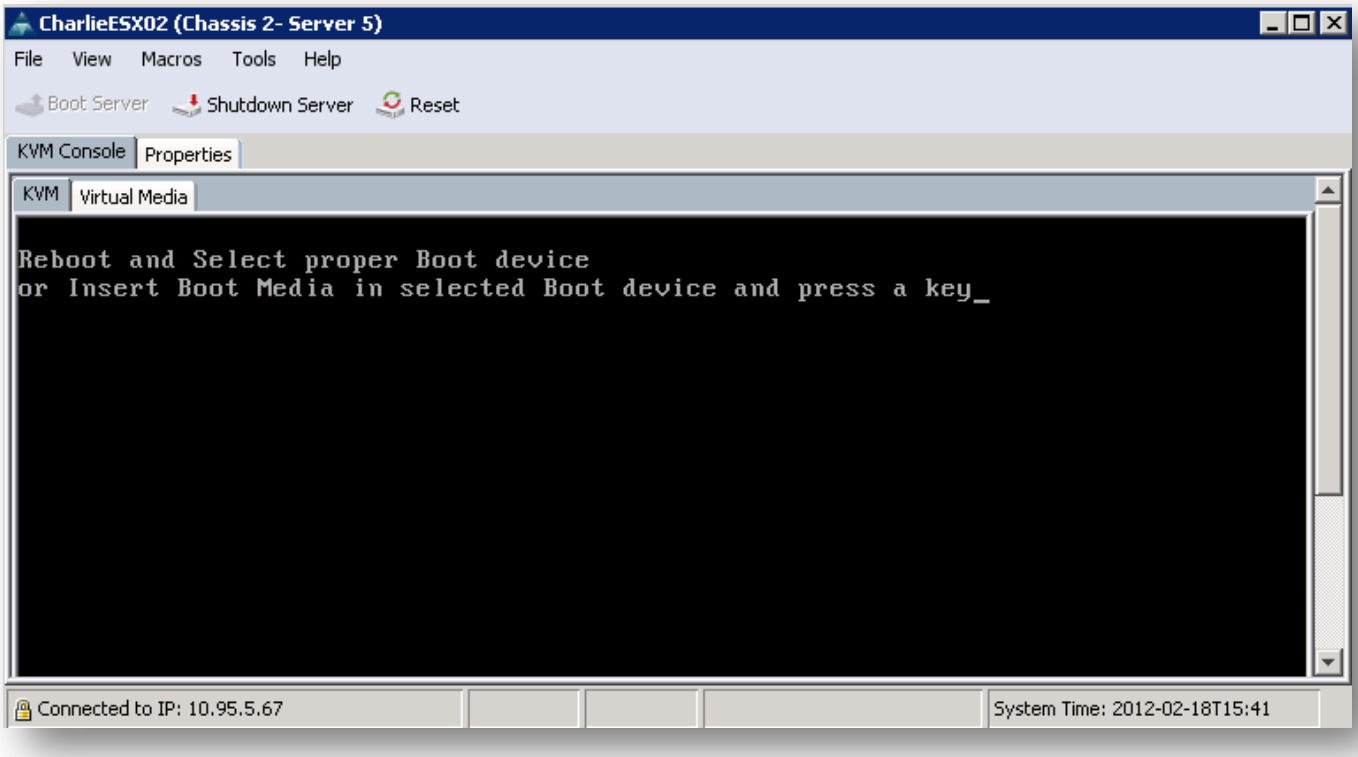
3 The Virtual Interface Controller subshell

3.1 The VIC and Option ROM:

When a Service Profile is first associated to a blade it will cycle through the PNUOS several times to run utilities that configure the WWPNs, MAC Addresses, Firmware, etc. On the final boot cycle the BIOS will be programmed according to the attached Boot Policy (See following section for discussion). During an initial "bringup" before an OS is installed the blades will stop at the "No boot device found" screen. At this point the adapter Option ROM is loaded and you can query the FC ports on the adapter to extract information about the upstream SAN devices. If everything is properly configured you will see the FC target/s and LUN/s configured in the Boot Policy. In Capitola (UCSM 2.0) a new utility, “lunlist” was added. This little utility is very useful when investigating SAN Boot problems and will be the focus of this article.

3.2 Exploring the M81KR/1280 subshell – Working Example

The following is an example from a Service Profile booting on a blade when the zoning, LUN masking, and Boot Policies are all correct but there is no OS on the LUN. This results in this unfortunately all too common prompt:



1. Connect to the adapter chassis/slot/card:

```
phx2-dc-ucs-A# connect adapter 2/5/1
```

2. Open rlogin session to M81KR interface:

```
adapter 2/5/1 # connect
```

3. Attach to the Fabric Link Services:

```
adapter 2/5/1 (top):1# attach-fls
```

4. Query vNIC interfaces to get device numbering:

   - Trivia – the “lif” is actually the vntag number given to the interface on this adapter.

```
adapter 2/5/1 (fls):1# vnic
---- ---- ---- ------- -------
vnic ecpu type state   lif
---- ---- ---- ------- -------
9    1    fc   active  5
10   2    fc   active  6
```

5. Have the interfaces logged into the target?

   - In this case yes they have, “PLOGI” indicates success.

```
adapter 2/5/1 (fls):2# lunmap 9
lunmapid: 0  port_cnt: 1
  lif_id: 6
    PORTNAME                 NODENAME                 LUN                PLOGI
    50:0a:09:83:8d:53:43:54  00:00:00:00:00:00:00:00  0000000000000000   Y

adapter 2/5/1 (fls):3# lunmap 10
lunmapid: 0  port_cnt: 1
  lif_id: 7
    PORTNAME                 NODENAME                 LUN                PLOGI
    50:0a:09:83:9d:53:43:54  00:00:00:00:00:00:00:00  0000000000000000   Y
```

6. Get port login information for each interface - target PWWN and FCID:

   - You may need the FCID later, but again, success, if it had failed the “FID” field would be all 0’s.

```
adapter 2/5/1 (fls):4# login 9
lifid: 6
  ID   PORTNAME                 NODENAME                 FID
    0: 50:0a:09:83:8d:53:43:54  00:00:00:00:00:00:00:00  0xac0600

adapter 2/5/1 (fls):5# login 10
lifid: 7
```

```
  ID    PORTNAME                  NODENAME                  FID
    0: 50:0a:09:83:9d:53:43:54   00:00:00:00:00:00:00:00   0xe60011
```

7. Use “lunlist” to view FC port database – Nameserver response, LUNID and Target information.

```
adapter 2/5/1 (fls):1# lunlist 9
vnic : 9 lifid: 6
  - FLOGI State : flogi est (fc_id 0x03000f)
  - PLOGI Sessions
     - WWNN 50:0a:09:83:8d:53:43:54 WWPN 50:0a:09:83:8d:53:43:54 fc_id 0xac0600
       - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
           LUN ID : 0x0000000000000000 (0x0, 0x4, NETAPP  , doatd4iJHIeA)
       - REPORT LUNs Query Response
           LUN ID : 0x0000000000000000
           LUN ID : 0x0001000000000000
           LUN ID : 0x0002000000000000
           LUN ID : 0x0003000000000000
           LUN ID : 0x0004000000000000
           LUN ID : 0x0005000000000000
           LUN ID : 0x0006000000000000
           LUN ID : 0x0007000000000000
           LUN ID : 0x000f000000000000
           LUN ID : 0x0010000000000000
           LUN ID : 0x0011000000000000
           LUN ID : 0x0012000000000000
  - Nameserver Query Response
     - WWPN : 50:0a:09:83:8d:53:43:54

adapter 2/5/1 (fls):2# lunlist 10
vnic : 10 lifid: 7
  - FLOGI State : flogi est (fc_id 0x23000c)
  - PLOGI Sessions
     - WWNN 50:0a:09:83:9d:53:43:54 WWPN 50:0a:09:83:9d:53:43:54 fc_id 0xe60011
       - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
           LUN ID : 0x0000000000000000 (0x0, 0x4, NETAPP  , doatd4iJHIeA)
       - REPORT LUNs Query Response
           LUN ID : 0x0000000000000000
           LUN ID : 0x0001000000000000
           LUN ID : 0x0002000000000000
           LUN ID : 0x0003000000000000
           LUN ID : 0x0004000000000000
           LUN ID : 0x0005000000000000
           LUN ID : 0x0006000000000000
           LUN ID : 0x0007000000000000
           LUN ID : 0x000f000000000000
           LUN ID : 0x0010000000000000
           LUN ID : 0x0011000000000000
           LUN ID : 0x0012000000000000
  - Nameserver Query Response
     - WWPN : 50:0a:09:83:9d:53:43:54
```

3.3 “lunlist” Breakdown

In this section we will break down the components of the “lunlist” command. Only one vnic will be broken down since they should both be the same. It is possible they could be different; this in itself tells you something. Also, some Active/Passive arrays will refuse SCSI enquiries on the passive path to the secondary vHBA, in which case you will see “access failure” on lines 6 and 7.

NOTES:

- Output from “REPORT LUNs” only lists the first 16 LUNs returned by the target query.
- Output from “Nameserver query response” only lists the first 16 targets returned by the Nameserver
- However - The Palo adapter syslog will contain log entries for all ports returned by the Nameserver and all LUNs returned by the target query.
- Also, if the LUN is provisioned and for some reason the Palo boot driver is not able to access it (example : passive path), syslog will show inquiry data for the LUN and the SCSI error status returned from the target when adapter tried to access it.

```
1  phx2-dc-ucsprod-B# connect adapter 2/5/1
2  adapter 2/5/1 # connect
3  adapter 2/5/1 (top):1# attach-fls
4  adapter 2/5/1 (fls):1# vnic
5  ---- ---- ---- ------- -------
6  vnic ecpu type state   lif
7  ---- ---- ---- ------- -------
8  9    1    fc   active  6
9  10   2    fc   active  7
10 adapter 2/5/1 (fls):1# lunlist 9
11 vnic : 9 lifid: 6

This section shows all FDISC sessions.  The Palo FC boot driver will login only to those target ports that are
configured in the lunmap table, which is defined by the primary and secondary targets in the Boot Policy.
=>
12   - FLOGI State : flogi est (fc_id 0x03000f)
13   - PLOGI Sessions
14      - WWNN 50:0a:09:83:8d:53:43:54 WWPN 50:0a:09:83:8d:53:43:54 fc_id 0xac0600
<=

The next lines are the response given by the target when the LUN configured in the Boot Policy is queried, if
successful the peripheral type, Vendor, and serial number are returned.  In this case it the Boot policy is configured for
LUN 0, the query was successful.
=>
15        - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
16           LUN ID : 0x0000000000000000 (0x0, 0x4, NETAPP  , doatd4iJHIeA)
<=

The LUNID and WWPN in the previous sections are from the Service Profile Boot Policy which can be viewed from
the CLI and compared:
17 phx2-dc-ucsprod-B# sh server boot-order 2/5 detail | grep -a 5 SAN
18         SAN Image:
19             Type: Primary
20             VHBA: vHBA-A
21
22             SAN Image Path:
23                 Type: Primary
24                 LUN: 0
25                 WWN: 50:0A:09:83:8D:53:43:54
26
27             Type: Secondary
28             VHBA: vHBA-B
```

```
29
30              SAN Image Path:
31                  Type: Primary
32                  LUN: 0
33                  WWN: 50:0A:09:83:9D:53:43:54
```

This next section is where the first 16 LUNs returned by the target are listed; any LUNs beyond 16 will be recorded in the syslog. This initiator has 12 LUNs mapped to it on the Array.

=>

```
34          - REPORT LUNs Query Response
35              LUN ID : 0x0000000000000000
36              LUN ID : 0x0001000000000000
37              LUN ID : 0x0002000000000000
38              LUN ID : 0x0003000000000000
39              LUN ID : 0x0004000000000000
40              LUN ID : 0x0005000000000000
41              LUN ID : 0x0006000000000000
42              LUN ID : 0x0007000000000000
43              LUN ID : 0x000f000000000000
44              LUN ID : 0x0010000000000000
45              LUN ID : 0x0011000000000000
46              LUN ID : 0x0012000000000000
```

<=

This section returns the targets available for login. If it is empty – check your zoning!

=>

```
47    - Nameserver Query Response
48       - WWPN : 50:0a:09:83:8d:53:43:54
```

<=

Secondary vHBA => in this case it is configured identically and the NetApp array is responding to SCSI inquiries on the alternate path.

=>

```
49  adapter 2/5/1 (fls):2# lunlist 10
50  vnic : 10 lifid: 7
51    - FLOGI State : flogi est (fc_id 0x23000c)
52    - PLOGI Sessions
53       - WWNN 50:0a:09:83:9d:53:43:54 WWPN 50:0a:09:83:9d:53:43:54 fc_id 0xe60011
54         - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
55             LUN ID : 0x0000000000000000 (0x0, 0x4, NETAPP  , doatd4iJHIeA)
56         - REPORT LUNs Query Response
57             LUN ID : 0x0000000000000000
58             LUN ID : 0x0001000000000000
59             LUN ID : 0x0002000000000000
60             LUN ID : 0x0003000000000000
61             LUN ID : 0x0004000000000000
62             LUN ID : 0x0005000000000000
63             LUN ID : 0x0006000000000000
64             LUN ID : 0x0007000000000000
65             LUN ID : 0x000f000000000000
66             LUN ID : 0x0010000000000000
67             LUN ID : 0x0011000000000000
68             LUN ID : 0x0012000000000000
69    - Nameserver Query Response
70       - WWPN : 50:0a:09:83:9d:53:43:54
```

4 Common SAN Boot Failure Scenarios

We start with the most common cases and the easiest for a UCS Admin to fix, and then move on to issues with the SAN switches and Storage.

It is worth mentioning again here – The most common reason for a failure to BFS after a successful OS install is a misconfigured Boot Policy. Using the "Principle of Least Astonishment and Most Likely Error" should lead you to immediately double check the Boot Policy before stating to dig around in the Palo subshell.

4.1 Boot Policy Issues

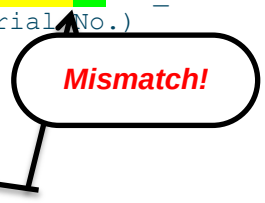
4.1.1 Wrong Boot Target in the Policy (typo)

No one has ever done this right? Of course not, but just in case, lunlist makes it really easy to check. Configured vs. Actual Boot Order is a strong hint and can point you there, but lunlist will confirm it.

You can see in the output below that the reported target (line 19), does not match the configured target (line 14). This would be a pretty common error, putting in 44 instead of 54, pretty easy to do. We get an fc_id for the initiator, but of course it is null for the target. And since the Target never responded, no LUNs are returned either. This one is pretty straightforward, and easily resolved. No need to dig any further. We really didn't even need to open up an SSH shell because the very first thing we should always check upon failure is the Boot Policy.

Result: No Target login, No LUNs Reported

```
1  phx2-dc-ucsprod-B# connect adapter 1/6/1
2  adapter 1/6/1 # connect
3  adapter 1/6/1 (top):1# attach-fls
4  adapter 1/6/1 (fls):1# vnic
5  ---- ---- ---- ------- -------
6  vnic ecpu type state   lif
7  ---- ---- ---- ------- -------
8  9    1    fc   active  6
9  10   2    fc   active  7
10 adapter 1/6/1 (fls):2# lunlist 9
11 vnic : 9 lifid: 6
12   - FLOGI State : flogi est (fc_id 0xac0e04)
13   - PLOGI Sessions
14      - WWNN 50:0a:09:83:8d:53:43:44 WWPN 50:0a:09:83:8d:53:43:44 fc_id 0x000000
15        - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
16            LUN ID : 0x0000000000000000  access failure
17        - REPORT LUNs Query Response
18   - Nameserver Query Response
19      - WWPN : 50:0a:09:83:8d:53:43:54
20
21 adapter 1/6/1 (fls):3# lunlist 10
22 vnic : 10 lifid: 7
23   - FLOGI State : flogi est (fc_id 0xe6001c)
24   - PLOGI Sessions
25      - WWNN 50:0a:09:83:9d:53:43:44 WWPN 50:0a:09:83:9d:53:43:44 fc_id 0x000000
26        - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
27            LUN ID : 0x0000000000000000  access failure
28        - REPORT LUNs Query Response
29   - Nameserver Query Response
```



```
30      - WWPN : 50:0a:09:83:9d:53:43:54
```

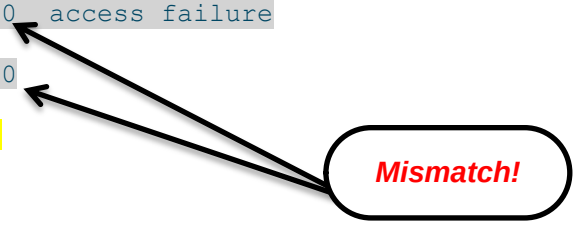
4.1.2 Wrong LUN ID in Boot Policy (case 1 – no extra LUNs mapped)

See 4.1.1 above ☺ A common and easy to make mistake.

The configured LUN (line 16) doesn't match the reported LUN (line 18), but notice the target is now correct so line 14 matches line 20. Note that incorrect LUN Masking would look exactly the same but for a different reason, the Option ROM driver will be looking for the configured LUN and not find it so the boot will fail and the output will look the same. Also note that this is a simple case where only one LUN is mapped to the Initiator – see case 2 for a more realistic configuration. What if both LUN 0 and LUN 1 are mapped to the Initiator? We will see this in the next example.

Result: Successful Target login, LUN access failure

```
1  phx2-dc-ucsprod-B# connect adapter 2/3/1
2  adapter 2/3/1 # connect
3  adapter 2/3/1 (top):1# attach-fls
4  adapter 2/3/1 (fls):1# vnic
5  ---- ---- ---- ------- -------
6  vnic ecpu type state   lif
7  ---- ---- ---- ------- -------
8  9    1    fc   active  6
9  10   2    fc   active  7
10 adapter 2/3/1 (fls):2# lunlist 9
11 vnic : 9 lifid: 6
12   - FLOGI State : flogi est (fc_id 0xac0e10)
13   - PLOGI Sessions
14      - WWNN 50:0a:09:83:8d:53:43:54 WWPN 50:0a:09:83:8d:53:43:54 fc_id 0xac0600
15        - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
16            LUN ID : 0x0001000000000000  access failure
17        - REPORT LUNs Query Response
18            LUN ID : 0x0000000000000000
19   - Nameserver Query Response
20      - WWPN : 50:0a:09:83:8d:53:43:54
21
22 adapter 2/3/1 (fls):3# lunlist 10
23 vnic : 10 lifid: 7
24   - FLOGI State : flogi est (fc_id 0xe6001d)
25   - PLOGI Sessions
26      - WWNN 50:0a:09:83:9d:53:43:54 WWPN 50:0a:09:83:9d:53:43:54 fc_id 0xe60011
27        - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
28            LUN ID : 0x0001000000000000  access failure
29        - REPORT LUNs Query Response
30            LUN ID : 0x0000000000000000
31   - Nameserver Query Response
32      - WWPN : 50:0a:09:83:9d:53:43:54
```



4.1.3 Wrong LUN ID in Boot Policy (case 2 – Many LUNs mapped)

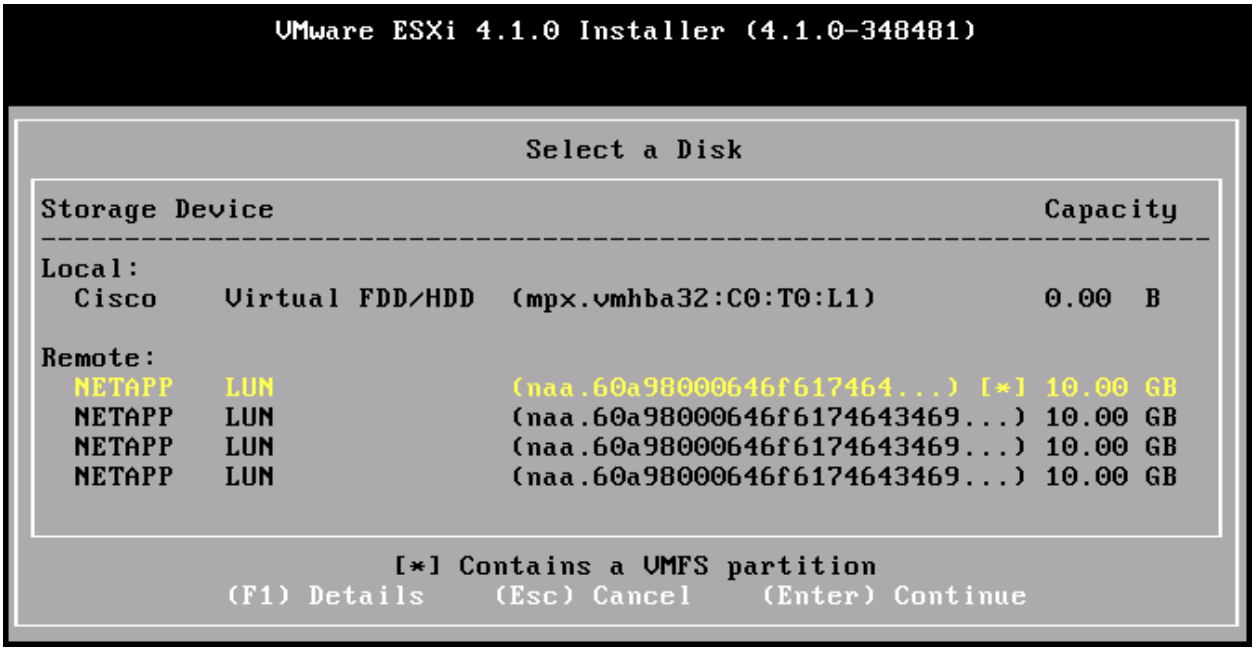
This is a more realistic example and a bit more subtle since there are no obvious problems at the Option ROM. Both paths look good (vnic 10 was omitted for brevities sake), the Target is responding and the LUNs are returned. What is the problem? Look closely and you will spot the "error" – hint - compare line 10 with 12 and 13.

```
1  phx2-dc-ucsprod-B# connect adapter 2/5/1
2  adapter 2/5/1 # connect
3  adapter 2/5/1 (top):1# attach-fls
4  adapter 2/5/1 (fls):1# lunlist 9
5  vnic : 9 lifid: 6
6    - FLOGI State : flogi est (fc_id 0xac0e0f)
7    - PLOGI Sessions
8       - WWNN 50:0a:09:83:8d:53:43:54 WWPN 50:0a:09:83:8d:53:43:54 fc_id 0xac0600
9         - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
10            LUN ID : 0x0001000000000000 (0x0, 0x4, NETAPP  , doatd4iJHJ1C)
11        - REPORT LUNs Query Response
12            LUN ID : 0x0000000000000000
13            LUN ID : 0x0001000000000000
14            LUN ID : 0x0002000000000000
15            LUN ID : 0x0003000000000000
16            LUN ID : 0x0004000000000000
17            LUN ID : 0x0005000000000000
18            LUN ID : 0x0006000000000000
19            LUN ID : 0x0007000000000000
20            LUN ID : 0x000f000000000000
21            LUN ID : 0x0010000000000000
22            LUN ID : 0x0011000000000000
23            LUN ID : 0x0012000000000000
24   - Nameserver Query Response
25      - WWPN : 50:0a:09:83:8d:53:43:54
```

No login failures and the LUNs are reported… hmmmm

⇒ We need to look at the installation to understand what happened:

- The SAN is ok, we are seeing LUNs:



- But – we chose LUN 0 to install on and the Boot Policy is telling the adapter to boot from LUN 1!

- So – This is either a mistake in the installation, or a bad Boot Policy, which it is depends on the situation. But it enforces the basic principle that at the Option ROM Driver level, there is no flexibility. The 3 elements of the device path defined in the Boot Policy; Controller (HBA), Target, and LUN, are explcit directives that define the int13h SCSI device to boot from. The bootstrap process expects to find an MBR there, if it doesn't, the boot will fail.

4.1.4 Wrong Boot Target in the Policy (Targets Reversed)

This scenario will look just like a typo in the Boot Policy. The Option ROM driver is looking for a specific ControllerID/Target#/LUN#, by reversing the targets the Target#/LUN# won't match up. If this were an Active/Active Array it might actually work since both Target ports will be presenting the same LUN and will respond to SCSI Inquiries.

4.2 LUN Masking is wrong

For this example we use a valid Boot Policy but modify the LUN Masking on the Array.

We need to be a bit careful here as Storage Arrays manage LUN Masking differently - example 1 on the blog "JeffSaidSo" is using Xiotech and is returning different results. In these examples we are using a NetApp FAS 3210. Data ONTAP LUN Masking is done through igroups and is "all or nothing". When no LUN is mapped to an igroup, the Array will return null when queried for a LUN by the initiator in the igroup.

4.2.1 No LUNs Mapped

```
1  phx2-dc-3120a> igroup show TrainBravoESX01
```

```
2       TrainBravoESX01 (FCP) (ostype: vmware):
3           20:00:04:ff:0c:00:00:0d (logged in on: 0c)
4           20:00:08:ff:0c:01:00:0d (logged in on: vtic)
5
6
7   phx2-dc-3120a> lun show -m
8   LUN path                                Mapped to          LUN ID  Protocol
9   ----------------------------------------------------------------------------
10  /vol/AlphaESX01/lun.1                   TrainAlphaESX01         0       FCP
11  /vol/AlphaESX02/lun.1                   TrainAlphaESX02         0       FCP
12  /vol/BravoESX01/lun.1                   TrainBravoESX01         0       FCP
13  /vol/BravoESX02/lun.1                   TrainBravoESX02         0       FCP
14
15  phx2-dc-3120a> lun unmap /vol/BravoESX01/lun.1               TrainBravoESX01
16
17  phx2-dc-3120a> lun show -m
18  LUN path                                Mapped to          LUN ID  Protocol
19  ----------------------------------------------------------------------------
20  /vol/AlphaESX01/lun.1                   TrainAlphaESX01         0       FCP
21  /vol/AlphaESX02/lun.1                   TrainAlphaESX02         0       FCP
22  /vol/BravoESX02/lun.1                   TrainBravoESX02         0       FCP
23
24  phx2-dc-3120a> igroup show TrainBravoESX01
25      TrainBravoESX01 (FCP) (ostype: vmware):
26          20:00:04:ff:0c:00:00:0d (logged in on: 0c)
27          20:00:08:ff:0c:01:00:0d (logged in on: vtic)
```

On line 15 above we unmap the LUN from the igroup and reboot the blade, which of course fails to boot, but it is logged in to the igroup as you can see on lines 24-27.

The output below is exactly what we would expect. The Option ROM driver successfully connects to the target as configured in the Boot Policy, but when it queries for LUNs, it is returned a null value – no LUNs. So we get access failure on line 10, nothing reported under “REPORT LUNS” (line 11), but the Nameserver responds with a valid target matching the Boot Policy (line 13).

```
1   phx2-dc-ucsprod-B# connect adapter 1/6/1
2   adapter 1/6/1 # connect
3   adapter 1/6/1 (top):1# attach-fls
4   adapter 1/6/1 (fls):1# lunlist 9
5   vnic : 9 lifid: 6
6     - FLOGI State : flogi est (fc_id 0xac0e04)
7     - PLOGI Sessions
8       - WWNN 50:0a:09:83:8d:53:43:54 WWPN 50:0a:09:83:8d:53:43:54 fc_id 0xac0600
9         - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
10            LUN ID : 0x0000000000000000  access failure
11        - REPORT LUNs Query Response
12    - Nameserver Query Response
13      - WWPN : 50:0a:09:83:8d:53:43:54
14
15  adapter 1/6/1 (fls):2# lunlist 10
16  vnic : 10 lifid: 7
17    - FLOGI State : flogi est (fc_id 0xe6001c)
18    - PLOGI Sessions
19      - WWNN 50:0a:09:83:9d:53:43:54 WWPN 50:0a:09:83:9d:53:43:54 fc_id 0xe60011
20        - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
21            LUN ID : 0x0000000000000000  access failure
22        - REPORT LUNs Query Response
```

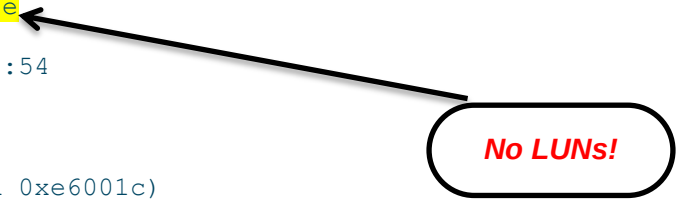
```
23    - Nameserver Query Response
24       - WWPN : 50:0a:09:83:9d:53:43:54
25
```

4.2.2 LUNs Mapped, no initiators in the igroup

What if we remove the initiators from the igroup but put the LUN back? The results are predictably the same. Successful response from the target but no LUNs returned. In this case lunlist isn't going to tell you exactly what is wrong with the LUN masking, but it does get you closer and let you blame the Storage Admin ☺

```
1  phx2-dc-3120a> igroup show TrainBravoESX01
2      TrainBravoESX01 (FCP) (ostype: vmware):
3
4  phx2-dc-3120a> lun show -m
5  LUN path                                Mapped to          LUN ID  Protocol
6  ----------------------------------------------------------------------------
7  /vol/AlphaESX01/lun.1                   TrainAlphaESX01         0       FCP
8  /vol/AlphaESX02/lun.1                   TrainAlphaESX02         0       FCP
9  /vol/BravoESX01/lun.1                   TrainBravoESX01         0       FCP
10 /vol/BravoESX02/lun.1                   TrainBravoESX02         0       FCP


11 adapter 1/6/1 # connect
12 adapter 1/6/1 (top):1# attach-fls
13 adapter 1/6/1 (fls):1# vnic
14 ---- ---- ---- ------- -------
15 vnic ecpu type state   lif
16 ---- ---- ---- ------- -------
17 9    1    fc   active  6
18 10   2    fc   active  7
19 adapter 1/6/1 (fls):2# lunlist 9
20 vnic : 9 lifid: 6
21   - FLOGI State : flogi est (fc_id 0xac0e04)
22   - PLOGI Sessions
23      - WWNN 50:0a:09:83:8d:53:43:54 WWPN 50:0a:09:83:8d:53:43:54 fc_id 0xac0600
24        - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
25            LUN ID : 0x0000000000000000  access failure
26        - REPORT LUNs Query Response
27   - Nameserver Query Response
28      - WWPN : 50:0a:09:83:8d:53:43:54
29
30 adapter 1/6/1 (fls):3# lunlist 10
31 vnic : 10 lifid: 7
32   - FLOGI State : flogi est (fc_id 0xe6001c)
33   - PLOGI Sessions
34      - WWNN 50:0a:09:83:9d:53:43:54 WWPN 50:0a:09:83:9d:53:43:54 fc_id 0xe60011
35        - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
36            LUN ID : 0x0000000000000000  access failure
37        - REPORT LUNs Query Response
38   - Nameserver Query Response
39      - WWPN : 50:0a:09:83:9d:53:43:54
```



4.3 Zone configuration is incorrect

If you were able to see the LUNs during the OS install but can't boot, you can check this possibility off of your list. If the zoning is incorrect on the switch you won't be getting any further, it is essentially the same as having

the cable unplugged. If you couldn't see any LUNs with the installer (and you have the correct drivers loaded), it isn't an Option ROM mapping issue, yet, you may still have a Boot Policy misconfiguration that gets revealed after fixing the zones. In this case you will get failures on both the Target login and LUN reporting.

4.4 Everything is correct but I still can't boot from the LUN

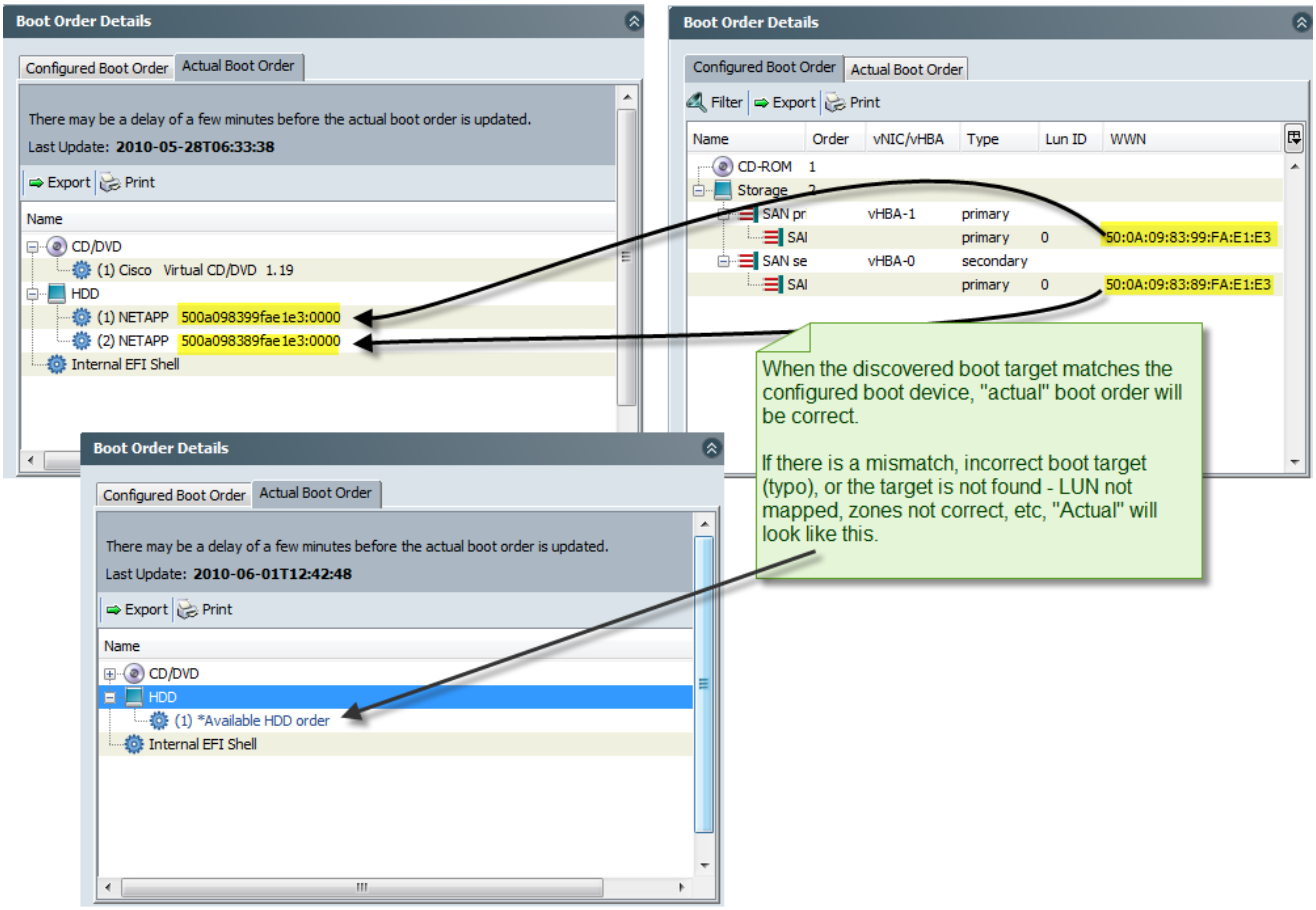
After going through all of the steps above, running every command you can think of and getting the SAN team mad at you for blaming everything on them, you still can't boot! The most likely root cause for this use case is an MBR placement problem (See 4.1.3 above). You are correctly defining the target and LUN in the Boot Policy, and installed the OS just fine, but the MBR was placed on LUN1 instead of LUN 0. It is possible that after the reboot the PCI devices got rearranged…. This is most common in Linux installations where you can choose the location for the MBR during install. But all bets are off with Windows….

5 Using UCSM GUI Tools

UCSM is a very full featured interface with many points of entry into the system and in many cases you will be better served by surfing through the GUI than diving into the CLI and subshells. The 2 most relevant tools for troubleshooting SAN Boot problems are “Boot Order Details”, and “VIF Paths”.

5.1 "Boot Order Details"

Is a real time view into the current status of the BIOS boot order configuration, rather than having to stop the boot process and enter a separate interface one can view the BIOS live. To prove this to yourself; observe it as a blade cycles you will see it transition several times - for example, during the PNUOS boot phase the "Actual Boot Order" will be a PXE boot from an internal IP interface. It can be accessed from 2 locations, the Equipment Tab – Blade Navigator, or the Service Profile Tab – “Server Details”.



There is a CLI Equivalent:

- What is the Boot Policy?

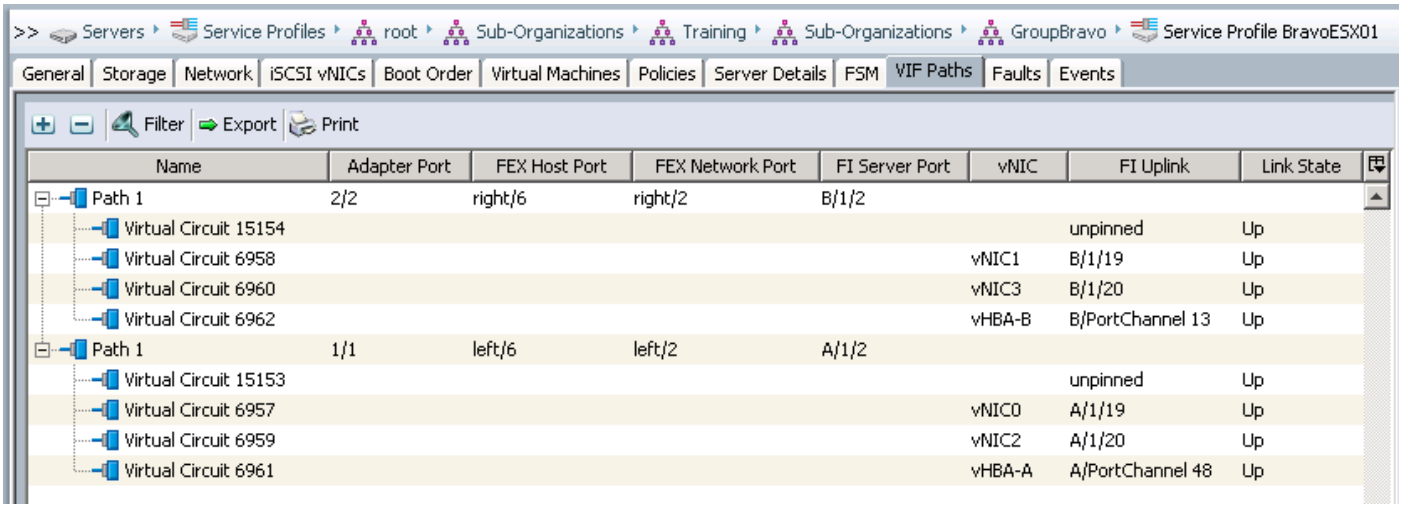
```
#  sh server boot-order #/# detail
```

- What is the actual boot order currently configured in the BIOS?

```
#  sh server actual-boot-order #/#
```

5.2 VIF Paths

This gives a view into the currently mapped virtual interfaces and how they are pinned to northbound resources. It will quickly tell you if all of the instantiated virtual interfaces are correctly connected and the current status. It contains a wealth of information in a very compact format and really deserves its own detailed description, which is outside the scope of this document at the moment. I may go into this in more depth in another paper and those familiar with US Internals should have no trouble interpreting the output.



Name	Adapter Port	FEX Host Port	FEX Network Port	FI Server Port	vNIC	FI Uplink	Link State
Path 1	2/2	right/6	right/2	B/1/2			
Virtual Circuit 15154						unpinned	Up
Virtual Circuit 6958					vNIC1	B/1/19	Up
Virtual Circuit 6960					vNIC3	B/1/20	Up
Virtual Circuit 6962					vHBA-B	B/PortChannel 13	Up
Path 1	1/1	left/6	left/2	A/1/2			
Virtual Circuit 15153						unpinned	Up
Virtual Circuit 6957					vNIC0	A/1/19	Up
Virtual Circuit 6959					vNIC2	A/1/20	Up
Virtual Circuit 6961					vHBA-A	A/PortChannel 48	Up

There is a CLI Equivalent:

- Display VNIC/VHBA logical names mapped to veth/vfc and status of pinning (same as VIF paths)

```
# sh service-profile circuit name <sp name>
# sh service-profile circuit server #/#
```

6 Appendix A - Trace the Path: Working outward from UCS to the Target

In the scenarios where you aren't seeing a Target or any LUNs which in lunlist look like Nameserver failures you may want to want to double check connectivity before rebooting, especially with large memory footprints since a reboot is costly time wise. After you double and triple check the Boot Policy you can use the following steps to verify that everything is correct within UCS first, then the switch, and finally the storage (this example is verifying the results in 4.1 above). Using the steps in 6.1 you can at least determine whether the problem is within the UCS itself.

6.1 Verify internal UCS Connections:

1. Find the vifs assigned to the adapter:

```
phx2-dc-ucsprod-B(nxos)# sh vifs int ethernet 1/1/6

Interface      MAX-VIFS VIFS
-------------- -------- ------------------------------------------------------
Eth1/1/6       30       Veth6958, Veth6960, Veth15154,
```

2. FC adapter instances are assigned veths in a very high number range, typically 8000 and above:

```
phx2-dc-ucsprod-B(nxos)# sh vifs int veth15154

Interface      MAX-VIFS VIFS
-------------- -------- ------------------------------------------------------
Veth15154      0        vfc6962,
```

3. Verify that the VSAN is correct and the interface is configured:

```
 phx2-dc-ucsprod-B(nxos)# sh int vfc6962 brief

--------------------------------------------------------------------------------
Interface  Vsan   Admin  Admin   Status          SFP    Oper  Oper   Port
                  Mode   Trunk                         Mode  Speed  Channel
                         Mode                                (Gbps)
--------------------------------------------------------------------------------
vfc6962    43     F      on      trunking         --     TF      auto --
```

4. Now query the UCS internal NPV flogi-table to find the FCID. Which you can use along with the WWPN to verify connectivity and zone configuration on the switch. In this case everything looks good on UCS. We move on to the FC Switch next.

   - ***NOTE:*** *If at this point the vfc has failed to FLOGI into the Fabric Interconnect you know that the problem is within UCS. The most common reason for a failure at this step is an incorrect VSAN configured in the Service Profile, or the VSAN in the SAN configuration is wrong. Of course if it is an incorrect VSAN you shouldn't see any LUNs from the OS installer – check both anyways.*

```
phx2-dc-ucsprod-B(nxos)# sh npv flogi-table
```

```
--------------------------------------------------------------------------------
SERVER                                                                  EXTERNAL
INTERFACE VSAN FCID             PORT NAME               NODE NAME       INTERFACE
--------------------------------------------------------------------------------
vfc6575   43   0xe60017 20:00:08:00:0b:01:00:0b 20:00:0c:00:0b:11:00:0f San-po13
vfc6588   43   0xe6002b 20:00:08:00:0c:01:00:07 20:00:0c:00:0c:11:00:07 San-po13
vfc6760   43   0xe60029 20:00:08:ff:0c:01:00:07 20:00:0c:ff:0c:11:00:07 San-po13
vfc6764   43   0xe60027 20:00:08:ff:0c:01:00:06 20:00:0c:ff:0c:11:00:06 San-po13
vfc6920   43   0xe60020 20:00:08:ff:0c:01:00:0f 20:00:0c:ff:0c:11:00:0f San-po13
vfc6926   43   0xe6001f 20:00:08:ff:0c:01:00:0e 20:00:0c:ff:0c:11:00:0e San-po13
vfc6944   43   0xe60028 20:00:08:ff:0c:01:00:04 20:00:0c:ff:0c:11:00:04 San-po13
vfc6950   43   0xe6002a 20:00:08:ff:0c:01:00:05 20:00:0c:ff:0c:11:00:05 San-po13
vfc6962   43   0xe6001c 20:00:08:ff:0c:01:00:0d 20:00:0c:ff:0c:11:00:0d San-po13
vfc6968   43   0xe6001d 20:00:08:ff:0c:01:00:0c 20:00:0c:ff:0c:11:00:0c San-po13
vfc6973   43   0xe60021 20:00:08:ff:0c:01:00:08 20:00:0c:ff:0c:11:00:08 San-po13
vfc6979   43   0xe60022 20:00:08:ff:0c:01:00:09 20:00:0c:ff:0c:11:00:09 San-po13

Total number of flogi = 12.


phx2-dc-ucsprod-B(nxos)# sh npv flogi-table interface vfc6962
--------------------------------------------------------------------------------
SERVER                                                                  EXTERNAL
INTERFACE VSAN FCID             PORT NAME               NODE NAME       INTERFACE
--------------------------------------------------------------------------------
vfc6962   43   0xe6001c 20:00:08:ff:0c:01:00:0d 20:00:0c:ff:0c:11:00:0d San-po13

Total number of flogi = 1
```

6.2 Verify MDS Configuration/Status

1. And we see that it is showing up in the FCNS database on the MDS - in the correct VSAN:

```
phx2-dc-9513-1# sh fcns database

VSAN 43:
--------------------------------------------------------------------------
FCID        TYPE  PWWN                    (VENDOR)        FC4-TYPE:FEATURE
--------------------------------------------------------------------------
0xe60011    N     50:0a:09:83:9d:53:43:54 (NetApp)        scsi-fcp
0xe60017    N     20:00:08:00:0b:01:00:0b                 scsi-fcp:init fc-gs
0xe6001c    N     20:00:08:ff:0c:01:00:0d
0xe6001d    N     20:00:08:ff:0c:01:00:0c
0xe6001f    N     20:00:08:ff:0c:01:00:0e
0xe60020    N     20:00:08:ff:0c:01:00:0f
0xe60021    N     20:00:08:ff:0c:01:00:08                 scsi-fcp fc-gs
0xe60022    N     20:00:08:ff:0c:01:00:09                 scsi-fcp fc-gs
0xe60024    N     20:41:54:7f:ee:0a:9e:00                 229
0xe60025    N     20:dc:00:25:b5:0b:01:df
0xe60026    N     20:dc:00:25:b5:0b:01:ff
0xe60027    N     20:00:08:ff:0c:01:00:06                 scsi-fcp:init
0xe60028    N     20:00:08:ff:0c:01:00:04
0xe60029    N     20:00:08:ff:0c:01:00:07                 scsi-fcp:init
0xe6002a    N     20:00:08:ff:0c:01:00:05
0xe6002b    N     20:00:08:00:0c:01:00:07                 scsi-fcp fc-gs
0xe6002c    N     24:0d:00:0d:ec:e2:85:80 (Cisco)         npv
```

2. A check of the active zoneset shows that the initiator, and the target, are zoned correctly, and logged in.

```
phx2-dc-9513-1# sh zoneset active
zoneset name UCS-Fabric-B vsan 43

<output removed for brevity>

zone name BravoESX01 vsan 43
  * fcid 0xe6001c [pwwn 20:00:08:ff:0c:01:00:0d]
  * fcid 0xe60011 [pwwn 50:0a:09:83:9d:53:43:54]

phx2-dc-9513-1#
```

6.3 Verify the Storage (Data ONTAP)

6.3.1 What are the Target WWPNs?

```
phx2-dc-3120a> fcp show adapters
Slot:                   1a
Description:            Fibre Channel Target Adapter 1a (Dual-channel, QLogic CNA
8112 (8152) rev. 2)
Adapter Type:           Local
Status:                 LINK NOT CONNECTED
FC Nodename:            50:0a:09:80:8d:53:43:54 (500a09808d534354)
FC Portname:            50:0a:09:81:8d:53:43:54 (500a09818d534354)
Standby:                No

Slot:                   1b
Description:            Fibre Channel Target Adapter 1b (Dual-channel, QLogic CNA
8112 (8152) rev. 2)
Adapter Type:           Local
Status:                 LINK DISCONNECTED
FC Nodename:            50:0a:09:80:8d:53:43:54 (500a09808d534354)
FC Portname:            50:0a:09:82:8d:53:43:54 (500a09828d534354)
Standby:                No

Slot:                   0c
Description:            Fibre Channel Target Adapter 0c (Dual-channel, QLogic 2432
(2462) rev. 2)
Adapter Type:           Local
Status:                 ONLINE
FC Nodename:            50:0a:09:80:8d:53:43:54 (500a09808d534354)
FC Portname:            50:0a:09:83:8d:53:43:54 (500a09838d534354)
Standby:                No

Slot:                   0d
Description:            Fibre Channel Target Adapter 0d (Dual-channel, QLogic 2432
(2462) rev. 2)
Adapter Type:           Local
Status:                 LINK NOT CONNECTED
FC Nodename:            50:0a:09:80:8d:53:43:54 (500a09808d534354)
FC Portname:            50:0a:09:84:8d:53:43:54 (500a09848d534354)
Standby:                No
```

6.3.2 What initiators are logged in?

When we get all the way to the end of the “virtual cable” it is clear that the initiator, while connected all of the way through, is failing to login to the target and we are back to the boot policy. It is our fault after all, now we owe the SAN Admin lunch for blaming it on them in the first place ☺

Notice – no Target login, no initiator login on the array, but everything up to this point looks good.

```
phx2-dc-3120a> fcp show initiators
Initiators connected on adapter 1a:
        None connected.

Initiators connected on adapter 1b:
        None connected.

Initiators connected on adapter 0c:
Portname                 Group
--------                 -----
20:00:04:ff:0c:00:00:0c   TrainBravoESX02; TrainBravo_VMFS; TrainMountAll_VMFS
20:00:04:ff:0c:00:00:05   TrainCharlieESX01; TrainMountAll_VMFS
20:00:04:ff:0c:00:00:08   TrainDelta_VMFS; TrainDeltaESX02
20:00:04:ff:0c:00:00:09   TrainDelta_VMFS; TrainDeltaESX01
20:00:04:ff:0c:00:00:04   TrainCharlieESX02; TrainMountAll_VMFS
20:00:04:ff:0c:00:00:0f   TrainAlphaESX01; TrainAlpha_VMFS; TrainMountAll_VMFS
20:00:04:00:0b:00:00:0b

Initiators connected on adapter 0d:
        None connected.

phx2-dc-3120a> igroup show TrainBravoESX01
    TrainBravoESX01 (FCP) (ostype: vmware):
        20:00:08:ff:0c:01:00:0d (not logged in)
        20:00:04:ff:0c:00:00:0d (not logged in)
```

So – everything is correct, after all of this work, we know what we knew already, we screwed up the Boot Policy!

7 Appendix B - Boot Policy Examples

I have included the Boot Policies I used if you are interested in setting these up yourself and don't want to go through the GUI to set up a bunch of them. You can simply modify for your targets and vHBA name and drop them into an SSH session.

7.1 Correct Example

```
scope org /
create boot-policy ValidPolicy
set descr "Example"
set reboot-on-update no
set enforce-vnic-name yes
create virtual-media read-only
set order 0
exit
create storage
create san-image primary
set vhba vHBA-A
create path primary
set wwn 50:0a:09:83:8d:53:43:54
set lun 1
exit
exit
create san-image secondary
set vhba vHBA-B
create path primary
set wwn 50:0a:09:83:9d:53:43:54
set lun 0
exit
exit
set order 2
exit
commit-buffer
top
```

7.2 Typo

```
scope org /
create boot-policy T-TargetTypo
set descr "Typo in the Boot Policy"
set reboot-on-update no
set enforce-vnic-name yes
create virtual-media read-only
set order 1
exit
create storage
create san-image primary
set vhba vHBA-A
create path primary
set wwn 50:0a:09:83:8d:53:43:44
set lun 0
exit
exit
create san-image secondary
set vhba vHBA-B
```

```
create path primary
set wwn 50:0a:09:83:9d:53:43:44
set lun 0
exit
exit
set order 2
commit-buffer
top
```

7.3 Wrong LUN

```
scope org /
create boot-policy T-WrongLUN
set descr "Wrong LUN for testing"
set reboot-on-update no
set enforce-vnic-name yes
create virtual-media read-only
set order 1
exit
create storage
create san-image primary
set vhba vHBA-A
create path primary
set wwn 50:0a:09:83:8d:53:43:54
set lun 1
exit
exit
create san-image secondary
set vhba vHBA-B
create path primary
set wwn 50:0a:09:83:9d:53:43:54
set lun 1
exit
exit
set order 2
exit
commit-buffer
top
```

7.4 Reversed Paths

```
scope org /
create boot-policy T-RevPaths
set descr "Paths are swapped but valid"
set reboot-on-update no
set enforce-vnic-name yes
create virtual-media read-only
set order 1
exit
create storage
create san-image primary
set vhba vHBA-A
create path primary
set wwn 50:0a:09:83:9d:53:43:54
set lun 0
exit
exit
```

```
create san-image secondary
set vhba vHBA-B
create path primary
set wwn 50:0a:09:83:8d:53:43:54
set lun 0
exit
exit
set order 2
exit
commit-buffer
top
```

7.5 vHBA Name not enforced

```
scope org /
create boot-policy T-NoHBACheck
set descr "No check vHBA name wrong"
set reboot-on-update no
set enforce-vnic-name no
create virtual-media read-only
set order 0
exit
create storage
create san-image primary
set vhba vHBA-0
create path primary
set wwn 50:0a:09:83:8d:53:43:54
set lun 0
exit
exit
create san-image secondary
set vhba vHBA-1
create path primary
set wwn 50:0a:09:83:9d:53:43:54
set lun 0
exit
exit
set order 2
exit
commit-buffer
top
```

7.6 No HBA Check, Paths Reversed

```
scope org /
create boot-policy T-RPathNoChk
set descr "No check vHBA name wrong"
set reboot-on-update no
set enforce-vnic-name no
create virtual-media read-only
set order 0
exit
create storage
create san-image primary
set vhba vHBA-0
```

```
create path primary
set wwn 50:0a:09:83:9d:53:43:54
set lun 0
exit
exit
create san-image secondary
set vhba vHBA-1
create path primary
set wwn 50:0a:09:83:8d:53:43:54
set lun 0
exit
exit
set order 2
exit
commit-buffer
top
```

7.7 Reversed Paths

```
scope org /
create boot-policy T-RevPathswLUN
set descr "Paths are swapped but valid"
set reboot-on-update no
set enforce-vnic-name yes
create virtual-media read-only
set order 1
exit
create storage
create san-image primary
set vhba vHBA-A
create path primary
set wwn 50:0a:09:83:9d:53:43:54
set lun 10
exit
exit
create san-image secondary
set vhba vHBA-B
create path primary
set wwn 50:0a:09:83:8d:53:43:54
set lun 10
exit
exit
set order 2
exit
commit-buffer
top
```

8 Appendix C - lunlist issues

I expect this list might grow.

8.1 Multi-Initiator Zones

When the target is in a zone with multiple initiators it looks like lunlist doesn't know how to properly report on the Nameserver information.

The Target is configured correctly, but the zone had 16 initiators in it, 5 of which were logged in. There were also 5 LUNs presented but only the first and the last ones showed up (lines 16,17, 29,30). Even worse, notice that the initiators and targets show up twice on Fabric B (lines 32-43)

```
1  adapter 1/5/1 (top):1# attach-fls
2  adapter 1/5/1 (fls):1# vnic
3  ---- ---- ---- ------- -------
4  vnic ecpu type state   lif
5  ---- ---- ---- ------- -------
6  9    1    fc   active  6
7  10   2    fc   active  7
8  adapter 1/5/1 (fls):2# lunlist 9
9  vnic : 9 lifid: 6
10   - FLOGI State : flogi est (fc_id 0x030002)
11   - PLOGI Sessions
12      - WWNN 50:0a:09:83:8d:53:43:54 WWPN 50:0a:09:83:8d:53:43:54 fc_id 0xac0600
13        - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
14            LUN ID : 0x0001000000000000  access failure
15        - REPORT LUNs Query Response
16            LUN ID : 0x0000000000000000
17            LUN ID : 0x0016000000000000
18   - Nameserver Query Response
19      - WWPN : 50:0a:09:83:8d:53:43:54
20
21 adapter 1/5/1 (fls):3# lunlist 10
22 vnic : 10 lifid: 7
23   - FLOGI State : flogi est (fc_id 0x230004)
24   - PLOGI Sessions
25      - WWNN 50:0a:09:83:9d:53:43:54 WWPN 50:0a:09:83:9d:53:43:54 fc_id 0xe60011
26        - LUN's configured (SCSI Type, Version, Vendor, Serial No.)
27            LUN ID : 0x0001000000000000  access failure
28        - REPORT LUNs Query Response
29            LUN ID : 0x0000000000000000
30            LUN ID : 0x0016000000000000
31   - Nameserver Query Response
32      - WWPN : 20:00:08:ff:0c:01:00:09
33      - WWPN : 20:00:08:ff:0c:01:00:06
34      - WWPN : 20:00:08:ff:0c:01:00:07
35      - WWPN : 20:00:08:ff:0c:01:00:05
36      - WWPN : 20:00:08:ff:0c:01:00:04
37      - WWPN : 50:0a:09:83:9d:53:43:54
38      - WWPN : 20:00:08:ff:0c:01:00:09
39      - WWPN : 20:00:08:ff:0c:01:00:06
40      - WWPN : 20:00:08:ff:0c:01:00:07
41      - WWPN : 20:00:08:ff:0c:01:00:05
42      - WWPN : 20:00:08:ff:0c:01:00:04
43      - WWPN : 50:0a:09:83:9d:53:43:54
```