SNIA DEVELOPER CONFERENCE

SDC 24

BY Developers FOR Developers

September 16-18, 2024
Santa Clara, CA

NVMe-oF™ Boot

NVMe-oF Booting, Industry Update - It's Real!

Doug Farley, Distinguished Engineer, Dell (Co-Chair, NVMe BTG)
Curtis Ballard, Distinguished Technologist, HPE (NVMe BOD)

Agenda

- Why Boot
- Groups: NVM Express, Timberland
- Reasons
- It’s Real! How you can get started NOW
- Evolving – Boot Specification 1.1
- Come help!

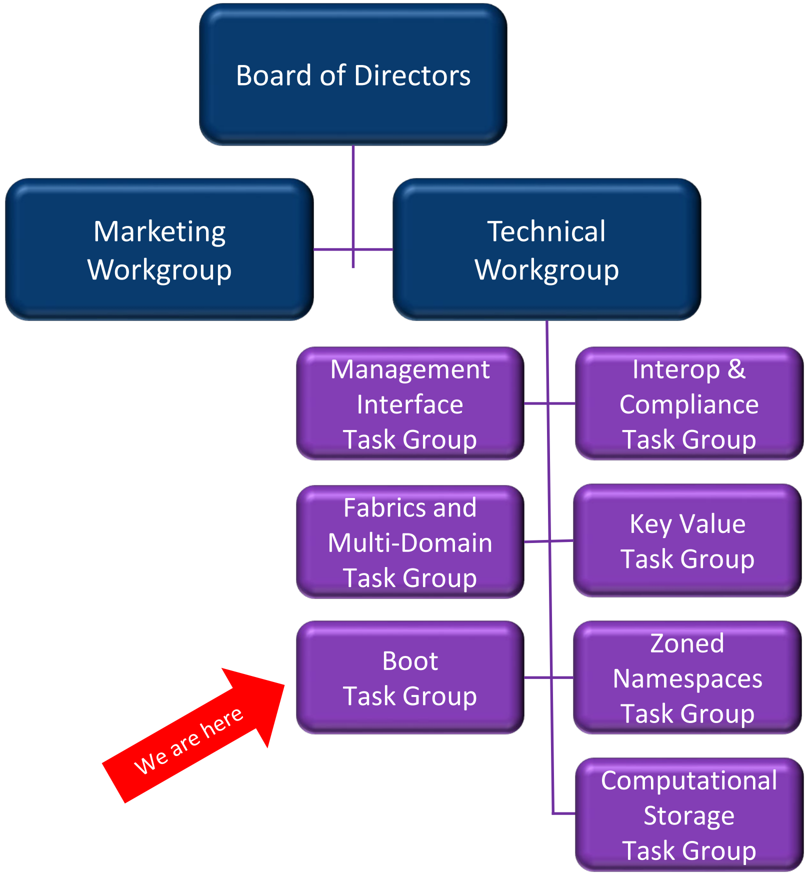
Why Boot

- Boot:
  - Developed by NVM Express, in the Boot Task Group, with 41 active Member Companies participating
  - The NVMe® Boot Specification v1.0 was released in November of 2022
  - The NVMe® Boot Specification v1.1 spec was released in August 2024
    - 1.1 content was entirely based on community feedback on 1.0 in the wild
- But Why Boot?
  - Furthers treating your compute as cattle and not pets
  - Get's you: Modern Orchestration, Diskless; Ultimately Stateless Compute
  - NVMe Boot and instance Storage can coexist

NVM Express Boot Task Group

- Membership: 41 companies

AMD
Avery Design Systems
Beijing MemBlaze Technology
Biwin Semiconductor (HK) Company
Broadcom
DapuStore Corporation
Dell Technologies*
Douyin Vision Co Ltd
FADU
Hewlett Packard Enterprise
Huawei Technologies
IBM
IEIT Systems Co., Ltd
InnoGrit Corp
Intel*
JetIO Technology
Kioxia
Lenovo
LightBits Labs
Marvell Semiconductor
Micron Technology
Microsoft
NVIDIA*
Oracle America
Phison Electronics
Qualcomm Incorporated
Samsung
ScaleFlux
Seagate Technology
Shenzhen Longsys Electronics
Shenzhen Unionmemory Info Sys
Silicon Motion
Solidigm
SUSE
Swissbit AG
Teledyne LeCroy
Toshiba America Electronic Comp.
University of New Hampshire
Western Digital
Wolley Inc.
Yangtze Memory Technologies


Board of Directors
Marketing Workgroup
Technical Workgroup
Management Interface Task Group
Interop & Compliance Task Group
Fabrics and Multi-Domain Task Group
Key Value Task Group
Boot Task Group
Zoned Namespaces Task Group
Computational Storage Task Group
We are here

*NVMe Boot Task Group Co-Chair

Timberland SIG: Public Reference Implementation Based on UEFI

- The Timberland SIG * partnered with NVM Express
- Created reference code for booting over NVMe-oF™ technology, based on:
  - the NVMe Boot Spec 1.0; and
  - open-source frameworks
    - Developed by a subset of NVM Express member companies including:

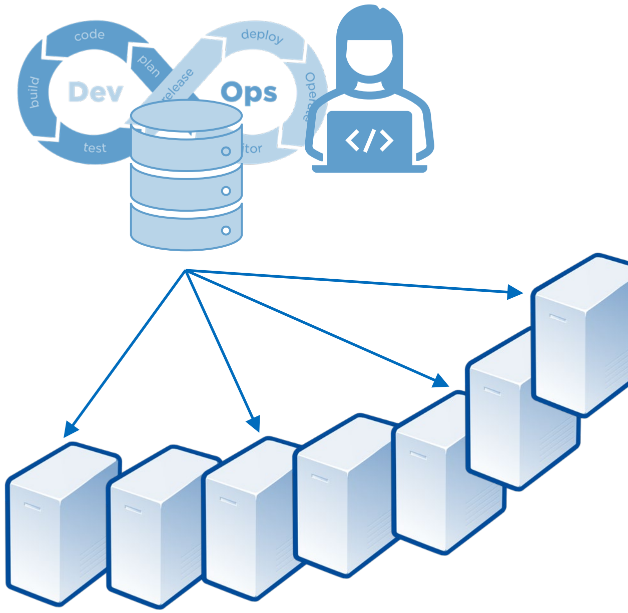
      DELL Technologies NVIDIA intel SUSE Red Hat Hewlett Packard Enterprise vmware by Broadcom
    - Released* and/or upstreamed under BSD-3-Clause (or other open-source license as required by components)

*https://github.com/timberland-sig

Why Boot from Fabrics is More Important in 2024

- Robust IaaS composability using diskless / stateless nodes
  - Provides at-scale solutions for remote management
  - Possible Opex/COGS savings by centralization
- Immutability – clone/redeploy from “golden” base image
- PlatformOps CI/CD and Lifecycle-Management (with Redfish!)
  - Prebuild your images
  - Don’t try to customize full deployments at the last mile when you can do that before a reboot and move the bits only as needed

Maintain Here



Not Here

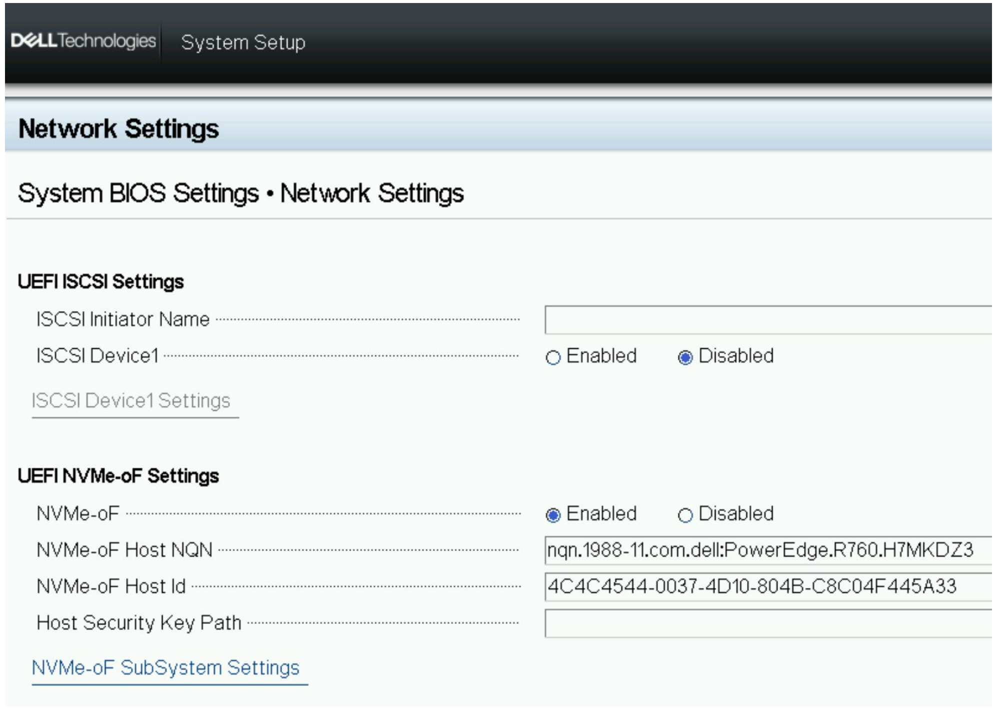
How Can I Use NVMe-oF™ Boot?

A specification and reference code is great,

but how does that affect a SNIA Developer?

It’s Real! How You Can Get Started NOW

- You just need a system that has:
  - UEFI BIOS with NVMe-oF™ Boot and with Redfish integration
- With a live NVM Subsystem setup it’s only 3 commands to boot from an existing image



DELL Technologies System Setup

Network Settings

System BIOS Settings • Network Settings

UEFI ISCSI Settings

ISCSI Initiator Name

ISCSI Device1 ○ Enabled ● Disabled

ISCSI Device1 Settings

UEFI NVMe-oF Settings

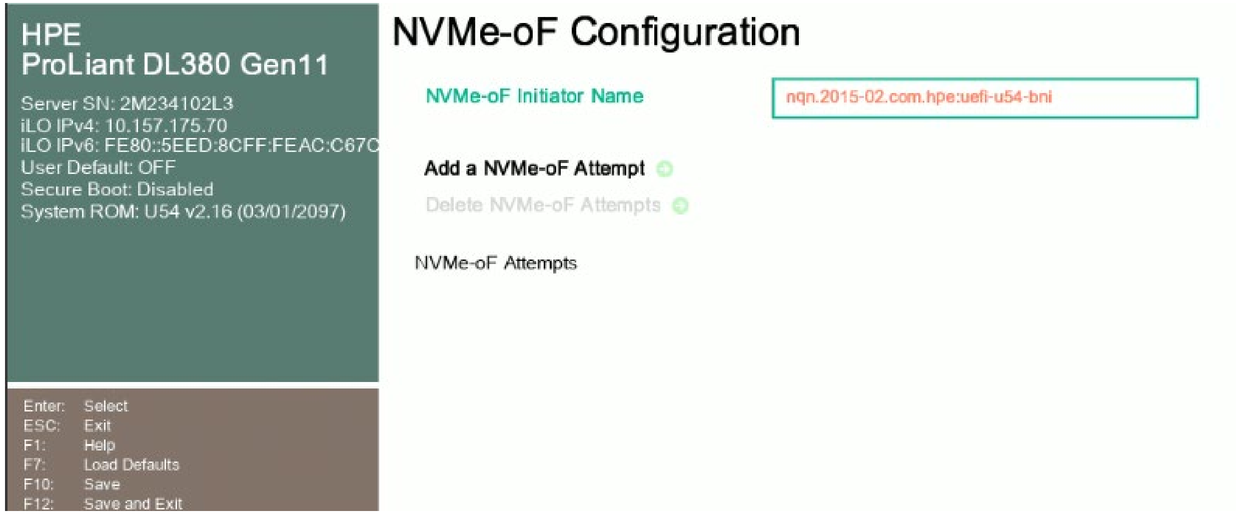
NVMe-oF ● Enabled ○ Disabled

NVMe-oF Host NQN nqn.1988-11.com.dell:PowerEdge.R760.H7MKDZ3

NVMe-oF Host Id 4C4C4544-0037-4D10-804B-C8C04F445A33

Host Security Key Path

NVMe-oF SubSystem Settings



HPE
ProLiant DL380 Gen11

Server SN: 2M234102L3
iLO IPv4: 10.157.175.70
iLO IPv6: FE80::5EED:8CFF:FEAC:C67C
User Default: OFF
Secure Boot: Disabled
System ROM: U54 v2.16 (03/01/2097)

```
Enter:  Select
ESC:    Exit
F1:     Help
F7:     Load Defaults
F10:    Save
F12:    Save and Exit
```

NVMe-oF Configuration

NVMe-oF Initiator Name nqn.2015-02.com.hpe:uefi-u54-bni

Add a NVMe-oF Attempt

Delete NVMe-oF Attempts

NVMe-oF Attempts

Tl;dr for IaaS Fleet Engineering in a Hurry

- Platform and IaaS Engineering Steps:
  - Select your base image, customize
    - This example uses openSUSE Leap 15.6 Minimal
  - Setup a Target Image on a NVMe Subsystem
    - This example uses Linux nvmet
  - Setup Target, get IP address
- Fleet Provisioning
  - Connect to Server BMC with Redfish
  - Setup Server IP, Capture Identity
  - Connect to Target, Reboot your system

3 Commands!

- Set BIOS Attributes

```
# curl -s -u root:calvin -k -X PATCH -H 'Content-Type: application/json' \
     -d '{
       "Attributes": {
           "NvmeofEnDis": "Enabled",
           "NvmeofSubsys1EnDis": "Enabled",
           "NvmeofSubsys1HostIP": "192.168.100.2",
           "NvmeofSubsys1ConInterface": "NIC.Integrated.3-1-1",
           "NvmeofSubsys1HostMask": "255.255.255.0",
           "NvmeofSubsys1Address": "192.168.100.1",
           "NvmeofSubsys1Nqn": "nqn.2014-08.org.nvmexpress:uuid:00112233-4455-6677-8899-
aabbccddeeff"
       }
     }' \
   "https://192.168.100.205/redfish/v1/Systems/System.Embedded.1/Bios/Settings"  | jq .
```

- Queue an “Activate” Job for the BIOS changes, and reboot

```
curl -s -u root:calvin -k -X POST -H 'Content-Type: application/json' \
     -d '{"TargetSettingsURI":"/redfish/v1/Systems/System.Embedded.1/Bios/Settings"}' \
   "https://192.168.100.205/redfish/v1/Managers/iDRAC.Embedded.1/Jobs"  | jq .

curl -s -u root:calvin -k -X POST -H 'Content-Type: application/json' -d '{"ResetType": "ForceRestart", "StartTime": "TIME_NOW" }'
"https://192.168.100.205/redfish/v1/Systems/System.Embedded.1/Actions/ComputerSystem.Reset"  | jq .
```

Where Do We Go From Here?

Evolving – Boot Specification 1.1

- Standardized SMBIOS HostNQN UUIDs displayed in an NBFT
  - TP4126
- Added support for IPV4/IPV6 DHCP Identifiers
  - TP8027
- Improved error codes for common Subsystem Connection failures
  - TP8027
- Clearer language in body and examples!
  - TP6036, ECN116, ECN120, ECN122

Participation

- Your help is welcome

- Places of interest:
  - Authentication Support with TLS and HMAC-CHAP
  - Improvements still needed in the ecosystem code and reference implementations
  - Automation of Server Discovery
  - Future standardization improvements (Transports, Bug Fixes, Examples)

References and Repositories

- **NVM Express® Specifications:** https://nvmexpress.org/specifications/
- **UEFI 2.10 Errata A:** https://uefi.org/specs/UEFI/2.10_A/10_Protocols_Device_Path_Protocol.html
- **ACPI 6.5:** https://uefi.org/specs/ACPI/6.5/05_ACPI_Software_Programming_Model.html
- **Open-Source Software Repos:** https://github.com/timberland-sig

  Note:
  - Most software has been pushed upstream.
  - For all software except edk2 use the latest upstream version.
  - For edk2 use the version off of the Timberland SIG github.
  - Timberland SIG is working towards upstream of edk2

- Full Configuration Walkthrough in Backup

Thank you!

Full Walkthrough
aka: Build your own lab

Select your Target Server That Will Host NVMe-oF™ Architecture

- I started with openSUSE for my Target, named `nvmet-server`
  - First get your image

```
# wget https://download.opensuse.org/repositories/Virtualization:/Appliances:/Images:/openSUSE-Leap-15.6/images/openSUSE-Leap-
15.6-Minimal-VM.x86_64-kvm-and-xen.qcow2
```

  - Then just convert qcow to raw to play nice with nvmet

```
# qemu-img convert -f qcow2 -O raw ./openSUSE-Leap-15.6-Minimal-VM.x86_64-kvm-and-xen.qcow2 openSUSE-Leap-15.6-Minimal-
VM.x86_64.raw
```

  - “Minimal” however is VERY minimal, we need to add some utilities, drivers, etc. To do that, you can just start it in qemu. If you wanted to do this for scale, try kiwi-ng.

```
# qemu-system-x86_64 -nographic -enable-kvm -drive if=pflash,format=raw,readonly=on,file=/usr/share/qemu/ovmf-x86_64-code.bin
-drive if=pflash,format=raw,file=/usr/share/qemu/ovmf-x86_64-vars.bin -drive file=./openSUSE-Leap-15.6-Minimal-
VM.x86_64.raw,format=raw -m 4098 -nic user,model=virtio-net-pci
```

  - Next. Inside Qemu, we’ll add packages.

```
# zypper in --force kernel-default [1]
# zypper in nvme-cli wicked-nbft vim less iputils
# dracut --add network-legacy --add nvmf --force --verbose --add-drivers "mlx5_core ice nvme-fabrics nvme-tcp" [2][3][4]
# shutdown –h now
```

  - You now have a usable image!

[1] Note: This example used the openSUSE Minimal image, which is VERY minimal – we needed additional drivers!

[2] Note: Pay attention to which drivers and or firmware you might need for your platform when building your image.

[3] If you are going to use this on multiple machines, cleanup instance data like /etc/nvme/{hostid,hostnqn} and run virt-sysprep – or build it in a real framework like kiwi-ng

[4] My version of dracut-059 didn’t auto pull in nvme modules, you might need to specify them manually as well

Pre-Orchestration

- Disk is ready:

```
nvmet-server:~/test-endpoint # fdisk -l ./openSUSE-Leap-15.6-Minimal-VM.x86_64.raw
Disk ./openSUSE-Leap-15.6-Minimal-VM.x86_64.raw: 24 GiB, 25769803776 bytes, 50331648 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: C5FD08EF-EE9C-4F52-B75B-C67CDB38BD9E

Device                                      Start      End  Sectors Size Type
./openSUSE-Leap-15.6-Minimal-VM.x86_64.raw1  2048     6143     4096   2M BIOS boot
./openSUSE-Leap-15.6-Minimal-VM.x86_64.raw2  6144    73727    67584  33M EFI System
./openSUSE-Leap-15.6-Minimal-VM.x86_64.raw3 73728 50331614 50257887  24G Linux filesystem
```

- Attach Disk to Loop:

```
nvmet-server:~/test-endpoint # losetup -f -P ./openSUSE-Leap-15.6-Minimal-VM.x86_64.raw
nvmet-server:~/test-endpoint # losetup -l
NAME       SIZELIMIT OFFSET AUTOCLEAR RO BACK-FILE                                                    DIO LOG-SEC
/dev/loop0         0      0         0  0 /root/test-endpoint/openSUSE-Leap-15.6-Minimal-VM.x86_64.raw   0     512
```

Pre-Orchestration

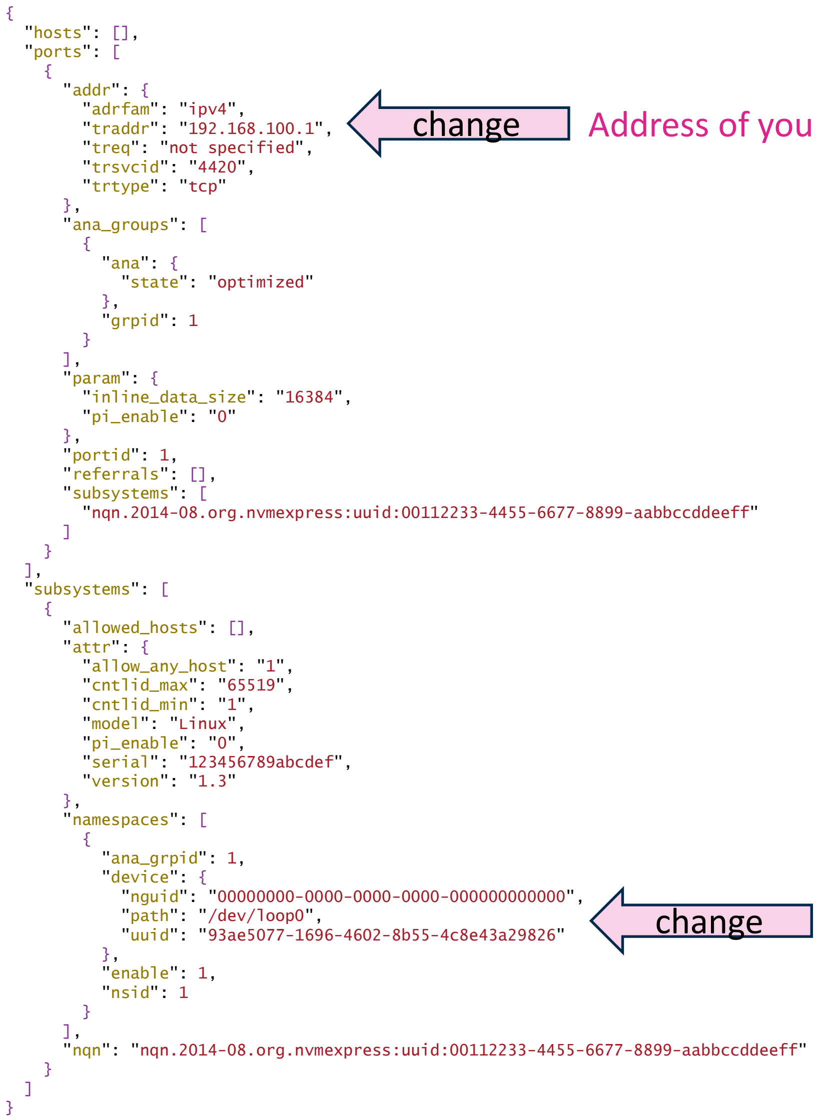
- Fill out a nvmetcli definition for your target.
  - Nvmetcli just interfaces to configfs, can just use natively or use nvmet python libraries
- You might need to load the target drivers

```
# modprobe nvmet
```

- Then reload the config above

```
# nvmetcli restore tcp.json
```

tcp.json



```
{
  "hosts": [],
  "ports": [
    {
      "addr": {
        "adrfam": "ipv4",
        "traddr": "192.168.100.1",
        "treq": "not specified",
        "trsvcid": "4420",
        "trtype": "tcp"
      },
      "ana_groups": [
        {
          "ana": {
            "state": "optimized"
          },
          "grpid": 1
        }
      ],
      "param": {
        "inline_data_size": "16384",
        "pi_enable": "0"
      },
      "portid": 1,
      "referrals": [],
      "subsystems": [
        "nqn.2014-08.org.nvmexpress:uuid:00112233-4455-6677-8899-aabbccddeeff"
      ]
    }
  ],
  "subsystems": [
    {
      "allowed_hosts": [],
      "attr": {
        "allow_any_host": "1",
        "cntlid_max": "65519",
        "cntlid_min": "1",
        "model": "Linux",
        "pi_enable": "0",
        "serial": "123456789abcdef",
        "version": "1.3"
      },
      "namespaces": [
        {
          "ana_grpid": 1,
          "device": {
            "nguid": "00000000-0000-0000-0000-000000000000",
            "path": "/dev/loop0",
            "uuid": "93ae5077-1696-4602-8b55-4c8e43a29826"
          },
          "enable": 1,
          "nsid": 1
        }
      ],
      "nqn": "nqn.2014-08.org.nvmexpress:uuid:00112233-4455-6677-8899-aabbccddeeff"
    }
  ]
}
```

change

Address of your nvmet Server

change

Loop device of .raw from previous step

Pre-Orchestration

- Server is now running locally

```
# nvme discover --transport=tcp --traddr=192.168.100.1 -s 4420

Discovery Log Number of Records 2, Generation counter 8
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified, sq flow control disable supported
portid:  1
trsvcid: 4420
subnqn:  nqn.2014-08.org.nvmexpress.discovery
traddr:  192.168.100.1
eflags:  none
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified, sq flow control disable supported
portid:  1
trsvcid: 4420
subnqn:  nqn.2014-08.org.nvmexpress:uuid:00112233-4455-6677-8899-aabbccddeeff
traddr:  192.168.100.1
eflags:  none
sectype: none
```

```
# nvmetcli ls
o- / ................................................................................................ [...]
  o- hosts .......................................................................................... [...]
  o- ports .......................................................................................... [...]
  | o- 1 ......................... [trtype=tcp, traddr=192.168.100.1, trsvcid=4420, inline_data_size=16384]
  |   o- ana_groups ................................................................................. [...]
  |   | o- 1 ............................................................................ [state=optimized]
  |   o- referrals .................................................................................. [...]
  |   o- subsystems ................................................................................. [...]
  |     o- nqn.2014-08.org.nvmexpress:uuid:00112233-4455-6677-8899-aabbccddeeff ..................... [...]
  o- subsystems ..................................................................................... [...]
    o- nqn.2014-08.org.nvmexpress:uuid:00112233-4455-6677-8899-aabbccddeeff  [version=1.3, allow_any=1,
serial=123456789abcdef]
      o- allowed_hosts .............................................................................. [...]
      o- namespaces ................................................................................. [...]
        o- 1 ............... [path=/dev/loop0, uuid=93ae5077-1696-4602-8b55-4c8e43a29826, grpid=1, enabled]
```

Orchestration

- Setup:
- Victim Server that we want to NVMe®/TCP boot has the following configuration:
  - IP: 192.168.100.2/24, using the NIC in PCIe slot 3, port 1

Orchestration

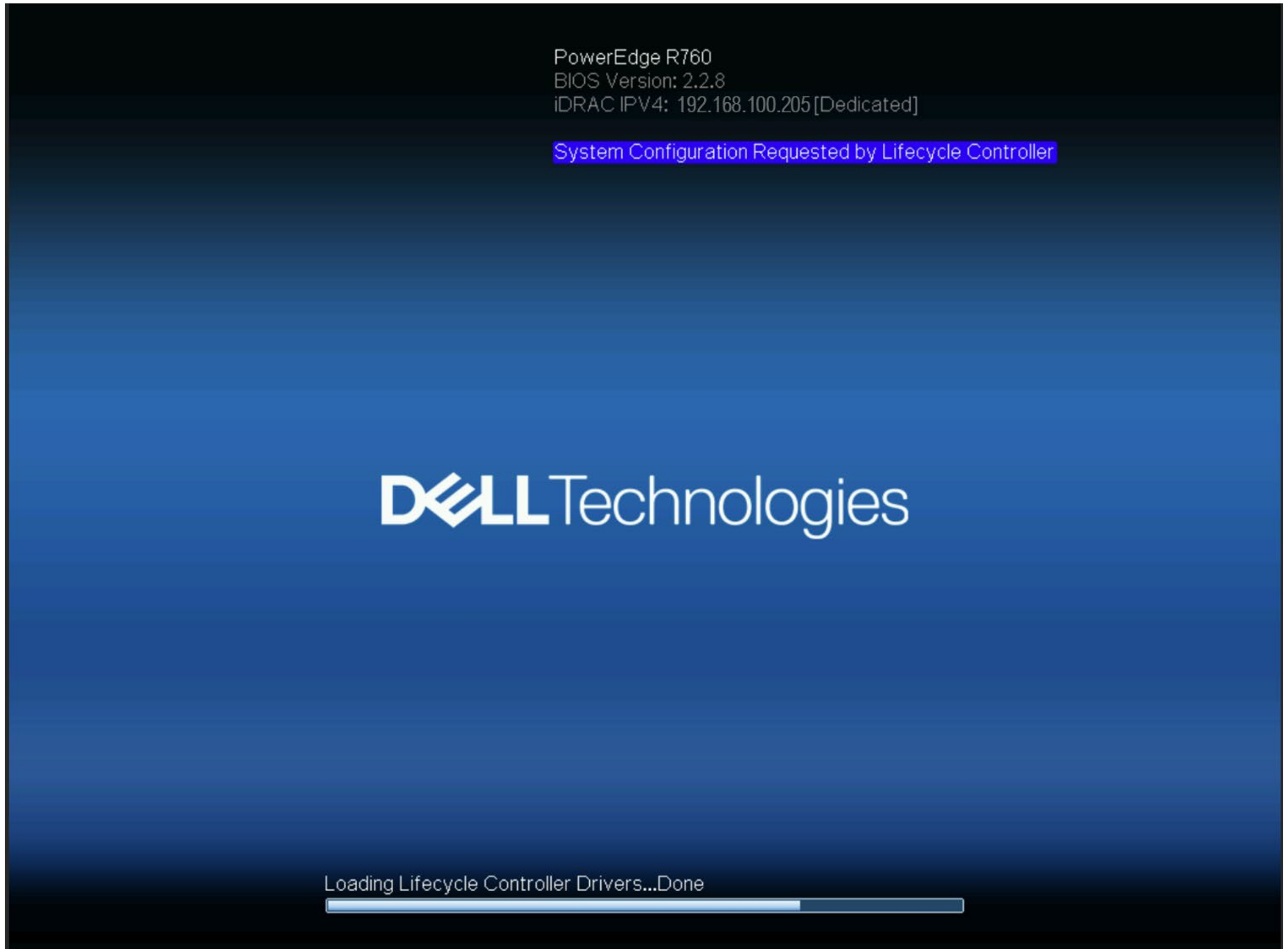
- Set BIOS Attributes

```
# curl -s -u root:calvin -k -X PATCH -H 'Content-Type: application/json' \
     -d '{
       "Attributes": {
           "NvmeofEnDis": "Enabled",
           "NvmeofSubsys1EnDis": "Enabled",
           "NvmeofSubsys1HostIP": "192.168.100.2",
           "NvmeofSubsys1ConInterface": "NIC.Integrated.3-1-1",
           "NvmeofSubsys1HostMask": "255.255.255.0",
           "NvmeofSubsys1Address": "192.168.100.1",
           "NvmeofSubsys1Nqn": "nqn.2014-08.org.nvmexpress:uuid:00112233-4455-6677-8899-
aabbccddeeff"
       }
     }' \
   "https://192.168.100.205/redfish/v1/Systems/System.Embedded.1/Bios/Settings"  | jq .
```

- Queue an “Activate” Job for the BIOS changes, and reboot

```
curl -s -u root:calvin -k -X POST -H 'Content-Type: application/json' \
     -d '{"TargetSettingsURI":"/redfish/v1/Systems/System.Embedded.1/Bios/Settings"}' \
   "https://192.168.100.205/redfish/v1/Managers/iDRAC.Embedded.1/Jobs"  | jq .

curl -s -u root:calvin -k -X POST -H 'Content-Type: application/json' -d '{"ResetType": "ForceRestart", "StartTime": "TIME_NOW" }'
"https://192.168.100.205/redfish/v1/Systems/System.Embedded.1/Actions/ComputerSystem.Reset"  | jq .
```

Server Reboots, in System Re-Configuration Cycle

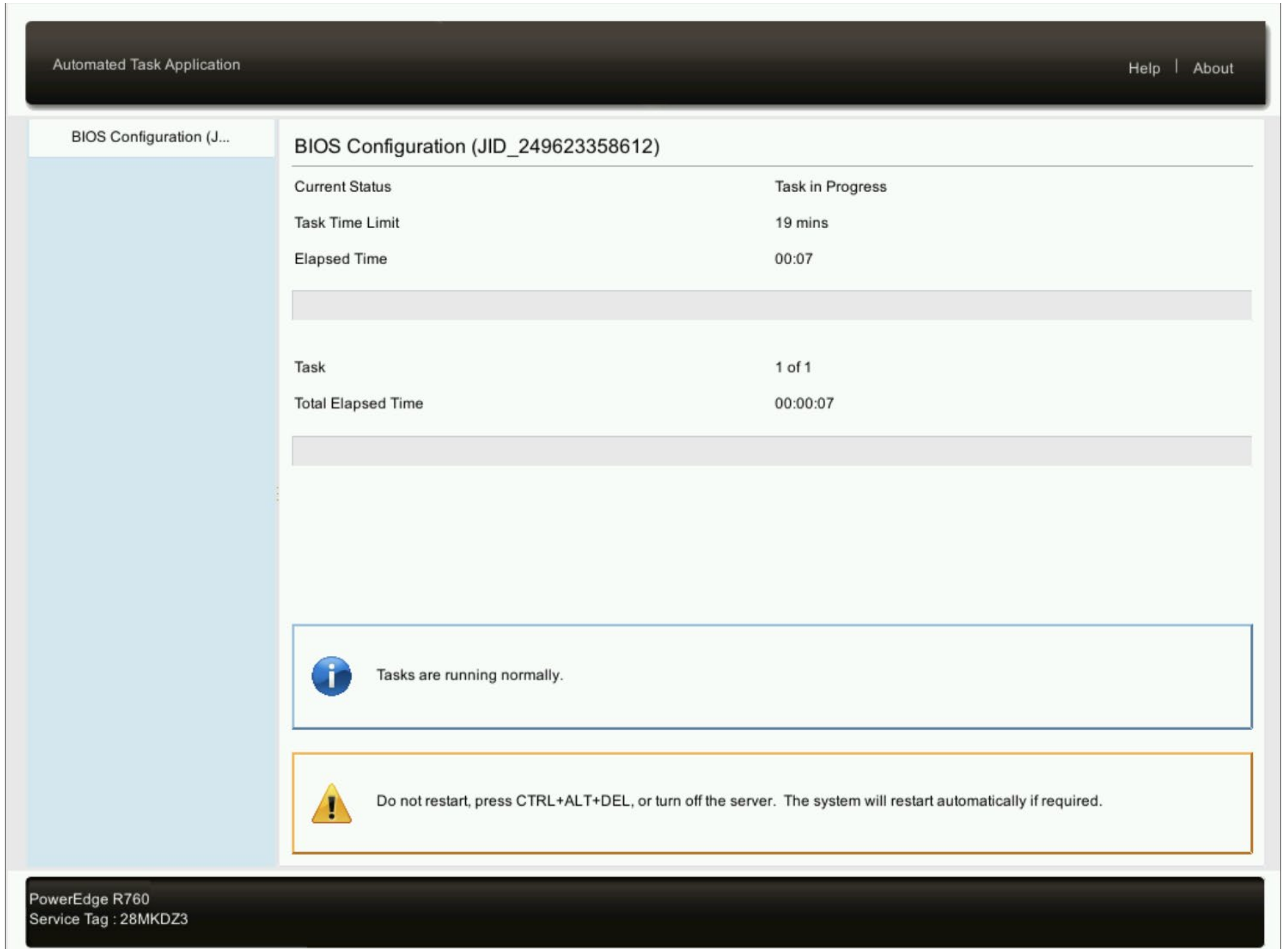


PowerEdge R760
BIOS Version: 2.2.8
iDRAC IPV4: 192.168.100.205 [Dedicated]

System Configuration Requested by Lifecycle Controller

Loading Lifecycle Controller Drivers...Done

In System Re-Configuration Cycle



BIOS Configuration (JID_249623358612)

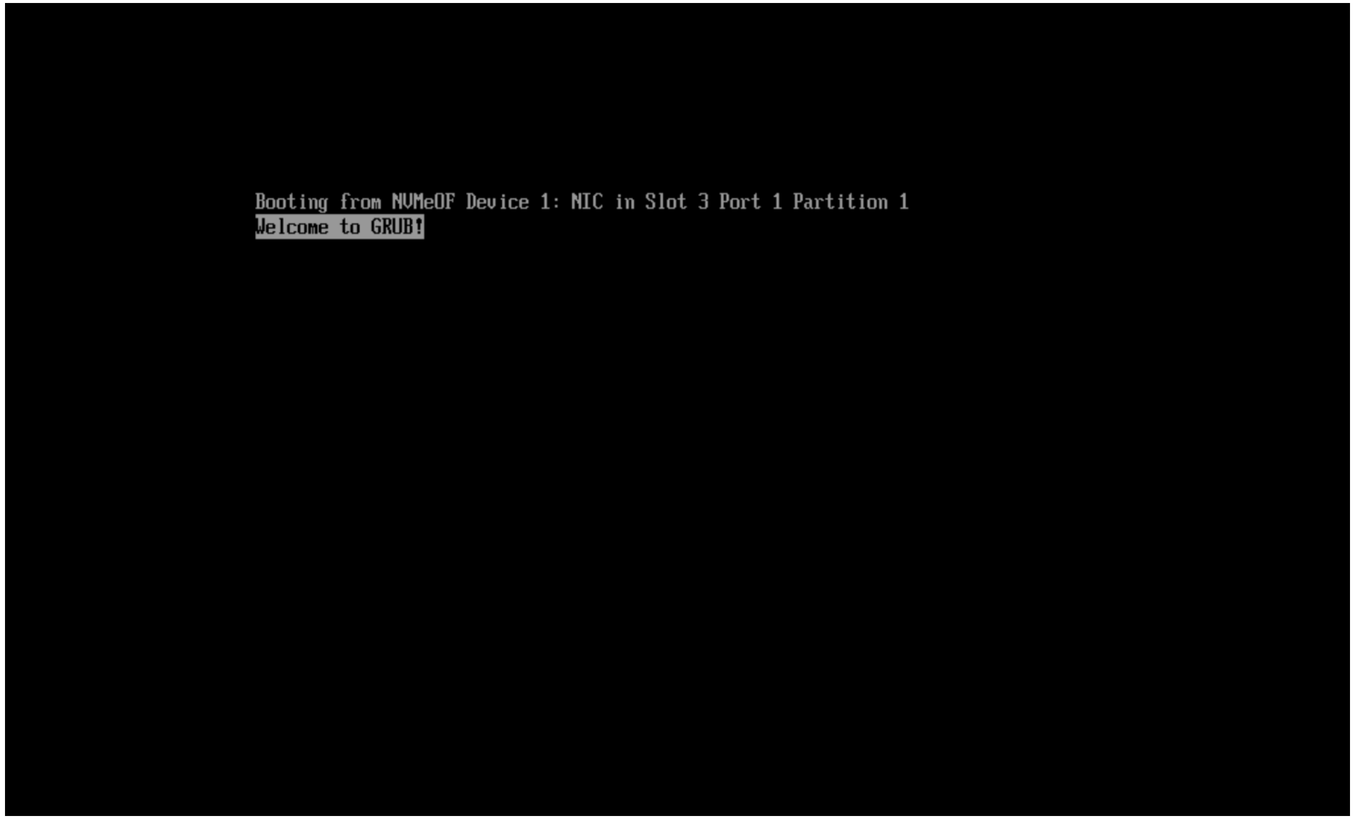
Current Status	Task in Progress
Task Time Limit	19 mins
Elapsed Time	00:07
Task	1 of 1
Total Elapsed Time	00:00:07

Tasks are running normally.

Do not restart, press CTRL+ALT+DEL, or turn off the server. The system will restart automatically if required.

PowerEdge R760
Service Tag : 28MKDZ3

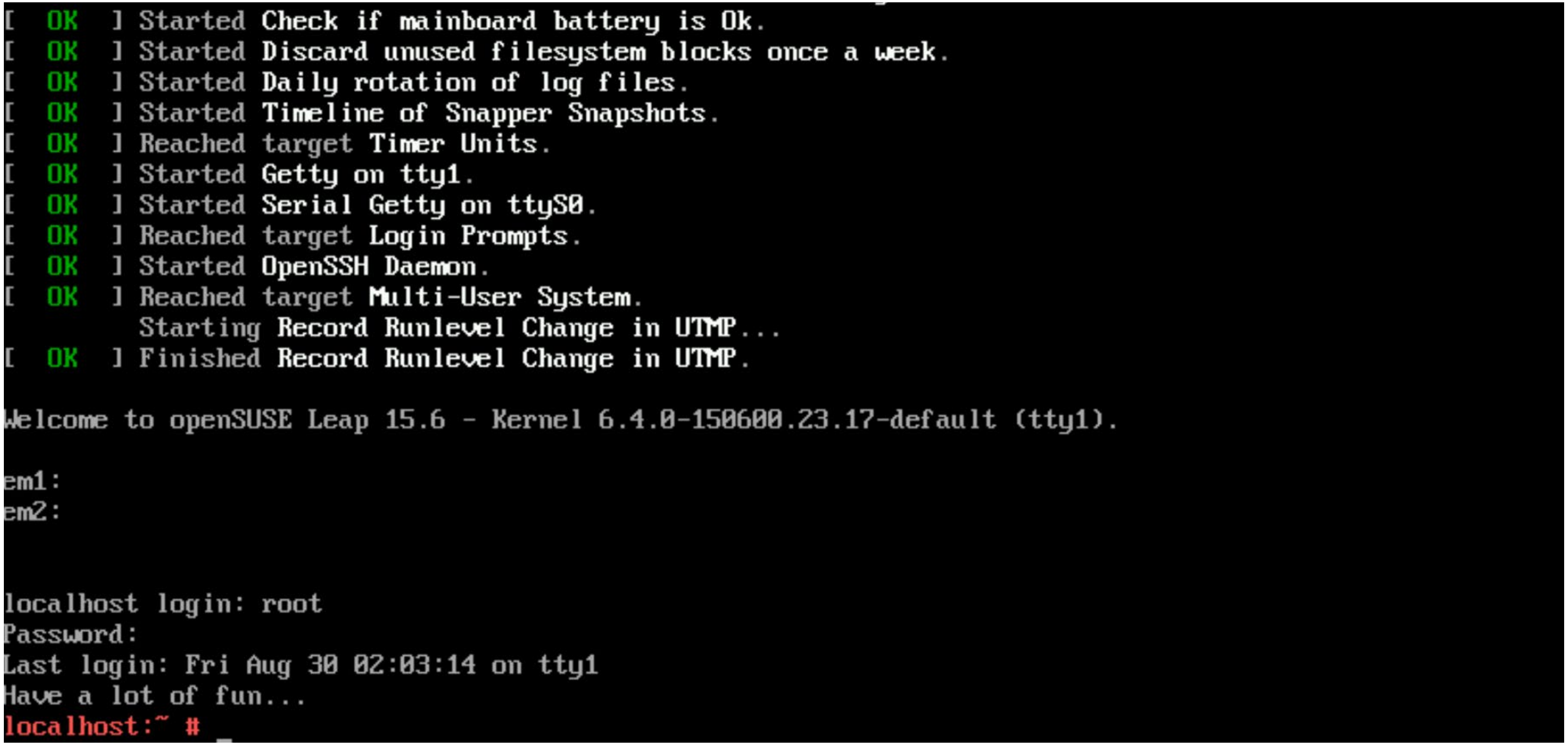
Boot’s into openSUSE GRUB2



```
Booting from NVMeOF Device 1: NIC in Slot 3 Port 1 Partition 1
Welcome to GRUB!
```

Implementers note: Pay attention to your console settings!

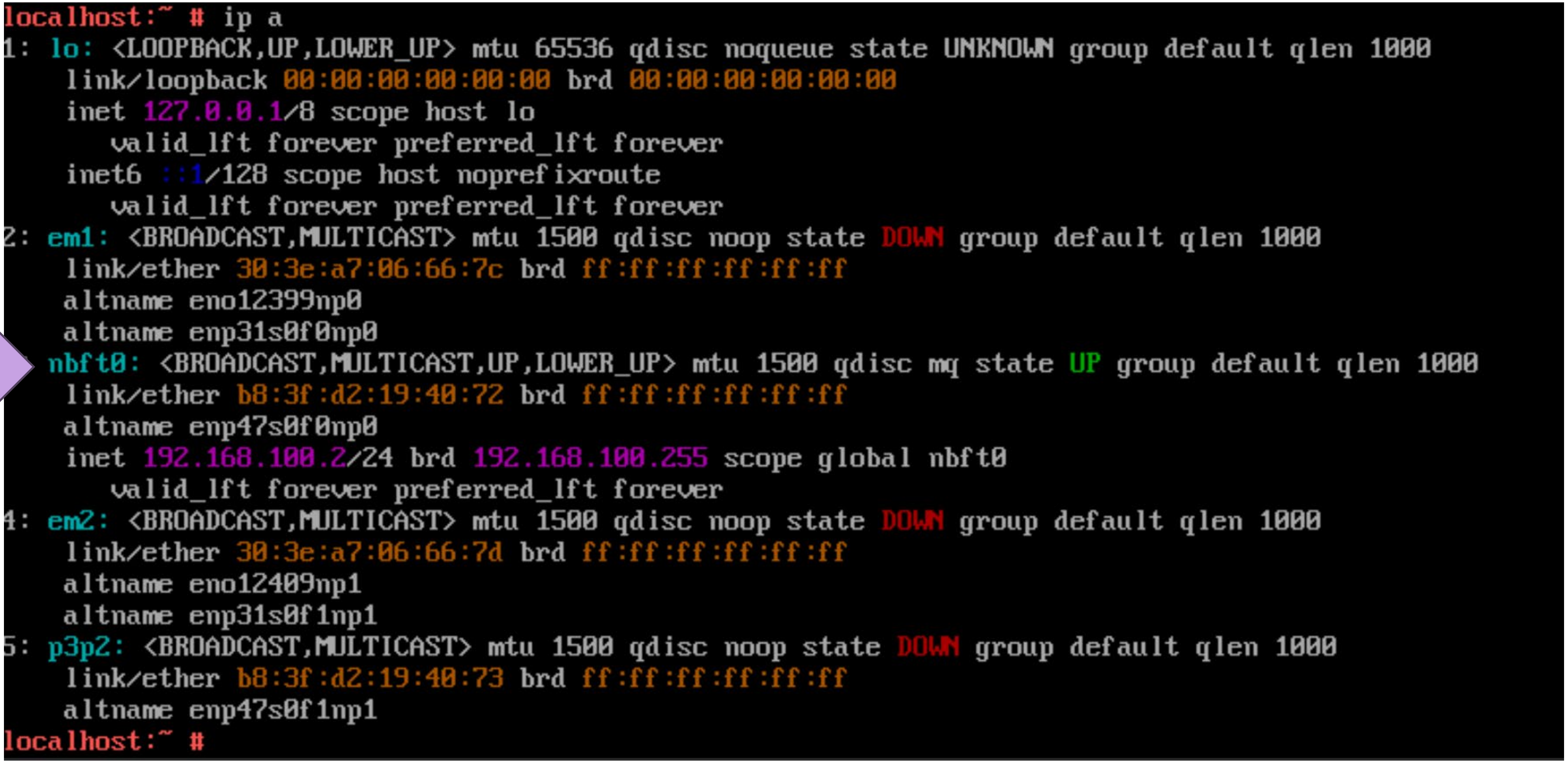
We Made it to a Booted OS!



```
[  OK  ] Started Check if mainboard battery is Ok.
[  OK  ] Started Discard unused filesystem blocks once a week.
[  OK  ] Started Daily rotation of log files.
[  OK  ] Started Timeline of Snapper Snapshots.
[  OK  ] Reached target Timer Units.
[  OK  ] Started Getty on tty1.
[  OK  ] Started Serial Getty on ttyS0.
[  OK  ] Reached target Login Prompts.
[  OK  ] Started OpenSSH Daemon.
[  OK  ] Reached target Multi-User System.
         Starting Record Runlevel Change in UTMP...
[  OK  ] Finished Record Runlevel Change in UTMP.

Welcome to openSUSE Leap 15.6 - Kernel 6.4.0-150600.23.17-default (tty1).

em1:
em2:


localhost login: root
Password:
Last login: Fri Aug 30 02:03:14 on tty1
Have a lot of fun...
localhost:~ # _
```

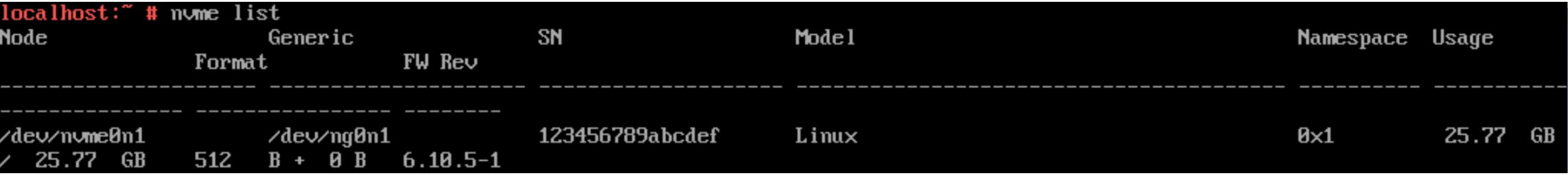
Has nbft0 Adapter from BIOS

nbft0 present

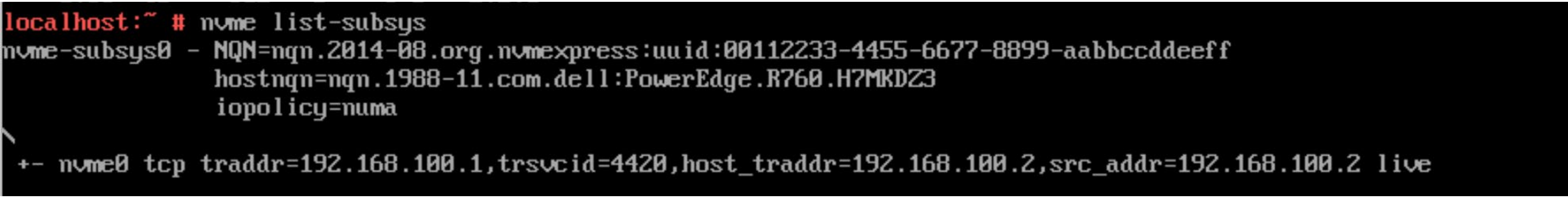


```
localhost:~ # ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: em1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 30:3e:a7:06:66:7c brd ff:ff:ff:ff:ff:ff
    altname eno12399np0
    altname enp31s0f0np0
   nbft0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether b8:3f:d2:19:40:72 brd ff:ff:ff:ff:ff:ff
    altname enp47s0f0np0
    inet 192.168.100.2/24 brd 192.168.100.255 scope global nbft0
       valid_lft forever preferred_lft forever
4: em2: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 30:3e:a7:06:66:7d brd ff:ff:ff:ff:ff:ff
    altname eno12409np1
    altname enp31s0f1np1
5: p3p2: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether b8:3f:d2:19:40:73 brd ff:ff:ff:ff:ff:ff
    altname enp47s0f1np1
localhost:~ #
```

Root Device Attached from Remote TCP Subsystem



```
localhost:~ # nvme list
Node                  Generic               SN                   Model                                    Namespace  Usage
                Format           FW Rev
--------------------- --------------------- -------------------- ---------------------------------------- ---------- ----------
--------------- ---------------- --------
/dev/nvme0n1          /dev/ng0n1            123456789abcdef      Linux                                    0x1         25.77  GB
/  25.77  GB    512   B +  0 B   6.10.5-1
```



```
localhost:~ # nvme list-subsys
nvme-subsys0 - NQN=nqn.2014-08.org.nvmexpress:uuid:00112233-4455-6677-8899-aabbccddeeff
               hostnqn=nqn.1988-11.com.dell:PowerEdge.R760.H7MKDZ3
               iopolicy=numa
\
 +- nvme0 tcp traddr=192.168.100.1,trsvcid=4420,host_traddr=192.168.100.2,src_addr=192.168.100.2 live
```