Boot Mode Considerations: BIOS vs. UEFI

An overview of differences between UEFI Boot Mode and traditional BIOS Boot Mode

Dell Engineering
June 2018

Revisions

Date	Description
October 2017	Initial release
June 2018	Added DHCP Server PXE configuration details.

The information in this publication is provided “as is.” Dell Inc. makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Use, copying, and distribution of any software described in this publication requires an applicable software license.

Dell believes the information in this document is accurate as of its publication date. The information is subject to change without notice.

Table of contents

Executive Summary

Dell EMC servers provide the option of using the traditional BIOS boot mode or UEFI boot mode. The boot mode determines how the system BIOS interacts with adapter card firmware and operating system software. Specific security features and boot mechanisms are available only when the system is configured for UEFI boot mode.

This Dell EMC Deployment and Configuration Guide has two goals. First, it informs readers of the benefits and shortcomings of the two boot modes, so they can choose the boot mode that is best for their environment. Second, this paper provides an overview of the configuration needed to use UEFI boot mode. It is assumed that the reader is familiar with the traditional BIOS boot mode, and likely has existing infrastructure that uses BIOS boot mode. This paper outlines changes needed to support UEFI boot mode in an existing datacenter infrastructure.

1 Introduction

Traditionally, the system BIOS performs initialization, boot, system management, and configuration tasks. The BIOS initializes the system's processors, memory, bus controllers, and I/O devices. After initialization is complete, the BIOS passes control to operating system (OS) software. The OS loader uses basic services provided by the system BIOS to locate and load OS modules into system memory. After booting the system, the BIOS and embedded management controllers execute system management algorithms, which monitor and optimize the condition of the underlying hardware. BIOS configuration settings enable fine-tuning of the performance, power management, and reliability features of the system.

The Unified Extensible Firmware Interface (UEFI) does not change the traditional purposes of the system BIOS. To a large extent, a UEFI-compliant BIOS performs the same initialization, boot, configuration, and management tasks as a traditional BIOS. However, UEFI does change the interfaces and data structures the BIOS uses to interact with I/O device firmware and operating system software. The primary intent of UEFI is to eliminate shortcomings in the traditional BIOS environment, enabling system firmware to continue scaling with industry trends.

Since 2010, Dell EMC has offered servers that support both the traditional BIOS boot mode and UEFI boot mode. However, the system administrator must choose the boot mode before deploying the server to its operating environment. This paper helps system administrators understand the implications of each boot mode. First, the paper explains the limitations of the traditional BIOS that UEFI resolves. Next, it describes functionality that is available in UEFI boot mode that is not available in BIOS boot mode. Finally, the paper provides considerations for deploying a server in UEFI boot mode in the midst of a traditional datacenter infrastructure.

2 Comparing UEFI and Traditional BIOS

This sections explains how UEFI corrects certain shortcomings in traditional BIOS implementations. The UEFI boot mode offers:

- Improved Partitioning scheme for boot media
  - Support for media larger than 2 TB
  - Redundant partition tables
- Flexible handoff from BIOS to OS
- Consolidated firmware user interface
- Enhanced resource allocation for boot device firmware

2.1 Partitioning Scheme for Boot Media

Traditional BIOS implementations use the Master Boot Record (MBR) scheme for partitioning boot media. Because it uses 32-bit addressing and 512-byte blocks, the MBR scheme limits the addressable storage in the boot media to 2 TB. The MBR scheme also limits the number of partitions to four, and expects bootstrap code to reside at specific locations in the media.

UEFI defines an improved partitioning scheme known as a GUID Partition Table (GPT). The GPT scheme uses 64-bit addressing, so the boot media can be much larger than 2 TB. Each entry in the table is identified by a 128-bit Globally Unique Identifier (GUID), so the scheme supports a large number of partitions. Bootstrap code is no longer required at fixed locations, and a backup partition table provides redundancy.

2.2 Handoff from BIOS to Operating System

After performing system initialization, the BIOS attempts to transfer control to an operating system. Traditional BIOS implementations maintain a prioritized list (“boot order”) of bootable media in the system, and attempt to launch boot software according to the list of media. For each entry in the list, the BIOS loads bootstrap code from a well-known location and passes control to it; if the attempt fails, the BIOS attempts subsequent entries in the list.

UEFI implementations also maintain a boot order, but each entry corresponds to an individual file instead of an entire bootable medium. This scheme allows for one medium (such as a hard disk) to contain multiple boot order entries (for example, multiple operating system loaders). Since each entry specifies the location of the boot file, UEFI also supports booting via Uniform Resource Identifiers (URIs).

Unlike traditional BIOS implementations, all bootable files (executable bootstrap images) must be formatted according to the Portable Executable / Common Object File Format (PE/COFF). This requirement applies to any code executed by the BIOS, including device firmware (traditionally called “option ROMs”), pre-boot execution environment (PXE) boot programs, and operating system loaders.

2.3 User Interfaces for Firmware

In a traditional BIOS, each boot device provides a separate user interface for its configuration settings. For example, a network boot device provides one interface for PXE settings, and a storage controller provides a separate interface for hard-disk or RAID configuration. Most boot devices require a system reboot after any firmware change, so system configuration requires multiple boots.

UEFI defines a shared user interface known as the Human Interface Infrastructure (HII). A user can configure all the firmware settings - including BIOS, onboard management controller, and boot devices – using a single user interface, without needing a reboot between changes for each device. HII also facilitates remote configuration of all firmware settings via baseboard management controller interfaces.

2.4 Resource Allocation for Boot Device Firmware

Traditional BIOS implementations offer limited memory space for boot device firmware. Boot devices such as storage controllers and network interface controllers require increasing amounts of memory to execute their firmware during the boot process. When a system contains multiple boot devices, a traditional BIOS may not allocate enough memory space for all of the device firmware to execute.

UEFI eliminates this limitation by defining standard interfaces for memory management. In UEFI boot mode, boot devices use these interfaces to request memory space from the BIOS memory manager. When a system contains multiple boot devices, UEFI boot mode allocates memory on-demand for each device's firmware.

3 Features Requiring UEFI Boot Mode

As UEFI grows in popularity, modern capabilities are implemented natively for UEFI Boot Mode instead of BIOS Boot Mode. In Dell EMC servers, the following features are available only in UEFI Boot Mode:

- UEFI Secure Boot
- Boot to Non-Volatile Memory Express (NVMe) devices
- Boot to Uniform Resource Identifier (URI)

3.1 UEFI Secure Boot

Most traditional BIOS implementations do not include mechanisms that verify the integrity of non-BIOS code modules (such as I/O device firmware or operating system loaders). A traditional BIOS may offer protection for the non-volatile memory where the BIOS code is stored, as well as defenses against unauthorized configuration changes. However, these implementations vary between vendors, and risks associated with each implementation may be difficult to assess.

UEFI defines a mechanism, named Secure Boot, which verifies the integrity of each pre-boot code module and allows only authorized code modules to execute. Users configure a Secure Boot Policy consisting of X.509 certificates and hash values for both authorized and unauthorized entities. The system BIOS enforces this policy when determining whether to execute pre-boot software including I/O device firmware and operating system loaders.

For more information on Dell's Secure Boot implementation and configuring the Secure Boot Policy, see the following documents:
Defining a Secure Boot Policy (Dell TechCenter)
Secure Boot Management on 14G Dell EMC PowerEdge Servers (Dell TechCenter)

3.2 Boot to Non-Volatile Memory Express (NVMe) Devices

Non-Volatile Memory Express (NVMe) refers to an interface for accessing non-volatile storage connected by PCI Express. In Dell EMC PowerEdge servers (beginning with the 13th generation), the NVMe boot firmware is developed by Dell as part of the BIOS firmware, instead of developed by individual NVMe device vendors. Dell EMC servers support booting to NVMe devices only when the server is configured for UEFI boot mode.

3.3 Boot to Uniform Resource Identifier (URI)

A Uniform Resource Identifier (URI) is a character string that a system can use to access a file. For example, the URI "http://mydomain.org/img/bootimage.efi" indicates that a file named "bootimage.efi" can be accessed using Hypertext Transfer Protocol (HTTP) at mydomain.org.

The Dell EMC PowerEdge BIOS supports booting to URIs only in UEFI boot mode. The bootable URI must use the HTTP protocol. Also, the bootable URI must refer to an .EFI image (PE/COFF format). The Boot URI can be configured in the System Setup utility or via remote management interfaces such as RACADM.

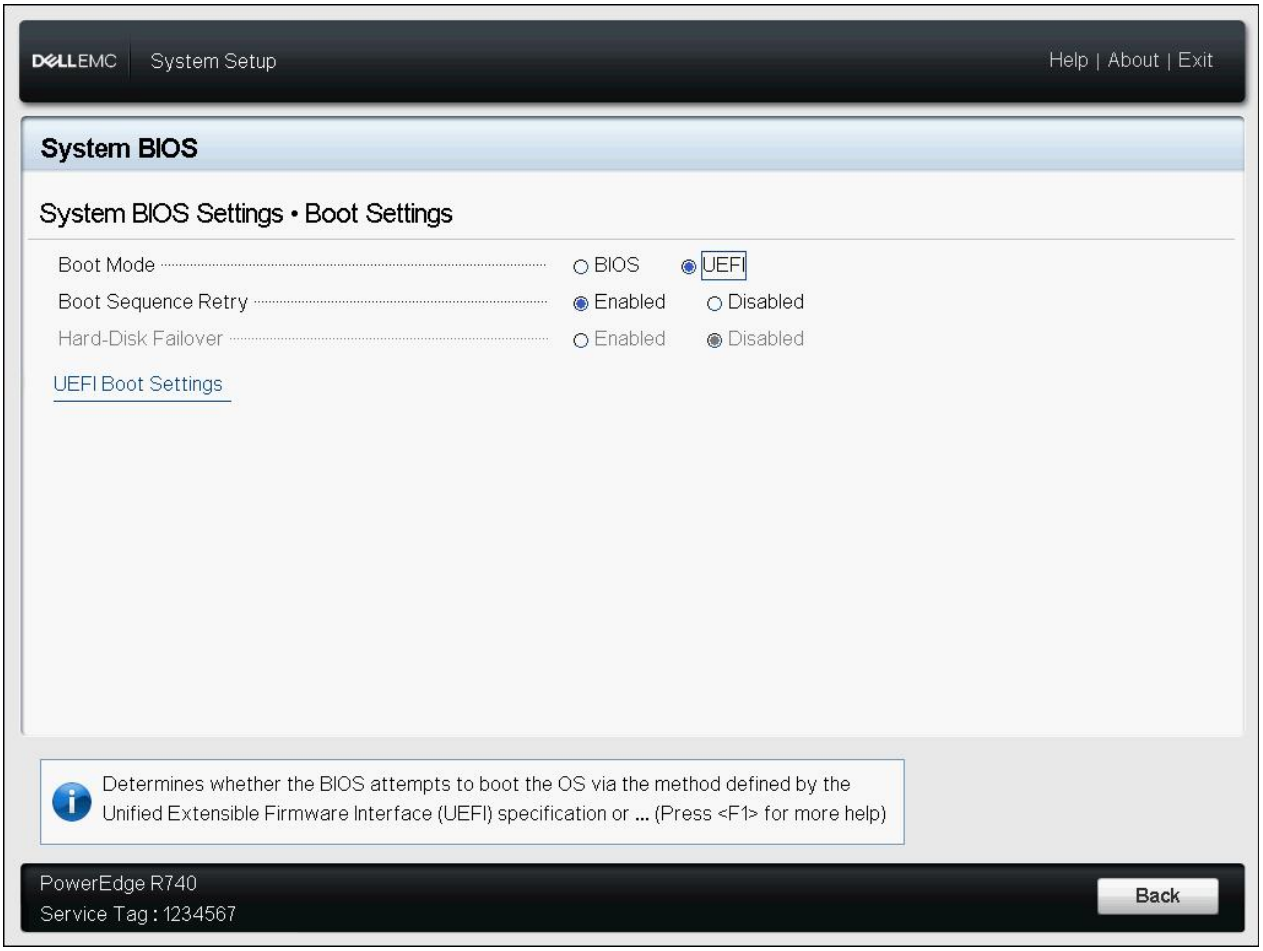
4 Configuration Settings for UEFI Boot Mode

This section provides an overview of the configuration changes needed to operate a Dell EMC PowerEdge server in UEFI Boot Mode. It is assumed that the reader is familiar with the traditional BIOS boot mode, and likely has existing infrastructure that uses BIOS boot mode.

Unless otherwise noted, all configuration settings are accessible through integrated Dell Remote Access Controller (iDRAC) interfaces such as RACADM, or locally through the System Setup utility. IDRAC interface documentation can be found at http://www.delltechcenter.com/iDRAC. The System Setup utility is accessed by pressing F2 at the prompt shown during the system boot process.

4.1 UEFI Boot Settings

The “Boot Mode” setting controls whether the system boots in the traditional BIOS mode or in UEFI mode. In System Setup, enter System BIOS > Boot Settings and set Boot Mode to UEFI (see Figure 1). In RACADM, set the BootMode attribute to UEFI (see Figure 2).



System BIOS

System BIOS Settings • Boot Settings

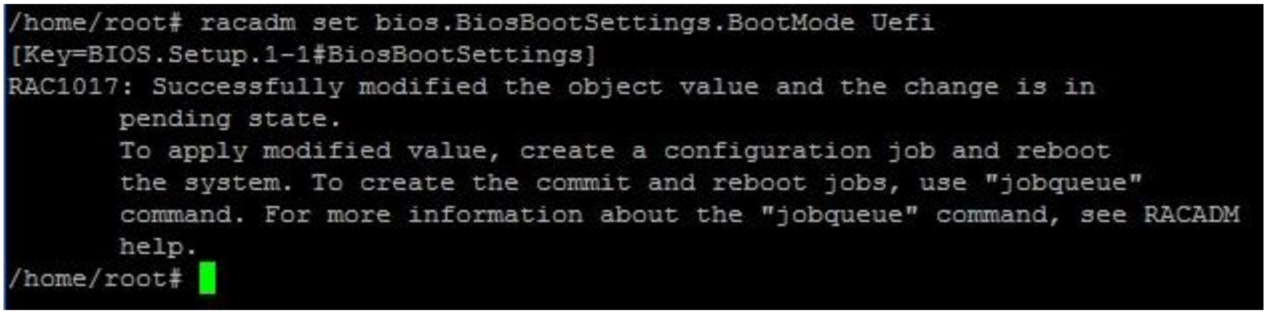
Boot Mode ○ BIOS ● UEFI
Boot Sequence Retry ● Enabled ○ Disabled
Hard-Disk Failover ○ Enabled ● Disabled

UEFI Boot Settings

Determines whether the BIOS attempts to boot the OS via the method defined by the Unified Extensible Firmware Interface (UEFI) specification or ... (Press <F1> for more help)

PowerEdge R740
Service Tag : 1234567

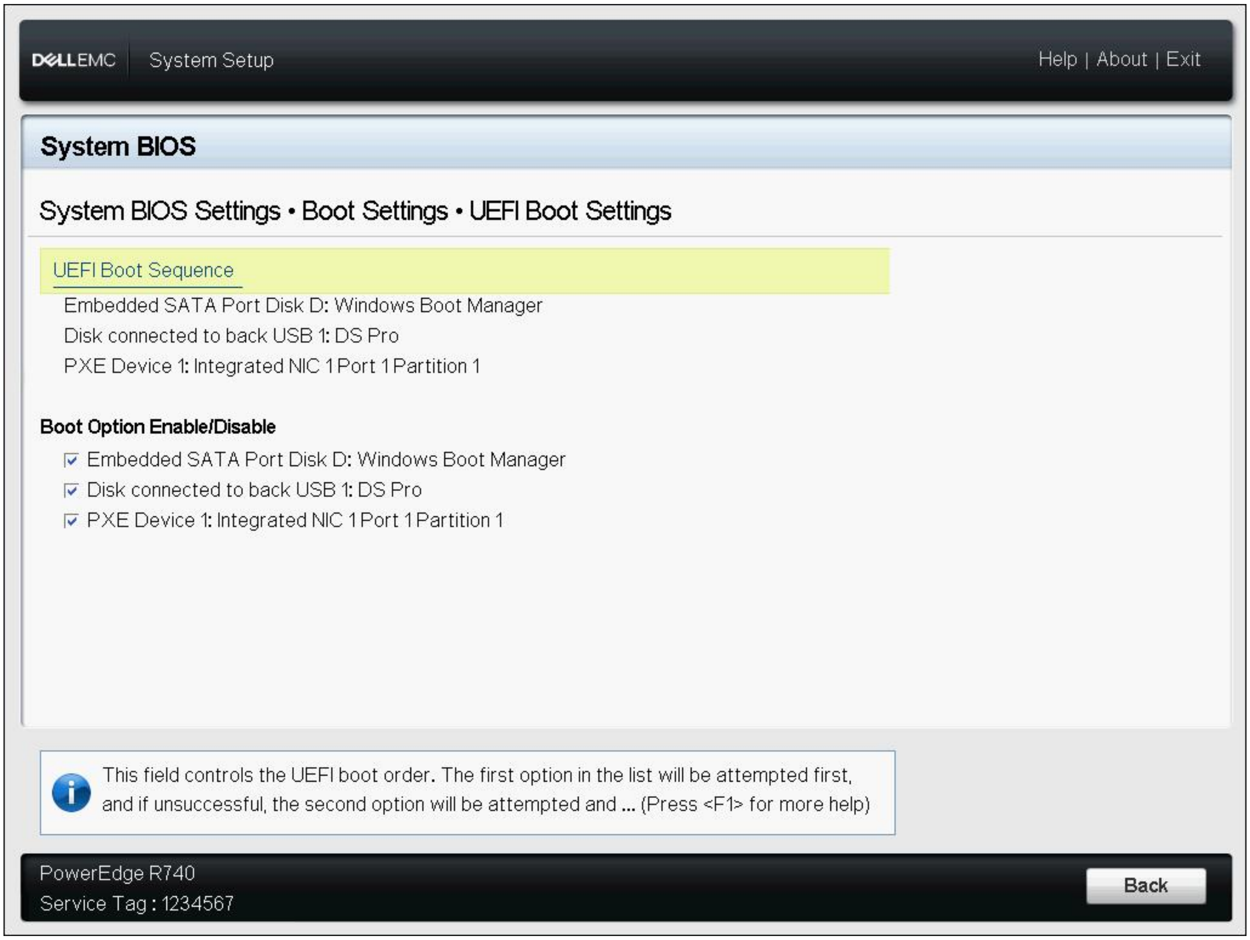
Figure 1 Setting Boot Mode in System Setup.



```
/home/root# racadm set bios.BiosBootSettings.BootMode Uefi
[Key=BIOS.Setup.1-1#BiosBootSettings]
RAC1017: Successfully modified the object value and the change is in
       pending state.
       To apply modified value, create a configuration job and reboot
       the system. To create the commit and reboot jobs, use "jobqueue"
       command. For more information about the "jobqueue" command, see RACADM
       help.
/home/root# █
```

Figure 2 Setting Boot Mode in RACADM.

When the system powers on with Boot Mode set to UEFI, the BIOS provides a list of available UEFI boot options. An administrator can view and edit the order of UEFI boot options. In System Setup, enter System BIOS > Boot Settings > UEFI Boot Settings (see Figure 3). Select UEFI Boot Sequence to edit the boot order. To disable specific boot options without changing the order, uncheck the desired options in the “Boot Option Enable/Disable” section on this page.



System BIOS Settings • Boot Settings • UEFI Boot Settings

UEFI Boot Sequence

Embedded SATA Port Disk D: Windows Boot Manager
Disk connected to back USB 1: DS Pro
PXE Device 1: Integrated NIC 1 Port 1 Partition 1

Boot Option Enable/Disable

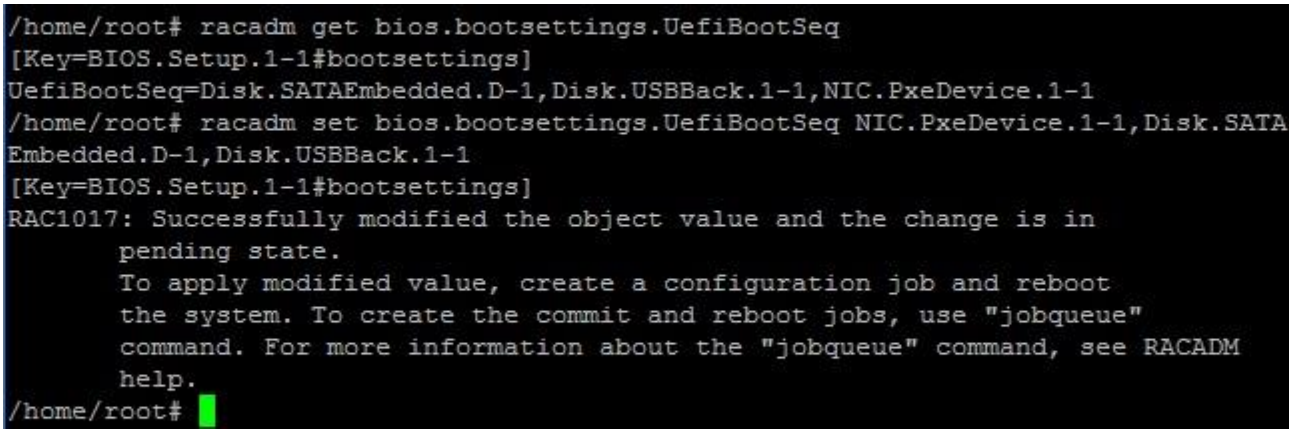
☑ Embedded SATA Port Disk D: Windows Boot Manager
☑ Disk connected to back USB 1: DS Pro
☑ PXE Device 1: Integrated NIC 1 Port 1 Partition 1

This field controls the UEFI boot order. The first option in the list will be attempted first, and if unsuccessful, the second option will be attempted and ... (Press <F1> for more help)

PowerEdge R740
Service Tag : 1234567

Figure 3 UEFI Boot Order Configuration.

In RACADM, the “UefiBootSeq” attribute controls the UEFI Boot Order. Figure 4 shows an example of moving the PXE boot device to the beginning of the boot order.



```
/home/root# racadm get bios.bootsettings.UefiBootSeq
[Key=BIOS.Setup.1-1#bootsettings]
UefiBootSeq=Disk.SATAEmbedded.D-1,Disk.USBBack.1-1,NIC.PxeDevice.1-1
/home/root# racadm set bios.bootsettings.UefiBootSeq NIC.PxeDevice.1-1,Disk.SATA
Embedded.D-1,Disk.USBBack.1-1
[Key=BIOS.Setup.1-1#bootsettings]
RAC1017: Successfully modified the object value and the change is in
       pending state.
       To apply modified value, create a configuration job and reboot
       the system. To create the commit and reboot jobs, use "jobqueue"
       command. For more information about the "jobqueue" command, see RACADM
       help.
/home/root#
```

Figure 4 UEFI Boot Order Configuration in RACADM.

4.2 UEFI Boot from Local Media

As a general rule, operating systems installed in a traditional BIOS environment will not be bootable in UEFI boot mode. There are no reliable means for converting or upgrading traditional bootable media to a UEFI-bootable form, other than re-installing the operating system when the BIOS is in UEFI boot mode.

The boot mode must be configured before installing operating systems or other bootable software. Operating system installers detect the current boot mode and provide tools for formatting the media accordingly. If the boot mode is UEFI, the installer will format the media using the GPT partitioning scheme. If the boot mode is BIOS, the installer uses the traditional MBR scheme. Operating systems also specify different boot loaders for the two boot modes.

4.3 UEFI PXE Boot Configuration

PXE is used to execute an operating system’s bootstrap program using a network connection. The PXE Client sends a DHCP request with PXE specific options. The DHCP server response contains the Network Bootstrap Program (NBP) filename and a list of TFTP boot servers. The PXE client downloads the NBP and then executes it to complete the boot process.

These are the primary differences between UEFI PXE and Legacy PXE:

- In UEFI boot mode the Network Bootstrap Program (NBP) must be a UEFI bootable image (PE/COFF format).
- If UEFI PXE Boot is being used to install an OS, that OS will be installed in UEFI boot mode. If the boot mode is changed later, the OS must be re-installed in the new boot mode.
- If a chainloader like iPXE is used it can take advantage of the Universal Network Device Interface (UNDI) embedded in the NIC to support network adapters that would not be supported in legacy mode.
- Because of the additional structure and security of UEFI, it will take a little longer to load the NBP.
- Legacy PXE firmware (option ROMs) on NICs may support options beyond PXE like IPv4/IPv6 HTTP boot and iSCSI boot. In UEFI boot mode, equivalent functionality is available through HTTP boot and

iSCSI boot configuration (see the UEFI HTTP boot and UEFI iSCSI boot sections in this paper for details).

Minimal changes are required for the PXE server and PXE client when transitioning from BIOS boot mode to UEFI boot mode. The following sections describe these changes.

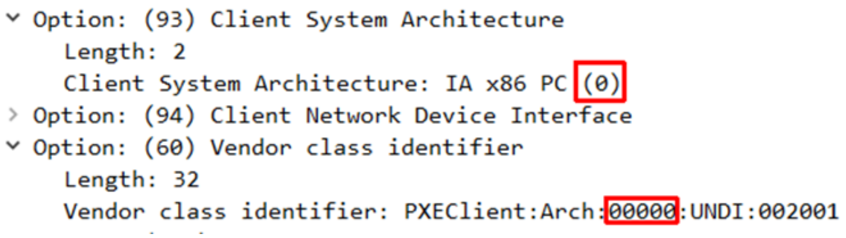
4.3.1 PXE Server Configuration

PXE server setup involves configuration of the DHCP server and boot server (a.k.a. TFTP server).

In UEFI boot mode the Network Bootstrap Program (NBP) must be a UEFI bootable image (PE/COFF format). For Linux environments, UEFI-capable NBPs include ELILO, grub2, and syslinux. Windows environments (Windows Server 2012 and later) use bootmgfw.efi. Alternatively, Windows Deployment Services (WDS) offers PXE server configuration capabilities for UEFI-based PXE clients.

4.3.2 DHCP Server Configuration for UEFI PXE

To support both legacy PXE and UEFI PXE in the same network, the DHCP server must supply different NBPs based on the Architecture type (RFC 4578) in the client's DHCP request. If the client sends Architecture type 0 ("Intel x86 PC") the system is in legacy boot mode. If it sends type 6, it is in UEFI 32-bit boot mode, but if it sends type 7, 8 or 9 it is in UEFI 64-bit boot mode. The architecture type will be sent in Option 93 and as part of the string in Option 60 ("Vendor class identifier"). The reason it is included in both options is backwards compatibility with older DHCP servers that do not support Option 93. The following is from a Wireshark capture of a PXE boot DHCP Discover:



```
⌄ Option: (93) Client System Architecture
    Length: 2
    Client System Architecture: IA x86 PC (0)
› Option: (94) Client Network Device Interface
⌄ Option: (60) Vendor class identifier
    Length: 32
    Vendor class identifier: PXEClient:Arch:00000:UNDI:002001
```

The following is an example of Linux DHCP server configuration that replies with different NBP files based on the Architecture type. Note that the Architecture type is encoded in the first 20 characters of the vendor class identifier:

```
subnet … {
    …
    class "UEFI64-7" {
    match if substring(option vendor-class-identifier, 0, 20) = "PXEClient:Arch:00007";
     filename "ipxe.efi";
    }
    class "UEFI64-8" {
    match if substring(option vendor-class-identifier, 0, 20) = "PXEClient:Arch:00008";
     filename "ipxe.efi";
    }
    class "UEFI64-9" {
    match if substring(option vendor-class-identifier, 0, 20) = "PXEClient:Arch:00009";
     filename "ipxe.efi";
    }
    class "Legacy" {
    match if substring(option vendor-class-identifier, 0, 20) = "PXEClient:Arch:00000";
```

```
        filename "undionly.kkpxe";
        }

    }
```

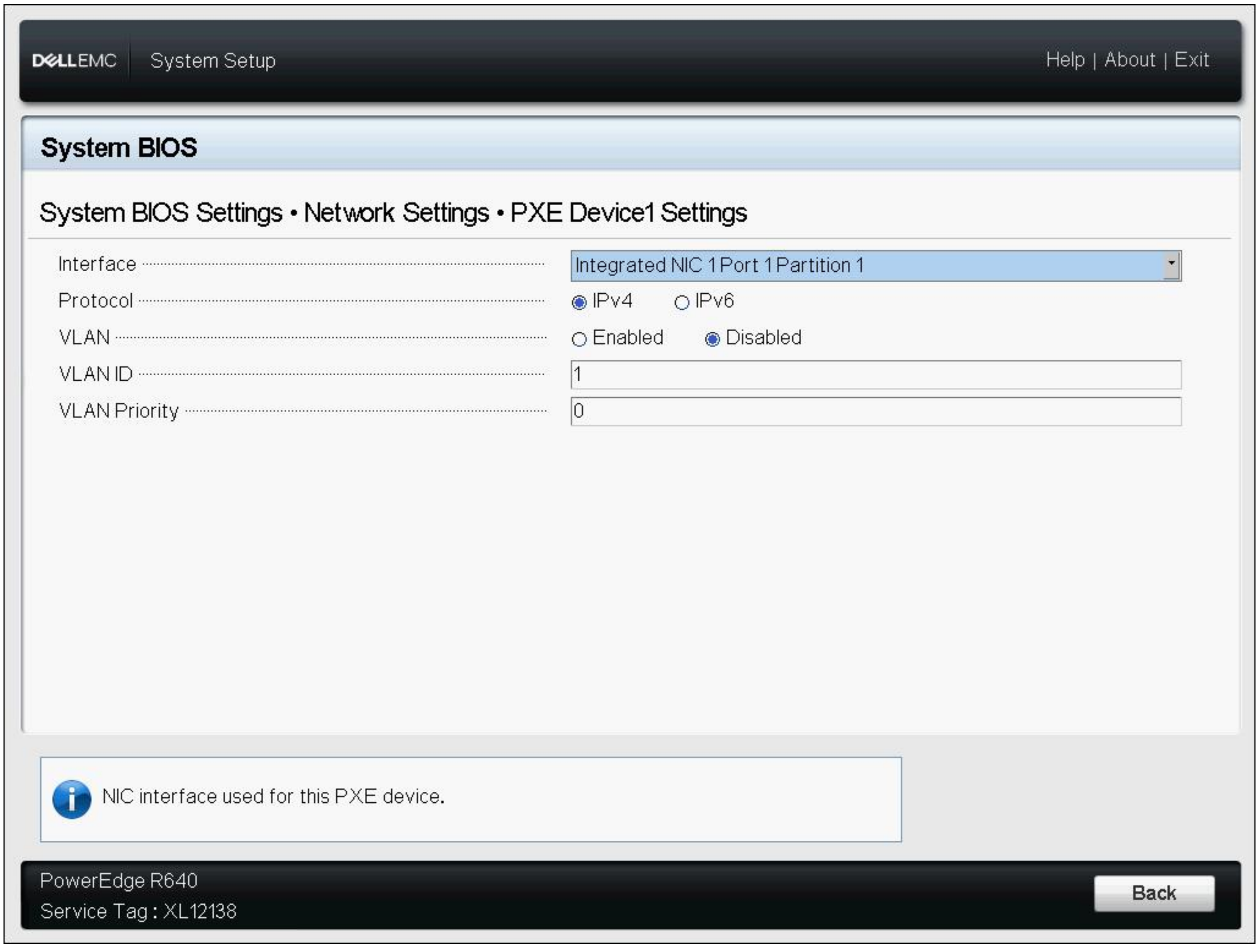
For detailed information on how to configure your DHCP server please consult the user’s guide for the DHCP server.

NOTE: Depending on the DHCP server configuration, the client may need to be configured for UEFI HTTP boot or UEFI iSCSI boot instead of PXE boot. If your current legacy boot DHCP server configuration contains an option 67 (filename) that is a URL instead of a simple filename, you need to configure the client UEFI HTTP boot settings instead of PXE boot. If you are using DHCPv6 option 59 (bootfile URL) in legacy mode you need to configure client UEFI HTTP boot settings instead of PXE boot. If you are using Option 17 (RootPath) you need to configure client UEFI iSCSI boot settings instead of PXE boot. See the UEFI HTTP boot and UEFI iSCSI boot sections in this paper for details.

4.3.3 PXE Client Configuration

As with other system settings, the PXE client configuration settings are accessible through integrated Dell Remote Access Controller (iDRAC) interfaces such as RACADM, or locally through the System Setup utility.

In BIOS boot mode, individual network devices provide the PXE settings in System Setup > Device Settings. However, in UEFI boot mode, PXE settings are configured in the BIOS pages (System Setup > System BIOS > Network Settings > PXE Device Settings). See Figure 5. The “Network Settings” option in the System BIOS page is available only in UEFI boot mode.



System BIOS

System BIOS Settings • Network Settings • PXE Device1 Settings

Interface	Integrated NIC 1 Port 1 Partition 1
Protocol	● IPv4 ○ IPv6
VLAN	○ Enabled ● Disabled
VLAN ID	1
VLAN Priority	0

NIC interface used for this PXE device.

PowerEdge R640
Service Tag : XL12138

Figure 5 UEFI PXE boot Configuration.

The parameters to be configured in this page are:

- Interface: the network interface in the PXE client to configure for PXE boot;
- Protocol: the Internet Protocol that will be used for PXE boot (IPv4 or IPv6);
- VLAN: ID and Priority for Virtual LAN if enabled.

4.4 UEFI HTTP Boot Configuration (Boot from URI)

UEFI HTTP boot is supported beginning with Dell PowerEdge 14G systems. The system that provides the Network Bootstrap Program (NBP) is known as the “HTTP boot server.” The “HTTP boot client” downloads and executes the NBP to complete the boot process.

The principle for HTTP boot is similar to PXE boot, except that HTTP boot uses HTTP (rather than TFTP) to transfer the NBP.

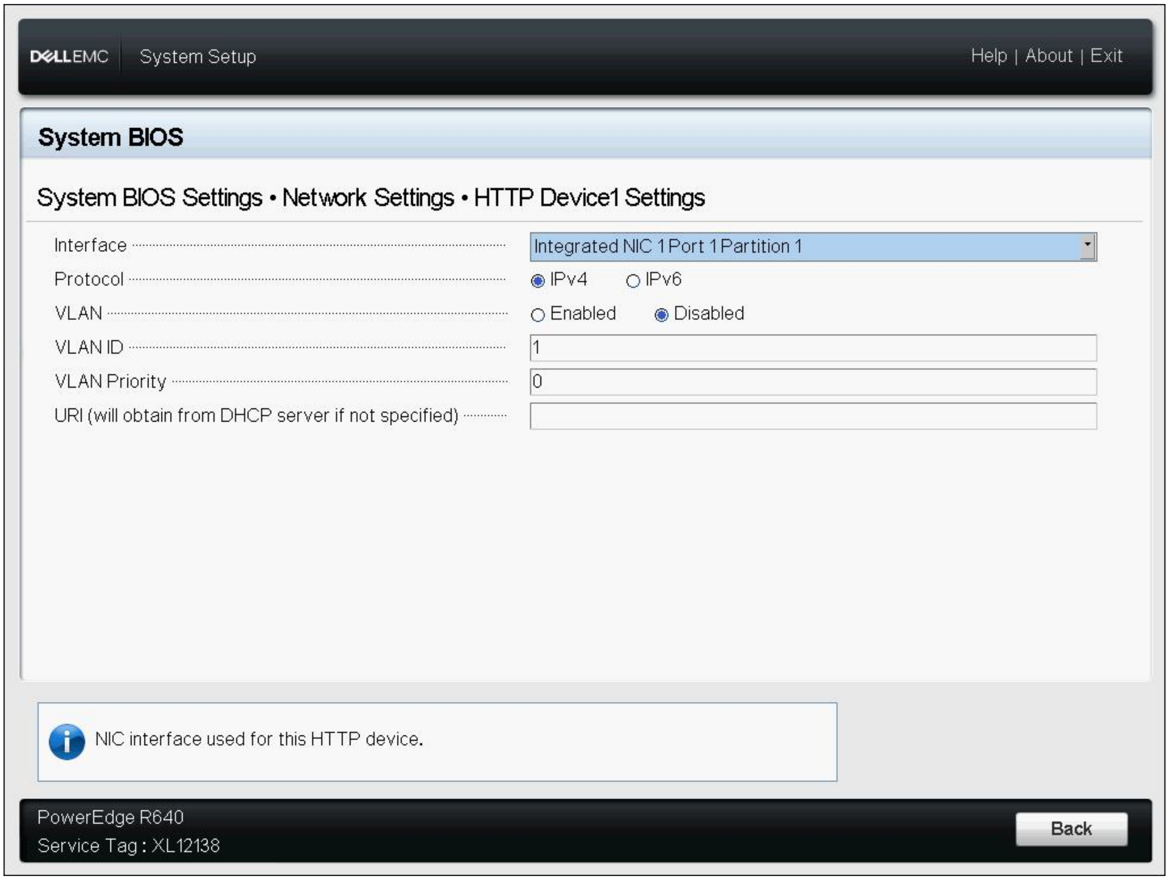
4.4.1 HTTP Boot Client Configuration

As with other system settings, the HTTP boot client configuration settings are accessible through integrated Dell Remote Access Controller (iDRAC) interfaces such as RACADM, or locally through the System Setup utility.

As shown in Figure 6, the UEFI HTTP Boot configuration page is found under System Setup > System BIOS > Network Settings > HTTP Device Settings. The settings are similar to PXE settings with the addition of the "URI" setting, which specifies the location of the bootstrap program. The URI must use the HTTP protocol, and must specify the name of the bootstrap program (for example, http://mydomain.org/img/bootimage.efi).

NOTE: If the "URI" setting is blank, the system will try to obtain the URI from the DHCP server (Option 67 - Bootfile_Name for DHCPv4; Option 59 - Bootfile_Url for DHCPv6).

HTTP Boot is supported only in UEFI boot mode. Also, the "Network Settings" option in the System BIOS page is available only in UEFI boot mode.



System BIOS

System BIOS Settings • Network Settings • HTTP Device1 Settings

Interface	Integrated NIC 1 Port 1 Partition 1
Protocol	● IPv4 ○ IPv6
VLAN	○ Enabled ● Disabled
VLAN ID	1
VLAN Priority	0
URI (will obtain from DHCP server if not specified)	

NIC interface used for this HTTP device.

PowerEdge R640
Service Tag : XL12138

Figure 6 UEFI PXE boot Configuration.

4.4.2 HTTP Boot Server Configuration

The HTTP boot server is composed of two major parts: DHCP server and HTTP server. A domain name system (DNS) server is necessary as well if the URI specifies the domain name instead of the IP address.

The bootstrap program provided by the HTTP boot server must be a UEFI bootable image (PE/COFF format). For Linux environments, UEFI-capable bootstrap programs include ELILO, grub2, and syslinux. Windows environments (Windows Server 2012 and later) use bootmgfw.efi.

4.5 UEFI iSCSI Boot Configuration

UEFI iSCSI Boot enables booting a system to a boot image located on a network-attached system. The network-attached system with the boot image is known as the "target." The other system, the "initiator," uses block transactions (similar to the way a hard-disk controller accesses a local hard-disk drive) to access the bootable software stored on the network-attached target.

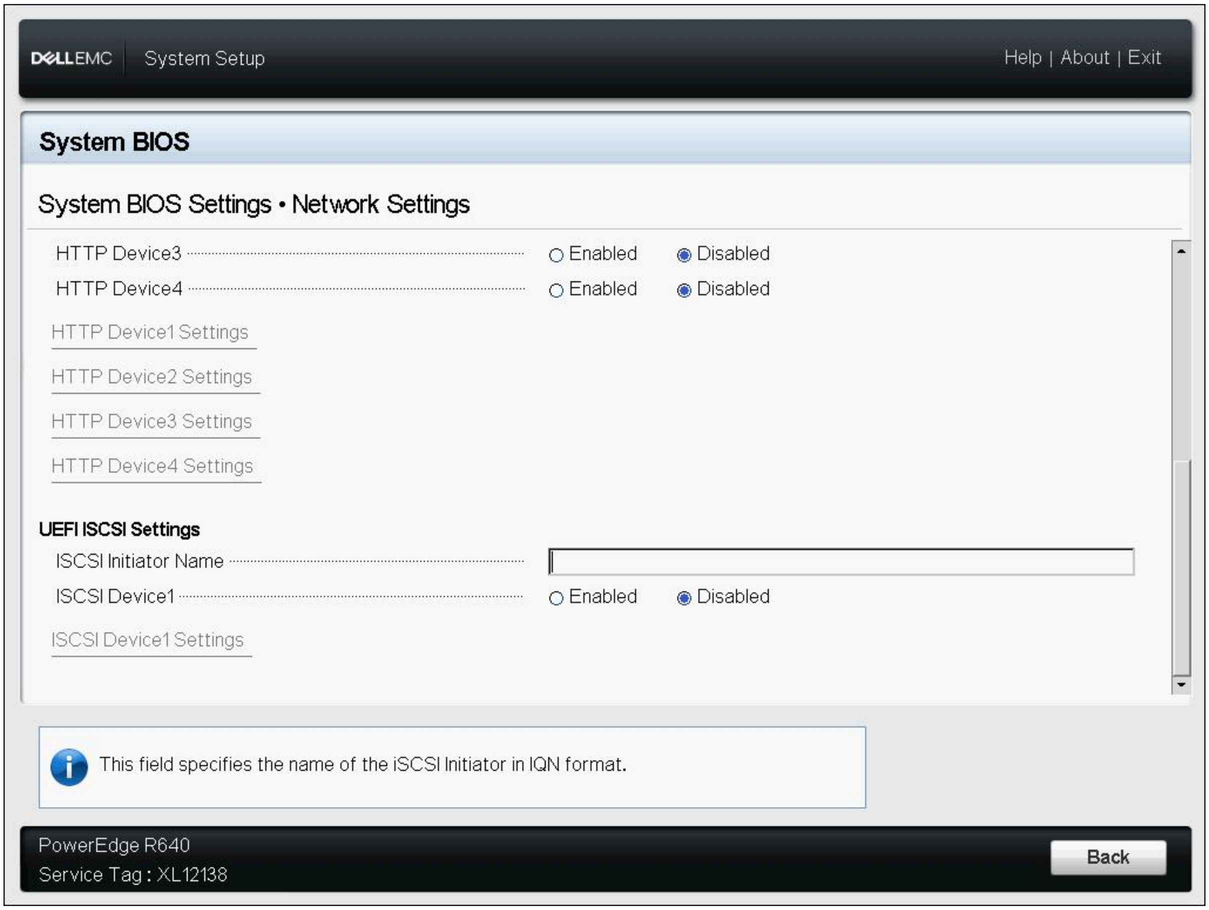
Minimal changes are required for the iSCSI initiator and iSCSI target when transitioning from BIOS boot mode to UEFI boot mode. The following sections describe these changes.

4.5.1 UEFI iSCSI Initiator Configuration

As with other system settings, the UEFI iSCSI configuration settings are accessible through integrated Dell Remote Access Controller (iDRAC) interfaces such as RACADM, or locally through the System Setup utility.

In BIOS boot mode, individual network devices provide the iSCSI settings in System Setup > Device Settings. However, in UEFI boot mode, iSCSI settings are configured in the BIOS pages (System Setup > System BIOS > Network Settings > UEFI iSCSI Settings). See Figure 7. The initiator name is configured on this page; this is the unique name (in IQN format) for the iSCSI initiator. The "Network Settings" option in the System BIOS page is available only in UEFI boot mode.

There are two iSCSI logical devices that can be configured for iSCSI boot. Each logical device appears as a separate entry in the UEFI boot order. When an iSCSI logical device is enabled, its settings are available in its iSCSI Device Settings menu. Figures 8 and 9 show the contents of the iSCSI Device Settings menu.



System BIOS

System BIOS Settings • Network Settings

HTTP Device3 ○ Enabled ● Disabled

HTTP Device4 ○ Enabled ● Disabled

UEFI ISCSI Settings

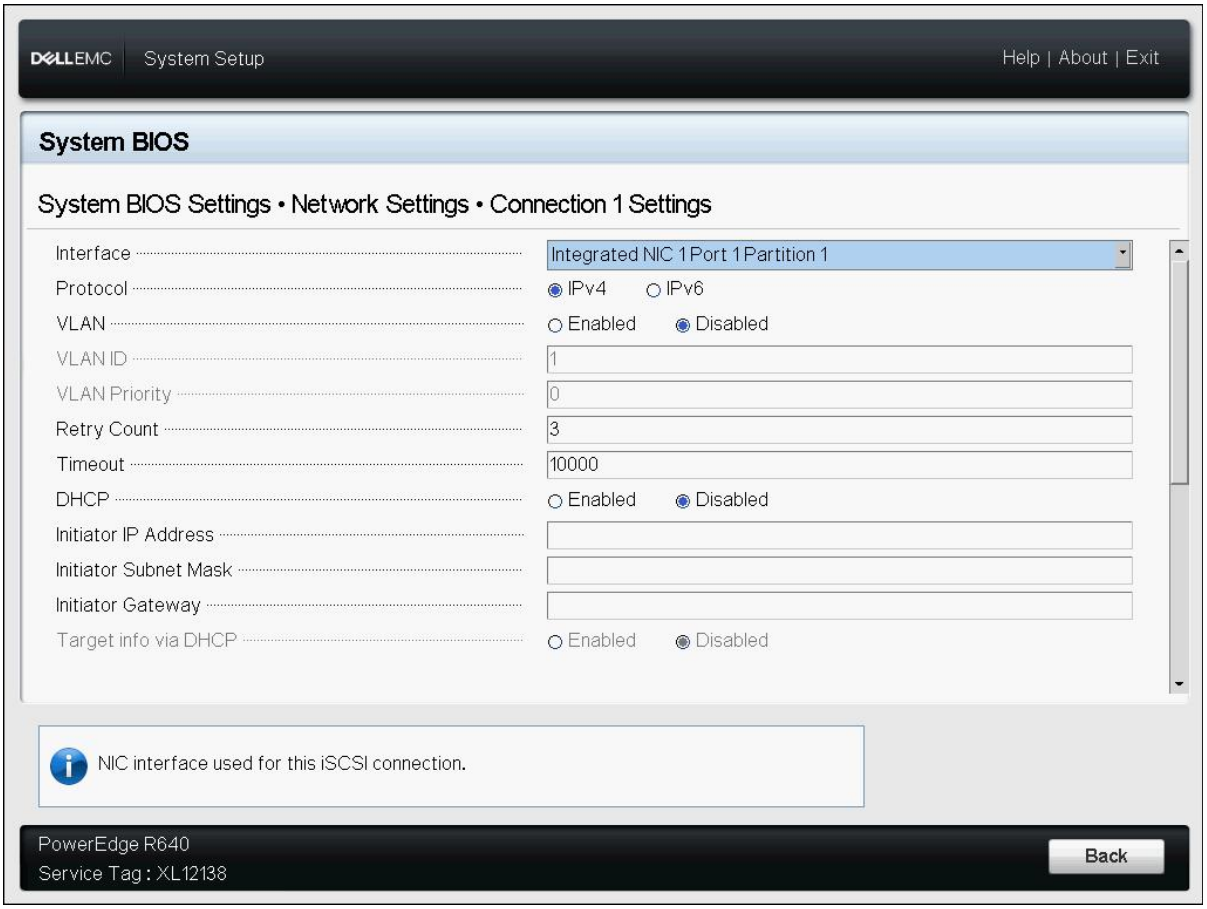
ISCSI Initiator Name

ISCSI Device1 ○ Enabled ● Disabled

This field specifies the name of the iSCSI Initiator in IQN format.

PowerEdge R640
Service Tag : XL12138

Figure 7 UEFI iSCSI boot Initiator Name Configuration



System BIOS

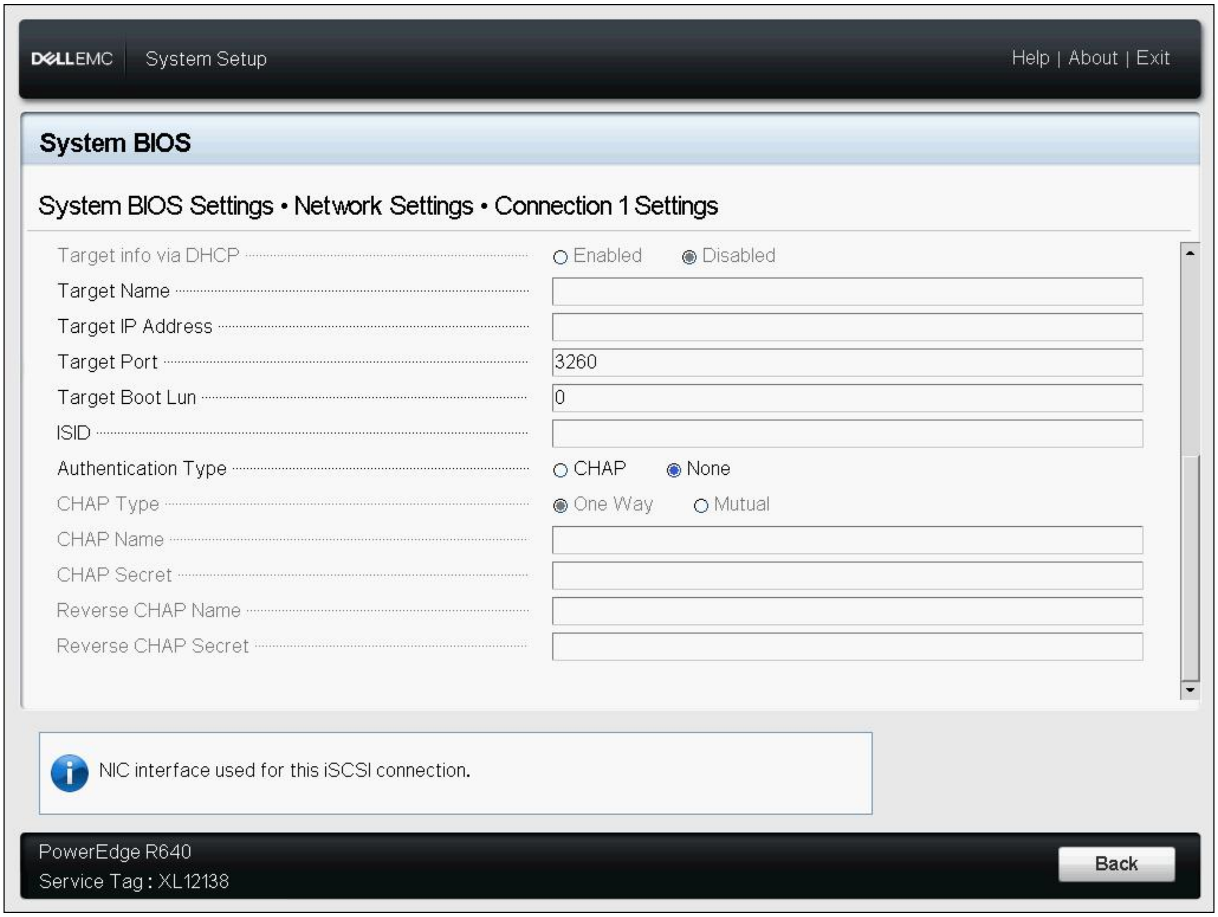
System BIOS Settings • Network Settings • Connection 1 Settings

Interface	Integrated NIC 1 Port 1 Partition 1
Protocol	● IPv4 ○ IPv6
VLAN	○ Enabled ● Disabled
VLAN ID	1
VLAN Priority	0
Retry Count	3
Timeout	10000
DHCP	○ Enabled ● Disabled
Initiator IP Address	
Initiator Subnet Mask	
Initiator Gateway	
Target info via DHCP	○ Enabled ● Disabled

NIC interface used for this iSCSI connection.

PowerEdge R640
Service Tag : XL12138

Figure 8 UEFI iSCSI boot Configuration.



System BIOS

System BIOS Settings • Network Settings • Connection 1 Settings

Target info via DHCP ○ Enabled ● Disabled
Target Name
Target IP Address
Target Port 3260
Target Boot Lun 0
ISID
Authentication Type ○ CHAP ● None
CHAP Type ● One Way ○ Mutual
CHAP Name
CHAP Secret
Reverse CHAP Name
Reverse CHAP Secret

NIC interface used for this iSCSI connection.

PowerEdge R640
Service Tag : XL12138

Figure 9 UEFI iSCSI boot Configuration (cont’d).

The parameters to be configured for an iSCSI logical device are:

- Interface: the network interface in system to be configured for PXE boot;
- Protocol: the Internet Protocol that will be used for PXE boot (IPv4 Vs IPv6);
- VLAN parameters: VLAN ID and Priority Virtual LAN if enabled;
- TCP parameters: Retry Count and Timeout to manage TCP handshake retries and timeout condition;
- iSCSI Initiator parameters: if set DHCP to Disabled, iSCSI initiator parameter fields (IP Address, Subnet Mask and Gateway) will need to be filled out;
- iSCSI Target parameters: if set Target info via DHCP to Disabled, iSCSI target parameter fields (Target Name, IP Address, Port and Lun) will need to be filled out;
- ISID: set the initiator session identifier;
- Authentication parameters: CHAP Type, CHAP Name/Secret and Reverse CHAP Name/Secret fields are used for authentication purposes.

4.5.2 iSCSI Target Configuration

The difference between iSCSI target configurations for UEFI and BIOS boot modes is the format of the bootable image on the target. In UEFI boot mode, the image must be a UEFI bootable image. iSCSI target configuration steps for Windows and Linux can be found on their respective official websites.

4.6 UEFI Secure Boot Configuration

There are three primary settings involved in configuring UEFI Secure Boot. All three settings are available in BIOS Setup (System Setup > System BIOS > System Security) and iDRAC interfaces such as RACADM. The Boot Mode must be set to UEFI; otherwise these settings are not configurable. Secure Boot is not available when the Boot Mode is set to BIOS.

To use UEFI Secure Boot, set the “Secure Boot” setting to “Enabled”, the “Secure Boot Policy” setting to “Standard”, and the “Secure Boot Mode” setting to “Deployed”. This configuration causes the BIOS to verify pre-boot code modules (such as adapter firmware and OS loaders) against an industry-standard set of certificates and hashes. The BIOS will execute only those modules signed by third parties trusted by Dell.

The first setting (Secure Boot) instructs the BIOS whether to perform integrity and authorization checks on pre-boot code modules. When this setting is set to “Enabled” the BIOS enforces the Secure Boot policy for each code module that is loaded during the boot process. When this setting is set to “Disabled” the BIOS loads code modules without performing integrity and authorization checks.

The second setting (Secure Boot Policy) tells the BIOS which Secure Boot policy to enforce. When this setting is set to “Standard” the BIOS uses an industry-standard set of certificates and hash values that authorize common operating systems and I/O adapter firmware. The Standard policy applies to a majority of server deployment environments. (The system BIOS will log and display an error message when a server component, such as an expansion card or operating system, does not satisfy the policy requirements.) When this setting is set to “Custom” the BIOS uses a set of certificates and hash values pre-defined by the system administrator. This setting is intended for advanced users who want additional assurance beyond the industry-standard policy. A custom policy enables the user to specify which pre-boot code modules are trusted and executed by the system BIOS.

The third setting (Secure Boot Mode) enables automated deployment capabilities for Secure Boot that are beyond the scope of this document. The most secure value for the Secure Boot Mode is its default value (Deployed Mode). Advanced users can find a description of the other values for Secure Boot Mode in the UEFI specification (version 2.7, section 31.3).

For more information on UEFI Secure Boot and configuring a custom Secure Boot policy, refer to the following documents:
Defining a Secure Boot Policy (Dell TechCenter)
Secure Boot Management on 14G Dell EMC PowerEdge Servers (Dell TechCenter)

4.7 Integrated Device Firmware

Dell PowerEdge servers include multiple integrated devices, such as network controllers and storage controllers. Each of these devices has firmware that initializes the hardware and contributes to the boot process.

Since the integrated device firmware supports both UEFI and BIOS boot modes, it is not necessary to update the firmware for one boot mode or the other. When the system is in UEFI boot mode, the BIOS automatically loads UEFI drivers for the devices instead of traditional option ROMs.

RAID containers and virtual disk configurations do not require re-configuration when the system is changed to UEFI boot mode. The RAID metadata is independent of the system boot mode.

5 Technical support and resources

Dell.com/support is focused on meeting customer needs with proven services and support.

Dell TechCenter is an online technical community where IT professionals have access to numerous resources for Dell EMC software, hardware and services.

The Dell Systems Management portion of Dell TechCenter provides guidance for deployment, update, and configuration tasks.

5.1 Related resources

Preboot Networking on Dell PowerEdge Servers – Dell whitepaper that discusses network boot operation in both traditional BIOS boot mode and UEFI boot mode.

Defining a Secure Boot Policy – Dell whitepaper that provides background on UEFI Secure Boot and explains how to configure a custom Secure Boot policy.

Secure Boot Management on 14G Dell EMC PowerEdge Servers – Dell whitepaper that describes remote management capabilities for Secure Boot certificates.