Dell OpenManage Deployment Toolkit Version 4.2 User's Guide

Notes, Cautions, and Warnings

NOTE: A NOTE indicates important information that helps you make better use of your computer.

CAUTION: A CAUTION indicates either potential damage to hardware or loss of data and tells you how to avoid the problem.

WARNING: A WARNING indicates a potential for property damage, personal injury, or death.

2012 - 12

Rev. A00

Contents

1 Introduction

The Dell OpenManage Deployment Toolkit (DTK) includes a set of utilities, sample scripts, and sample configuration files that you need to deploy and configure your Dell system. You can use DTK to build script-based and RPM-based installation for deploying large number of systems on a preoperating system environment in a reliable way, without changing their current deployment processes.

In addition to the command line utilities used to configure various system features, DTK also provides sample scripts and configuration files to perform common deployment tasks and documentation. These files and scripts describe the use of the DTK in Microsoft Windows Pre-installation Environment (Windows PE) and Linux environments.

What's New In This Release

The new features for this release include:

- Using DTK utilities you can deploy the following operating systems:
  - Red Hat Enterprise Linux 6.3 (64-bit)
  - Red Hat Enterprise Linux 5.8 (32-bit and 64-bit)
  - SUSE Linux Enterprise Server 11 SP2 (64-bit)
  - SUSE Linux Enterprise Server 10 SP4 (64-bit)
  - Microsoft Windows Server 2012
- You can install DTK Linux RPM utilities on the following operating systems:
  - Red Hat Enterprise Linux 6.3 (64-bit)
  - Red Hat Enterprise Linux 5.8 (32-bit and 64-bit)
  - SUSE Linux Enterprise Server 11 SP2 (64-bit)
  - SUSE Linux Enterprise Server 10 SP4 (64-bit)
- Support for Windows PE 4.0 (64-bit) for deploying Windows Server 2012 using **W2K12INST.BAT** script. For more information, see Deployment Scripts For Windows.
- DTK tool support for Embedded Linux ISO image (64-bit).
- Improved **syscfg** performance on Dell PowerEdge *yx2x* (12G) systems and reduction in server provisioning time.
- Syscfg supports the following:
  - Fully Qualified Domain Name (FQDN) as SNMP trap destination.
  - BIOS configuration using XML file on Dell PowerEdge yx2x (12G) systems.

  NOTE: For more information, see *Dell OpenManage Deployment Toolkit Command Line Interface Reference Guide* available at **dell.com/support/manuals**.
- Raidcfg supports the following:
  - Blinking and unblinking virtual disks and array disks.
  - Enumerating virtual disk.
  - Setting a virtual disk as bootable device.
  - Setting virtual disk name.

  - Listing global hot spares.

  NOTE: For more information, see *Dell OpenManage Deployment Toolkit Command Line Interface Reference Guide* available at **dell.com/support/manuals**.

- Support for BIOS and RAID configurations in the following Dell PowerVault Network Attached Servers (NAS) :
  - NX3000
  - NX3100
  - NX300
  - NX200
  - NX3300
  - NX3200
  - NX400
- Support for creating and reporting RAID in following Dell PowerVault enclosures:
  - MD1000
  - MD1200
  - MD1220
- Deprecated support for the following operating systems:
  - Windows PE 2.0 (32-bit and 64-bit)
  - Red Hat Enterprise Linux 6.2

Other Documents You May Need

In addition to this guide, you can access the following guides available at **dell.com/support/manuals**. On the **Manuals** page, click **Software → Systems Management**. Click the appropriate product link on the right-side to access the documents.

- *The Dell OpenManage Deployment Toolkit Installation Guide* provides information about installing, deploying, and upgrading the DTK on supported Dell systems. The guide is available as part of the DTK download and at support.dell.com.
- The *Dell OpenManage Deployment Toolkit Command Line Interface Reference Guide* provides information about configuring the BIOS, Baseboard Management Controller (BMC), Remote Access Controller (RAC), RAID controllers, and hard-drive partitions on supported Dell systems.
- The*Dell OpenManage Installation and Security User's Guide* provides additional information about performing an unattended installation of Server Administrator on systems running supported Windows, Red Hat Enterprise Linux Server, and SUSE Linux Enterprise Server operating systems.
- The *Dell OpenManage Legacy Compatibility Guide* provides compatibility information about Server Administrator installation and operation on various hardware platforms (or systems) running supported Windows, Red Hat Enterprise Linux Server, and SUSE Linux Enterprise Server operating systems.
- The *Dell Systems Software Support Matrix* provides information about the various Dell systems and the operating systems supported by these systems.
- The *Dell Systems Build and Update Utility Quick Reference Guide* is an introduction to the Dell Systems Build and Update Utility, and provides tips on how to make the product effective in helping you streamline your deployment and update procedures for Dell systems.
- The *Dell Remote Access Controller 5 User's Guide* provides comprehensive information about using the RACADM command line utility to configure a DRAC 5.
- The *Dell Chassis Management Controller User's Guide* provides comprehensive information about using the controller that manages all modules in the chassis containing your Dell system.
- The *Command Line Reference Guide for iDRAC6 and CMC* provides information about the RACADM subcommands, supported interfaces, property database groups and object definitions for iDRAC6 and CMC.

- The *Integrated Dell Remote Access Controller 7 (iDRAC7) User's Guide* provides information about configuring and using iDRAC7 for yx2x rack, tower, and blade servers to remotely manage and monitor your system and its shared resources through a network.
- The *Integrated Dell Remote Access Controller 6 (iDRAC6) Enterprise for Blade Servers User Guide* provides information about configuring and using an iDRAC6 for yx1x blade servers to remotely manage and monitor your system and its shared resources through a network.
- The *Integrated Dell Remote Access Controller 6 (iDRAC6) User Guide* provides complete information about configuring and using an iDRAC6 for yx1x tower and rack servers to remotely manage and monitor your system and its shared resources through a network.
- The *Dell Baseboard Management Controller Utilities User's Guide* provides information about configuring a managed system to use the BMC Management Utility to manage your system through its BMC. The Dell Update Packages User's Guide provides information about obtaining and using Dell Update Packages as part of your system update strategy.
- The *Glossary* provides information about the terms used in this document.

Additionally, the Deployment Toolkit **readme.txt** file, which is available as part of the DTK download and at support.dell.com, provides the latest available information for the installation and operation of the DTK components.

Other Supporting Documents You May Need

Besides the Dell-provided documentation, there are numerous other resources to aid you in planning and executing a DTK-assisted deployment.

- Operating system documentation to prepare for and execute the unattended installation process. In addition, you should consult the available web-based resources such as:
  - The Microsoft Tech Net database at **microsoft.com/technet**
  - The Red Hat Enterprise Linux support pages at **redhat.com**
  - The SUSE Linux Enterprise Server support pages at **novell.com**
- Windows PE 3.0, Windows PE 4.0, Windows Automated Installation Kit (Windows AIK), Windows Assessment and Deployment Kit (ADK), and Windows Deployment Services (WDS) documentation.
- Imaging software documentation, if you are creating a bootable media or are planning to deploy from an image (deploying from an image is not covered in this guide).

Contacting Dell

NOTE: If you do not have an active Internet connection, you can find contact information on your purchase invoice, packing slip, bill, or Dell product catalog.

Dell provides several online and telephone-based support and service options. Availability varies by country and product, and some services may not be available in your area. To contact Dell for sales, technical support, or customer service issues:

1. Visit **www.dell.com/support**.
2. Select your support category.
3. Verify your country or region in the Choose a Country/Region drop-down menu at the top of page.
4. Select the appropriate service or support link based on your need.

2

Prerequisites And Deployment

The Deployment ToolKit (DTK) is a collection of Windows PE and Linux-based utilities and scripts that assist in performing operating system tasks and deployment on Dell systems. An advanced knowledge of Windows PE and Linux is required for users who want to use the DTK utilities to perform pre-operating system and post-operating system configuration tasks or to run a scripted deployment in the respective environments.

A pre-operating system environment is defined as the environment used to configure system hardware *before* a major operating system, such as Microsoft Windows or Linux, is installed. DTK utilities and scripts, for example, are executed in a Windows PE or Linux environment for hardware configuration.

Because of the open nature of the utilities and scripts and the environment in which they can be used, it is strongly recommended that DTK users have an advanced understanding of the Windows PE and Linux environment and scripting knowledge for the respective environments. DTK users with an advanced understanding of Windows PE and Linux can take full advantage of the utilities and exploit their capabilities above and beyond what the sample scripts can provide.

CAUTION: Some of the DTK utilities can destroy data if used incorrectly. To avoid the potential risk of data loss, take all necessary precautions to protect data so that mission-critical systems are not disrupted in the unlikely event of a failure. See the *Dell OpenManage Deployment Toolkit Command Line Interface Reference Guide* available at dell.com/support/manuals for complete information about the capabilities of each DTK utility.

Supported Systems

For a complete list of supported Dell PowerEdge systems, see the *Dell Systems Software Support Matrix* available at **dell.com/support/manuals**. On the Manuals page, click **Software → Systems Management → Dell OpenManage Releases**. Click the appropriate product version to access the Support Matrix.

Supported Operating Systems

For Windows, the DTK utilities support Windows PE 3.0 (32-bit and 64-bit) and Windows PE 4.0. Using the DTK utilities, we can deploy the following Windows operating systems:

- Microsoft Windows Server 2012 (64–bit)
- Windows Server 2008 SP2 (32-bit and 64-bit)
- Windows Server 2008 R2 SP1

For Linux, the DTK utilities support the following operating systems:

- Red Hat Enterprise Linux 6.3 (64-bit)
- Red Hat Enterprise Linux 5.8 (32-bit and 64-bit)
- SUSE Enterprise Linux 11 SP2 (64-bit)
- SUSE Enterprise Linux 10 SP4 (64-bit)

You can install DTK Linux RPM utilities on the following operating systems:

- Red Hat Enterprise Linux 6.3 (64-bit)
- Red Hat Enterprise Linux 5.8 (32-bit and 64-bit)

- SUSE Enterprise Linux 11 SP2 (64-bit)
- SUSE Enterprise Linux 10 SP4 (64-bit)

NOTE: The RPMs are available in the RPMs folder of DTK ELI ISO and also at **linux.us.dell.com/repo/hardware/OMSA_7.1.1/**.

Supported BIOS Firmware Versions

The latest (**n**) version or the **n-1** version of BIOS firmware that is available at **support.dell.com**.

Prerequisites And Requirements

The prerequisites and requirements to run DTK utilities are as given below.

NOTE: The target system must not have Fibre Channel (FC) devices.

See Also:

- Installation Prerequisites
- Deployment Prerequisites

Installation Prerequisites

Before installing the DTK components for Windows PE, ensure that you have the following:

- The DTK self-extracting zip file (**DTKX.X-WINPE-XX.exe**), which contains the utilities, sample scripts, sample configuration files, and documentation you need to deploy your Dell system.
- A Windows workstation or server that has 512 MB of RAM.

Before installing DTK components for embedded Linux, ensure that you have the following:

- The DTK ISO image, which contains the utilities, sample scripts, sample configuration files, RPMs, and documentation you need to deploy your Dell system.
- A Linux workstation that has at least 200 MB of free hard-drive space.

Deployment Prerequisites

Before beginning the deployment process for Windows PE, ensure that you have all of the following tools, software, and information ready to use or consult:

- Windows Automated Installation Kit (AIK) for Windows Server 2008 R2 to build Windows PE 3.0.
- Windows Assessment and Deployment Kit (ADK) for Windows Server 2012 to build Windows PE 4.0.
- An unzip utility.
- Working knowledge to build Windows PE 3.0 and Windows PE 4.0 (see the Microsoft documentation to customize Windows PE).
- Working knowledge of Microsoft Remote Installation Services (RIS) and Automated Deployment Services (ADS) (including setting up of RIS and ADS environments) or any other third party deployment system or tool for Windows PE.
- Working knowledge of Windows Deployment Services (WDS) or any other third party deployment system or tool for Windows PE.
- A workstation with the following capabilities:
  - Writable media drive

  - Network access
- A target system with a media drive, if performing a local deployment.
- A target system with a media drive and network access, if performing a network deployment.
- All DTK utilities, sample scripts, and sample configuration files.
- *Dell Systems Management Tools and Documentation* DVD.
- Your operating system software and documentation.
- An optimally configured source system with network access.

NOTE: You can download the latest drivers from **support.dell.com.**

For Linux, ensure that you have all of the following tools, software, and information ready to use or consult:

- Advanced knowledge of Linux and Linux scripting (bash), Linux networking, installing and working with RPM Package Managers, and creating and modifying loop file systems.
- Any third party deployment system or tool.
- A workstation with the following capabilities:
  - A writable media drive
  - Network access
- A target system with a media drive, if performing a local deployment.
- A target system with a media drive and network access, if performing a network deployment.
- All DTK utilities, sample scripts, sample configuration files, and RPM packages.
- All operating system RPM packages that DTK RPMs require.

NOTE: Tools such as Yellowdog Updater, Modified (YUM), Yet Another Setup Tool (YAST), and Advanced Packaging Tool (APT) can be used to manage RPM dependency issues.

- All other utilities and files necessary to perform the deployment, including all required Linux drivers, operating system drivers, and the Dell utility partition file.
- *Dell Systems Management Tools and Documentation* DVD.
- Your operating system software and documentation.
- An optimally configured source system with network access.

NOTE: You can download the latest drivers from **support.dell.com.**

Drivers

The following table describes the essential and optional drivers required for Dell-supported hardware to build your Windows PE environment.

NOTE: The drivers provided have been tested successfully in the Windows PE environment.

Table 1. Drivers Required to Build Your Windows PE Environment

Drivers	Purpose
common\hapi	Essential drivers for DTK tools to work.
common\raidcfg	Essential drivers for **raidcfg** to work.
winpe3.x\WINPE3.0_driverinst.bat	Used to install Dell-provided drivers, from **\Dell\x32\Drivers or \Dell \x64\Drivers**, into the customized Windows PE 3.0 image.

Drivers	Purpose
winpe4.x\WINPE4.0_driverinst.bat	Used to install Dell-provided drivers, from **\Dell\x32\Drivers or \Dell \x64\Drivers**, into the customized Windows PE 4.0 image.

Other Supporting Tools And System Files

While DTK provides all the necessary utilities and scripts for deploying Dell systems, there are additional applications, utilities, and system files needed to create the pre-operating system environment in which this toolkit is used.

NOTE: Ensure that the tools are 32-bit compatible.

The following table lists the Windows system files and other tools that you need to use DTK utilities and its sample scripts.

Table 2. Supporting Tools and System Files for Windows

Additional Tools	Purpose	Source
Windows AIK for Windows Server 2008 R2 to build Windows PE 3.0	Create a bootable media for Windows PE 3.0.	**microsoft.com**
Windows ADK for Windows Server 2012 to build Windows PE 4.0	Create a bootable media for Windows PE 4.0.	**microsoft.com**

The following table is a matrix of Linux system files and other tools that you need to use with DTK utilities and its sample scripts.

Table 3. Supporting Tools and System Files for Linux

Additional Tools	Purpose	Source	Save in DTK location
mkfs.*	Formats hard-drive partitions.	Any system running the Linux operating system.	In the `PATH` variable.
fdisk	To partition the hard drive.	Any system running the Linux operating system.	In the `PATH` variable.
unzip	To unzip files in Linux.	Any system running the Linux operating system.	In the `PATH` variable.
upimg	Contains system-specific Dell utility partition files. **NOTE:** In the previous releases, each Dell system required a unique **upimg** file. The **upimg** file is common for all Dell systems.	**/opt/dell/toolkit/systems**, or the Dell Systems Build and Update Utility on the *Dell Systems Management Tools and Documentation* DVD.	**/opt/dell /toolkit/ systems**
kernel image	Used during a Linux unattended installation.	Available on the Linux media.	**/opt/dell /toolkit /systems/ linux**
initial ramdisk	Used during a Linux unattended installation.	Available on the Linux operating system media or on the Systems Build and Update Utility on the *Dell Systems Management Tools and Documentation* DVD.	**/opt/dell /toolkit /systems**

Additional Tools	Purpose	Source	Save in DTK location
	NOTE: For legacy operating systems, you may need to use the driver disk. At the end of the Red Hat Enterprise Linux Server installation, you may need to install the drivers available on the Systems Build and Update Utility.		

Deploying Operating Systems

After the system hardware is versioned and configured, an operating system can be installed using unattended setup files, samples of which are provided as part of DTK. While these sample configuration files are operating system specific, they have been enhanced for Dell systems. See your operating system documentation for more information about how to modify these unattended installation files.

For Windows PE, the sample configuration files contain examples to deploy the following operating systems:

- Windows Server 2008 R2 with SP1
- Windows Server 2008 SP2 (32-bit and 64-bit)

NOTE: For the latest supported operating systems, see the Windows PE documentation.

For Linux, the sample configuration files contain examples to deploy the following operating systems:

- Red Hat Enterprise Linux Server version 6.3 and 6.2 (64-bit)
- Red Hat Enterprise Linux Server version 5.8 (32-bit and 64-bit)
- SUSE Linux Enterprise Server version 11 SP2 (64-bit)
- SUSE Linux Enterprise Server version 10 SP4 (64-bit)

Deployment

Dell systems can be deployed using several methods. The most common method is using the Systems Build and Update Utility (SBUU) on the *Dell Systems Management Tools and Documentation* DVD that is shipped with your system. Currently, this method is used to deploy a single system at a time with the installation wizard.

DTK, on the other hand, offers a complete set of utilities, sample scripts, and RPM packages that can be used to automate deployment on large numbers of Dell systems. This guide is designed to help you through some of the most basic planning considerations, logistical preparations, and deployment procedures to get you started using DTK to deploy Dell systems.

A well-thought-out deployment plan is critical to the success of your deployment effort and includes the following considerations:

- Assessing your existing IT environment
- Selecting an operating system
- Selecting the optimum Dell OpenManage systems management software configuration for your system
- Choosing a deployment method

DTK is designed to be flexible enough to fit into almost any deployment plan.

DTK components can be used in many different ways to assist IT staff in deploying Dell systems in large numbers. Because all the tools and scripts are task oriented, many of them can be used separately to perform a specific task on many systems or collectively to perform many tasks on many systems at once. For information on the deployment methods, see Running The Deployment Scripts.

Deployment Matrix

Depending on the scope, current business needs, network setup, and process, you can select any of the suggested methods of deployment. The following table provides the configuration complexity and the pros and cons of each deployment method.

Table 4. Deployment Method Matrix

Deployment Methods	Complexity	Advantages	Disadvantages
Local deployment (bootable media)	Relatively simple	• Necessary if a network connection is not available or is too slow. • Can install supported Windows operating systems. • Can perform a minimal installation of supported Linux operating systems.	• Any change requires the creation of a new bootable media. • Limited space on media for some operating system installation files. • Deployment tasks must be performed at the individual target system. • Media are read-only and requires the creation of a RAMDISK for temporary data storage. • Installation time is high.
Remote deployment over a network (bootable Windows PE or Linux media with network stack loaded)	Moderate	• Everything is in one place in a network share. • Easy to manage—changes have to be made in a single location. • Data captured (profiles) can be stored in a network share. • Data can be replicated from a network share. • Can install supported Windows or Linux operating systems.	• Must have network connection. • Deployment tasks must be performed at the individual target system. • Must locate the appropriate Windows PE or embedded Linux network drivers. • Not supported from a mapped NetWare system.
Deployment solution framework from a third-party vendor	Mixed	• DTK can be used in context of third-party	Must either acquire and learn to use or have an existing third-party

Deployment Methods	Complexity	Advantages	Disadvantages
		deployment solution framework. • Third-party deployment solution framework is used as the deployment transport mechanism. • Tasks and scripts can be pushed to the target systems.	deployment solution framework.
PXE boot for Linux	High	• Everything is in one place on a network share. • Easy to manage—changes have to be made in only in one place. • High flexibility for remote deployment. • No media required. • Faster deployment.	• Must have high-speed connectivity to network (LAN). • Set up time is longer.

After you have determined which deployment method best suits the needs of your organization, you are ready to begin building the DTK deployment directory structure on a network volume. For instructions on installing the DTK components and populating the deployment directory structure, see Preparing The Script Files.

DTK Deployment Overview

The deployment process can be separated into two main tasks: preparation and the actual deployment. For more details, see Preparing For Deployment and Deploying.

Preparing For Deployment

For using DTK utilities and scripts to deploy on a target system:

1. On systems running Windows, copy or extract DTK utilities, sample scripts, sample configuration files, and drivers provided (in the zip file) to the Windows PE image. On systems running Linux, obtain the DTK Linux ISO image, which is a self-contained bootable ISO image.
2. On systems running Windows, organize DTK utilities, scripts, and configuration files, the operating system installation files, and the requisite system files and drivers on a network share or local media. On systems running Linux, use and customize the sample scripts per your requirements.
3. Set up an optimally configured source system by using the Dell Systems Build and Update Utility on the *Dell Systems Management Tools and Documentation* DVD (to install your operating system) and the Systems Service and Diagnostics Tools (to load drivers). This source system acts as the master server that is used to replicate settings to target servers.

   NOTE: You can also download the latest drivers from **support.dell.com.**
4. Generate a system BIOS, BMC (Baseboard Management Controller), RAID, and/or RAC configuration profile from the optimally configured source system. Copy the generated configuration files to a read/write share onto the workstation.

NOTE: You can obtain the system BIOS, BMC, RAID, and/or RAC configuration files from DTK Windows PE ISO image and running the **SYSCAP.BAT, RAIDCAP.BAT,** and **RACCAP.BAT** scripts or for Linux from DTK Linux ISO image and running the **syscap.sh, raidcap.sh,** and **raccap.sh** scripts.

NOTE: In previous releases, you could use a system profile generated on a specific Dell system to deploy only that same system model. From DTK version 2.4 and later, you can use a system BIOS configuration profile generated for all systems belonging to the same generation. For example, **syscfg_x9xx.ini** for *x9xx* systems.

5. Create an operating system answer file that contains unattended operating system software installation information.
6. Edit the DTK sample script files that read the system configuration files to set up the system BIOS, BMC, RAID, and RAC and then install an operating system on a target system.

Deploying

After you have prepared all scripts, files, and utilities, you are ready to proceed with the deployment process. While there are many ways to use DTK utilities to deploy Dell systems, this guide focuses on the two most common methods for Windows PE and Linux: media-based deployment (local) and network-based deployment (remote).

Configuring A Source System

You need an optimally configured system, both for testing purposes and as the source for the **SYSCFG** and **RAC** configuration files, to be used in the deployment process. For instructions on using the sample DTK scripts, SYSCAP.BAT or syscap.sh and RACCAP.BAT or raccap.sh, to generate SYSCFG and RAC configuration files that are used to configure the target system, see Using DTK Sample Scripts To Capture Configuration Information.

You can use the **SBUU** and **Systems Service and Diagnostics Tools** available on the *Dell Systems Management Tools and Documentation* DVD supplied with your system to:

- Streamline the operating system installation.
- Setup and configure your system, including automatic discovery and configuration of Dell-provided RAID controllers and network adapters.
- Access and load the latest utilities, drivers, and diagnostics for your system.
- Install Dell OpenManage systems management software.

NOTE: You can also download the latest drivers from **support.dell.com**.

For complete information about using the **Systems Build and Update Utility and the Dell Systems Service and Diagnostics Tools**, see the *Dell Systems Build and Update Utility User's Guide* and the *Dell Systems Service and Diagnostics Tools Quick Installation Guide*, available at **dell.com/support/manuals** and on the *Dell Systems Management Tools and Documentation* DVD.

After you have installed the operating system and all applicable device drivers from the Systems Build and Update Utility and Systems Service and Diagnostics Tools, check **support.dell.com** for the latest drivers. Install all necessary drivers to be used in the deployment in the system directory for the target system. For example, if your target system is a PowerEdge R720, copy all necessary files in the **\Toolkit\Systems\peR720** directory (Windows) or **/opt/dell/toolkit/systems/pe** R720 directory (Linux).

RAC And BIOS Configuration

Configure RAC on your source system using the **RACADM.EXE** (Windows) or **racadm** (Linux) utility, if applicable. From 12G onwards, BIOS options can be configured using **racadm**.

NOTE: For more information on configuring RAC and BIOS options on your source system, see the *Integrated Dell Remote Access Controller 7 (iDRAC7) User Guide*.

4. DTK carries racadm for DRAC/iDRAC configurations – one of the important feature that racadm picks up the BIOS configurations available with racadm for 12G Servers, this needs to be highlighted as we want Customers to start adopting racadm for 12G BIOS configurations as we move forward to an Unified CLI. This needs to be covered with examples & syntax for the BIOS configurations on 12G – You may want to cover this high level and for complete details you could point to the section of 12G BIOS configurations from the racadm(iDRAC7) Guide Page 19 ug ,

After you have successfully prepared your directory structure for a deployment and fully configured your source system, you are ready to write, review, or edit the deployment sample scripts. For instructions on editing the sample scripts necessary to run a full deployment, see Preparing The Script Files.

How To Obtain Dell Utility Partition Files

Obtain the Dell utility partition files from **/opt/dell/toolkit/systems**, or the Systems Build and Update Utility on the *Dell Systems Management Tools and Documentation* DVD.

3

Downloading DTK

This section describes some of the procedures necessary to begin the deployment process, including:

- Downloading and unzipping the Deployment Toolkit (DTK) components, including a comprehensive matrix of DTK components.
- Setting up a directory structure to facilitate a successful deployment, including a list of the supporting components that are not included with DTK, but are needed to use DTK.
- Setting up an optimally configured source system.
- Configuring supported RAID controllers.

Extracting DTK Components On Systems Running Windows

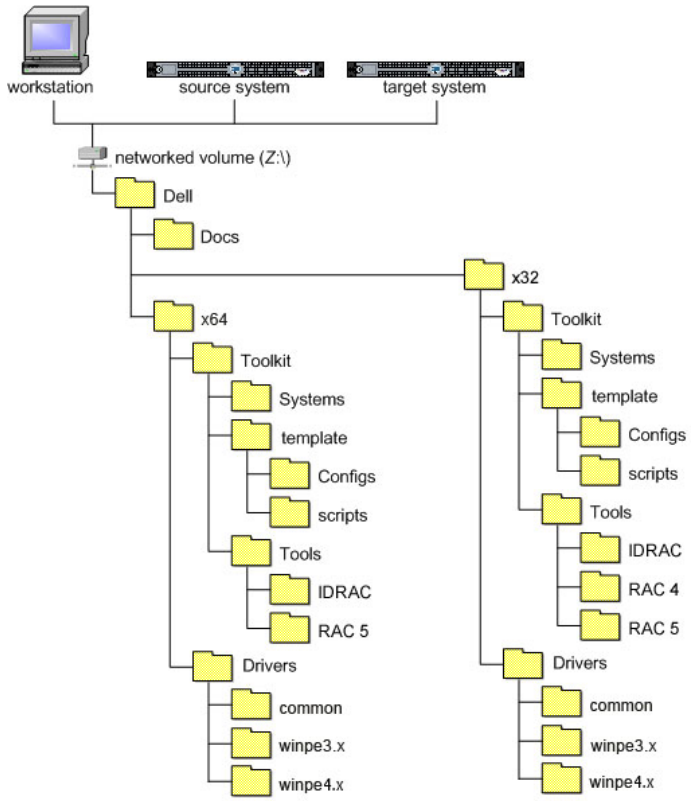
DTK components are provided in a self-extracting zip file at **support.dell.com**. The self-extracting file can be opened on any system running Microsoft Windows operating system. By default, **DTKX.XWINPE- XX.exe** files are extracted to the root directory of your local hard drive, **C:**. This location can be changed by giving a different path when extracting the files, but it is strongly recommended that you keep the DTK default directory structure if you plan to use the sample deployment scripts provided with DTK.

1. Download the DTK file **DTKX.X-WINPE-XX.exe** from **support.dell.com** and save it on your system.
2. After the file downloads, double-click the file. The **Dell OpenManage Deployment Toolkit for WindowsPE** screen containing the product and download information appears.
3. Click **OK**.
4. Click **Unzip**.

   By default, files are unzipped to **C:**. You can unzip the DTK components to your local hard drive and then copy them later to a network volume, or you can change the installation path to an available network share. For example, you can unzip the files directly to the network share (for example, **Z:**) that you plan to use in the deployment process.

   NOTE: Do not copy the DTK components to a Novell NetWare network volume. The DTK network-based deployment is not supported from mapped Novell NetWare systems. The DTK utilities do not run in Windows PE when Novell's Client32 is loaded.

   The files are unzipped in the directory structure shown in the following figure.



DTK Components On Systems Running Windows

Workstation is the system on which the DTK components are extracted.

Source system is an optimally configured system from which the system settings are captured.

Target system is the system on which the settings captured from the source system are replicated.

DTK utilities for 32-bit systems are located in the following directories:

- Utilities: **\Dell\x32\Toolkit\Tools**
- Sample scripts: **\Dell\x32\Toolkit\Template\Scripts**
- Sample configuration files:**\Dell\x32\Toolkit\Template\Configs**

DTK utilities for 64-bit systems are located in the following directories:

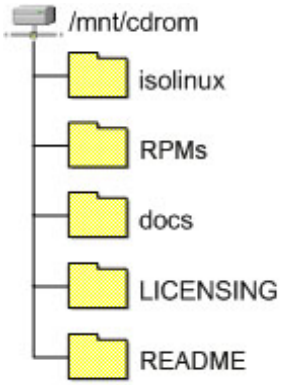
- Utilities: **\Dell\x64\Toolkit\Tools**
- Sample scripts: **\Dell\x64\Toolkit\Template\Scripts**
- Sample configuration files: **\Dell\x64\Toolkit\Template\Configs**

The directory **\Dell\x32\Toolkit\Systems** or **\Dell\x64\Toolkit\Systems** is a placeholder that can be used for all system information that is captured using sample scripts and tools (for example, the system specific BIOS) and Baseboard Management Controller (BMC) configuration files used by the DTK utilities for configuring a system). For more information on the deployment directory structure, see Setting Up A Directory Structure For Deployment.

Extracting DTK Components On Systems Running Linux

DTK components are provided as an ISO image at **support.dell.com**. You can use a CD/DVD burning software to burn the ISO image on a bootable CD/DVD or loop mount the ISO image to access the image contents. Perform the following steps to extract DTK components to a workstation running a Linux operating system.

1. Download the ISO image **dtk_X.X_XXX_Linux.iso** from **support.dell.com** and save it on your system (where *X* is the latest version of DTK and *XXX* is the latest build for DTK release).
2. After the image downloads, burn it on a CD/DVD. The directory structure on the CD/DVD (after it is mounted) is as follows:



DTK Components On The Mounted Media

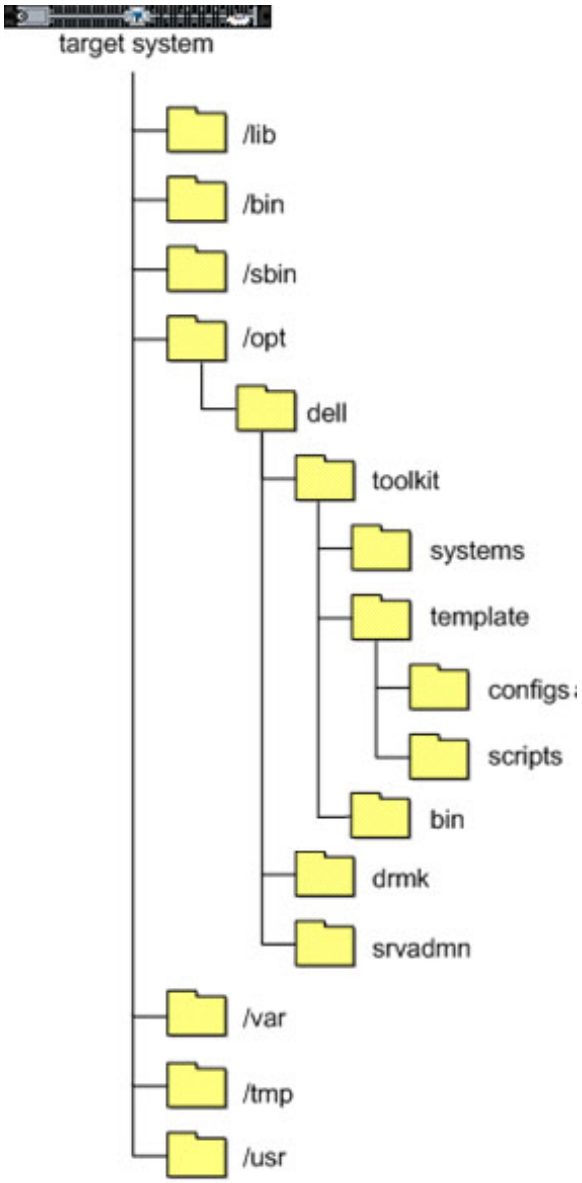
The following table describes the files, components, and other contents of the mounted media.

Table 5. Mounted Media Contents

Contents	Details	Location
docs	Contains *Quick Installation Guide.*	**/mnt/cdrom/docs**
LICENSING	Contains all the licenses for the open source components used in this media.	**/mnt/cdrom/LICENSING**
README	Provides the latest product information.	**/mnt/cdrom/README**
sa.1	Contains the Linux kernel images.	**/mnt/cdrom/isolinux**
sa.2	Initial RAM disk containing the embedded Linux component and DTK tools.	**/mnt/cdrom/isolinux**
RPMs	Contains DTK RPMs and dependency RPMs	**/mnt/cdrom/RPMs**

Directory Structure On The Target System

The following figure displays the directory structure on the target system, when you boot the media on a target system.



The following table describes the contents of the directory structure on the target system.

Table 6. Target System Directory Structure Contents

Contents	Details
/lib	The libraries and drivers required for embedded Linux to function.
/bin	Basic utilities (**ls**, **chmod**, and so on).
/sbin	Utilities that are required by the system superuser to perform administrative tasks.
/opt	The libraries, binaries, and scripts required for the toolkit to function. **NOTE:** The **/opt/dell/srvadmin** directory must be writable.
/var	Any run-time variable data that might be required for embedded Linux and tools to function.

Contents	Details
/tmp	The only safe, writable area. However, all data in this folder is lost every time you reboot.
/usr	Miscellaneous tools and libraries for embedded Linux to function.

DTK Contents

DTK contains Windows PE-based or Linux-based utilities, sample scripts, sample configuration files, drivers, RPMs, and documentation for automating the deployment on Dell systems. For a more information on DTK utilities, see the *Dell OpenManage Deployment Toolkit Command Line Interface Reference Guide*. For more information on sample configuration files and scripts, see Editing The Sample Batch Files To Perform A Complete System Deployment and Preparing The Script Files.

System Utilities And Supporting Files For Windows

The following table lists the files located in **Toolkit\Tools** on systems running the Windows operating system. These are the core tools (utilities) that perform pre-operating system configuration and deployment tasks. See the *Dell OpenManage Deployment Toolkit Command Line Interface Reference Guide* at **dell.com/support/manuals** for complete information about the capabilities of each DTK utility.

Table 7. System Utilities and Supporting Files for Windows

Toolkit Components	Purpose
mr2kserv.exe	Service required to run (in 32-bit systems) **raidcfg.exe** functions.
raidcfg.exe	Creates and reports Redundant Array of Independent Disks (RAID) information.
syscfg.exe	• Reports and configures – BIOS settings and options – BMC settings and options – RAC 5 settings and options – IDRAC settings and options • Maintains system state between reboots • Reports hardware devices and other system-related information.
sysdrmk.exe	Used by **UPINIT.BAT** to patch the boot sector and Master Boot Record (MBR).
config.ini	Configuration file used by **raidcfg.exe**. **CAUTION: Do not edit this file. Incorrect edits might cause raidcfg.exe to fail.**
stsvc.ini	Configuration file used by **raidcfg.exe**. **CAUTION: Do not edit this file. Incorrect edits might cause raidcfg.exe to fail.**
sys.ini	A **.ini**file used with **syscfg.exe**.
IDRAC	Configures iDRAC6. **NOTE:** This feature may not be supported on all systems.
IDRAC7	Configures iDRAC7.

Toolkit Components	Purpose
	NOTE: This feature may not be supported on all systems.
RAC 4	Configures Dell Remote Access Controller (DRAC) 4/P and DRAC 4/I.
RAC 5	Configures DRAC 5/iDRAC.
pci.ids	Reports PCI devices when used with **syscfg.exe**.
disclaimer.txt	Displays the disclaimer for the tools.
racadmErrorCodes.txt	Lists the error codes and messages for the **racadm.exe** tool.
RaidcfgErrorCodes.txt	Lists the error codes and messages for the **raidcfg.exe** tool.
SyscfgErrorCodes.txt	Lists the error codes and messages for the **syscfg.exe** tool.

System Utilities And Supporting Files For Linux

The following table lists the files, located at **/opt/dell/toolkit/bin** and the supporting libraries located in **/opt/dell/toolkit/lib** on systems running the Linux operating system. These are the core tools (utilities) that perform pre-operating system configuration and deployment tasks. See the *Dell OpenManage Deployment Toolkit Command Line Interface Reference Guide* at **dell.com/suport/manuals** for complete information about the capabilities of each DTK utility.

Toolkit Components	Purpose
pci.ids	Used with **syscfg** to report PCI devices.
racadm	Wrapper script which invokes **racadm4**, **racadm5**, or **racadm6** depending on the Remote Access Controller (RAC) on your system.
racadm4	Configures DRAC 4/P and DRAC 4/I RACs.
racadm5	Configures DRAC 5 RACs.
racadm6	Configures iDRAC6.
	NOTE: This feature may not be supported on all systems.
racadmErrorCodes.txt	Lists the error codes and messages for the racadm tool.
raidcfg	Creates and reports RAID information.
RaidcfgErrorCodes.txt	Lists the error codes and messages for the raidcfg tool.
stsvc.ini	Configuration file used by raidcfg.
syscfg	• Reports and configures BIOS, BMC, and DRAC. • Maintains system state between reboots. • Reports hardware devices and other system-related information.
SyscfgErrorCodes.txt	Lists the error codes and messages for the **syscfg** tool.
sys.ini	A .ini file used with **syscfg**.
UpinitErrorCodes.txt	Lists the error codes and messages for **upinit.sh**.

Sample Configuration Files For Windows

The following table describes examples of Windows configuration files used to perform a scripted deployment. The Windows files for 32-bit and 64-bit supported systems are located in the following directories:

- **\Dell\x32\Toolkit\Template\Configs**
- **\Dell\x64\Toolkit\Template\Configs**

NOTE: All configuration files are provided as examples only. These sample configuration files must be edited before they can be used in any deployment tasks.

Table 8. Sample Configuration Files for Windows

Toolkit Components	Purpose	Location
raccfg.ini	Sample configuration file for DRAC 4 remote access controllers.	**\Toolkit\Systems\<system>**
raidcfg.ini	Sample output file to enable RAID replication.	**\Toolkit\Systems\<system>**
syscfg.ini	Sample configuration file for SYSCFG utility to configure the BIOS and BMC settings.	**\Toolkit\Systems\<system>**
winbom.ini	Sample file used to start the racsvc and **mr2kserv** services.	**\Toolkit\Template\Configs**
winpeoem.sif	Sample file to add support for Dell mass storage drivers.	**\Toolkit\Template\Configs**

Sample Configuration Files For Linux

The following table describes examples of Linux configuration files used to perform a scripted deployment. The Linux files are located at**opt/dell/toolkit/template/configs**.

NOTE: All configuration files are provided as examples only. These sample configuration files must be edited before they can be used in any deployment tasks.

Table 9. Sample Configuration Files for Linux

Toolkit Components	Purpose
autoinst.xml	Sample file used with unattended installation of SUSE Linux Enterprise Server operating systems.
autoinst-sles11.xml	Sample file used with unattended installation of SUSE Linux Enterprise Server (version 11) operating systems.
autoinst-sles11sp2.xml	Sample file used with unattended installation of SUSE Linux Enterprise Server (version 11 SP2) operating systems.
ks.cfg	Sample file used with unattended installation of Red Hat Enterprise Linux Server (versions 3 and 4) operating systems.
ks-rhel5.cfg	Sample file used with unattended installation of Red Hat Enterprise Linux Server (version 5) operating system.
ks-rhel6.cfg	Sample file used with unattended installation of Red Hat Enterprise Linux Server (version 6) operating system.
raccfg.ini	Sample configuration file for DRAC 4 RACs.

Toolkit Components	Purpose
raidcfg.ini	Sample output file to enable RAID replication.
syscfg.ini	Sample configuration file for SYSCFG utility to configure the BIOS, DRAC, and BMC settings.

Sample Scripts For Windows

The following table lists Windows sample scripts necessary to perform a scripted deployment. The sample scripts for Windows are located at **Toolkit\Template\Scripts**.

NOTE: All DTK scripts are provided as examples only. The batch scripts and configuration files must be modified to reflect the unique information for each deployed system.

Table 10. Sample Scripts For Windows

Toolkit Components	Purpose
createup.cfg	This file is used by **UPINIT.BAT**.
diskpartclr.cfg	Used by **PARTCFG.BAT** to clear the partitions on the selected disks.
diskpartos.cfg	Used by **PARTCFG.BAT** to create deployment partition to install the operating system.
LISTUP.CFG	This file is used by **UPINIT.BAT**.
MOUNTUP.CFG	This file is used by **UPINIT.BAT**.
UMOUNTUP.CFG	This file is used by **UPINIT.BAT**.
DTKRUNALL.BAT	Applies BIOS, BMC, RAC, and RAID settings saved in the files to the current system.
ERRHNDL.BAT	Handles errors returned in DTK scripts.
PARTCFG.BAT	Creates and populates the Dell utility partition and creates the operating system partition on a specified hard drive. This script cleans out all existing partitions in your system before creating and populating the partition.
RACCAP.BAT	Captures Remote Access Controller (RAC) settings to a file for DRAC 4 remote access controllers.
RACREP.BAT	Applies (replicates) the RAC settings saved in a file to the target system.
RAIDCAP.BAT	Captures the RAID settings and saves them to a file.
RAIDCFG.BAT	Configures the selected RAID controllers on the system.
RAIDREP.BAT	Applies (replicates) RAID settings saved in a file to the target system.
SYSCAP.BAT	Captures BIOS and BMC configuration settings and saves them to a file.
SYSREP.BAT	Applies (replicates) BIOS and BMC configuration settings and configures the target system.
TKENVSET.BAT	Sets the path of DTK installation and variables for the specific system being deployed.
UPINIT.BAT	Creates and populates the Dell utility partition.

Sample Scripts For Linux

The following table lists Linux sample scripts necessary to perform a scripted deployment.

NOTE: All DTK scripts are provided as examples only. The batch scripts and configuration files must be modified to reflect the unique information for each deployed system.

Table 11. Sample Scripts For Linux

Toolkit Components	Purpose
errhndl.sh	Handles errors returned in DTK scripts.
lininst.sh	Performs an unattended installation of a supported Red Hat Enterprise Linux Server operating system.
partcfg.sh	Creates and populates the Dell utility partition and creates the deployment partition on a specified hard drive. Before doing so, however, this script cleans out all existing partitions in your system.
raccap.sh	Captures RAC settings to a file.
racrep.sh	Applies (replicates) the RAC settings saved in a file to the target system.
raidcfg.sh	Configures all RAID controllers detected in a system.
suseinst.sh	Performs an unattended installation of a supported SUSE Linux Enterprise Server operating system.
syscap.sh	Captures BIOS and BMC configuration settings and saves them to a file.
sysdrmk	Used by **upinit.sh**to patch the boot sector.
sysrep.sh	Applies (replicates) BIOS and BMC configuration settings and configures the target system.
tkenvset.sh	Sets the path of DTK installation and variables for the specific system being deployed.
upinit.sh	Creates and populates the Dell utility partition.

RPMs

The following table describes all DTK RPMs located at **mnt/cdrom/RPMs**.

Table 12. RPMs

RPMs	Purpose
syscfg	Configure server BIOS, BMC/DRAC settings, DTK state settings, PCI device detection, and so on.
raidcfg	Configures all supported RAID controllers.
racadm	Configures RAC.

NOTE: Besides the listed RPMs, a set of dependency RPMs are also available at this location.

Documentation

The following table describes documents containing reference information for each DTK utility and instructions for using DTK utilities and sample scripts. On systems running the Windows operating system, documentation can be found at **\Dell\Docs**. On systems running the Linux operating system, documentation can be found at **mnt/cdrom/docs**.

Table 13. Documentation

Toolkit Components	Purpose
Dell OpenManage Deployment Toolkit Installation Guide	Contains information about installing and deploying DTK on supported Dell systems.
README	Contains the latest information about supported systems, known issues, and important notes. On systems running the Linux operating system, it is located at **/mnt/cdrom** as well.

Drivers And Dell Real Mode Kernel

Hardware application programing interface (HAPI) drivers are essential for DTK tools to work in the embedded Linux environment. Dell Real Mode Kernel (DRMK) contains the **mbr** file necessary to create a bootable Dell Utility Partition.

4

Setting Up A Directory Structure For Deployment

To perform a Deployment ToolKit (DTK)-enabled deployment, you must create a deployment directory structure on either a network share or your local workstation hard drive. The deployment directory structure is a central repository for all deployment files that can be used for network deployment and to build a bootable deployment media. It can also be used successively as a testing space. For a network-based deployment, you must set up the directory structure on a network volume that is accessible from your workstation, the source system (to generate configuration information), and the target system (to deploy).

If you plan to run a scripted deployment from DTK sample scripts, make sure that the DTK files are correctly structured in the deployment directory. DTK sample scripts use this structure to set the default paths for deployment. If you make changes to the directory structure, you must also make changes to the sample scripts.

The deployment directory structure consists of the following files:

- DTK Utilities And Files
- System-Specific Files
- Operating System Installation Files

NOTE: Do not create your DTK directory structure on a NetWare network volume. DTK network-based deployment is not supported from mapped NetWare systems.

DTK Utilities And Files

All the necessary DTK files are provided as part of the initial installation. It is recommended that you use the default directory structure created in the initial installation of DTK components. The sample scripts provided with DTK are set up to use these default paths.

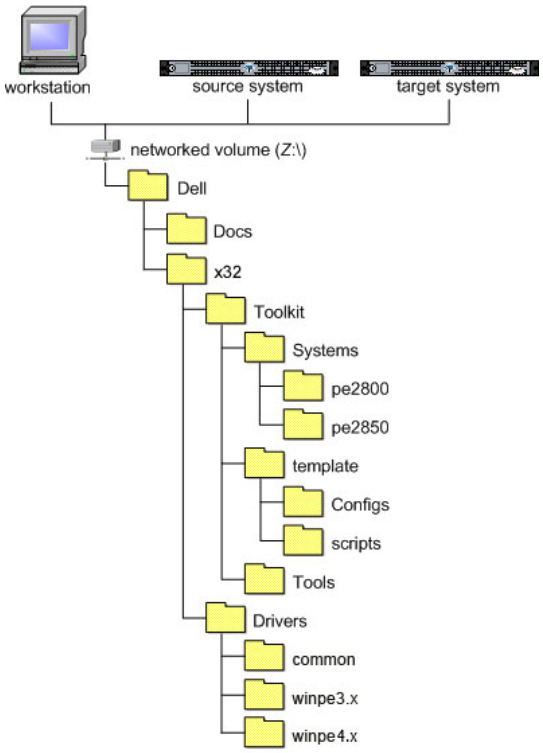
NOTE: If you choose to create a directory structure for DTK components that is different from the one provided in the initial installation, carefully review and edit each sample script (wherever necessary) to reflect the different directory structure.

A 32-bit unzip utility is required to create the Dell utility partition. Make sure that you copy this utility to one of the directories of your Windows PE image, preferably **\windows\system32** folder for Windows or **/systems** folder for Linux.

System-Specific Files

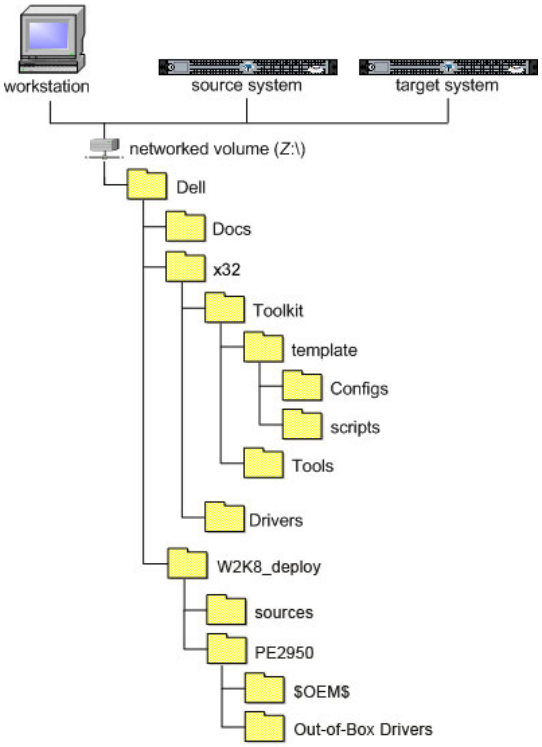
The directory **Toolkit\Systems** in systems running Microsoft Windows and **/opt/dell/toolkit/systems/system name** in systems running Linux acts as a central repository for all the system-specific deployment information. On systems running Linux, the **/systems** folder acts as a repository for the common files required for the systems. On systems running Windows, you can store all profile configuration files for an optimally configured Dell PowerEdge R720 in the directory **Toolkit\Systems\peR720** as shown in the following figure. These files can then be accessed from this directory when deploying multiple PowerEdge R720 systems.

NOTE: If you choose to create a directory structure for the system-specific files that is different from the one suggested in this guide, ensure that you carefully review and edit each sample script (wherever necessary) to reflect the different directory structure.


workstation
source system
target system
networked volume (Z:\)
Dell
Docs
x32
Toolkit
Systems
pe2800
pe2850
template
Configs
scripts
Tools
Drivers
common
winpe3.x
winpe4.x

Operating System Installation Files

The operating system file structure for Windows Server 2008 is shown in the following figure.



Creating Operating System Installation Directories On Systems Running Windows Server 2008 Operating System

NOTE: The following procedures are provided as best practice guidelines for creating operating system distribution folders compatible with DTK sample scripts. For complete instructions on setting up an installation directory structure for an unattended installation, see your operating system documentation.

To deploy on Windows Server 2008 operating systems:

1. Create an operating system-specific directory in the root directory of the network share. For example, **W2K8_deploy**.
2. Create two directories under the **W2K8_deploy** directory: **sources** and <*platform name*>, for example, **PER720**. Copy the contents of the operating system media to **sources**.

   NOTE: The platform name must be a DTK standard name. To know the platform name, run **TKENVSET.BAT** located in the **x:\Dell\32\Toolkit\Template\Scripts** directory, and then run the **SET** command. In the **Details** section, see the **DT_PLATFORM** name.
3. Under **PER720**, create two directories: **OEM** and **Out-of-Box Drivers**.
4. Copy all the drivers to **W2K8_deploy\PER720\ Out-of-Box Drivers**. You can copy each driver to a different sub-folder.
5. Copy **unattend.xml** from **\Toolkit\Template\Configs** to **W2K8_deploy\PER720**.

NOTE: If required, modify the **unattend.xml** for **DiskID** under the tags **DiskConfiguration** and **ImageInstall**. Default value is 0 for **DiskID**.

NOTE: Make sure that **DiskID** specified in the **unattend.xml** is actually available to install the operating system. You can run the **DiskPart** utility to check disk details.

6. Boot your system with DTK Windows PE image.
7. Run the following command to connect the target system to the shared folder **W2K8_deploy**.

   ```
   X:\Dell\Toolkit\Tools net use z: \\<IP address of the management station>
   \W2K8_deploy
   ```

8. From your system, run **TKENVSET.BAT** under **\Toolkit\Template\Scripts**.
9. On your system, set the environment variable **DT_DRIVE=Z:**.
10. Navigate to **Toolkit\Template\Scripts** and run **W2K8INST.BAT**.

Creating Operating System Installation Directories For Systems Running Windows Server 2012 Operating System

NOTE: The following procedures are provided as best practice guidelines for creating operating system distribution folders compatible with DTK sample scripts. For complete instructions on setting up an installation directory structure for an unattended installation, see your operating system documentation.

To deploy on Windows Server 2012 operating systems:

1. Create an operating system-specific directory in the root directory of the network share. For example, **W2K12_deploy**.
2. Create two directories under the **W2K12_deploy** directory: **sources** and **<** *platform name* **>**, for example, **PER720**. Copy the contents of the operating system media to **sources**.

   NOTE: The platform name must be a DTK standard name. To know the platform name, run **TKENVSET.BAT** located in the **x:\Dell\Toolkit\Template\Scripts** directory, and then run the **SET** command. In the **Details** section, see the **DT_PLATFORM** name.

3. Under **PER720**, create two directories: **OEM** and **Out-of-Box Drivers**.
4. Copy all the drivers to **W2K812_deploy\PER720\Out-of-Box Drivers**. You can copy each driver to a different sub-folder.
5. Copy **unattend.xml** from **\Toolkit\Template\Configs** to **W2K12_deploy\PER720**.

   NOTE: If required, modify the **unattend.xml** for `DiskID` under the tags **DiskConfiguration** and **ImageInstall**. Default value is **0** for `DiskID`. Make sure that the `DiskID` specified in the **UnattendW2K12.xml** is actually available to install the operating system. You can run the **DiskPart** utility to check disk details. You can also edit the edit the **MetaData** tag to select the required operating system in **UnattendW2K12.xml**.

6. Boot your system with DTK Windows PE image.
7. Run the following command to connect the target system to the shared folder **W2K12_deploy**.

   ```
   X:\Dell\Toolkit\Tools net use z: \\<IP address of the management station>
   \W2K12_deploy
   ```

8. From your system, run **TKENVSET.BAT** under **\Toolkit\Template\Scripts**.
9. On your system, set the environment variable **DT_DRIVE=Z:**.
10. Navigate to **Toolkit\Template\Scripts** and run **W2K12INST.BAT**.

Creating Operating System Installation Directories For Systems Running Supported Linux Operating Systems

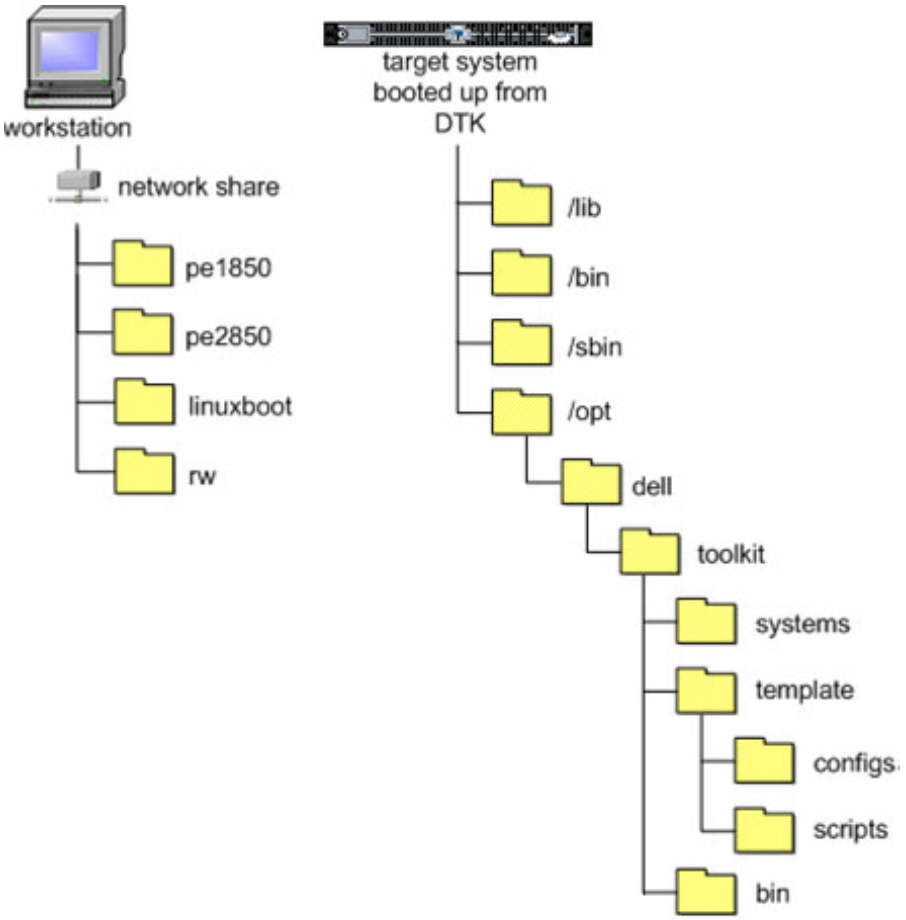
NOTE: The following procedures are provided as a best practices guideline for creating operating system distribution folders compatible with DTK sample scripts. For complete instructions on setting up an installation directory structure for an unattended installation, see your operating system documentation.

For deployment on systems running supported Linux operating systems, copy the ISO images for the operating system installation to a folder on your network (for example, **/linuxboot**). This folder must be accessible after DTK completes its tasks.

Also, copy the following into the **/linuxboot** folder:

- **kernel image** (from the Linux operating system media)
- **initial ramdisk** from the Dell Systems Build and Update Utility on the *Dell Systems Management Tools and Documentation* DVD

For a sample directory structure for deployment on a supported Linux operating system, see the following figure:



NOTE: If you choose to create a directory structure for the Linux installation files that is different from the one suggested in this guide, be sure to carefully review and edit each sample script (where necessary) to reflect the different directory structure.

5

Preparing The Script Files

This chapter describes the steps required to use the Deployment Toolkit (DTK) sample scripts to perform a full deployment on a target system in the Microsoft Windows PE and embedded Linux environment.

There are four main tasks:

- Using The Sample DTK Scripts To Run A Full Deployment
- Editing The Sample Batch Files To Perform A Complete System Deployment
- Using DTK Sample Scripts To Capture Configuration Information
- Preparing The Operating System Installation Scripts

Using The Sample DTK Scripts To Run A Full Deployment

A scripted DTK deployment relies on the master deployment file to integrate all the applicable DTK task scripts and utilities and perform pre-operating system configuration tasks before launching the operating system installation files. The master deployment file, along with the **TKENVSET.BAT** (for Windows) or **tkenvset.sh** (for Linux) scripts, makes up the master set of instructions that drives the deployment process. This file is not provided with the sample scripts.

A sample **DEPLOY.BAT** is given below:

```
::
*****************************************************
::Call TKENVSET.BAT to discover platform information
::as well as set paths and environment variables for
::Deployment Toolkit Tools and Scripts.
*****************************************************
*****************************************************
:: call RAIDCFG.BAT to configure RAID.
::
*****************************************************
*****************************************************
:: Replicate the system settings to the current SUT!.
CALL SYSREP.BAT or DTKRUNALL.BAT.
::
*****************************************************
*****************************************************
::Partitioning the system hard disk. call PARTCFG.BAT.
*****************************************************
*****************************************************
:: Put hdd as first in boot sequence (1-removable, 2-
cdrom, 3-hdd)
:: If the system configuration has changed (i.e. a
::boot device was added, removed, or enabled in BIOS)
::the following line should be modified to reflect the
::new hdd device number.
:: This is calling Windows installer scripts.
*****************************************************
::%DT_TOOLS%\syscfg --bootseq=hdd.emb.0
::call WIN2K8.BAT
```

Editing The Sample Batch Files To Perform A Complete System Deployment

The sample scripts utilize DTK utilities, the additional supporting utilities, the source system configuration files, the target system hardware drivers, and the operating system installation files to perform a full deployment on the target system.

The sample scripts provided with DTK can be edited and customized using any ASCII text editor that conforms to the hardware available on the target systems. While it is recommended that you use DTK sample scripts to automate your deployment tasks, you can create your own system deployment batch files using any ASCII text editor by incorporating the various DTK utilities and commands in the batch file.

NOTE: For more information about scripting batch files and specific batch file commands, see your Windows PE or Linux operating system documentation.

Each of the utilities, scripts, and system configuration files are described below. Depending on your requirements, you may use all or only some of them in your deployment. You can write your master deployment file to perform a full deployment on a system with the help of utilities, scripts, and configuration files. This file is not provided with the sample scripts.

Scripts For Deployment On Systems Running Windows

The following table lists the capture scripts for Windows.

Table 14. Capture Scripts for Windows

Script Name	Description	Associated Files
RACCAP.BAT	Captures RAC settings to a file for Dell Remote Access Controller 4 (DRAC 4).	• TKENVSET.BAT — Provides information to the tasks scripts about the location of the DTK utilities, scripts, and configuration files. • RACADM.EXE — Retrieves RAC information and configure settings.
SYSCAP.BAT	Captures BIOS and Baseboard Management Controller (BMC) configuration settings and saves them to a file.	• TKENVSET.BAT — Provides information to the tasks scripts about the location of the DTK utilities, scripts, and configuration files. • SYSCFG.EXE — Configures the BIOS and BMC settings.
RAIDCAP.BAT	Captures the RAID settings and saves them to a file	• TKENVSET.BAT — Provides information to the tasks scripts about the location of the DTK utilities, scripts, and configuration files. • RAIDCFG.EXE— Retrieves RAID information and configure settings.

Scripts For Deployment On Systems Running Linux

The following table lists the capture scripts for Linux.

Table 15. Capture Scripts for Linux

Script Name	Description	Associated Files
raccap.sh	Captures RAC settings to a file for DRAC 4.	• **tkenvset.sh** — Used by **raccap.sh** to inform the task scripts where to find the DTK utilities, scripts, and configuration files. • **racadm** — Used by **raccap.sh** to retrieve RAC information and to configure settings. The wrapper script, **racadm**, invokes **racadm4**, **racadm5**, or **racadm6** depending on the system configuration. • **syscfg**— Used by **raccap.sh** to configure DRAC.
raidcap.sh	Captures the RAID settings and saves them to a file.	• **tkenvset.sh** — Used by **raidcap.sh** to inform the task scripts where to find the DTK utilities, scripts, and configuration files. • **raidcfg** — Used by **raidcap.sh** to retrieve RAID information and to configure settings.
syscap.sh	Captures BIOS, DRAC, and BMC configuration settings and saves them to a file.	• **tkenvset.sh** — Used by **syscap.sh** to inform the task scripts where to find the DTK utilities, scripts, and configuration files. • **syscfg** — Used by **syscap.sh** to configure the BIOS, DRAC, and BMC settings.

Replication Scripts For Deployment On Systems Running Windows Operating System

The following table lists the replication scripts for Windows.

Table 16. Replication Scripts for Windows

Script Name	Description	Associated Files
RACREP.BAT	Applies the RAC configuration settings saved in a file to the current system.	• **TKENVSET.BAT** — Used by **RACREP.BAT** to inform the task scripts about where to find the DTK utilities, scripts, and configuration files. • **RACADM.EXE** — Used by **RACREP.BAT** to replicate RAC information and to configure settings.
RAIDREP.BAT	Applies the RAID settings saved in a file to the current system.	• **RAIDCFG.EXE** — Used by **RAIDREP.BAT**to retrieve RAID information and to configure settings.
SYSREP.BAT	Applies the BIOS and BMC settings saved in a file to the current system.	• **TKENVSET.BAT** — Used by **SYSREP.BAT** to inform the task scripts where to find the DTK utilities, scripts, and configuration files. • **SYSCFG.EXE** — Used by **SYSREP.BAT** to configure the BIOS and BMC settings.

Replication Scripts For Deployment On Systems Running Linux Operating System

The following table lists the replication scripts for Linux.

Table 17. Replication Scripts for Linux

Script Name	Description	Associated Files
racrep.sh	Applies the RAC configuration settings saved in a file to the current system.	• **tkenvset.sh** — Used by **racrep.sh** to inform the task scripts about where to find the DTK utilities, scripts, and configuration files. • **racadm** — Used by **racrep.sh** to replicate the RAC information and to configure settings. The wrapper script, **racadm** invokes **racadm4** , **racadm5** , or **racadm6** depending on the system configuration. • **syscfg** — Used by **racrep.sh** to configure DRAC 5.
sysrep.sh	Applies the BIOS and BMC settings saved in a file to the current system.	• **tkenvset.sh** — Used by **sysrep.sh** to inform the task scripts about where to find the DTK utilities, scripts, and configuration files. • **syscfg** — Used by **sysrep.sh** to configure the BIOS settings.

Deployment Scripts For Systems Running Windows Operating Systems

The following table lists the deployment scripts for Windows.

Table 18. Deployment Scripts For Windows

Script Name	Description	Associated Files
W2K8INST.BAT	Performs an unattended installation of supported Windows Server 2008 operating systems.	• **TKENVSET.BAT** — Used by **W2K8INST.BAT** and **W2K12INST.BAT** to inform the task scripts about where to find the DTK utilities, scripts, and configuration files.
W2K12INST.BAT	Performs an unattended installation of supported Windows Server 2012 operating systems.	• **SETUP.EXE** — Used by **W2K8INST.BAT** and **W2K12INST.BAT** to install the operating system in conjunction with the **UNATTEND.XML** answer file. • **UNATTEND.XML** — Used by **W2K8INST.BAT** and **W2K12INST.BAT**in conjunction with **SETUP.EXE** to install the operating system files.

Deployment Scripts For Systems Running Linux Operating Systems

The following table lists the deployment scripts for Linux.

Table 19. Deployment Scripts For Linux

Script Name	Description	Associated Files
lininst.sh	Performs an unattended installation of a supported Red Hat Enterprise Linux operating system.	• **tkenvset.sh** — Used by **lininst.sh** to inform the task scripts about where to find the DTK utilities, scripts, and configuration files. • **grub** — Used to install the operating system in conjunction with the unattended answer file. It also uses the following images in conjunction with the unattended answer file to perform unattended operations:

Script Name	Description	Associated Files
		– initial ramdisk — Available on the Dell Systems Build and Update Utility. – kernel image — Available on the Linux operating system media. • **ks.cfg** — Used by **lininst.sh** in conjunction with grub to install the Red Hat Enterprise Linux Server (versions 3 and 4) operating system files. • **ks-rhel5.cfg** — Used by **lininst.sh** in conjunction with grub to install the Red Hat Enterprise Linux Server (versions 5) operating system files. • **ks-rhel6.cfg** — Used by **lininst.sh** in conjunction with grub to install the Red Hat Enterprise Linux Server (versions 6) operating system files.
suseinst.sh	Performs an unattended installation of a supported SUSE Linux Enterprise Server operating system.	• **tkenvset.sh** — Used by **suseinst.sh** to inform the task scripts about where to find the DTK utilities, scripts, and configuration files. • **grub** — Used to install the operating system in conjunction with the unattended answer file. It also uses the following images in conjunction with the unattended answer file to perform unattended operations: – initial ramdisk — Available on the SUSE Linux Enterprise Server operating system media – kernel image — Available on the SUSE Linux Enterprise Server operating system media • **autoinst.xml** — Used by **suseinst.sh** in conjunction with **grub** to install the operating system files. • **autoinst-sles11.xml** — Used by **suseinst.sh** in conjunction with **grub** to install the SUSE Linux Enterprise Server (version 11) operating system files. • **auto-sles11sp2.xml** — Used by **suseinst.sh** in conjunction with grub to install the SUSE Linux Enterprise Server (version 11 SP2) operating system files.

Other Scripts For Systems Running Windows Opearting Systems

The following table lists the other scripts for systems running Windows opearting systems.

Table 20. Other Scripts for Windows

Script Name	Description	Associated Files
TKENVSET.BAT	Sets the path of DTK installation and variables for the system to be deployed and configured. The variables and paths specified in this file must be specified before any of the other supporting scripts can be used.	• **SYSCFG.EXE** — Used by **TKENVSET.BAT** to discover the system type.
RAIDCFG.BAT	Configures the selected RAID controller detected in the system.	• **TKENVSET.BAT** — Used by **RAIDCFG.BAT** to inform the task scripts about where to find the DTK utilities, scripts, and configuration files.

Script Name	Description	Associated Files
		• **RAIDCFG.EXE** — Used by **RAIDCFG.BAT** to retrieve RAID information and configure settings.
PARTCFG.BAT	Creates and populates the Dell utility partition (UP) and the operating system partition on a specified disk.	• **TKENVSET.BAT** — Used by **PARTCFG.BAT** to inform the task scripts about where to find the DTK utilities, scripts, and configuration files. • **DISKPART.EXE** — Provided by Windows; this script is used to create partitions on your disk. When creating a Windows PE image, ensure that the **DISKPART.EXE** tool is present in the Windows PE image you created. This script uses the following files for unattended operations: – **UPINIT.BAT**— Used by **PARTCFG.BAT** to create and populate the Dell utility partition. – **FORMAT.EXE**— Provided by Windows to format the partition.

Other Scripts For Systems Running Linux Operating Systems

The following table lists the other scripts for Linux.

Table 21. Other Scripts for Linux

Script Name	Description	Associated Files
tkenvset.sh	Sets the path for DTK installation and variables for the system to be deployed and configured. The variables and paths in this file must be specified before any of the other supporting scripts can be used.	• **syscfg** — Used by **tkenvset.sh** to discover the system type.
raidcfg.sh	Configures the selected RAID controller detected in the system.	• **tkenvset.sh** — Used by **raidcfg.sh** to inform the task scripts about where to find the DTK utilities, scripts, and configuration files. • **raidcfg** — Used by **raidcfg.sh** to retrieve RAID information and configure settings.
partcfg.sh	Creates and populates the Dell utility partition and the deployment partition on a specified disk. The **partcfg.sh**script uses a **/tmp**folder to store a few temporary files critical to its execution.	• **tkenvset.sh** — Used by **partcfg.sh** to inform the task scripts about where to find the DTK utilities, scripts, and configuration files. • **fdisk** — Used to create partitions on your disk. When creating an embedded Linux ISO image, ensure that **fdisk** is present in the embedded Linux environment you created. This script uses the following file for unattended operations: • **upinit.sh** — Used by **partcfg.sh** to create and populate the Dell utility partition. **NOTE:** Do not make any modifications to the **upinit.sh** script.

Script Name	Description	Associated Files
upinit.sh	Creates and populates the Dell utility partition. **NOTE:** Do not make any modifications to the **upinit.sh** script.	• **tkenvset.sh** — Used by **upinit.sh** to inform the task scripts about where to find the DTK utilities, scripts, and configuration files. • **fdisk** — Used to create partitions on your disk. • **mbr** — File necessary to create a bootable Dell utility partition. • **sysdrmk** — Used by **upinit.sh** to patch the boot sector. • **unzip** — Used by **upinit.sh** to populate the Dell utility partition. • **mke2fs** —Used to create a **ext2** file system. • **upimg.bin** — Contains the zipped contents of the Dell utility partition.

Editing Example Of TKENVSET And tkenvset

The majority of editing required for DTK sample scripts involves the **TKENVSET.BAT** (for Windows) or **tkenvset.sh** (for Linux) script. As a result, the scripting discussion focuses on this script**.** Other task scripts may require edits in certain situations. These optional edits are discussed in Editing DTK Task Scripts. The **TKENVSET.BAT** or **tkenvset.sh** script sets the path of DTK utilities and scripts and sets the variables for the system to be deployed and configured. The variables and paths specified in this file before other supporting scripts are used.

Specifying DTK Installation Paths

The **TKENVSET.BAT** or **tkenvset.sh** sample script is provided with the following default values. These values must be confirmed in some cases, and modified in other cases, before a deployment. Locate the Deployment Toolkit Installation Paths heading to find the following lines:

On systems running Windows:

```
set DT_DRIVE=X:
set DT_PATH=%DT_DRIVE%\Dell\Toolkit
set DT_TOOLS=%DT_PATH%\Tools
set DT_SYSTEMS=%DT_PATH%\Systems
set DT_SCRIPTS=%DT_PATH%\Template\Scripts
set DT_STOP_ON_ERROR=FALSE
```

On systems running Linux:

```
DT_PATH=$dt_drive/opt/dell/toolkit
DT_TOOLS=$dt_path/bin
DT_SYSTEMS=$dt_path/systems
DT_SCRIPTS=$dt_path/template/scripts
DT_STOP_ON_ERROR=FALSE
DT_DRMK=$dt_path/opt/dell/drmk
```

Confirming Or Editing DTK Installation Path Variables

Perform the following steps to confirm and/or edit the installation path variables:

1. On systems running Windows, set the variable that defines the path to the full set of DTK deployment files (**DT_DRIVE**). On systems running Windows, this variable is set to **X:** by default in the sample script. For Windows, this variable is set to **X:** by default in the sample script. On systems running Linux, this variable is set to

$dt_drive/opt/dell/toolkit by default in the sample script. Edit this drive letter to reflect the actual drive letter for DTK deployment files, if necessary.

2. Set the variable that defines the path to the full set of DTK deployment files **(DT_PATH)**. This variable is set to **%DT_DRIVE%\Dell\Toolkit** (Windows) or **$dt_drive/opt/dell/toolkit** (Linux) by default in the sample script. Edit this path to reflect the actual location of DTK deployment files, if necessary.
3. Set the variable that defines the directory containing DTK utilities (**DT_TOOLS**). This variable is set to **%DT_PATH% \Tools** (Windows) or **$dt_path/bin** (Linux) by default in the sample script. Edit this path to reflect the actual location of DTK utilities, if necessary.
4. Set the variable that defines the directory containing the configuration files for your Dell systems (**DT_SYSTEMS**). This variable is set to **%DT_PATH%\Systems** (Windows) or **$dt_path/systems** (Linux) by default in the sample script. Edit this path to add a subdirectory for the Dell system you plan to deploy. For example, **$dt_path/systems/pe**R720.
5. Set the variable that defines the directory containing DTK deployment scripts (**DT_SCRIPTS**). This variable is set to **%DT_PATH%\Template\Scripts** (Windows) or **$dt_path/template/scripts** (Linux) by default in the sample script. Edit this path to reflect the actual location of DTK deployment scripts, if necessary.
6. Set the variable that indicates whether you want the deployment process to exit when any error is returned (**DT_STOP_ON_ERROR**). This variable is set to **FALSE** by default in the sample script, indicating that errors are handled by the **ERRHNDL.BAT** (Windows) or **errhndl.sh** (Linux) script. Set the value to **TRUE** only if you want the deployment to exit when any error is returned.
7. For Linux, set the variable that defines the directory containing the DRMK system files necessary to create a Dell utility partition (**DT_DRMK**). This variable is set to **$dt_path/opt/dell/drmk** by default in the sample script. Edit this path to reflect the actual location of the DRMK files, if necessary.

Specifying The SYSCFG Variables

The SYSCFG replication file is set to **syscfg.ini** by default. If you used the **SYSCAP.BAT** (for Windows) or **syscap.sh** (for Linux) sample script to generate the file, there is no need to edit these default settings. If you have made any modifications to the sample scripts or to the suggested directory structure, you must confirm that **syscfg.ini** is correctly named and that the path is set correctly for deployment.

To edit the SYSCFG capture variables, locate the following lines:

In **SYSCAP.BAT** on systems running Windows:

```
set DT_SYS_CAPFILE=%DT_SYSTEMS%\%DT_PLATFORM%\syscfg.ini
```

In **syscap.sh** on systems running Linux:

```
set DT_SYS_CAPFILE=$dt_systems/$dt_platform/syscfg.ini
```

To edit the SYSCFG replication variables, locate the following lines:

In **SYSREP.BAT** on systems running Windows::

```
set DT_SYS_REPFILE=%DT_SYSTEMS%\%DT_PLATFORM%\syscfg.ini
```

In **syscap.sh** on systems running Linux:

```
DT_SYS_REPFILE=$dt_systems/$dt_platform/syscfg.ini
```

NOTE: If the replication requires a change in the memory redundancy mode, you must reboot the target server and execute the script again to complete the replication.

Specifying The RAC Configuration Variables

The IP address for your target system RAC is set in the sample script as **10.98.8.121** by default. Edit this value to reflect the actual IP address of your target system RAC, if applicable. Additionally, the RAC configuration file used to configure your RAC is set by default to **raccfg.ini**. If you use the **RACCAP.BAT** or **raccap.sh** sample script to generate the **raccfg.ini** file, there is no need to edit this default setting. If you have made any modifications to the sample scripts or to the suggested directory structure, you must confirm that **raccfg.ini** is correctly named and that the path is correctly set for your deployment.

To edit the RAC configuration variables, locate the following lines:

In **RACCAP.BAT**:

```
set DT_RAC_CAPFILE=%DT_SYSTEMS%\%DT_PLATFORM%\raccfg.ini
```

In **raccap.sh:**

```
DT_RAC_CAPFILE=$dt_systems/$dt_platform/raccfg.ini
```

To edit the RAC configuration variables, locate the following lines:

In **RACREP.BAT**:

```
:: RAC section
set DT_NICCFG=10.98.8.121
set DT_RAC_CAPFILE=%DT_SYSTEMS%\%DT_PLATFORM%\raccfg.ini
```

In **racrep.sh**:

```
# RAC section
DT_NICCFG=10.98.8.121
set DT_RAC_CAPFILE=$dt_systems/$dt_platform/raccfg.ini
```

Specifying Variables Used To Create Disk Partitions And Install The Dell Utility Partition

NOTE: If the primary deployment state (**DT_PARTN**) is set to **OFF**, the partitioning deployment state subsection is skipped.

The **PARTCFG.BAT** (Windows) or **partcfg.sh** (Linux) sample script provides the following disk partitioning default values. These values must be confirmed in some cases, and modified in other cases, before performing a deployment. Locate the **partn** label to find the following lines:

In **PARTCFG.BAT**:

```
set DT_PARTN_UP=ON
set DT_MOUNT=H

:: Default Hard Drive
set DT_HD=0

:: PARTCFG section
set DT_OS_SIZE=4000
set DT_OS_FSTYPE=FAT32

if not "%DT_PARTN_UP%"=="ON" goto ospart

:: DELLUP section
set DT_UP_SIZE=32
set DT_UP_IMAGE=%DT_SYSTEMS%\UPIMG.BIN

:: RAIDCFG section:: default size for virtual disk
set DT_VD_SIZE=10480
:: default size for virtual disk
```

In **partcfg.sh**:

```
DT_PARTN_UP=on
DT_MOUNT=/tmp/up

# Default Hard Drive
DT_HD=/dev/sda

# PARTCFG section
set DT_OS_SIZE=4000
# DELLUP section
set DT_UP_SIZE=32
set DT_UP_IMAGE=$dt_systems/upimg.bin
```

```
# RAIDCFG section
DT_VD_SIZE=10480
# default size for virtual disk
```

CAUTION: DT_MOUNT specifies the mount point of the Dell utility partition. If your RAMDRIVE, hard drive, or other device already owns this designation, mount does not override the setting and the utility partition is not created.

Editing The Default Partition Configuration Values

Perform the following steps to edit the default partition configuration default values, as necessary:

1. Edit the default utility partitioning variable (**DT_PARTN_UP=ON**), if necessary. This variable sets if you want to install the Dell utility partition.

   NOTE: It is recommended that you install the Dell utility partition.
2. Edit the default variable value for the default hard drive (**DT_HD=0** for Windows and **DT_HD=sda** for Linux), if applicable.

   NOTE: Ensure that the default variable value for **DT_HD** is set to a valid non-removable disk, and not to removable media such as virtual media.
3. Edit the default variable value (in MB) for the primary operating system partition (**DT_OS_SIZE=4000**), if applicable.
4. Edit the default variable value for file system type (**DT_OS_FSTYPE=FAT32**), if applicable.

   NOTE: The valid values are **FAT32** and **NTFS**.
5. If the **DT_PART_UP** variable is set to **ON**, edit the default variable value (in MB) for the utility partition (**DT_UP_SIZE=32**), if applicable.
6. If the **DT_PART_UP** variable is set to **ON**, edit the default variable value for the utility partition installation package location (`DT_UP_IMAGE=%DT_SYSTEMS%\UPIMG.BIN` for Windows and `DT_UP_IMAGE=$dt_systems/upimg.bin` for Linux), if applicable.

   NOTE: Edit the **UPIMG_BIN** variable only if you have changed the name or path of the utility partition installation package.

Editing DTK Task Scripts

The task scripts called by the master batch file do not require any edits (with the following exceptions) if the default variable values are utilized in the **TKENVSET.BAT** or **tkenvset.sh** scripts. However, if you make any changes to the default values set by the **TKENVSET.BAT** or **tkenvset.sh**, you must make the same edits to the matching default values set in the appropriate task scripts. Locate the **USER MODIFICATION REQUIRED** heading in the task scripts to find the variable values you may need to edit. Each of the following task scripts can be called by the user-written master batch script during a full deployment:

- **RACREP.BAT** or **racrep.sh**
- **RAIDCFG.BAT** or **raidcfg.sh**
- **SYSREP.BAT** or **sysrep.sh**
- **PARTCFG.BAT** or **partcfg.sh**

RACREP.BAT Or racrep.sh

The **RACREP.BAT** or **racrep.sh** sample script applies RAC options to the target system based on configuration information that has been saved in the configuration file. User input is optional, depending on the variables set in the **RACREP.BAT** or **racrep.sh** sample script. On systems running Windows, **RACREP.BAT** script uses **RACADM.EXE** to configure RAC in the target system. On systems running Linux the **racrep.sh** script uses **racadm** wrapper script or the

syscfg utility (depending on the version of RAC on your system) to configure RAC in the target system. The **raccfg.ini** is passed as the first parameter to these scripts. If this parameter is not passed, the default variables values used in these scripts are set in the **TKENVSET.BAT** or **tkenvet.sh** scripts. You can specify the IP address to be used to configure the RAC as an optional parameter. The **raccfg.ini** file is generated with the **RACCAP.BAT** or **raccap.sh** sample script. For information about using the **RACCAP.BAT**or **raccap.sh** sample script to capture RAC configuration information, see RACCAP.BAT Or raccap.sh.

NOTE: For *x9xx* or later systems, **racrep.sh** and **sysrep.sh** have duplicate functions. If you run **racrep.sh** followed by **sysrep.sh**, the latter overwrites the settings of the former.

RAIDCFG.BAT Or raidcfg.sh

The **RAIDCFG.BAT** or **raidcfg.sh** sample script configures RAID controllers detected on the system. User input is optional to run this script. The input parameter is a name or path to a log file. This parameter is not set to any value by default, and no log file is generated. The default variable values used in this script are set in the **TKENVSET.BAT** or **tkenvest.sh** scripts. This script uses the **RAIDCFG.EXE** or **raidcfg.sh** utility to automatically configure the detected controllers on your target system. For example, for the first controller discovered, if **RAIDCFG.EXE** or **raidcfg.sh** discovers only one attached hard drive, the script tries to create RAID 0; if two hard drives are discovered, a RAID 1 configuration is enabled; if three or more hard drives are discovered, a RAID 5 configuration is enabled.

NOTE: On the detected controller, a RAID virtual disk is created only if the array disks connected to that particular controller have space available and, where applicable, are not part of already existing array group.

SYSREP.BAT Or sysrep.sh

The **SYSREP.BAT** or **sysrep.sh** sample script applies SYSCFG options to the target system based on configuration information that has been saved in a configuration file. User input is optional to run this script, depending on the variable settings in the **SYSREP.BAT** or**sysrep.sh** scripts. This script uses the **SYSCFG** utility to configure the BIOS and BMC in the target system using the configuration file **syscfg.in**. The**syscfg.in** is the first parameter passed to this script. If this parameter is not passed, the default variables values used in this script are set in the **SYSREP.BAT** or **sysrep.sh** scripts. The **syscfg.ini** file is generated with the **SYSCAP.BAT** or **syscap.sh** sample script. For information about using the **SYSCAP.BAT** or **syscap.sh** sample script to capture BIOS and BMC settings, see SYSCAP.BAT Or syscap.sh. The second parameter is a name or path to a log file. This parameter is not set to any value by default, and no log file is generated.

NOTE: If the replication requires a change in the memory redundancy mode, you must reboot the target server and execute the script again to complete the replication.

NOTE: For *x9xx* or later systems, **racrep.sh** and **sysrep.sh** have duplicate functions. If you run **racrep.sh** followed by **sysrep.sh**, the latter overwrites the former settings.

PARTCFG.BAT Or partcfg.sh

The **PARTCFG.BAT** or **partcfg.sh** sample script creates and populates the Dell utility partition using **UPINIT.BAT** or **upinit.sh** and creates the deployment partition on a specified disk. For more information about using the UPINIT.BAT or **upinit.sh** tool, see UPINIT.BAT Or upinit.sh. User input is not required to run this script. The default variable values used in this script are set in the **TKENVSET.BAT** or **tkenvset.sh** scripts.

NOTE: Ensure that **DT_HD** is set to the required disk to be configured before you execute the **PARTCFG.BAT** or **partcfg.sh** script.

Using DTK Sample Scripts To Capture Configuration Information

The SYSCFG and RACADM utilities can read the source system configuration and duplicate that configuration on a target system using a sample DTK script file. These configuration files are used during a full deployment to configure the BIOS, BMC, and RAC settings on the target system. The following sections provide instructions for:

- Capturing The System BIOS And BMC Configuration With The SYSCAP Or syscap Sample Script
- Capturing The RAC Configuration With The RACCAP Or raccap Sample Script

Capturing The System BIOS And BMC Configuration With The SYSCAP Or syscap.sh Sample Script

You can replicate the system BIOS and BMC configurations from a source Dell system to an identical target Dell system. For example, from one PowerEdge R720 to another Dell PowerEdge R720.

NOTE: For the script to complete successfully, the source system must have access to the network share that includes the pre-built DTK directory structure.

SYSCAP.BAT Or syscap.sh

The **SYSCAP.BAT** or **syscap.sh** sample script captures BIOS and BMC configuration settings and saves them to a specified location. An optional user input can be passed as a parameter. You can provide the path or file name as an input to this script. The default variable values used in this script are set to the correct default values to perform a scripted deployment. The **SYSCAP.BAT** or **syscap.sh** script uses the SYSCFG utility to retrieve BIOS and BMC configuration settings from the source system and to save the settings in a file named **syscfg.ini** in the **Z:\Dell\Toolkit \Systems\<target system>** directory.

Running The SYSCAP.BAT Or Syscap.sh

To run the **SYSCAP.BAT** or **syscap.sh** sample script to capture the BIOS and BMC configuration information in the **syscfg.ini** file:

1. Edit the **startnet.cmd** file (on your Windows PE image) or start-up script file (on your embedded Linux image) to call the **SYSCAP.BAT** or **syscap.sh** script after the network share with the pre-built DTK directory structure is mounted. For example, add:

   On systems running Windows:

   ```
   call Z:\Dell\Toolkit\Template\Scripts\SYSCAP.BAT
   ```

   On systems running Linux:

   ```
   /opt/dell/toolkit/template/scripts /syscap.sh
   ```

2. Boot the source system into Windows PE or embedded Linux.

   The script runs and saves the configuration information in the **\Toolkit\Systems\<target system>** directory in Windows and **/opt/dell/toolkit/systems/<target system>** directory in Linux.

3. Edit the **startnet.cmd** file (Windows PE image) or start-up script file (embedded Linux image) to remove the text you added in step 1.

   NOTE: For *x9xx* or later systems, **raccap.sh** and **syscap.sh** have duplicate functions. If you run **raccap.sh** followed by **syscap.sh**, the latter overwrites settings of the former.

Capturing The RAC Configuration With The RACCAP Or raccap.sh Sample Script

If your system has a RAC, you can replicate the RAC configuration from the RAC of a source Dell system to an identical target Dell system with an identical RAC. For example, from one PowerEdge R720 with iDRAC7 to another PowerEdge R720 with iDRAC7.

NOTE: For the script to complete successfully, the source system must have access to the network share that includes the pre-built DTK directory structure.

Running RACCAP.BAT Or raccap.sh

To run the **RACCAP.BAT** or **raccap.sh** sample script to capture the RAC configuration information in the raccfg.ini file:

1. Edit the **startnet.cmd** file (on your Windows PE image) or start-up script file (on your embedded Linux image) to call the **RACCAP.BAT** or **raccap.sh** script after the network share with the pre-built DTK directory structure is mounted. For example, add:

   In Windows:

   ```
   Z:\Dell\x32\Toolkit\Template\Scripts\RACCAP.BAT\Z:\raccfg.ini
   ```

   In Linux:

   ```
   /opt/dell/toolkit/template/scripts/raccap.sh
   ```

2. Boot the source system into Windows PE or embedded Linux.

   The script runs and saves the configuration information in the **\Toolkit\Systems\<target system>** directory in Windows and **/opt/dell/toolkit/systems/<target system>** directory in Linux.

3. Edit the **startnet.cmd** file (Windows PE image) or start-up script file (embedded Linux image) to remove the text you added in step 1.

   NOTE: The RAC configuration file is referred to as the **.cfg** file in the *Integrated Dell Remote Access Controller 6 (iDRAC6) Enterprise for Blade Servers User Guide and the Dell Remote Access Controller 4 User's Guide*. See these guides for additional information about manually creating a RAC configuration file.

   NOTE: Do not replicate the RAC IP address when creating a **.cfg** file. Replicating the RAC IP address can leave the system inaccessible because multiple systems are configured with the same IP address.

   NOTE: For *x9xx* or later systems, **raccap.sh** and **syscap.sh** have duplicate functions. If you run **raccap.sh** followed by **syscap.sh**, the latter overwrites the former settings.

RACCAP.BAT Or raccap.sh

The **RACCAP.BAT** or **raccap.sh** sample script captures RAC configuration settings and saves them to a specified location. An optional user input can be passed as a parameter. You can provide the path or file name as an input to this script. The default variable values used in this script are set to the correct default values to perform a scripted deployment. The **RACCAP.BAT** or **raccap.sh** script uses the **RACADM.EXE** utility in Windows and the **racadm** wrapper script or **syscfg** utility (depending on the RAC version on your system) to retrieve RAC configuration settings from the source system and saved it in a file named **raccfg.ini** at the **\Toolkit\Systems\<target system>** directory in Windows and **/opt/dell/toolkit/systems/<target> system** directory in Linux.

NOTE: Ensure that you run **RACCAP.BAT** in a writable environment. If you run **RACCAP.BAT** in a read-only environment, DTK displays the message **<** *filename.ini* **> has been generated** even though no new file is created.

UPINIT.BAT Or upinit.sh

The **UPINIT.BAT** or **upinit.sh** is a tool used to create and populate the Dell utility partition.

NOTE: You can use this script to create a new Dell utility partition only if there are no existing partitions on your hard disk.

You can also use this script to upgrade or downgrade an existing Dell utility partition. Upgrades or downgrades can be made only to existing Dell utility partitions to capture BIOS and BMC settings using the overwrite option.

CAUTION: The existing Dell utility partition must be the first primary partition on the hard disk. Also, the Dell utility partition must be greater than or equal to 32 MB and less than 2 GB.

For details on the command line arguments to be used for **UPINIT.BAT** or **upinit.sh**, see the *Dell Deployment Toolkit Command Line Interface Reference Guide T **dell.com/support/manuals***.

Preparing The Operating System Installation Scripts

The W2K8INST.BAT And W2K12INST.BAT script (on supported Windows operating systems), lininst.sh sample script (on supported Red Hat Enterprise Linux Server operating system) and suseinst.sh sample script (on supported SUSE Linux Enterprise Server operating system) are used to perform an installation. The variables and paths must be specified before the operating system can be installed.

CAUTION: It is recommended that you consult your Windows or Red Hat Enterprise Linux Server operating system documentation and unattended deployment documentation to develop a thorough understanding of the unattended installation process before attempting to perform a full scripted deployment.

NOTE: While installing Linux, ensure that you install **grub**in the boot partition. Otherwise, you cannot boot to the utility partition by pressing the <F10> key during reboot.

W2K8INST.BAT And W2K12INST.BAT

The **W2K8INST.BAT** and **W2K12INST.BAT** sample scripts are used to perform an unattended installation of the Windows Server 2008 and Windows Server 2012 operating systems respectively. The variables and paths must be specified before the operating system can be installed.

CAUTION: It is recommended that you consult your Windows operating system documentation and unattended deployment documentation to develop a thorough understanding of the Windows unattended installation process before attempting to perform a full scripted deployment.

The syntax for **W2K8INST.BAT** and **W2K12INST.BAT** are **w2k8inst.bat** and **w2k12inst.bat** respectively.

Utilities Used

SETUP.EXE — Used to install the operating system in conjunction with the **UNATTEND.XML** answer file.

External Dependencies

W2K8INST.BAT and **W2K12INST.BAT** scripts uses the uses two environment variables: `DT_PLATFORM` and `DT_DRIVE`.

The `DT_PLATFORM` variable is set by **TKENVSET.BAT** using platform discovery.

The `DT_DRIVE` denotes the network drive letter to the network shared path containing the Windows Server 2008 or the Windows Server 2012 operating system source and the **UNATTEND.XML** file.

The **W2K8INST.BAT** and W2K12INST.BAT scripts uses the **UNATTEND.XML** file in conjunction with **SETUP.EXE** utility to install the operating system files. For more information about preparing the **UNATTEND.XML** file, see UNATTEND.XML.

Specifying The Windows Installation Variables And Installation Paths

To edit the Windows operating system installation variables, locate the following lines in **W2K8INST.BAT** or **W2128INST.BAT**:

```
:: Location of Unattend.xml file

set DT_UNATTEND=%DT_SYSTEMS%\%DT_PLATFORM%\unattend.xml
```

UNATTEND.XML

The **UNATTEND.XML** file is the default name of the answer file that you use to automate Windows setup during an unattended installation. A sample copy of the **UNATTEND.XML** file is provided as part of the DTK download in the **\Toolkit\Template\Configs** directory.

UNATTEND.XML contains the headings and parameters that instruct Windows setup to perform various configuration tasks. In **UNATTEND.XML**, you want to install. Before a full deployment can be run, you must edit all appropriate headings and parameters, as necessary.

NOTE: See your operating system documentation for instructions on modifying the options in the operating system unattended installation file to customize the scripted installation of your operating system.

In Windows Server 2008 operating system, when you have completed all necessary modifications to the **UNATTEND.XML** file, save the completed file in the **\Toolkit\W2K8_deploy\<target system>** directory.

After you have successfully edited your deployment scripts, you are ready to create your deployment media and run your deployment. For instructions on creating the deployment media and running a full deployment, see Running The Deployment Scripts.

lininst

The **lininst.sh** sample script is used to perform an unattended installation of a supported Red Hat Enterprise Linux Server operating system. The variables and paths must be specified before the operating system can be installed.

Utilities Used By lininst

grub — Used in conjunction with the **kernel image** and **initial ramdisk** files, along with the **unattended** answer file to install the operating system.

Variables Used:

- **DT_OS_DISK** — Sets the disk on which to deploy the Red Hat Enterprise Linux Server operating system.
- **DT_OS_SRC** — Sets the location from where to copy the **kernel image** and **initial ramdisk** files.

External Dependencies Of lininst.sh

The **DT_PLATFORM** variable is set during deployment by **tkenvset.sh** using platform discovery.

The **lininst.sh** script uses the **unattended answer** file in conjunction with the **grub** utility and **kernel image** and **initial ramdisk** files to install the operating system files. For more information about preparing the **unattended answer** file, see Answer Files.

Answer Files (ks.cfg, ks-rhel5.cfg, And ks-rhel6.cfg)

Depending on the version of the Red Hat Enterprise Linux Server operating system, the **ks.cfg**, **ks-rhel5.cfg**, or **ks-rhel6.cfg** files are the default names of the answer files that you use to automate an unattended Red Hat Enterprise Linux Server installation. A sample copy of the **ks.cfg/ks-rhel5.cfg/ks-rhel6.cfg** files are provided as part of the DTK download in the **/opt/dell/toolkit/template/configs** directory. The **ks.cfg/ks-rhel5.cfg/ks-rhel6.cfg** files contain the headings and

parameters that instruct the installation utility to perform various configuration tasks. Before a full deployment can be run, you must edit all appropriate headings and parameters, as necessary.

NOTE: The sample file provided to install Red Hat Enterprise Linux Server (version 5) is named **ks-rhel5.cfg**. To use the sample **ks-rhel5.cfg** file to install Red Hat Enterprise Linux Server (version 5) using **lininst.sh**, ensure that you rename the file to **ks.cfg**.

NOTE: See your operating system documentation for instructions on modifying the options in the **ks.cfg/ks-rhel5.cfg/ks-rhel6.cfg** file to customize the scripted installation of your operating system.

When you have completed all necessary modifications of the **ks.cfg/ksrhel5. cfg/ks-rhel6.cfg** file, save the completed file in the **/opt/dell/toolkit/systems/<target system>** directory.

Installing Linux Using lininst.sh

NOTE: See your operating system documentation for instructions on modifying the options in the **ks.cfg/ks-rhel5.cfg/ ks-rhel6.cfg** file to customize the scripted installation of your operating system.

Perform the following steps to install Red Hat Enterprise Linux using the **lininst.sh** script:

1. On the source system, navigate to the location of the required Red Hat Enterprise Linux operating system image.
2. Create NFS share using the following command: `vi /etc/exports`.
3. Share **/home/rhelshare**.

   NOTE: If the folder you want to share is **/home**, then the **/etc/exports** file must contain the **"/home *(rw)"** line.
4. Restart the NFS service.
5. Copy the operating system image to the NFS share.
6. Copy **initrd.img** for Red Hat Enterprise Linux to NFS share.
7. Copy **vmlinuz** from Red Hat Enterprise Linux opearting system image to NFS share.
8. Boot DTK on the target machine.
9. Create the virtual disk using the **raidcfg** command.
10. Run the **partcfg.sh** (located at **/opt/dell/toolkit/template/scripts/**) script to create the Dell utility partition and OS partition.

    Ensure that the correct partition is set to **DT_HD**.
11. Create a directory under **/tmp**. For example, **os_src**.
12. Mount the NFS Share to **/tmp/os_src**.
13. Copy **ks.cfg** file from **/opt/dell/toolkit/template/configs** to NFS share on the source machine.
14. Edit **ks.cfg** on the source system. Set the NFS IP Address where Red Hat Enterprise Linux images are available and the NFS share path.
15. Export the environment variables as follows:
    a) Run export `DT_OS_DISK=/dev/sda`, the disk device on which the Dell utility partition was created.
    b) Run export `DT_OS_SRC=/tmp/os_src`, where **vmlinuz**, **initrd.img**, and **ks.cfg** are available.
    c) Run export `DT_HD=/dev/sda`.
16. From **/opt/dell/toolkit/template/scripts**, run **/lininst.sh** script.
17. The server reboots to the **grub** prompt. The Red Hat Enterprise Linux operating system unattended installation proceeds.

    For more information on installing, see **en.community.dell.com/techcenter/systems-management/w/wiki/1772.dell-openmanage-deployment-toolkit.aspx**.

suseinst

The **suseinst.sh** sample script is used to perform an unattended installation of a supported SUSE Linux Enterprise Server operating system. The variables and paths must be specified before the operating system can be installed.

CAUTION: It is recommended that you consult your SUSE Linux Enterprise Server operating system documentation and unattended deployment documentation to develop a thorough understanding of the unattended installation process before attempting to perform a full scripted deployment.

NOTE: While installing Linux, ensure that you install **grub** in the boot partition. Otherwise, you cannot boot to the utility partition by pressing the **<F10>** key during reboot.

Utilities Used By suseinst

grub — Used in conjunction with the **kernel image** and **initial ramdisk** files, along with the **unattended** answer file to install the operating system.

Variables Used

- **DT_OS_DISK** — Sets the disk on which to deploy the SUSE Linux Enterprise Server operating system.
- **DT_OS_SRC** — Sets the location from where to copy the **kernel image** and **initial ramdisk** files.
- **DT_OS_PART** — Sets the valid partition to install the grub files.
- **DT_OS_IMG_PATH** — Sets the network share where the operating system files are located.
- **DT_SUSE_AUTOINST** — Sets the network share from where the **autoinst.xml** file can be accessed.

External Dependencies Of suseinst.sh

The **DT_PLATFORM** variable is set during deployment by **tkenvset.sh** using platform discovery.

The **suseinst.sh** script uses the **unattended answer** file in conjunction with the **grub** utility and **kernel image** and **initial ramdisk** files to install the operating system files.

autoinst.xml For SUSE Linux Enterprise Server

The **autoinst.xml** file is the default name of the answer file that you use to automate an unattended SUSE Linux Enterprise Server installation. A sample copy of the **autoinst.xml** file is provided as part of DTK download in the **/opt/dell/toolkit/template/configs** directory. The **autoinst.xml** file contains the headings and parameters that instruct the installation utility to perform various configuration tasks. Before a full deployment can be run, you must edit all appropriate headings and parameters, as necessary.

NOTE: After installing SUSE Linux Enterprise, change the login password by editing the user node of the root user in **autoinst.xml** . You can also encrypt the password file by changing `false` to `true` in `<encrypted config:type="boolean">false</encrypted>`.

NOTE: See your operating system documentation for instructions on modifying the options in the **autoinst.xml** file to customize the scripted installation of your operating system.

When you have completed all necessary modifications of the **autoinst.xml** file, save the completed file in the **/opt/dell/toolkit/systems/<target system>**directory.

After you have successfully edited your deployment scripts, you are ready to create the deployment media and run the deployment. For instructions on creating the deployment media and running a full deployment, see Deployment Using Dell Provided Embedded Linux .

NOTE: During an unattended SUSE Linux Enterprise Server installation, if the installer lists out missing packages, delete the listed packages from **autoinst.xml**.

6

Running The Deployment Scripts

This chapter provides best practices, procedures, and scenarios for using the Deployment Toolkit (DTK) to perform pre-operating system configuration tasks and to install supported operating systems on supported Dell systems.

After you have populated the deployment directory structure with all necessary files and carefully edited the deployment scripts and configurations files, you are ready to begin the final stage of the deployment process. Before running the deployment, however, you need to create specific deployment media to facilitate your chosen deployment method. The common deployment scenarios are:

- Media Based Local Deployment For Windows
- Media Based Local Deployment With Networking Enabled For Windows
- Deployment Using Removable Boot Media With A Network Connection For Windows
- Deployment Using Removable Boot Media Without A Network Connection Or Media Based Connection For Windows
- Network Based Deployment For Windows
- Using A Third Party Deployment Solution Framework For Windows
- Deployment Using Dell Provided Embedded Linux
- Deployment Using Customized Embedded Linux
- Using A Third-Party Deployment Solution Framework For Linux

Media-Based Local Deployment For Windows

This deployment method is easy to assemble, but is the least flexible. Any change to the system configuration, for example, requires the creation of a new bootable media. This method does not provide a writable media, making it necessary to use predefined and tested configuration files.

For the media-based deployment methods, the following tasks need to be completed:

- Creating And Customizing Images For Windows PE 3.0
- Creating And Customizing Images For Windows PE 4.0
- Integrating The DTK Directory Structure
- Creating A Bootable Media For WIN PE 3.0
- Running The Image

Creating And Customizing Images For Windows PE 3.0

If you are using Windows PE 3.0, download Windows Automated Installation Kit (Windows AIK) from **microsoft.com**. By default, Windows AIK is copied to the **C:\Program Files\Windows AIK** directory. You must include the Dell-customized drivers and instrumentation drivers into your Windows PE media-based operating system.

Creating And Customizing Images For Windows PE 4.0

If you are using Windows PE 4.0, download Windows Assessment and Deployment Kit (ADK) from **microsoft.com**. By default, Windows ADK is copied to the **C:\Program Files\Windows Kits** directory.

Integrating DTK Directory Structure

DTK provides **WINPE3.0_driverinst.bat** (for Windows PE 3.0) and **WINPE4.0_driverinst.bat** (for Windows PE 4.0) , to pre-install the Dell drivers into a base Windows PE 3.0 or Windows PE 4.0 image (**winpe.wim**). To execute this script in 32-bit or 64-bit systems:

NOTE: Make sure you have administrator privileges before running the scripts.

1. At the command prompt, change the directory to the location of the script (**WINPE3.0_driverinst.bat** for Windows PE 3.0 and **WINPE4.0_driverinst.bat** for Windows PE 4.0).

   Example for 64-bit supported systems:

   - **cd C:\Dell\x64\Drivers\winpe3.x**
   - **cd C:\Dell\x64\Drivers\winpe4.x**

   Example for 32-bit supported systems:

   - **cd C:\Dell\x32\Drivers\winpe3.x**
   - **cd C:\Dell\x32\Drivers\winpe4.x**

2. Execute the script.

   On Windows PE 3.0:

   ```
   WINPE3.0_driverinst.bat <WIMPATH> <DTKPATH>
   ```

   On Windows PE 4.0:

   ```
   WINPE4.0_driverinst.bat <WIMPATH> <DTKPATH>
   ```

   where <*WIMPATH*> is the destination path to create the directory structure for Windows PE 3.0 and Windows PE 4.0 respectively and <*DTKPATH*> is the path for the Dell drivers in the extracted DTK toolkit. For example:

   On Windows PE 3.0:

   ```
   WINPE3.0_driverinst.bat C:\winpe_30
   C:\DELL\x64\DRIVERS
   ```

   On Windows PE 4.0:

   ```
   WINPE4.0_driverinst.bat C:\winpe_40
   C:\DELL\x64\DRIVERS
   ```

   This pre-installs the Dell drivers into **winpe.wim**. The successful execution of the above commands creates a bootable ISO image for Windows PE 4.0 at <*WIMPATH*>. For creating a bootable media for Windows PE 3.0, see Creating A Bootable Media For WIN PE 3.0.

   NOTE: The destination folder (**C:\winpe_30 for Windows PE 3.0** and **C:\winpe_40** for Windows PE 4.0) is created as part of the process, and must not be an existing directory. The destination path and the path to the Dell drivers must not contain any blank space.

Creating A Bootable Media For Windows PE 3.0

To create a bootable media:

1. Click **Start**, navigate to **All Programs** → **Microsoft Windows AIK** .
2. Click **Windows PE Tools Command Prompt** to open a command prompt window.
3. Navigate to **C:\program files\Windows AIK\Tools\x32** or **C:\program files\Windows AIK\Tools\amd64** directory based on the system.
4. Run the following command: `oscdimg -n -bc:\winpe_30\etfsboot.com c:\winpe_30\ISO c:\winpe_30\WinPE3.0.iso`

   WinPE3.0 iso, a media bootable ISO image is created.

You can use any CD or DVD burning software to burn the image onto a CD or DVD. After burning the ISO image, ensure that it boots from the CD or DVD drive for all the supported Dell systems you plan to deploy. After it boots, you are advised to test all the tools and scripts on these systems to make sure that the integration is successful and that there are no issues with hardware components not being recognized.

Running The Image

You are now ready to use your bootable media to access the deployment components from the directory structure on the media:

1. Boot the target system with the bootable deployment media.
2. Execute the master batch file, which calls individual task scripts and utilities from the media to complete the deployment process.

Media-Based Local Deployment With Networking Enabled For Systems Running Windows

This method provides greater flexibility and is highly recommended in large deployments. The prerequisites are the availability of network bandwidth and all target systems connected to the network.

NOTE: DTK network-based deployment is not supported from mapped Novell NetWare systems.

This deployment method is easy to assemble and provides great flexibility in making changes to the scripts and configuration files. Any change to the system configuration, for example, does not require re-creation of the bootable media. This method also provides access to the remote share as writable media. Hence, the configuration files captured during the deployment automation process can be saved to this remote share.

Bootable Windows PE Media With Networking Enabled

This process includes:

- Creating a bootable Windows PE media with the appropriate network drivers. This step enables networking services to start, get an IP address, and bring the target system into a functioning network.

  NOTE: For more details on how to create a customized version of a single Windows PE bootable media that works across all Dell supported systems and provides an underlying networking stack, see Running The Deployment Scripts.
- Creating scripts to automatically *map* to a predefined network share.
- Accessing scripts, configuration files, and operating system installation files from the network.

Preparing And Populating The Network Share

This process includes:

- Creating a network share on a system that is always available over the network for the target systems to be deployed.
- Ensuring that a large amount of space is available for storing operating system installation files.
- Ensuring that proper permissions are assigned to this share so that the target systems can read files from and write files to the share.

You can also use the bootable media to call the deployment components from the directory structure on the media and the remaining components from the network share.

1. Boot the target system with the bootable deployment media.
2. Execute the master batch file, which calls individual task scripts and utilities from the network share to complete the deployment process.

Deployment Using Removable Boot Media With A Network Connection (Media-Based) For Windows

For media-based deployment with a network connection:

1. Create a deployment media containing a bootable image with the appropriate media and network drivers, along with any utilities needed to connect to a network share. The deployment media initiates the deployment process by mapping to the network share where the deployment directory structure resides.
2. Insert the bootable media into the appropriate drive of the system to be deployed.
3. Boot or reboot the system.

   The configuration process begins and the following tasks are completed:

   - Windows PE or Linux, as the case may be, is installed from the media.
   - Network shares are mapped.
   - The DTK scripts execute the necessary DTK utilities from the network share or the media itself.
   - Configuration information is read from the network share.
   - The operating system is installed from a network share. This installation may happen after your system reboots.

Deployment Using Removable Boot Media Without A Network (Media-Based) Connection For Windows

For media-based deployment without a network connection:

1. Create a deployment media containing a bootable image with the appropriate drivers. The media must also include the complete deployment directory structure, which contains all DTK utilities, scripts, and configuration files, an operating system installation answer file, and the required operating system installation files and drivers.
2. Insert the bootable media into the media drive of the system to be deployed (the target system).
3. Boot or reboot the target system.

   The deployment process begins and the following tasks are completed:

   - Windows PE or embedded Linux, as the case maybe, is installed from the media.
   - The DTK scripts execute the necessary DTK utilities from the media.
   - Configuration information is read from the media.
   - The operating system is installed from the media.

Network Based Deployment For Windows

For Windows PE environments, deployment using Remote Installation Services (RIS) is recommended. For details, see the Microsoft RIS documentation. You can also use any other deployment tool such as Automated Deployment Services (ADS).

Using A Third Party Deployment Solution Framework For Windows

You can use DTK with any existing third-party deployment solution framework that provides an Automated Deployment Services (ADS) booting infrastructure for Windows Deployment Services (WDS) that can be used as the transport mechanism for DTK utilities. Because each third-party deployment framework is unique, these solutions fall outside the scope of this document. If you plan to utilize a third-party deployment solution framework, keep in mind that DTK is a Windows PE-based set of tools and scripts, so the deployment solution framework must also support Windows PE as a pre-operating system environment.

Deployment Using Dell Provided Embedded Linux

The two common scenarios for deployment using Dell provided Embedded Linux are following:

- Network Based Deployment
- Media Based Deployment

Network-Based Deployment

The procedure enables you to boot DTK over network and pass parameters (startup script location and name) during boot using the Pre-boot eXecution Environment (PXE) configuration file. This automatically launches the start-up script off the network share, when DTK is booted.

1. Obtain the ISO image of the embedded Linux available at **support.dell.com**.
2. Burn the ISO image using any commonly available CD or DVD burning software.

   NOTE: When you create a bootable CD/DVD, this image is supplied to the CD/DVD burning software along with the complete deployment directory structure, which includes all necessary supporting files to perform the deployment.

   The CD or DVD burning software creates a self-bootable CD or DVD.
3. Set up the Trivial File Transfer Protocol (TFTP) services and the Dynamic Host Configuration Protocol (DHCP) services on your network to boot the target system using PXE.
4. Copy the contents of the embedded Linux DTK ISO image (**SA.1**, **SA.2**, and **isolinux.cfg**) from **/mnt/cdrom/isolinux** to the **tftp** folder.
5. The **isolinux.cfg** file has various boot options. Select the method you want, and edit the **isolinux.cfg** file for the following:
   - IP address of the network share
   - Share path of the network share
   - Name of your startup script
6. The **isolinux.cfg** file is ready for PXE-booting. Copy this **isolinux.cfg** file into the PXE configuration folder.
7. Create your own start-up script and place it in the network share specified in your modified **isolinux.cfg** file.

   The modified script is picked up and executed during the boot process.

NOTE: The network share on which you have the scripts and DTK files are mounted to **/opt/dell/toolkit/systems**.

NOTE: Ensure that you specify the correct **RAMDISK** size in the **PXE** config file before booting through PXE. A sample pxe config file is provided in **/mnt/cdrom/isolinux/isolinux.cfg**.

Media-Based Deployment

The two ways of media deployment are as follows:

- Using Custom Scripts
- Using RPMs

Using Custom Scripts

1. Obtain the ISO image of the embedded Linux at **support.dell.com**.
2. Extract the contents of the ISO image to a folder on your hard drive.
3. Copy the custom scripts into the same folder.

   NOTE: Your scripts must also take care of copying miscellaneous items to unlock the media, so that the media can be mounted and ejected as required by the operating system's installation process.
4. In **/mnt/cdrom/isolinux.cfg**, edit the cd *install* section to point to your customized start-up script. In other words, add the **share_script** option; for example:

   ```
   label 1
   kernel SA.1
   append initrd=SA.2 root=/dev/ram0 rw DEBUG=0 share_type=cdrom loglevel=1
   BUILD=430 init=/init selinux=0
   ```

   Then, use **mkisofs**to create the ISO image; for example:

   ```
   mkisofs -o output.iso -b isolinux/isolinux.bin -c isolinux/boot.catalog -no-
   emul-boot -boot-loadsize 4 -pad -r -J -hide-joliet-trans-tbl -bootinfo-
   table \isoimage
   ```

   NOTE: It is important to use **mkisofs** to make your customized image because the ISO Linux recognizes only the **isolinux.cfg** (and all files within **/isolinux**). If you do not use **mkisofs**, the HAPI libraries of the DTK cannot load and most of the Dell Update Packages do not work.

   NOTE: The scripts that you copy into the media are copied to the RAM disk and run from the RAM disk. This task is done to make sure that the media is not locked. Ensure that your sample scripts have valid path names.
5. Burn the ISO contents to a media.
6. Your ISO image is ready for booting.

Using RPMs

You can install DTK using Red Hat Package Manger (RPM), Yellowdog Updater, Modified (YUM) or any other RPM Installation Manager.

Installing Using RPM

To install DTK RPMs using RPM:

1. Mount the DTK media at **/mnt/cdrom**.
2. Navigate to the **/mnt/cdrom/RPMs** directory.

   NOTE: The **/mnt/cdrom/RPMs** directory contains all DTK RPMs and the dependency RPMs.
3. Install the required DTK RPMs.

NOTE: To resolve the RPM dependency related issues, install the missing RPMs from **/mnt/cdrom/RPMs** directory. If the RPMs are not available in this directory, install these RPMs from the operating system media.

Installing Using YUM

To install DTK RPMs using YUM:

1. Install YUM and the dependency RPMs.
2. Mount the DTK iso to **/mnt/cdrom**.
3. Create a repository file (for example, **dtk.repo**) for the required operating system in a writable location. The corresponding RPMs for the operating system is located at **/mnt/cdrom/RPMs/<OS>**, where OS can be Red Hat Enterprise Linux 5 or Red Hat Enterprise Linux 6.

   For example, to deploy DTK RPMs on RHEL 6.3, you can use the following content in **dtk.repo** file:

   ```
   [DTK_RHEL6]
   name=DTK_RHEL6
   baseurl=file:///mnt/cdrom/RPMs/rhel6/
   enabled=true
   ```

4. Use the following command to install DTK RPMs: `yum install -c dtk.repo -y <RPM>`, where <*RPM*> can be DTK binaries.

   NOTE: For both installations, the **raidcfg** and **syscfg** binaries and libraries are installed at **/opt/dell/toolkit/bin**. The **racadm** binaries are installed at **/opt/dell/srvadmin/bin** and **/opt/dell/srvadmin/sbin**.

Post Installation Of RPMs

After installing the RPMs, perform the following:

1. For the binaries **syscfg** and **raidcfg**, export the PATH environment variable **/opt/dell/toolkit/bin**.
2. To start the services, navigate to **/opt/dell/srvadmin/sbin**, and type: `srvadmin-services.sh start`

   NOTE: To execute the commands to configure the system as per the requirements, see the *Dell OpenManage Deployment Toolkit Command Line Interface Reference Guide* available at **dell.com/support/manuals**.

Deployment Using Customized Embedded Linux

Make sure that you have the following basic libraries, Dell toolkit libraries, tools, and utilities required for DTK and integrate them to your embedded Linux deployment environment to proceed with the deployment:

- Drivers for all hardware installed in the embedded Linux. These drivers are available at **support.dell.com**.
- Serial port configuration utilities (**setserial** and **stty**) for **racadm**.
- Installed and working instrumentation drivers from **/mnt/cdrom/tools**.
- Point-to-Point Protocol (PPP) tools.
- Basic libraries and utilities for Linux to execute customized scripts.
- Installed DTK and the dependency RPMs.
- Necessary entries in the **ld.so.config** file so that the libraries are installed.

Integrate all the above mentioned libraries, Dell toolkit libraries, tools, and utilities into your embedded Linux environment and proceed with deployment.

NOTE: For more information on how Dell-provided utilities and drivers are installed, see **start-stage3.sh** , **start-hapi.sh**, or **start-raid.sh** available in **/mnt/cdrom/tools**.

NOTE: The **/opt/dell/srvadmin/shared**and **/opt/dell/srvadmin/hapi** directories must have read-write permissions. For more details, see **start-stage3.sh** and **start-hapi.sh**.

Using A Third-Party Deployment Solution Framework For Linux

You can use DTK with any existing third-party deployment solution framework that provides a PXE booting infrastructure that can be used as transport mechanism for the DTK utilities. Because each third-party deployment framework is unique, these solutions are beyond the scope of this document. If you plan to utilize a third-party deployment solution framework, make sure that the deployment solution framework supports embedded Linux as a pre-operating system environment.

7

Running Dell Update Packages On Systems Running Embedded Linux

You can run the Dell Update Packages in Embedded Linux environments on supported Dell systems. The common scenarios for running update packages are:

- Running Update Packages In Dell-Provided Embedded Linux
- Running Update Packages In Customized Embedded Linux

Running Update Packages In Dell-Provided Embedded Linux

1. Obtain the required update packages from the *Dell Server Updates* DVD or from **support.dell.com**.
2. Save the update packages on a network share.
3. Mount the network share where you saved the update packages and run the individual packages.

NOTE: See the *Dell Update Packages for Linux Operating Systems User's Guide* available at **dell.com/support/manuals** for help on using Dell Update Packages and information on error codes.

Running Update Packages In Customized Embedded Linux

Before running update packages in customized embedded Linux environment, ensure that you meet the dependencies listed in the following table.

Table 22. Dependencies for Dell Update Packages to Run in Customized Embedded Linux

Update Packages	Dependency	Dependencies Available On
BIOS	Instrumentation drivers dell_rbu	**/mnt/cdrom/tools** Kernel
ESM	Instrumentation drivers	**/mnt/cdrom/tools**
PERC	**/etc/dataeng** megaraid driver 2.4.2.0 SCSI drivers	**/mnt/cdrom/tools** **/mnt/cdrom/tools** Kernel
RAC4	Instrumentation drivers racser4.sh	**/mnt/cdrom/tools** **/mnt/cdrom/tools**
RAC 5	Instrumentation drivers	**/mnt/cdrom/tools**
iDRAC	Instrumentation drivers	**/mnt/cdrom/tools**
Dell PowerVault 220S	SCSI drivers	Kernel
SAS	SAS drivers	Kernel
Any Update Package	RPMs sysvinit	Linux media or any open source website

Update Packages	Dependency	Dependencies Available On
	grep	
	sed	
	awk	
	less	
	fmt	
	tar.gz	
	compat-libstdc	
	proc-mail	
	libxml2	

After you have met all the dependencies, run the update packages in your customized embedded Linux environment. For more information, see Running Update Packages In Dell-Provided Embedded Linux.

8

Known Issues And Frequently Asked Questions

This section describes known issues with the Deployment Toolkit (DTK) utilities and scripts, including answers to some frequently asked questions.

Known Issues

The following issues are organized by DTK utility or other function.

General Issues

- Virtual disk creation and/or deletion takes a long time on Microsoft Windows PE.

RAIDCFG Issues

- Due to a rounding limitation for RAID 0, RAID 1, and RAID 5, when creating a virtual disk, RAIDCFG can accept a disk size that is 1 MB greater than the maximum allowable virtual disk size limit. However, RAIDCFG creates the maximum virtual disk size and does not display an error. If you want to use the maximum allowable virtual disk size, it is recommended that you do not provide the size in the CLI and let RAIDCFG calculate the size for the RAID type.

SYSCFG Issues

- Setup passwords and system passwords cannot be cleared using DTK on systems prior to Dell PowerEdge *yx2x* systems.
- On PowerEdge 1950 and 2950 systems with internal USB, you can configure BIOS settings for internal USB Port only if the User Accessible/external (UA) USB ports are set to **All Ports On**. If the UA USB ports are not set to **All Ports On** and you try to configure BIOS setting for the internal USB port using SYSCFG, the configuration appears to be successful but the changes does not take effect after the next reboot. If you set the UA USB port from **All Ports On** to **All Ports Off** or **Only Back Ports On**, the USB port is automatically set to **Off** during the next reboot.

Windows PE Installation Issues

- If virtual flash is enabled and does not contain a valid image (for example, if the virtual flash contains a corrupt or random image), you may not be able to install Windows Server 2008 locally or remotely. To fix this issue, install a valid image on virtual flash or disable virtual flash if it is not used during the installation procedure.

Embedded Linux Installation Issues

- If internal SD card is present in the server, you may not be able to install the operating system using default partition. To fix this, remove the SD card from the server.

Frequently Asked Questions

The following questions are organized by DTK utility or other function.

General Deployment Questions

Q: While running SYSCFG.EXE on a Windows PE image, the SYSCFG.EXE tool silently fails.

A: Ensure that you have built your Windows PE image with the **/WMI** option. For more details, see Running The Deployment Scripts.

Q: When I use the sample DTK scripts to deploy multiple systems, do I need to edit the configuration files to reflect unique information (such as unique system names, IP address, and BIOS asset tags) for each system?

A: The sample scripts are provided as examples for users who want to develop their own deployment process. The scripts may work perfectly in your environment. If not, you may need to develop your own scripts entirely from scratch. If you are deploying multiple systems, for example, you need to provide unique information for each system when appropriate. To perform this task, you need to modify each **.ini** (or other configuration script) file to reflect the unique information for each system you are deploying (such as the remote access controller (RAC) IP addresses and BIOS asset tags). There are many options available to optimize this process and it is suggested that you conduct an Internet search for available tools.

Q: When I change my hard drive controller from SCSI to RAID (or RAID to SCSI), the system prompts me for confirmation during POST. How can I stop this from happening?

A: Use the **--noraidprompt** option with the **--embscsiraid** option to prevent the system from prompting during POST. Data loss results from changing the state of the disk controller, so you must be certain before skipping the prompt. There is no method for using the **--noraidprompt** in the input file. If you want to change the controller state through an input file, you must configure your script to call the **SYSCFG.EXE** utility twice, once to specify the input file, and once to change the controller state with the **--noraidprompt** option.

Q: What do I do if the PARTCFG script fails?

A: Ensure that **DT_HD** is set to the required disk to be configured before you execute the PARTCFG script.

RAIDCFG Questions

Q: Why does RAIDCFG display an error message when I use an invalid read, write, and cache policies and/or stripe sizes for a particular controller?

A: All RAID controllers have their own default read, write, and cache policies and stripe sizes. It is possible that if you provide an invalid policy or a stripe size for a particular controller, RAIDCFG may not give an error but creates the virtual disk with the default policy and/or stripe size.

Q: When I execute RAIDCFG with the RAID level of "01," RAIDCFG creates RAID 1, not RAID 01.

A: This behavior occurs because the RAIDCFG CLI parser ignores zeros preceding any parameter value. For example, **-r=01** is parsed as **-r=1**.

Q: What is strict creation?

A: Strict creation is an optional flag that has been added to the **create virtual disk** command. It allows you to create virtual disks only if the array disks (that is, hard drives) are within a specified percentage of disk space of each other.

Q: Why is the drive location of array disks displayed like 0:0:0?

A: The nexus used to display array disks is "channel:target:lun." The Logical Unit Number (LUN) is always "0." The channel and the target IDs are necessary to identify array disks when dealing with SCSI devices.

When dealing with SAS devices, the array disk location is displayed as **channel:target:enclosure**. It is possible to have the enclosure value as non-zero. If the enclosure has a non-zero value, then all three numbers (channel, target, and enclosure) have to be included on the command line.

Q: Do I need to specify RAID type and size when creating virtual disk(s)?

A: No. The default RAID type is RAID 0. If RAID size is not provided, the maximum size allowed for virtual disk is created. The mandatory fields required to create virtual disks are the controller slot ID and the array disks.

Q: What is a hotspare (failover drive)?

A: A hotspare is an extra and unused disk drive that is part of an array disk subsystem. A hotspare is always in standby mode. If a disk failure occurs, the hotspare replaces the failed drive without interrupting the system.

Q: What is mr2kserv.exe?

A: This is a service needed to configure all LSI RAID controllers in Windows PE. It provides Plug and Play support.

Q: What features have changed from the MS-DOS version of RAIDCFG to the Windows PE and embedded Linux versions?

A: Windows PE and embedded Linux version of RAIDCFG support new controllers. For the list of supported controllers, see *Dell Systems Software Support Matrix* at **dell.com/support/manuals**.

Features that have been removed from the MS-DOS version include:

- The --name and --wait options in the create command
- The --runlocation and the --scratchlocation options
- The --getpercentcomplete environment variables

Q: What is the minimum virtual disk size that I can create?

A: For PERC 5 controllers:

- RAID 0: 100 MB
- RAID 1: 100 MB
- RAID 5: 100 MB

For PERC 6 controllers:

- RAID 0: 100 MB
- RAID 1: 100 MB
- RAID 5: 100 MB
- RAID 6: 100 MB
- RAID 60: 100 MB

Q: Why can I not see any controllers using RAIDCFG?

A: Ensure that the RAID controller(s) can be seen in the BIOS during POST. If the BIOS does not detect the controller, then RAIDCFG does not detect it. Also ensure that the controller is set to RAID mode and not any other mode in the controller BIOS.

Q: What is span length?

A: Span length is the field needed when you try to create a RAID 50. The number represents the span size of the RAID 5.

Q: PARTCFG cannot enumerate the virtual disk(s) created by RAIDCFG. What must be done?

A: A reboot is required for PARTCFG to enumerate virtual disks after creating a virtual disk using the RAIDCFG command.

SYSCFG Questions

Q: Can I use the SYSCFG utility to configure Point-to-Point Protocol (PPP) over the serial port to access my RAC?

A: No. The Baseboard Management Controller (BMC) firmware does not support PPP configurations for the serial port.

Q: The SYSCFG utility --lancfgparams option has suboptions that configure the IP address, Gateway, and Subnet Mask for the system BMC. There is also a MAC address. Are these the same as the operating system network parameter of the managed system?

A: No. The Baseboard Management Controller (BMC) has its own IP address, Gateway, Subnet Mask, and MAC address values.

Q: What is the minimum configuration needed to support BMC Platform Event Filtering?

A: You must enable LAN-channel access **pefalerting** and ensure that you set the BMC IP address and Gateway values for alerting.

Q: I installed the factory defaults for the BMC of my system. Then, I enabled BMC user ID 3. What is the username for this user ID?

A: The default username for user IDs 3 to 10 is **NULL**.

Q: Can I use this NULL username to remotely log in to a BMC of a system?

A: No. Remote login using a **NULL** username is not allowed. It is recommended that you always provide a non-null, valid username for the user ID when you enable a BMC user ID.

Q: Are duplicate usernames allowed in the BMC?

A: No. If you enter a username value that already exists, the **SYSCFG** utility returns an error code of 89, which means **This username is already in use. Enter a unique username**.

Q: In --solcfgparams, when I set the Serial Over LAN (SOL) character send threshold using solcharsendthreshold to be 225, I get a Hardware subsystem error. Invalid data field in request. What are the valid values?

A: The range of valid values for **--solcfgparams** and **--solcharsendthreshold** are from 1 to 220.

Q: How do I enable console redirection on a modular system?

A: Set the **--serialcomm** option value to **enableconred**. For example, **syscfg -serialcomm=enableconred**.

Embedded Linux Questions

Q: After installing Linux, I am not able to boot to the utility partition by pressing the <F10> key during reboot.

A: Re-create the utility partition using the **upinit.sh** script and install **grub** in the boot partition.

Q: When booting DTK through PXE, my system displays error messages, unknown behavior, and also leads to a kernel panic sometimes. However, booting from the media causes no problems.

A: Ensure that you specify the correct **RAMDISK** size in the pxe config file before booting through PXE. A sample pxe config file is provided in **/mnt/cdrom/isolinux/isolinux.cfg**.

Q: Can we use the same sample scripts that we have from earlier versions of DTK?

A: The scripts written for the previous versions of DTK does not work in the later versions for embedded Linux. For the sample scripts, see the directory **/opt/dell/toolkit/template/scripts** .

Q: After the DTK deployment is complete, the Red Hat Enterprise Linux Server installation keeps going into attended mode or halts with the message `ks.cfg not found`.

A: If the **initrd.img** is not available in the operating system, get the latest update of Red Hat Enterprise Linux Server or prepare driver disks for your network or disk controller.

Q: How to apply driver disks during operating system installation?

A: You can apply driver disks as follows:

1. During operating system installation, after the first reboot, attach the **driver-disk.iso** file to the DVD-ROM or virtual media.
2. At the **grub** prompt, press <a> followed by <dd> and, then press <Enter>.

Windows PE Questions

Q: My system gets the error `Illegal or Missing File Types Specified in Section Files.SCSI.Name` during an unattended installation of a supported Windows operating system on supported Dell systems.

A: This behavior can occur when the line in the **Txtsetup.oem** file under the [Files.SCSI. *name*] heading is not a supported file type. To resolve this behavior, you must remove the line in the **Txtsetup.oem** file. The dynamic-link library (DLL) file can be copied to the correct location of the installation by placing it in the **I386\$Oem$\$$\OEMDIR** folder. (**OEMDIR** is the destination folder where the file would normally be located, if installed to a running operating system. For example, INF files are normally found in the **%SystemRoot%\INF** folder. The correct **OEMDIR** destination can be found by searching the INF file used to install the device or driver.) For more information, see Microsoft Knowledge Base Article 275334 available at Microsoft website.

Q: In Windows, I see a blue screen when I delete a virtual disk and create a new one without initialization and attempt to format the disk.

A: Initialize (fast init) the drives before formatting the disk. This prevents the issue.