

# Automated OS Deployment

## Using the Dell OpenManage Deployment Toolkit and Microsoft WinPE

Rapid OS deployment has always been a challenging task in enterprise IT environments. In particular, deploying a server OS can be complicated because it may require first configuring BIOS, RAID, and remote access settings. This article shows how Microsoft® Windows® Preinstallation Environment can be combined with the Dell™ OpenManage™ Deployment Toolkit to help deploy Windows operating systems on Dell PowerEdge™ servers across the enterprise.

BY RAVIKANTH CHAGANTI AND JATIN N. MUDDU

### Related Categories:

*Dell OpenManage*

*Enterprise management*

*Microsoft Windows*

*System deployment*

*Systems management*

*Windows pre-installation environment (WinPE)*

Visit [www.dell.com/powersolutions](http://www.dell.com/powersolutions) for the complete category index.

IT administrators can efficiently install operating systems on multiple servers by scripting most of the deployment operations. Automating pre-OS installation activities, such as the configuration of hardware-specific settings, can help streamline server deployment and often can be completed using original equipment manufacturer (OEM)–supplied utilities.

The Dell OpenManage Deployment Toolkit (DTK) 2.0 is designed to provide a simple interface for scripted and automated configuration of Dell PowerEdge servers before OS deployment. The DTK contains a set of tools and sample scripts that can help accomplish most deployment tasks. The DTK 2.0 for Microsoft Windows Preinstallation Environment (WinPE) is designed to be integrated into a WinPE environment<sup>1</sup> and is available for download

as a self-extractable zip file on the Dell support Web site ([support.dell.com](http://support.dell.com)). This zip file contains DTK tools, sample scripts, template answer files, drivers, and documentation.

Activities involved in a typical server deployment scenario include the following:

- Configuration of the BIOS and baseboard management controller (BMC) settings of the server
- Configuration of the remote access controller (RAC) if a RAC is present
- Configuration of the RAID controller to create virtual disks if a RAID controller is present
- Configuration of the hard disk (creation of partitions)
- Deployment of the OS

<sup>1</sup> The DTK 2.0 has been developed and validated using WinPE 2005 only.

Tool	Purpose
Syscfg	Configure and replicate BIOS and BMC settings
Racadm	Configure and replicate RAC settings
Raidcfg	Create virtual disks on a RAID controller

Figure 1. Tools available within the Dell OpenManage Deployment Toolkit

The DTK can help administrators accomplish these tasks. Furthermore, by integrating the DTK and customized scripts with WinPE, administrators can automate OS deployments.

### Configuring BIOS, BMC, RAC, and RAID settings with the DTK

Figure 1 lists the tools available within the DTK. These tools—when run from the WinPE command-line interface (CLI)—can be used to set or retrieve specific settings on a Dell PowerEdge server and are designed to be used in scripts to automate deployment operations. For example, executing the following command retrieves the boot sequence settings from the system:

```
syscfg --bootseq>
```

Executing the following command sets the first alert destination of the BMC to the specified IP address:

```
syscfg lcp --alertdest=1 --destipaddr=ipaddress
```

When several options need to be set, administrators can write a simple batch script that contains all the commands for those options. When executed, the script will automatically set the required options. For a list of supported commands and options, refer the *Dell OpenManage Deployment Toolkit Command Line Interface Reference Guide* at [support.dell.com/support/edocs/software/dtk/1.4/en/CLI/index.htm](http://support.dell.com/support/edocs/software/dtk/1.4/en/CLI/index.htm).

Another method for configuring settings on multiple PowerEdge servers is to manually configure a server, referred to as the master server, with all the required optimal settings and then use the `--outfile` option to capture these settings to a file. The captured settings can then be replicated on another PowerEdge server, referred to as the target server, by using the `--infile` option.

To help IT administrators perform this activity, the DTK contains two scripts for capturing the BIOS, BMC, and RAC settings to a file (SYSCAP.BAT and RACCAP.BAT) and two more scripts to replicate the captured settings from the master server to the target server (SYSREP.BAT and RACREP.BAT). *Note:* The DTK currently does not support replication of RAID settings.

The sample scripts provided with the DTK can perform very basic (atomic) operations and should be customized. This may involve changing a few environment variables or even rewriting the entire script. A master deployment script that can initiate and control the deployment process should also be written. This script can be very simple—calling the sample scripts provided with the DTK one after another, depending on the requirements—or it can be very complicated, performing more than just server configuration. The master deployment script should be capable of mounting a network share so that the captured settings from the master server can be stored in a file on the network share, which is then accessible when replicating the same settings on the target systems. This script should also have logic coded into it so that it can differentiate the master server from the target servers. This will make the script usable for both capture and replication purposes. Figure 2 shows an example code snippet demonstrating this logic.

The `raidcfg` utility included in the DTK can help IT administrators create virtual disks on the target server. Administrators can write scripts to achieve this goal by using the `raidcfg` utility and the `RAIDCFG.BAT` sample script. The DTK also contains sample scripts to partition the hard disk on a target server (`PARTCFG.BAT`) and to initiate the OS installation on a server (`WININST.BAT`). Best practices recommend modifying these scripts to meet specific deployment needs.

### Running DTK scripts from WinPE

After writing the required scripts and determining the appropriate deployment strategy, administrators should integrate the scripts into a WinPE image. WinPE is designed to help administrators perform Windows OS deployment and maintenance activities such as virus scanning, data backup, disk imaging, and hardware configuration and diagnostics.<sup>2</sup> WinPE is based on the Windows kernel running in protected mode and includes a minimal Win32 subsystem with limited services. This section outlines the general procedure for creating a customized WinPE image, integrating DTK components, and finally creating an ISO image that can be used with Microsoft Remote Installation Services (RIS) to boot servers into WinPE over a network.

```
for /F %i in ('X:\Dell\Toolkit\Tools\SYSCFG
--svctag') set %i
If "%svctag%" == "service tag of the Master
server" goto capturesettings
.... code to perform deployment .....
```

Figure 2. Code snippet from an example master deployment script

<sup>2</sup> For more information about WinPE, visit [www.microsoft.com/whdc/system/winpreinst/default.msp](http://www.microsoft.com/whdc/system/winpreinst/default.msp).

## Creating a WinPE image

Software requirements for creating a WinPE image include the following:

- WinPE CD
- Microsoft Windows OS installation CD<sup>3</sup>
- Additional mass-storage drivers that are not part of the Windows OS installation CD
- Additional drivers (for the network adapter, instrumentation, storage, and RAC) that are not a part of the Windows OS installation CD

The next step is to build the WinPE directory structure. Administrators can do so by performing the following steps:

1. Place the Windows OEM Preinstallation Kit (OPK) CD in the CD drive of the management station.
2. Create a directory on the hard disk of the management station to store the WinPE build tools, denoted as *build\_location*. For example:

```
md c:\build_x86
```

3. Remove the Windows OPK CD and place the WinPE CD in the CD drive, denoted as *cd\_drive*.
4. Copy *cd\_drive\WinPE* and all subdirectories to *build\_location*. For example:

```
xcopy e:\WinPE c:\build_x86 /s
```

5. Remove the WinPE CD and place the Windows OS installation CD in the CD drive.
6. Navigate to *build\_location*. For example:

```
cd c:\build_x86
```

7. Run the `mkimg` command with the two required options: the path to the OS installation CD and the target WinPE directory. For example:

```
mkimg.cmd e:\ c:\WinPE
```

## Integrating DTK components and drivers

The DTK zip file can be extracted onto the hard disk of a management station—preferably the same system where the WinPE image is built. Best practices recommend that the files be extracted to a folder that is shared across the network and that this network

share be mounted by the master deployment script, as discussed earlier in this article. The contents of the DTK zip file can be extracted to the WinPE image's root folder as well. Refer to the DTK documentation for more details about the recommended deployment directory structure.

The DTK zip file contains drivers for instrumentation, network adapters, mass storage, and RACs within the Dell\Drivers folder. Most of these drivers are required for the DTK tools to function correctly. The DRIVERINST.BAT script, which is also located in the Dell\Drivers folder, can be used to install these drivers into the WinPE image. Using this script is very simple—for example, executing the following command will install all the drivers into the WinPE image:

```
c:\WinPE\dell\drivers\driverinst.bat c:\WinPE\dell\drivers c:\WinPE
```

*Note:* Simply extracting the drivers into the WinPE image is not sufficient; the drivers need to be installed using the DRIVERINST.BAT script. Also, some services need to be started during boot-up. Refer to the DTK documentation for information about how to start these services automatically at startup.

Next, administrators should copy the master deployment script to the WinPE image and modify `winbom.ini` (located in the root WinPE image folder—for example, `c:\WinPE`) to automatically start executing the master deployment script after booting the system into WinPE.

## Creating an ISO image

After using `mkimg` to create the WinPE image, administrators can customize this image. They also can use the `oscdimg` tool (available in the `c:\WinPE` folder) to create an ISO image file from the customized image, which they can then burn to a CD.

1. Run the `oscdimg` command with the `-b` option, which specifies location, and the `-n` option, which specifies the path to the WinPE directory and the image file path and name. For example:

```
oscdimg -b c:\build_x86\etfsboot.com -n c:\WinPE c:\WinPEx86.iso
```

*Note:* In this example, the `-b` option specifies the path to the EL Torito boot sector file and the `-n` option enables support for long file names. This command creates an ISO file in the location specified.

2. Use any CD-recording software, such as Roxio, to burn the ISO image file to a blank CD and create a bootable WinPE CD.

<sup>3</sup> WinPE 2005 supports the 32-bit version of Microsoft Windows Server™ 2003 Service Pack 1 (SP1) as well as Windows Server 2003 x64 Editions; however, the DTK 2.0 supports only the 32-bit version of Windows Server 2003 SP1.

```
[SetupData]
BootDevice = "ramdisk(0)"
BootPath = "\platform\System32\"
OsLoadOptions = "/noguiboot /fastdetect /minint /rdexportascd /rdpath=bootimage"
Architecture = "platform"
[OSChooser]
Description = "Dell OpenManage Deployment Tool Kit"
Help = "This option will load WinPE RAMDISK containing DTK tool set & scripts"
LaunchFile = "%INSTALLPATH%\%MACHINETYPE%\templates\startrom.com"
ImageType = Flat
Version = "5.2 (0)"
```

Figure 3. Example Winnt.sif file

### Integrating the WinPE ISO image with RIS


Windows RIS provides network administrators with the capability to easily install a base OS onto a new PC or to replace a system that has failed. RIS allows administrators to perform these tasks without having to visit each individual machine. RIS also provides a centralized location to integrate maintenance and troubleshooting tools that are accessible through a network boot.

Combining WinPE with Windows RIS enables the WinPE image to serve as the pre-OS environment for the network boot machines. This implementation is designed to help deploy Windows operating systems over a TCP/IP network. To integrate the WinPE image with RIS, administrators should perform the following steps:

1. Browse to the location where RIS images are installed—for example, `\\Server_name\REMINST\Setup\Language\Images`.
2. Create a folder to which to copy the WinPE image.
3. Navigate to the newly create folder.
4. Create another folder named "i386."
5. Copy the WinPE image created earlier to the i386 folder.
6. Create a folder named "templates" within the i386 folder.
7. Navigate to the *build\_location* folder created earlier—for example, `c:\build_x86`.
8. Copy `ntdetect.com` to the templates folder.
9. Copy `setupldr.exe` to the templates folder and rename it "ntldr."
10. Create a text file named "Winnt.sif" within the templates folder and modify it to meet the specific requirements of the deployment. Figure 3 shows an example Winnt.sif file. *Note:* In this example file, *platform* refers to the machine architecture (for example, i386 for x86 or IA-32 architecture), and *bootimage* refers to the path to the WinPE ISO image file.
11. Restart the `binlsv` service.

Once these steps are completed, administrators should reboot all the required servers using Preboot Execution Environment (PXE). The servers should then boot into the customized WinPE image, and the master deployment script should begin executing automatically either to capture the settings from a master system or to replicate captured settings to target systems.

### Enhancing server deployment across the enterprise

The Dell OpenManage Deployment Toolkit used in conjunction with Microsoft Windows Preinstallation Environment can help simplify the process of deploying multiple similarly configured servers. By scripting common tasks with the DTK tools and integrating scripts into a WinPE image, administrators can minimize the amount of time required to deploy servers as well as the potential errors that can occur during a manual deployment. 

**Ravikanth Chaganti** is a senior engineer with the Factory Install Development team at the Dell Bangalore Development Center. He has a bachelor's degree in science from Andhra University in India and has various IT certifications, including Microsoft Certified Systems Engineer (MCSE), Cisco Certified Network Associate (CCNA), and Sun Certified System Administrator (SCSA).

**Jatin N. Muddu** is an engineer with the DTK development team at the Dell Bangalore Development Center. His areas of interest include system programming and server architecture.

#### FOR MORE INFORMATION

##### Microsoft WinPE:

[www.microsoft.com/whdc/system/winpreinst/WindowsPE\\_over.msp](http://www.microsoft.com/whdc/system/winpreinst/WindowsPE_over.msp)