Cisco Custom Linux Preboot Image User Guide

For Altiris Deployment Solution 6.9

May 17, 2013

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: OL-29711-01

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Cisco Custom Linux Preboot Image User Guide

CONTENTS

Preface

This section discusses the objectives, audience, conventions, and organization of the *Cisco Custom Linux Preboot Image User Guide For Altiris Deployment Solution 6.9*.

Objectives

This guide describes the tasks and commands to install the Cisco custom Linux preboot image for the Altiris 6.9, which supports the Cisco Unified Computing System servers and its sub-components.

Audience

This publication is intended primarily for administrators and users who are engaged in deploying the Altiris Deployment Solution 6.9 on the Cisco UCS servers.

Organization

This chapter describes the contents of each chapter in this document.

Table 1 Organization

Chapter	Title	Description
Chapter 1	Introduction	Overview of the Cisco UCS custom LinPE and the prerequisites for installing the Cisco custom LinPE.
Chapter 2	Cisco Custom Linux Preboot Tasks	Describes the various tasks involved in deploying and installing the Cisco custom LinPE.
Chapter 3	Storage Area Network Support (DM-Multipath)	Overview of the DM-Multipath and how to perform scripted SAN backup and deployment.
Appendix 1	Acronyms and Abbreviations	Lists the acronyms and abbreviations used in the document.
Appendix 2	Sample Kickstart File for RHEL	Provides a sample script file demonstrating the kickstart installation.

Command Syntax Conventions

Table 2 describes the syntax used with the commands in this document.

Table 2 Command Syntax Guide

Convention	Description
boldface	Commands and keywords.
italic	Command input that is supplied by you.
[ ]	Keywords or arguments that appear within square brackets are optional.
{ x | x | x }	A choice of keywords (represented by x) appears in braces separated by vertical bars. You must select one.
^ or Ctrl	Represent the key labeled *Control*. For example, when you read *^D* or *Ctrl-D*, you should hold down the Control key while you press the D key.
`screen font`	Examples of information displayed on the screen.
boldface screen font	Examples of information that you must enter.
< >	Nonprinting characters, such as passwords, appear in angled brackets.
[ ]	Default responses to system prompts appear in square brackets.

CHAPTER 1

Introduction

This chapter contains the following sections:

- Overview, page 1-1
- Prerequisites, page 1-1

Overview

This guide describes the tasks and commands to install the Cisco Custom Linux Preboot Image (LinPE) for Altiris 6.9, which supports the Cisco Unified Computing System servers and its sub-components. This LinPE image contains the latest network and storage drivers needed for Cisco Unified Servers (UCS) server and its sub-components.

Before integrating Cisco UCS with the Altiris DS, it is important to download the Cisco UCS specific custom Linux preboot image that supports all the current Cisco UCS servers and its sub-components. This image also supports multipath and related binaries which can be used by the end user.

Prerequisites

The prerequisites to installing the Cisco Custom LinPE includes collating the following information which assists to boot the Cisco custom LinPE.

- Preboot Execution Server IP— server IP on which the PXE services are running.
- Altiris DS Server Share Domain— domain name required for accessing the default Altiris eXpress share.
- Altiris DS Server Share user name—username required for accessing the default Altiris eXpress share.
- Altiris DS Server Share password—password required for accessing the default Altiris eXpress share.

CHAPTER 2

Cisco Custom Linux Preboot Tasks

This chapter contains the following sections:

- Deploying Custom LinPE, page 2-1
- Creating PXE Boot Menu, page 2-4
- Using a Custom Preboot Image, page 2-7
- Creating a Scripted/Network OS Install Job, page 2-10
- Creating Imaging Jobs, page 2-17

Deploying Custom LinPE

Contents of the Install Zip Folder

The install zip folder contains the following two zip files and a pdf file:

- Cisco_Custom_LinPE_DS69_SP5_Image_x.x.x.zip

  This zip file contains the following three files related to the Linux Preboot image:

  - rootfs.gz—the custom root file system.
  - linux—the custom Linux kernel image.
  - pxelinux.cfg/default—the default PXE configuration file.
- Cisco_Custom_LinPE_DS69_SP5_Sample_Jobs_and_Scripts_x.x.x.zip

  This zip file contains one file and one folder:

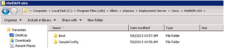
  - Cisco_Custom_LinPE_DS69_SP5_Sample_Jobs_x.x.x.bin—contains sample jobs for user reference which can be imported in Altiris Deployment Console.
  - Cisco—this folder contains the desired Linux OS and a file osinstall.sh. This folder should be placed in the Deployment Server subfolder within the Altiris.

  The Linux OS folder contains two folders:

  - Boot
  - SampleConfig

Cisco-Confidential

This folder structure is required to execute the Scripted/Network OS Install Job.

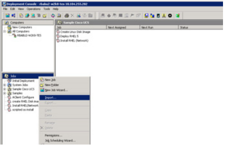


- CiscoLinPE_AltirisDS_User_Guide.pdf

  This guide describes the Altiris deployment solution provided by Cisco System Inc.

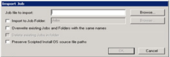
Importing Sample Jobs and Scripts

Step 1 In the Altiris Deployment Console window, right-click **Jobs**.



Cisco-Confidential

Step 2 Click **Import**. The **Import Job** window opens.


Import Job

Job file to import:

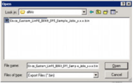
□ Import to Job Folder: Jobs

□ Overwrite existing Jobs and Folders with the same names

□ Delete existing Jobs in folder

□ Preserve Scripted Install OS source file paths

Step 3 Click **Browse** and choose the location of the job file "Cisco_Custom_LinPE_DS69_SP5_Sample_Jobs_x.x.x.bin".


Open

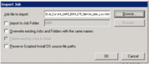
Look in: altiris

Cisco_Custom_LinPE_DS69_SP5_Sample_Jobs_x.x.x.bin

File name: Cisco_Custom_LinPE_DS69_SP5_Sample_Jobs_x.x.x.bin

Files of type: Export Files (*.bin)

Step 4 Click **Open**.


Import Job

Job file to import: Cisco_Custom_LinPE_DS69_SP5_Sample_Jobs_x.x.x.bin

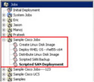
□ Import to Job Folder: Jobs

□ Overwrite existing Jobs and Folders with the same names

□ Delete existing Jobs in folder

□ Preserve Scipted Install OS source file paths

Step 5 Click **OK** to import all the sample jobs to Altiris Deployment console.



Note The sample jobs are available for each operation, which assist in creating and managing the jobs in Altiris.

Creating PXE Boot Menu

Step 1 Launch the Altiris Deployment Console.

Step 2 From the menu bar, click the **Tools** tab.

Step 3 From the Tools menu, choose **PXE Configuration**.

The **PXE Configuration Utility** window opens.

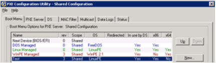
Tools
Help
Boot Disk Creator
PC Transplant Editor
ImageExplorer
Remote Agent Installe
PXE Configuration

Step 4 Click the **Boot Menu** tab.

Step 5 Click **New** to create a new boot configuration.

Cisco-Confidential

The **New Shared Menu Option** window opens.

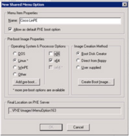

PXE Configuration Utility - Shared Configuration

Boot Menu Options for PXE Server: Shared Configuration

Name	rev	Scope	OS	Redirected	In use by DS	x86	x64
Next Device (BIOS/EFI)	0	Shared					
DOS Managed	0	Shared	FreeDOS		Yes	Yes	
Linux Managed	0	Shared	LinuxPE		Yes	Yes	Yes
WinPE Managed	3	Shared	WinPE 2.1		Yes	No	Yes
Test	3	Shared	LinuxPE		Yes	Yes	No

Step 6 In the New Shared Menu Option window, enter the name for the PXE boot option. In the Pre-boot Image Properties area, choose the options as shown in the following figure.

Step 7 Click **Create Boot Image** to create a new boot image.


New Shared Menu Option

Menu Item Properties

Name: Cisco LinPE

☑ Allow as default PXE boot option

Pre-boot Image Properties

Operating System & Processor Options

○ DOS
● Linux *
○ WinPE
○ Other

□ x86
☑ x64
□ ia64 *

* more pre-boot options are available

Image Creation Method

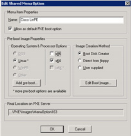
● Boot Disk Creator
○ Direct from floppy
○ User supplied

Final Location on PXE Server

...\PXE\Images\MenuOption163

Cisco-Confidential

Step 8 After the boot image is created, click **OK** to create a new PXE boot option.



Edit Shared Menu Option

Menu Item Properties

Name: Cisco LinPE

☑ Allow as default PXE boot option

Pre-boot Image Properties

Operating System & Processor Options

○ DOS
● Linux *
○ WinPE
○ Other

☐ x86
☑ x64
☐ ia64 *

Add pre-boot...

* more pre-boot options are available

Image Creation Method

● Boot Disk Creator
○ Direct from floppy
○ User supplied

Edit Boot Image...

Final Location on PXE Server

...\PXE\Images\MenuOption163

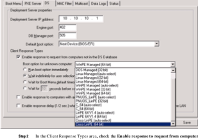
OK Cancel

Note Creation of a boot image is a simple wizard-based process and guides the user through every step.

Cisco-Confidential

Using a Custom Preboot Image

Step 1 After the boot option is created in the PXE Configuration Utility window, click the **DS** tab.


PXE Configuration Utility - Shared Configuration

Deployment Server properties

Deployment Server IP address: 10 . 10 . 10 . 1

Engine port: 402

DB Manager port: 505

Default boot option: Next Device (BIOS/EFI)

Client Response Types

☑ Enable response to request from computers not in the DS Database

Boot option for unknown computer: WinPE Managed (64-bit)

○ Run boot option immediately

● Wait indefinitely for user selection

○ Wait for Boot Menu default timeo

○ Wait for 10 seconds before c

□ Enable response to computers with a

□ Enable response delay (1/2 sec.) whe ... e LAN

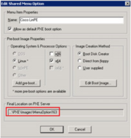
DOS Managed (32-bit)
Linux Managed (auto-select)
Linux Managed (32-bit)
Linux Managed (64-bit)
WinPE Managed (auto-select)
WinPE Managed (32-bit)
WinPE Managed (64-bit)
PNUOS_LinPE (auto-select)
PNUOS_LinPE (32-bit)
C_64 (auto-select)
C_64 (64-bit)
LinPE 64 V1.4 (auto-select)
LinPE 64 V1.4 (64-bit)
Cisco LinPE (auto-select)
Cisco LinPE (64-bit)

Step 2 In the Client Response Types area, check the **Enable response to request from computers not in the DS Database** check box.

Step 3 In the Boot option for unknown computer area, click the **Wait for Boot Menu default time out** radio button.

Step 4 From the **Boot option for unknown computer** drop-down list, choose **Cisco LinPE**.

Step 5 Click **OK** to close the PXE Configuration Utility window.



Edit Shared Menu Option

Menu Item Properties

Name: Cisco LinPE

☑ Allow as default PXE boot option

Pre-boot Image Properties

Operating System & Processor Options

○ DOS
● Linux *
○ WinPE
○ Other

□ x86
☑ x64
□ ia64 *

Add pre-boot...

* more pre-boot options are available

Image Creation Method

● Boot Disk Creator
○ Direct from floppy
○ User supplied

Edit Boot Image...

Final Location on PXE Server

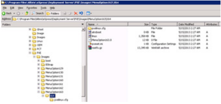
...\PXE\Images\MenuOption163

OK Cancel

This defines the default pre-execution environment. After the processing is complete, the PXE boot option image is saved in a folder on the PXE Server location.

The folder created is saved at the following location: <Altiris Install Path>\eXpress\Deployment Server\PXE\Images\MexuOptionxxx>.

```
For example - C:\Program Files\Altiris\eXpress\Deployment
Server\PXE\Images\MenuOption163
```


C:\Program Files\Altiris\eXpress\Deployment Server\PXE\Images\MenuOption163\X64
File Edit View Favorites Tools Help
Back Search Folders
Address C:\Program Files\Altiris\eXpress\Deployment Server\PXE\Images\MenuOption163\X64
Folders
Ghost
Image
Images
Linux
OEM
PCT
PXE
Images
boot
BStrap
MenuOption129
MenuOption130
MenuOption131
MenuOption160
MenuOption161
MenuOption162
MenuOption163
X64
pxelinux.cfg

Name	Size	Type	Date Modified	Attributes
pxelinux.cfg		File Folder	5/15/2013 2:27 AM	
atrsboot	0 KB	File	5/15/2013 2:27 AM	A
linux	1,358 KB	File	5/15/2013 2:27 AM	A
MenuOption163.0	12 KB	0 File	5/15/2013 2:27 AM	A
pxeset.ini	1 KB	Configuration Settings	5/15/2013 2:27 AM	A
rootfs.gz	13,395 KB	WinRAR archive	5/15/2013 2:27 AM	A

Step 6 Copy and then extract the zip folder (**Cisco_Custom_LinPE_DS69_SP5_Image x.x.x.zip**) contents to the **MenuOption163** subfolder.

Copy the following files and folders to the following location **MenuOption163>X64**:

a. rootfs.gz—The custom root file system.

b. linux—The custom Linux kernel image.

c. pxelinux.cfg/default—The default PXE configuration file.

Step 7 Manually update the default file in the **pxelinux.cfg** folder.

Update the following variables in this file:

a. PXE_PATH—This relative path to the Images folder is where the PXE boot process searches for the Linux kernel (linux) and root file system (rootfs.gz).

```
For example:
The original Altiris LinPE image is generated at the following location:
C:\Program Files\Altiris\eXpress\Deployment Server\PXE\Images\MenuOption163
The image created is stored as MenuOption163, and the value for this variable is as
follows:
PXE_PATH= MenuOption163\X64
```

b. PXE_SERVER—DNS name of the PXE server.

c. PXE_IP—IP address of the PXE server.

d. PXE_RID—This ID (XXX) should be same as that in the MenuOptionXXX.

e. SHARE_USER—The username for the Altiris eXpress share.

f. SHARE_PASSWORD—The password for the Altiris eXpress share.

g. SHARE_DOMAIN—The windows domain name required for accessing the Altiris eXpress share.

Cisco-Confidential

Creating a Scripted/Network OS Install Job

The Scripted/Network OS install task performs remote, automated, and unattended operating system installations over the network using answer files to input configuration and installation-specific values. Scripted/Network installations allow you to deploy server and client computers across the network from installation files and perform post-installation configuration tasks.

The Kickstart file is essential for installing the Linux through Scripted/Network OS. This file lists all the answers to the queries that appear during a typical installation. This method is called kickstart installation.

Kickstart installations can be performed using either a local CD-ROM, a local hard drive, or through file transfer protocols such as NFS, FTP, or HTTP.

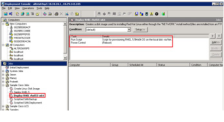
Performing a Kickstart Installation

Step 1 Create a kickstart file.

To view a sample kickstart file, see Appendix 2, "Sample Kickstart File for RHEL".

Step 2 Make the kickstart file available on the network.

Step 3 Make the installation tree available.



Deploy RHEL rhel55-x64

Description: Creates a disk image used for installing Red Hat Linux either through the "NETWORK" install method (files are installed from an FTP

Condition: (default)

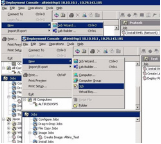
Task	Details
Run Script	Script for provisioning RHEL 5.5 x64 OS on the local disk via ftwo
Power Control	(Reboot)

Creating a RHEL Scripted/Network OS Install Task

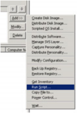
Step 1 In the Deployment Console window, choose **File> New> Job**.

Cisco-Confidential

Step 2 Enter the job name, and double-click on the Job created to open the Task View pane.

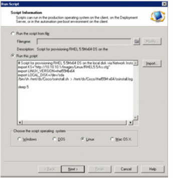


Step 3 In the Task View pane, click **Add**, and choose **Run Script**.



Step 4 Enter the following script in the **Run Script** text box.

```
#Script for provisioning RHEL 5.5Mx64 OS on the local disk using Network Install
export KS="http://10.10.10.1/Images/Linux/RHEL5.5/ks.cfg"
export LINUX_VERSION=rhel55M-x64
export LOCAL_DISK=/dev/sda
/bin/sh /mnt/ds/Cisco/osinstall.sh> /mnt/ds/Cisco/rhel55M-x64/osinstall.log
sleep 5
```



Run Script

Script Information

Scripts can run in the production operating system on the client, on the Deployment Server, or in the automation pre-boot environment on the client.

○ Run the script from file

Filename:

Description: Script for provisioning RHEL 5.5Mx64 OS on the

● Run this script

```
# Script for provisioning RHEL 5.5Mx64 OS on the local disk via Network Insta
export KS="http://10.10.10.1/images/Linux/RHEL5.5/ks.cfg"
export LINUX_VERSION=rhel55Mx64
export LOCAL_DISK=/dev/sda
/bin/sh /mnt/ds/Cisco/osinstall.sh > /mnt/ds/Cisco/rhel55M-x64/osinstall.log

sleep 5
```

Choose the script operating system

○ Windows ○ DOS ● Linux ○ Mac OS X

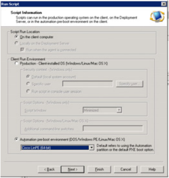
Step 5 Click the **Linux** radio button to choose the operating system.

Step 6 Click **Next**.

Step 7 From the Automation pre-boot environment (DOS/WinPE/Linux) drop-down list, choose the pre-boot environment created previously. For information on the preboot environment, see Creating PXE Boot Menu.

Cisco-Confidential

Step 8 Click **Next** and then click **Finish.**



Run Script

Script Information

Scripts can run in the production operating system on the client, on the Deployment Server, or in the automation pre-boot environment on the client.

Script Run Location

● On the client computer

○ Locally on the Deployment Server

☑ Run when the agent is connected

Client Run Environment

○ Production - Client-installed OS (Windows/Linux/Mac OS X)

Security context - (Windows only)

● Default (local system account)

○ Specific user

○ Run script in console user session

Script Options - (Windows only)

Script Window: Minimized

Script Options - (Windows/Linux/Mac OS X)

Additional command-line switches:

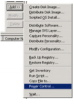
● Automation pre-boot environment (DOS/Windows PE/Linux/Mac OS X)

Cisco LinPE (64-bit)

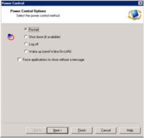
Default refers to using the Automation partition or the default PXE boot option.

Adding Power Control

Step 1 In the Task View pane, click **Add** and choose **Power Control**.



Step 2 Click the **Restart** radio button to select the power control method.



Power Control

Power Control Options
Select the power control method

● Restart
○ Shut down (if available)
○ Log off
○ Wake up (send Wake-On-LAN)

□ Force applications to close without a message.

Step 3 Click **Next** and then click **Finish**.

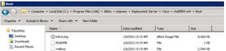
Note The ks.cfg file must be modified according to the shared path (ftp or http) with correct IP and other parameters.

Unzipping the Cisco_Custom_LinPE_DS69_SP5_Sample_Jobs_and_Scripts_x.x.x.zip file will also give a folder named "Cisco". Place the Cisco folder within the Altiris eXpress folder (<Altiris Install Path>\eXpress\Deployment Server\), which contains a folder for the particular Linux version (RHEL version 5.5 so the folder name is rhel55MM-x64) and a file osinstall.sh. This OS folder should contain two folders:

- Boot—This folder is empty, by default. You need to copy the PXE preboot images (initrd.img and vminuz) from the Linux OS which is to be installed.

  In this example, RHEL 5.5 is the Linux OS being installed. Copy the pxeboot folder contents to the Boot folder (Cisco > rhel56M-x64 > Boot folder).

  Following image shows the contents of the pxeboot folder from RHEL 5.5 that is copied to the Boot folder.

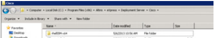

Boot
Computer ▸ Local Disk (C:) ▸ Program Files (x86) ▸ Altiris ▸ eXpress ▸ Deployment Server ▸ Cisco ▸ rhel55M-x64 ▸ Boot
Organize ▾
Include in library ▾
Share with ▾
New folder
Favorites
Desktop
Downloads
Recent Places
Libraries

Name	Date modified	Type	Size
initrd.img	1/6/2011 8:19 AM	Altiris Image File	8,060 KB
README	1/6/2011 8:19 AM	File	1 KB
vmlinuz	1/6/2011 8:19 AM	File	2,046 KB

– Sample Config—The SampleConfig folder contains the ks.cfg file for reference. This file can be modified as per the requirements.

Note

- In the script, ensure that the LOCAL_DISK contains the desired disk information where the Linux is to be installed.
- The LINUX_VERSION value is the name of folder under Cisco folder.


Cisco
Computer ▸ Local Disk (C:) ▸ Program Files (x86) ▸ Altiris ▸ eXpress ▸ Deployment Server ▸ Cisco ▸
Organize ▾
Include in library ▾
Share with ▾
New folder
Favorites
Desktop
Downloads
Recent Places

Name	Date modified	Type	Size
rhel55M-x64	5/6/2013 10:56 AM	File folder	
osinstall.sh	5/7/2013 2:59 PM	SH File	7 KB

- In case of the variable KS, there is a shared path (either http or ftp) for ks.cfg file. Ensure that the ks.cfg file can be accessed through the given path with proper credentials.

Creating Imaging Jobs

Creating a Disk Image

This task creates an image of a computer's hard disk. The disk image is saved in any of the formats such as, an .IMG, .EXE, .WIM, .DMG, or .GHO file.

To create a disk image task, perform the following tasks sequentially:

1. Create Image (disk image)
2. Reboot to Production

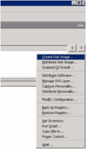
Step 1 In the Deployment Console, choose **File> New> Job**.

Step 2 Enter the job name as **Create Linux Disk Image**.

Step 3 Double-click on the created job to open the Task View pane.

Step 4 In the Task view pane, click **Add.**

Step 5 From the drop-down menu, choose **Create Disk Image**.



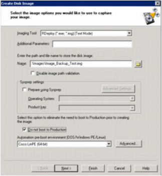
Step 6 From the **Imaging Tool** drop-down list, choose **RDeploy (Text Mode)** option.

Step 7 Browse to the path and filename to store the disk image file.

Step 8 Check the **Do not boot to Production** checkbox.

This step creates an image of the hard disk while it is booted to Automation, without first booting to Windows to save the network settings (TCP/IP settings, SID, computer name).

Step 9 From the **Automation pre-boot environment (DOS/WinPE/Linux)** drop-down list, choose the preboot environment created previously. For information on the preboot environment, see Creating PXE Boot Menu.



Create Disk Image

Select the image options you would like to use to capture your image.

Imaging Tool: RDeploy (*.exe; *.img) (Text Mode)

Additional Parameters:

Enter the path and file name to store the disk image.

Name: .\Images\Image_Backup_Test.img

☐ Disable image path validation.

Sysprep settings

☐ Prepare using Sysprep

Operating System:

Product key:

Select this option to eliminate the need to boot to Production prior to creating the image.

☑ Do not boot to Production

Automation pre-boot environment (DOS/Windows PE/Linux)

Cisco LinPE (64-bit)

Step 10 Click **Next**.

Step 11 Click **Finish** to complete the task creation.

Distributing a Disk Image

This task allows users to distribute an RDeploy, ImageX, Mac, or Ghost image file to managed computers, to deploy a previously created hard disk image.

To perform a deploy image task, perform the following tasks sequentially:

1. Wipe the hard drive clean

2. Distribute the disk image
3. Reboot to production

Step 1 In the Deployment Console, choose **File> New> Job**.

Step 2 Type the job name as **Distribute Linux Disk Image** and double-click on the job to open the Task View pane.

Step 3 Erase the system disk using Run Script task.

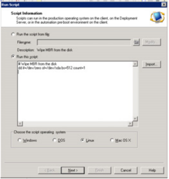
Step 4 In the Task View pane, click **Add**.

Step 5 From the drop-down menu, choose **Run Script**.

As a prerequisite, it is important to wipe the hard drive to clean up partition errors, bad installations, or privacy concerns.

Step 6 Enter the following commands in the Run this script textbox:

```
# Wipe MBR from the disk
dd if=/dev/zero of=/dev/sda bs=512 count=1
```



Script Information

Scripts can run in the production operating system on the client, on the Deployment Server, or in the automation pre-boot environment on the client.

○ Run the script from file

Filename:

Description: Wipe MBR from the disk

● Run this script:

```
# Wipe MBR from the disk
dd if=/dev/zero of=/dev/sda bs=512 count=1
```

Choose the script operating system

○ Windows ○ DOS ● Linux ○ Mac OS X

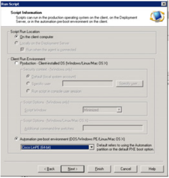
Step 7 Click on the **Linux** radio button to select the operating system.

Step 8 Click **Next**.

Step 9 Click the **On the Client computer** radio button to define the Script Run Location.

Step 10 From the **Automation pre-boot environment (DOS/WinPE/Linux)** drop-down list, select the preboot environment created previously. For information on the preboot environment created, see Creating PXE Boot Menu.

Step 11 Click **Next**.



Run Script

Script Information

Scripts can run in the production operating system on the client, on the Deployment Server, or in the automation pre-boot environment on the client.

Script Run Location

● On the client computer

○ Locally on the Deployment Server

☑ Run when the agent is connected

Client Run Environment

○ Production - Client-installed OS (Windows/Linux/Mac OS X)

Security context - (Windows only)

● Default (local system account)

○ Specific user

○ Run script in console user session

Script Options - (Windows only)

Script Window: Minimized

Script Options - (Windows/Linux/Mac OS X)

Additional command-line switches:

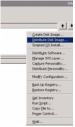
● Automation pre-boot environment (DOS/Windows PE/Linux/Mac OS X)

Cisco LinPE (64-bit)

Default refers to using the Automation partition or the default PXE boot option.

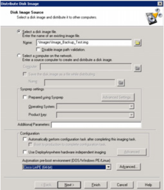
Step 12 Click **Finish** to complete the task creation.

Step 13 Click **Add** and choose the **Distribute Disk Image** from the drop-down list.



Step 14 Browse to the disk image file and choose the image file created. For information on the image created, see Creating a Disk Image.

Step 15 From the **Automation pre-boot environment (DOS/WinPE/Linux)** drop-down list, select the preboot environment created. For information on the preboot environment, see Creating PXE Boot Menu.



Distribute Disk Image

Disk Image Source

Select a disk image and distribute it to other computers.

● Select a disk image file.
Enter the name of an existing image file.

Name: \Images\Image_Backup_Test.img

□ Disable image path validation.

○ Select a computer on the network.
Enter a source computer to create and distribute a disk image.

Computer:

□ Save the disk image as a file while distributing

Name:

Sysprep settings

□ Prepared using Sysprep

Operating System:

Product key:

Additional Parameters:

Configuration

□ Automatically perform configuration task after completing this imaging task.

□ Boot to production to complete configuration task.

□ Use DeployAnywhere hardware independent imaging

Automation pre-boot environment (DOS/Windows PE/Linux)

Cisco LinPE (64-bit)

Step 16 Click **Next**.

Step 17 Click **Finish** to complete the task creation.

CHAPTER 3

Storage Area Network Support (DM-Multipath)

This chapter contains the following:

- Device Mapper Multipathing, page 3-1
- Scripted SAN Backup, page 3-2
- Scripted SAN Deployment, page 3-5

Device Mapper Multipathing

The Cisco UCS custom LinPE supports multipath workflows and binaries by default. The LinPE kernel supports Device Mapper Multipathing (DM-Multipath), which allows you to configure multiple I/O paths between server nodes and storage arrays into a single device.

The Cisco UCS custom LinPE by default checks for the existence of the multipath and then adds them to the device mapper directory (*/dev/mapper*). The local disks are automatically excluded by detecting and blacklisting them in the multipath configuration.

This section enumerates the following:

- Kernel modules
- Binaries
- Multipath Configuration
- Services

Kernel Modules

The following kernel modules reroutes I/O and supports failover for paths and path groups:

- dm-mod.ko
- dm-multipath.ko
- dm-round-robin.ko

Binaries

The following binaries are included in the Cisco UCS custom LinPE image:

- Multipath Configuration tool (*/sbin/multipath*)—Provides commands to configure, list, and flush multipath devices.
- Multipath daemon (*/sbin/multipathd*)—Monitors the path status. When paths revert, the multipath daemon can also initiate path group switches to ensure that the optimal path group is used.
- kpartx utility (*/sbin/kpartx*)— Reads partition tables on the specified device and creates device maps over the detected partitions. The kpartx utility is called from the hotplug whenever device maps are created and deleted.

Multipath Configuration

The multipath daemon behavior can be updated and controlled by using the */etc/multipath.conf* file.

This configuration file can be used for the following multipath configurations:

- blacklist—specific devices that are not considered for multipath. By default, all devices are blacklisted. Usually, the default blacklist section is commented out.
- blacklist_exceptions—multipath candidates that would otherwise be blacklisted according to the parameters of the blacklist section.
- defaults—general default settings for DM-Multipath.
- multipaths—settings for the characteristics of individual multipath devices. These values overwrite what is specified in the defaults and devices section of the configuration file.
- devices—settings for the individual storage controllers. These values overwrite what is specified in the defaults section of the configuration file. If any of the storage array used is not supported by default, you must create a devices subsection for that array.

Service

The DM multipath daemon is started by default using the /etc/init.d/multipath-tools script. This script can be used with various options for starting, stopping, restarting, reloading, and force-reloading the daemon.

The various binaries listed can be used to run several scripted tasks as per the requirement.

The disk number can be in form of dm-0 or dm-1 etc for the SAN attached disks.

To take the scripted SAN backup you can run the sample job explained in the following section.

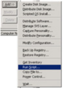
Note The Altiris imaging methods do not support any path based (/dev/mapper/mpathxx) image creation or deployment, therefore you must specify the disk number requires a backup or deployment.

Scripted SAN Backup

Step 1 In the Deployment Console, choose file **Menu> Job> New**.

Step 2 Enter the job name and double-click on the created Job to open the Task View Pane.

Step 3 In the Task view pane, click **Add** and choose **Run Script**.



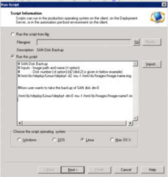
Step 4 Enter the following script to take a SAN backup.

```
#SAN Disk Backup
#Inputs - Image path and name (-f option)
#- Disk number (-d option) (d2 (disk2) is given in below example)
#/mnt/ds/rdeploy/Linux/rdeployt -d2 -mu -f /mnt/ds/Images/Image-name.img

#Now user wants to take the backup of SAN disk dm-0
/mnt/ds/rdeploy/Linux/rdeployt -dm-0 -mu -f /mnt/ds/Images/Image-name1.img
```

Step 5 Click the **Linux** radio button to select the operating system.

Step 6 Click **Next**.

Cisco-Confidential



Run Script

Script Information

Scripts can run in the production operating system on the client, on the Deployment Server, or in the automation pre-boot environment on the client.

○ Run the script from file:

Filename:

Description: SAN Disk Backup

● Run this script:

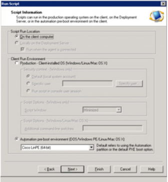
```
# SAN Disk Backup
# Inputs - Image path and name (-f option)
#          - Disk number (-d option) (d2 (disk2) is given in below example)
#/mnt/ds/rdeploy/Linux/rdeployt -d2 -mu -f /mnt/ds/Images/Image-name.img


#Now user wants to take the backup of SAN disk dm-0

/mnt/ds/rdeploy/Linux/rdeployt -dm-0 -mu -f /mnt/ds/Images/Image-name1.im
```

Choose the script operating system

○ Windows ○ DOS ● Linux ○ Mac OS X

Note In this section a backup of SAN disk dm-0 was created, and here you can also change the disk number.

Step 7 From the **Automation pre-boot environment (DOS/WinPE/Linux)** drop-down list, select the preboot environment created previously. For information on the preboot environment, see Creating PXE Boot Menu.



Run Script

Script Information

Scripts can run in the production operating system on the client, on the Deployment Server, or in the automation pre-boot environment on the client.

Script Run Location

● On the client computer

○ Locally on the Deployment Server

☑ Run when the agent is connected

Client Run Environment

○ Production - Client-installed OS (Windows/Linux/Mac OS X)

Security context - (Windows only)

● Default (local system account)

○ Specific user

○ Run script in console user session

Script Options - (Windows only)

Script Window: Minimized

Script Options - (Windows/Linux/Mac OS X)

Additional command-line switches:

● Automation pre-boot environment (DOS/Windows PE/Linux/Mac OS X)

Cisco LinPE (64-bit)

Default refers to using the Automation partition or the default PXE boot option.

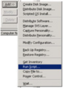
Step 8 Click **Next**, and then click **Finish**.

Scripted SAN Deployment

Step 1 In the Deployment Console, choose file **Menu> Job > New**.

Step 2 Enter the job name and double-click on the job to open the Task View Pane.

Step 3 In the Task view pane, click **Add** and choose **Run Script**.

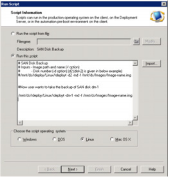


Step 4 Enter the following script to take backup.

```
#SAN Disk Backup
#Inputs - Image path and name (-f option)
#- Disk number (-d option) (d2 (disk2) is given in below example)
#/mnt/ds/rdeploy/Linux/rdeployt -d2 -md -f /mnt/ds/Images/Image-name.img
#Now the user wants to take the backup of SAN disk dm-1
/mnt/ds/rdeploy/Linux/rdeployt -dm-1 -md -f /mnt/ds/Images/Image-name.img
```

Step 5 Click the **Linux** radio button to select the operating system.

Step 6 Click **Next**.



Script Information

Scripts can run in the production operating system on the client, on the Deployment Server, or in the automation pre-boot environment on the client.

○ Run the script from file

Filename:

Description: SAN Disk Backup

● Run this script

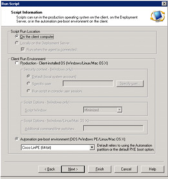
```
# SAN Disk Backup
# Inputs - Image path and name (-f option)
#          - Disk number (-d option) (d2 (disk2) is given in below example)
#/mnt/ds/rdeploy/Linux/rdeployt -d2 -md -f /mnt/ds/Images/Image-name.img


#Now user wants to take the backup of SAN disk dm-1

/mnt/ds/rdeploy/Linux/rdeployt -dm-1 -md -f /mnt/ds/Images/Image-name.img
```

Choose the script operating system

○ Windows ○ DOS ● Linux ○ Mac OS X

Note In this section we deployed the backup on SAN disk dm-1, and here you can also change the disk number.

Step 7 From the **Automation pre-boot environment (DOS/WinPE/Linux) drop-down list**, choose the preboot environment created previously. For information on the preboot environment, see Creating PXE Boot Menu.



Run Script

Script Information

Scripts can run in the production operating system on the client, on the Deployment Server, or in the automation pre-boot environment on the client.

Script Run Location

● On the client computer

○ Locally on the Deployment Server

☑ Run when the agent is connected

Client Run Environment

○ Production - Client-installed OS (Windows/Linux/Mac OS X)

Security context - (Windows only)

● Default (local system account)

○ Specific user

○ Run script in console user session

Script Options - (Windows only)

Script Window: Minimized

Script Options - (Windows/Linux/Mac OS X)

Additional command line switches:

● Automation pre-boot environment (DOS/Windows PE/Linux/Mac OS X)

Cisco LinPE (64-bit)

Default refers to using the Automation partition or the default PXE boot option.

Step 8 Click **Next** and then click **Finish**.

APPENDIX 1

Acronyms and Abbreviations

The following table describes the acronyms and abbreviations used in the document.

Abbreviation	Translation
PXE	Preboot Execution Environment
PE	Production Environment (Client-installed OS)
AE	Automation Environment (PXE boot)
MBR	Master Boot Record
DS	Deployment Solution
UCS	Unified Computing System
OS	Operating System
NFS	Network File System
FTP	File Transfer Protocol
HTTP	Hypertext Transfer Protocol
RHEL	Red Hat Enterprise Linux
DM-Multipath	Device Mapper Multipathing

APPENDIX 2

Sample Kickstart File for RHEL

Following is a sample kickstart installation script for RHEL.

```
# Kickstart file for RHEL
# Please note the following:
#
# 1- You must select and modify an Installation source below.
# 2- The installation of the Symantec NS Agent depends on the
# ability to perform a smbfs mount to the NSServer's NSCap
# share. If you modify the group package selections below and
# remove all the groups that include samba, you will need to do
# one of the following:
#  2.1-Manually install the NS agent
#  2.2-Specify the samba packages in the %packages section
#  2.3-Modify the "mount -t smbfs..." statement below to connect
#     in some other way to the NSCap. (NFS, etc.)
# 3- You need to set the username/password used to make the smbfs
# connection. Search for the "mount -t smbfs" command line below.
# It defaults to administrator/altiris.
# 4- In the "%packages" section, verify packages to be installed.

# ==========================================================
# Passwords/Authentication configuration.
# Both passwords are pre-encrypted and set to altiris.
# Please change your passwords once Installation is complete.
# ----------
authconfig --enableshadow --enablemd5
bootloader --md5pass=$1$oAsWAvbt$3tt06w9ZsS.A9DRU7IJw31 --location=mbr
rootpw --iscrypted $1$oAsWAvbt$3tt06w9ZsS.A9DRU7IJw31

# ==========================================================
# Installation Source
# uncomment and make the necessary changes to the line that
# describes the type of installation source you will use.
# ----------
install

#nfs --server=<server ip address> --dir=/<sharename>/<directory>
#url --url http://<server ip address>/<directory>
#url --url ftp://<server ip address>/<directory>
url --url ftp://10.10.10.1/RedHat_ISO/


# ==========================================================
# Driver Disks
# Path to dd.img
# ----------
# driverdisk --source=ftp://path/to/dd.img
# driverdisk --source=http://path/to/dd.img
```

Cisco-Confidential

```
# driverdisk --source=nfs:host:/path/to/dd.img

device eth eth0

# ==========================================================
# Network configuration
# Uncomment the line for the needed network configuration.
# Static configuration options MUST all be on one line!
# ----------
#network --bootproto=dhcp --hostname=Altiris_Test --device=eth0
#network --bootproto=static --ip=192.168.232.2 --netmask=255.255.255.0
--gateway=192.168.231.1 --nameserver=172.27.172.10 --hostname=0013D407A52B
--device=eth0
network --onboot=yes --device=eth0 --bootproto=dhcp --noipv6


# ==========================================================
# Command Sections
# ----------
#System  language
lang en_US
#Language modules to install
langsupport --default=en_US
#System keyboard
keyboard us
#System mouse
mouse generic3ps/2
#Sytem timezone
timezone --utc America/Denver
#Reboot after installation
reboot
#Use text mode install
text
#Install Red Hat Linux instead of upgrade
install
#Firewall configuration
firewall --disabled
#Do not configure XWindows
#skipx
#Installation Code
#key %INSTALLATION_CODE%
key --skip

#XWindows configuration information
#Probe for video card, Probe for monitor
xconfig --depth 16 --resolution 1024x768 --defaultdesktop=KDE --startxonboot

# ==========================================================
# The following is a standard partition configuration. It
# assumes you have no existing partitions or at the most a
# Dell Utility Partition.
# Note that if you uncomment "clearpart --all" and thus clear
# all partitions, it will remove the Dell Utility Partition
# if it exists.
# clearpart --linux will remove any existing Linux partitions,
# leaving the Dell Utility partition. This is so the install
# partition is removed.
# ----------
clearpart --all --initlabel
#clearpart --linux --initlabel
part /boot --size=100 --fstype ext3 --ondisk=sda
part swap --recommended --ondisk=sda
part / --size=3072 --grow --fstype ext3 --ondisk=sda
#autopart
```

```
zerombr yes

# ===========================================================
# Package Installation Selection
# Items with an @ are Groups defined in /RedHat/base/comps
# Individual packages are defined on separate lines.
# ----------
%packages

# --- Packages ---
@ X Window System
@ GNOME Desktop Environment
#@ KDE (K Desktop Environment)
@ Sound and Video

@ Administration Tools
#@ Authoring and Publishing
#@ Editors
#@ Engineering and Scientific
#@ Games and Entertainment
#@ Graphics
#@ Office/Productivity
#@ Printing Support
#@ Server Configuration Tools
@ System Tools

# Internet
#@ Graphical Internet
#@ Text-based Internet

# Development
#@ Ruby
#@ Java
#@ Java Development
#@ Development Libraries
#@ Development Tools
#@ X Software Development
#@ GNOME Software Development
#@ KDE Software Development

# Servers
#@ MySQL Database
#@ DNS Name Server
#@ FTP Server
#@ Legacy Network Server
#@ Mail Server
#@ Network Servers
#@ News Server
#@ Windows File Server
#@ Web Server

# ===========================================================
# Pre-Installation Section
# ----------
%pre

# ===========================================================
# Check for FAT and Linux partitions and delete if found. This is used mainly for
# launching the scripted installs. This may seem a little verbose, but it was
necessary
# to do it this way to support all possible shell environments.
# ----------

for VAR in FAT Linux
```

```
do
    foo=`fdisk -l /dev/sda | grep "$VAR" | head -n 1 | sed -e 's| .*$||' -e
's|^[^(0-9)]*||'`
    if [ "$foo" != "" ] ; then
        if [ "$foo" = "1" ] ; then
            echo "d" > /tmp/foo.txt
            echo "w" >> /tmp/foo.txt
        else
            echo "d" > /tmp/foo.txt
            echo "${foo}" >> /tmp/foo.txt
            echo "w" >> /tmp/foo.txt
        fi

        fdisk /dev/sda < /tmp/foo.txt
    fi
done

# ==========================================================
# Post-Installation Section
# Run any additional script commands here as needed.
# Use the --nochroot option if you need to be in a non chroot'ed
# environment
# ----------
%post
# ==========================================================
# Wait till the system establishes a network conncection and
# then restart the network to make sure that its working.
# ----------
sleep 30
service network restart
sleep 10

# ==========================================================
# Create a script to install the altiris-agent
# This script(ddp-post-install.sh) will be called by rc.local
# when the system boots for the first time.
# ----------
mkdir /tmp/ddp
cat > /tmp/ddp/ddp-post-install.sh << _EOF
#!/bin/bash

#                        Altiris Agent Automatic Install               #
#                        ===============================               #



########################################################################
#                                                                      #
#           This script can be used as a sample to automatically       #
#           install the Altiris agent and connect it to Altiris        #
#           Deployment Solution Console                                #
#                                                                      #
#           Apply the necessary settings according to your envir       #
#           onment :                                                   #
#                                                                      #
#                     set the agent version used in sections           #
#           "Agent Version" and "Package Download"                     #
#                                                                      #
#                    set your FTP settings in section                  #
#           "FTP Access and Security Settings"                         #
#                                                                      #
#                                                                      #
########################################################################
```

```
## Common variables

DATE=`date +%Y%m%d`
CACHE=/Altiris
LOG=\$CACHE/\$DATE.log

## Agent Information

PKG=altiris-adlagent-2.6-92.i386.bin
CONF=/opt/altiris/deployment/adlagent/conf
SERVICE=adlagent

# FTP Access and Security Settings

HOST=10.10.10.1
USER=anonymous
PASSWD=anonymous

## Make Altiris install cache and logs directory

if [ -d \$CACHE ];

   then echo "Cache Already Exists !"> /dev/null

else

mkdir \$CACHE && echo "Cache Created !"> \$LOG

fi

## Package Download

cd \$CACHE

if [ -f \$PKG ];

    then echo "Altiris package has already been retrieved"> \$LOG

else

ftp -inv \$HOST <<END_SCRIPT
quote USER \$USER
quote PASS \$PASSWD
binary
get \$PKG
get adlagent.conf
bye
END_SCRIPT

fi

if [ -f \$PKG ]

    then echo "Altiris package has been successfully retrieved !"> \$LOG

else

echo "Altiris package could not be retrieved, check your network
connection !"> \$LOG && exit 0
```

```
fi

## Altiris agent install and configuration

if [ -f /etc/init.d/adlagent ];

    then echo "Altiris agent is already installed !"> \$LOG

else

sh \$PKG && cp -Rfv adlagent.conf \$CONF

service \$SERVICE start && echo "Agent successfully installed !"> \$LOG

fi

if ps ax | grep -v grep | grep \$SERVICE > /dev/null

    then echo "Altiris Agent is running !">> \$LOG

else

echo "Check agent installation or manually start the agent by running the following
command: service adlagent start">> \$LOG

fi

## Cleanup

exit 0

_EOF
# ==========================================================
# Change the mode of ddp-post-install.sh so that it is
# executable and add it to rc.local
# ----------

chmod +x /tmp/ddp/ddp-post-install.sh
cat >> /etc/rc.d/rc.local << _EOF
/tmp/ddp/ddp-post-install.sh
_EOF
```