Appendixes

XI

The following sections are accessed from other sections of the *Novell® ZENworks® 7 Linux Management Administration Guide*:

Bundle and Policy Schedules

<div align="right">

A

</div>

Using Novell® ZENworks® Linux Management - Dell Edition, you can schedule when bundles are deployed to or installed on assigned devices. You can also schedule when policies are applied to assigned devices.

The following scheduling options are available:

A.1 No Schedule

Use this option to indicate no schedule; no action occurs.

A.2 Date Specific

Select one or more dates on which to run the scheduled event and set other restrictions that might apply.

NOTE: If you schedule an event in the past, the scheduled event occurs when the assigned device refreshes.

Start Dates(s): Click the plus (+) symbol to display a calendar from which you can select dates to run the event on. Click the arrows next to the month to display the previous or next month's calendar; click the arrows next to the year to display the previous or next year's calendar.

Run Event Every Year: Select this option to run the event every year on the dates that you selected in the *Start date(s)* field.

Select When Schedule Execution Should Start: Select one of the following options:

- **Start Immediately at Start Time:** The scheduled event runs immediately at the time that you specify in the *Start Time* box.

- **Start at a Random Time between Start Time and End Time:** This option randomly spreads out the scheduled event times so the scheduled event does not run at the same time on multiple devices. You can use this option to avoid possible network overload. For example, if you want to distribute or install a bundle to 100 users, you could use the *Start at a random time between start time and end time* option to specify a one-hour block of time (starting at the scheduled start time) in which to randomly deploy or install the bundle to the various devices.

StartTime\End Time: Use the down-arrows to select the start and end times of the scheduled event.

IMPORTANT: Be aware that ZENworks Linux Management uses the *End Time* as the "expiration time." If a bundle or policy is in the middle of being executed, execution stops at the specified time.

Use Greenwich Mean Time (GMT): Usually, a schedule is based on the device's local time zone. If your network spans different time zones and you schedule an application to run at 1:00 p.m., it runs at 1:00 p.m. in each time zone. This option lets you specify a single time across the globe.

You can, for example, select this option to have bundles deployed or installed on devices at the same time regardless of their time zones.

A.3 Day of the Week Specific

Select one or more days of the week to run the scheduled event on and set other restrictions that might apply.

Select the Days of the Week to Run the Scheduled Event: Select one or more days, Sunday to Saturday, on which you want to run the scheduled event. By default, no days are selected; a day is selected when the check box is checked.

Restrict Schedule Execution to the Following Date Range: Use the *Start date* and *End date* fields to restrict the scheduled event to the dates between the start and end dates. Click the *Calendar* icon to display a calendar from which you can select the respective dates.

Select When Schedule Execution Should Start: Select one of the following options:

- **Start Immediately at Start Time:** The scheduled event runs immediately at the time that you specify in the *Start Time* box.

- **Start at a Random Time between Start Time and End Time:** This option randomly spreads out the scheduled event times so the scheduled event does not run at the same time on all devices. You can use this option to avoid possible network overload. For example, if you want to distribute or install a bundle to 100 users, you could use the *Start at a random time between start time and end time* option to specify a one-hour block of time (starting at the scheduled start time) in which to randomly deploy or install the bundle to the various devices.

- **Start Immediately at Start Time, and then Repeat until End Time Every:** Use the *Hours* and *Minutes* fields to specify how often you want the scheduled event repeated until deployment or installment of the bundle is successful.

Start Time\End Time: Use the down-arrows to select the start and end times of the scheduled event.

IMPORTANT: Be aware that ZENworks Linux Management uses the *End Time* as the "expiration time." If a bundle or policy is in the middle of being executed, execution will stop at the specified time.

Use Greenwich Mean Time (GMT): Usually, a schedule is based on the device's local time zone. If your network spans different time zones and you schedule an application to run at 1:00 p.m., it runs at 1:00 p.m. in each time zone. This option lets you specify a single time across the globe.

You can, for example, select this option to have bundles deployed or installed on devices at the same time regardless of their time zones.

Set the "Black Out" Time Ranges when Execution Should Not Occur: Click *Add* to display the Specify Black-Out Time Period dialog box. Use the *Start/End date* and the *Start/End time* options to specify the time period in which you do not want the scheduled event run. You can use this option to minimize network traffic during a certain time period.

A.4 Event

Select the event that this schedule should be triggered on:

- ◆ **Runlevel Change:** The Linux operating system has different modes of operation, or runlevels, the operating system can run in. These runlevels are similar to the safe mode or command-prompt-only mode in Microsoft* Windows. The *Runlevel change* option lets you trigger the event schedule when a user or administrator changes the runlevel on the device.

- ◆ **User Login:** The *User login* option lets you trigger the event schedule when the user logs in to the device.

A.5 Monthly

Select the day of the month to run the scheduled event on and set other restrictions that might apply.

Day of the Month: Select one of the following options:

- ◆ **Start the Scheduled Event on a Specific Day of the Month:** Specify the day of the month on which to run the scheduled event.

- ◆ **Start the Scheduled Event on the Last Day of the Month:** Select this option to run the scheduled event on the last day of the month. For example, for the month of February, the event runs on the 28th (except for leap years, in which case it runs on the 29th); for the month of December, the event runs on the 31st.

Select When Schedule Execution Should Start: Select one of the following options:

- ◆ **Start Immediately at Start Time:** The scheduled event runs immediately at the time that you specify in the *Start Time* box.

- ◆ **Start at a Random Time between Start Time and End Time:** This option randomly spreads out the scheduled event times so the event is not run at the same time on all devices. You can use this option to avoid possible network overload. For example, if you want to distribute or install a bundle to 100 users, you could use the *Start at a random time between start time and end time* option to specify a one-hour block of time (starting at the scheduled start time) in which to randomly deploy or install the bundle to the various devices.

Start Time\End Time: Use the down-arrows to select the start and end times of the scheduled event.

IMPORTANT: Be aware that ZENworks Linux Management uses the *End Time* as the "expiration time." If a bundle or policy is in the middle of being executed, execution stops at the specified time.

Use Greenwich Mean Time (GMT): Usually, a schedule is based on the device's local time zone. If your network spans different time zones and you schedule an application to run at 1:00 p.m., it runs at 1:00 p.m. in each time zone. This option lets you specify a single time across the globe.

You can, for example, select this option to have bundles deployed or installed on devices at the same time regardless of their time zones.

Set the "Black Out" Time Ranges when Execution Should Not Occur: Click *Add* to display the Specify Black-Out Time Period dialog box. Use the *Start/End date* and the *Start/End time* options to specify the time period in which you do not want the scheduled event run. You can use this option to minimize network traffic during a certain time period.

A.6 Relative to Refresh

Select the initial delay and repeat frequency to run the scheduled event and set other restrictions that might apply.

Schedule Execution: Select one of the following options:

- ◆ **Start Immediately on Refresh:** Select this option to run the scheduled event when the device refreshes (looks for new bundles, policies, and so forth). The event runs on the first device refresh only and does not run on subsequent refreshes. By default, the system global refresh schedule is every two hours.

    To change the default refresh schedule, click the *Configuration* tab > *Device Refresh Schedule*.

    If the user performs a manual refresh on the device by running the `rug ref` command, the scheduled action runs on the first manual refresh only and not on subsequent refreshes (manual refreshes or global refreshes).

- ◆ **Delay Execution after Refresh:** Select this option to run the scheduled event for a specified number of days, hours, or minutes after the device refreshes (looks for new bundles, policies, and so forth).

After Executing, Repeat Every: Select this option and specify the number of days, hours, and minutes after which you want to repeat execution of the scheduled event after a successful execution.

Set the "Black Out" Time Ranges when Execution Should Not Occur: Click *Add* to display the Specify Black-Out Time Period dialog box. Use the *Start/End date* and the *Start/End time* options to specify the time period in which you do not want the scheduled event run. You can use this option to minimize network traffic during a certain time period.

Naming Conventions in the ZENworks Control Center

<div style="text-align:right">B</div>

When you name an object in the ZENworks Control Center (folders, bundles, bundle groups, catalogs, and so forth), ensure that the name adheres to the following conventions:

- The name must be unique in the folder.
- Uppercase and lowercase letters, as well as underscores and spaces, are displayed as you first entered them, but they aren't distinguished. Leading and trailing underscores are also removed. For example, Bundle_1, BUNDLE 1, and _bundle_1_ are all considered identical.
- If you use spaces, you must enclose the name in quotes when entering it on the command line. You must enclose bundle 1 in quotes ("bundle 1") when entering it in the zlman utility, for example.

Imaging Utilities and Components

<div style="text-align: right; font-size: large;">C</div>

The following sections provide reference information on Novell® ZENworks® Linux Management imaging utilities, commands, and configuration settings.

C.1 Image Explorer (imgexp.exe)

Use the Image Explorer utility at a Windows workstation to view or modify workstation images, create add-on images, compress image files, and split images.

Although ZENworks Imaging Explorer looks and functions like Microsoft Windows Explorer in most situations, some functionality differences exist between the two programs. The following describes the key differences between how ZENworks Image Explorer and Microsoft Windows Explorer function:

- **Replacing Files in an Image:** During the life cycle of an image, files might be deleted or updated using Image Explorer. When you replace an existing file in an image by using Image Explorer, the original file is not deleted from the image. Image Explorer purges only deleted files; it does not purge files that have been updated.

  When files are added to an image where the file already exists, Image Explorer appends the entry to the end of the image. When images are restored, all files that have been previously updated are sequentially restored.

  To avoid performance problems, you should manually delete and purge each instance of a duplicate file in order to have the duplicates purged from the image. In Windows Explorer, replaced files are automatically deleted.

- **Dragging Files from Image Explorer:** You cannot drag files from Image Explorer in order to extract them, which you can do in Windows Explorer. However, you can drag and drop files and folders into an image using Image Explorer.

IMPORTANT: When editing a base image, do not exclude BPB files from it or the device won't be able to boot the new operating system after receiving the image.

The following sections describe the tasks that you can perform using the Image Explorer:

C.1.1 Starting Image Explorer (imgexp.exe)

The Image Explorer utility must be run on a Windows device. You need Samba running on the Linux imaging server where the utility file is located in order for the Windows device to have access to it.

There are no command line parameters for the Image Explorer utility.

1 To start Image Explorer, run the following file:

```
/opt/novell/zenworks/zdm/imaging/winutils/imgexp.exe
```

C.1.2 Opening an Image

1 Start Image Explorer.

2 Click on the toolbar, browse for and select the image (`.zmg`) file, then click *Open*.

Large image files might take a few moments to open.

C.1.3 Adding a File or Folder to an Open Image

1 Start Image Explorer.

2 In the left pane, browse to the partition or directory where you want to add the file or folder.

3 Click or on the toolbar, browse to the file or folder, then click *Add* or *OK*.

C.1.4 Creating a Folder in an Open Image

1 Start Image Explorer.

2 In the left pane, browse to the partition or folder where you want to create the folder, click 📁⃰, type the name of the folder, then click *OK*.

C.1.5 Excluding a File or Folder from a File Set in the Open Image

1 Start Image Explorer.

2 Select the file or folder, click *Edit*, click *File sets*, then select the file sets that you want the file or folder to be excluded from.

   This image has 10 possible file sets, labeled Set 1, Set 2, and so on. The files and/or folders that you selected in the main window is excluded only from the file sets that you select in this dialog box.

C.1.6 Marking a File or Folder for Deletion in the Open Image

1 Start Image Explorer.

2 Select the file or folder, click *Image*, then click *Delete*.

IMPORTANT: Deleting a file in the Image Explorer merely marks it for deletion; it can still be retrieved. A file marked as deleted is not removed from the image until the image is purged; files and folders marked as deleted are not restored during imaging.

C.1.7 Purging Files and Folders Marked for Deletion from the Open Image

1 Start Image Explorer.

2 Ensure that the open image has been saved, click *File*, then click *Purge deleted files*.

3 Browse to the image filename or specify a new image filename, then click *Save*.

C.1.8 Extracting a File or Directory from the Open Image to a Folder

1 Start Image Explorer.

2 Click the file or directory, click *File > Extract > As files*, browse to and select a folder, then click *OK*.

C.1.9 Extracting a File or Directory from the Open Image As an Add-On Image

1 Start Image Explorer.

2 Click the file or directory, click *File > Extract > As add-on image*, type the name of the new add-on image, then click *OK*.

C.1.10 Viewing a File from the Open Image in its Associated Application

1 Start Image Explorer.

2 Click the file, then click *File > Extract and view*.

C.1.11 Saving Your Changes to the Open Image

1 Start Image Explorer.

2 Click 🖫 on the toolbar.

C.1.12 Creating an Add-On Image

1 Start Image Explorer.

2 Click ☐ on the toolbar, open Windows Explorer, browse to the files and folders you want the add-on image to contain, drag the files and folders into the right pane from Windows Explorer, then click *Save*.

C.1.13 Adding a Partition to a New Add-On Image

1 Start Image Explorer.

2 Click ☐ on the toolbar, click the root of the image, click *Image*, then click *Create partition*.

You cannot add a partition to an existing add-on image or to any base image.

C.1.14 Compressing an Image

You can set compression options so that it takes less time to restore the image file and less space to store the file on your imaging server. You can compress an uncompressed image (including images created by previous versions of ZENworks) by 40 to 60 percent of the original file size.

The ZENworks Linux Management Image Explorer provides the following types of image compression:

- ◆ "Compressing an Open Image" on page 480
- ◆ "Compressing Any Image without Waiting for the File to Fully Load into Image Explorer" on page 481

Compressing an Open Image

1 Start Image Explorer.

2 Browse for the image (`.zmg`) file, then click *Open*.

Large image files might take a few moments to open.

3 Click *File > Compress image*.

4 Browse to a folder, specify a new image filename, then select a compression option:

Optimize for Speed: Takes the least amount of time to compress, but creates the largest compressed image file.

Balanced (Recommended): Represents a compromise between compression time and image file size. This option is used by default when an image is created.

Optimize for Space: Creates the smallest image file, but takes longer to compress.

5 Click *Compress*.

Files marked for deletion in the image are removed during the compression operation.

Compressing Any Image without Waiting for the File to Fully Load into Image Explorer

You can set compression options to quickly compress an image file without waiting for the file to fully load into Image Explorer.

To use QuickCompress:

1 Start Image Explorer.

2 Click *Tools* > click *QuickCompress*.

3 Browse to the image file, browse to a folder, specify a new image filename, then select a compression option:

Optimize for Speed: Takes the least amount of time to compress, but creates the largest compressed image file.

Balanced (Recommended): Represents a compromise between compression time and image file size. This option is used by default when an image is created.

Optimize for Space: Creates the smallest image file, but takes longer to compress.

4 Click *Compress*.

Files marked for deletion in the image are removed during the compression operation.

C.1.15 Splitting an Image

You can split an image file into separate files so that you can span the entire image across several CDs or DVDs.

When you split a device image and span it across several CDs or DVDs, you are essentially creating a base image on the first CD or DVD. The remaining CDs or DVDs are add-on images.

To restore a device image that has been spanned across several CDs or DVDs you should restore the first CD or DVD before restoring the remaining CDs or DVDs containing the add-on images. For more information, see .

Restoring split Images is a manual task and can only be automated using scripted imaging. For more information, see .

To split an image:

1 Start Image Explorer.

2 Click *Tools* > *Image split*.

3 Specify an existing base image file to split, specify the directory in which to store the split images, then specify the maximum file size of each split-image file.

Because images are split by placing individual files into different images, an image cannot be split if it contains any single file that is larger than the specified maximum file size.

4 Click *Split*.

C.1.16 Resizing a Partition in an Image

For base images, you can edit the value in the *Original size* field to allow you to change how big the ZENworks Imaging Engine makes the partition when the image is restored.

For example, suppose you create a base image of a device with a 20 GB hard drive and you want to then put that image on a new device with a 60 GB hard drive. If you do not increase the size of the partition, the partition will be 20 GB, thus making the remaining 40 GB unusable.

However, if you increase the number in the *Original size* field to match the size of the new hard drive, the ZENworks Imaging Engine expands the partition when the image is restored so that you can use the entire drive.

To resize a partition:

1 Start Image Explorer.

2 Right-click a partition in the left frame, then click *Properties*.

3 Increase or decrease the value in the *Original size* field.

You cannot decrease the number in the *Original size* field to a smaller value that what is in the *Minimum size* field.

The *Original size* field is not applicable for add-on images and cannot be modified for them.

C.2 Novell ZENworks Linux Management Imaging Agent (novell-zislnx)

The Novell ZENworks Linux Management client (which includes novell-zislnx) should be installed on devices where you want to apply images. For information on installing the client on your devices, see "Setting Up Managed Devices" in the *Novell ZENworks 7 Linux Management - Dell Edition Installation Guide*.

Installing the Linux Management client automatically installs the Novell ZENworks Linux Management Imaging Agent (novell-zislnx). The Imaging Agent's purpose is to save certain device-unique data (such as IP addresses and host names) to an area on the hard disk that is safe from imaging. The Imaging Agent records this information when you install it on the device. Then the agent restores this information from the image-safe area after the device has been imaged. This allows the device to use the same network identity as before.

The Imaging Agent is installed on your imaging server by default when you install ZENworks Linux Management.

If a device is new and does not contain a unique network identity, the default settings that you have configured for the ZENworks Management Zone are applied when you image the device using a Preboot Services Imaging bundle.

The data that the Imaging Agent saves to (or restores from) the image-safe area includes the following:

◆ Whether a static IP address or DHCP is used

- If a static IP address is used:
  - IP address
  - Subnet mask
  - Default gateway (router)
- DNS settings
  - DNS suffix
  - DNS hostname
  - DNS servers

The novell-zislnx daemon is generally run automatically. However, if you want to run it manually, for the command line arguments that can be used with the Imaging Agent, see "Understanding Script Arguments" on page 499.

C.3 Image-Safe Data Viewer and Editor (zisview and zisedit)

After booting a device from an imaging boot media (PXE, CD, DVD, or ZENworks partition), you can enter `zisedit` and `zisview` at the Linux bash prompt to edit and view the image-safe data for that device.

The following sections contain additional information:

- Section C.3.1, "Information Displayed by the Image-Safe Data Viewer," on page 483
- Section C.3.2, "Using the Image-Safe Data Viewer," on page 485
- Section C.3.3, "Using the Image-Safe Data Editor," on page 486

C.3.1 Information Displayed by the Image-Safe Data Viewer

After booting a device from an imaging boot media, enter `zisview` at the Linux bash prompt to view the image-safe data for that device.

The image-safe data viewer (`zisview`) displays the following information about the device:

Table C-1 *zisview Information*

Category	Information
Image-safe Data	◆ **Version:** The version number of the Novell ZENworks Linux Management Imaging Agent (novell-zislnx). ◆ **Just Imaged Flag:** If this is set to False, the Imaging Agent reads data from Linux and writes it to the image-safe data store. If this is set to True, the Imaging Agent reads data from the image-safe data store and writes it to Linux. ◆ **Scripted Image Flag:** If this option is set to True, the last imaging operation was a scripted image. If this option is set to False, the last imaging operation was not a scripted image. ◆ **Last Base Image:** The last base image that was restored to the device. ◆ **Last Base Image Time:** The time stamp of the last base image that was restored to the device. ◆ **Last Base Image Size:** The size of the last base image that was restored to the device. ◆ **Last Base Image Address:** The IP address of the last base image that was restored to the device. ◆ **Script Checksum:** Displays the checksum value representing the last script run. The ZENworks Imaging Engine uses the checksum to prevent the same script from re-running on the device unless you specify in the ZENworks Control Center that you want to rerun the same script.
Device Identity Information	◆ **Zone GUID:** The ZENworks Management Zone that contains the device, if it has been imported. ◆ **Device GUID:** The Globally Unique Identifier of this computer's device. ◆ **Device Index:** The device identification number. ◆ **Win 9x Computer Name:** The computer name for the device. [1] ◆ **Windows Workgroup:** The Microsoft network workgroup of the device. [1] ◆ **Windows SID:** The Windows Security ID of the device, a unique number that identifies this device in Windows. [1]

Category	Information
Network Information	◆ **DHCP:** Displays whether this device uses DHCP to obtain its IP address. ◆ **IP Address:** Displays the static IP address that this device uses. ◆ **Gateway:** Displays the gateway that this device uses. ◆ **Subnet Mask:** Displays the subnet mask that this device uses. ◆ **DNS Servers:** The number of DNS nameservers used for DNS name resolution. ◆ **DNS Server [0]:** The IP address of the DNS server. This line is repeated, numbering from 0, 1, 2, 3, and so on for each DNS name server. For example, if the DNS Servers number is 3, there will be three of these lines, numbered from 0 through 2. ◆ **DNS Suffix:** The DNS context of the device. ◆ **DNS Hostname:** The DNS local hostname of the device. Use this field to change the computer name of Linux devices.

[1] The *Win 9x Computer Name*, *Windows Workgroup*, and *Windows SID* device identity information fields are present for imaging compatibility with ZENworks Desktop Management. These fields are not relevant to Linux devices.

C.3.2 Using the Image-Safe Data Viewer

To use zisview, enter any of the following commands at the Linux bash prompt:

Table C-2 *Data Viewer Commands*

Command	Explanation
zisview	Displays all image-safe data.

Command	Explanation
`zisview -z` *`field`*	Displays information about a specific field or fields. *field* is one or more field names separated by a space. *field* is not case-sensitive.
	All of the following are valid field names (the corresponding minimum names that can also be entered on the command line follow each field name in parenthesis):
	JustImaged (J) ScriptedImage (SC) LastBaseImage (L) Zone GUID (T) Device GUID (ObjectDN) Device Index (N) Windows WorkGroup (WorkG) Windows SID (SI) WorkstationID (Works) DHCP (DH) IP (I) Gateway (Gateway) Mask (M) DNSServerCount (DNSServerC) DNSServer (DNSServer) DNSSuffix (DNSSu) DNSHostName (DNSH)
`zisview -s`	Creates a script that can be used to generate environment variables that contain all of the image-safe data fields.
`zisview -h`	Displays help for `zisview`.

C.3.3 Using the Image-Safe Data Editor

After booting a device from an imaging boot media, you can enter `zisedit` at the Linux bash prompt to change, clear, or remove information the image-safe data for that device.

To use `zisedit`, enter any of the following commands at the Linux bash prompt:

Table C-3 *zisedit Commands*

Command	Explanation
`zisedit`	Displays a screen showing all of the image-safe data fields. You can add or change any of the information in the fields.

Command	Explanation
zisedit *field*=*new_information*	You can change the information for one field using this syntax, where *field* is any valid field name and *new_information* is the information you want this field to contain. *field* is not case sensitive.
	For example, enter zisedit Mask=255.255.252.0 to enter this information in the *subnet mask* field.
	All of the following are valid field names (the corresponding minimum names that can also be entered on the command line are shown in parenthesis after each field name):
	JustImaged (J) ScriptedImage (SC) LastBaseImage (L) Zone GUID (T) Device GUID (ObjectDN) Device Index (N) Windows WorkGroup (WorkG) Windows SID (SI) WorkstationID (Works) DHCP (DH) IP (I) Gateway (Gateway) Mask (M) DNSServerCount (DNSServerC) DNSServer1 (DNSServer1) DNSSuffix (DNSSu) DNSHostName (DNSH) PXEWorkRevision (PXEWorkR) PXEWorkObject (PXEWorkO) PXETaskID (PXETaskI) PXETaskState (PXETaskS) PXETaskRetCode (PXETaskR)
zisedit -c	Clears all image-safe data fields.
zisedit -r	Removes the image-safe data store.
zisedit -h	Displays help for zisedit.

C.4 ZENworks Imaging Floppy Boot Disk Creator (zmediacreator.exe)

You can use this utility to do the following:

* Create a floppy boot diskette to help devices that cannot boot from their CD or DVD to do so
* Manage the settings.txt file
* Create a Preboot Bootable Image (PBI)

The ZENworks Imaging Media Creator utility must be run on a Windows device. You need Samba running on the Linux imaging server in order for the Windows device to have access to the utility.

The `zmediacreator.exe` file is located at `/opt/novell/zenworks/zdm/imaging/winutils/zmediacreator.exe` on your ZENworks Linux Management imaging server.

For instructions on using the utility, see Section 26.2.3, "Using the ZENworks Imaging Media Creator," on page 297.

C.5 Imaging Configuration Parameters (settings.txt)

The `settings.txt` file contains parameters that control how the imaging boot process occurs. A copy is located in the `/opt/novell/zenworks/zdm/imaging/winutils` directory on the imaging server where ZENworks Linux Management is installed. You should maintain the working copy of `settings.txt` at the root of the imaging boot device (CD or DVD, or ZENworks partition).

`Settings.txt` is a plain text file that contains various parameters, each on a separate line. Each parameter has the general format of PARAMETER=value. Lines that begin with a pound sign (#) signify comments and are ignored during the imaging boot process.

You can edit this file manually in a text editor, or by making configuration changes in the `zimgboot.exe` utility (see Section C.4, "ZENworks Imaging Floppy Boot Disk Creator (zmediacreator.exe)," on page 487).

IMPORTANT: If you manually edit the `settings.txt` file to provide any paths to executables, make sure that you provide the full path, or the executable might not run.

The format and function of each parameter in the `settings.txt` file are described in Table C-4:

Table C-4 *Settings.txt File Parameters*

Parameter	Specifies
PROMPT	Specifies whether to prompt for each configuration setting when you boot a device from the imaging boot media.
	If you leave this parameter commented out or set it to No, the device boots using the configuration settings specified in `settings.txt` and you can't override the settings when booting, unless you type `config` at the boot prompt before the Linux operating system begins to load.
	If you set this parameter to Yes, you are automatically prompted for each configuration setting when booting.

Parameter	Specifies
MANUALREBOOT	Specifies whether you must reboot a device manually after it was booted from the imaging boot media in automatic mode. If the device was booted from the imaging boot media in manual mode, you must always reboot the device manually. If you boot a device from the imaging boot media and you let the boot process proceed in automatic mode, the ZENworks Imaging Engine starts and checks the imaging server to see if an imaging operation should be performed on the device. If so, it performs the imaging operation and quits. If not, it quits without doing anything. What happens next depends on how you set this parameter: ◆ If you leave it commented out or set it to No, you are prompted to remove the imaging boot media (if necessary) and press any key to reboot the device to the native operating system. ◆ If you set this parameter to Yes, the device doesn't reboot automatically, but instead displays the Linux bash prompt, allowing you to perform additional imaging-related tasks using the Linux menu or at the command line. This is helpful if you want to do things such as check the current partition information or the image-safe data before booting to the native operating system. Example: MANUALREBOOT=YES
PARTITIONSIZE	Specifies the number of megabytes to allocate to the ZENworks partition if you choose to create one locally on a device when you boot the device from the imaging boot media. The default size is 150 MB. The minimum partition size is 50 MB. The maximum size allowed is 2048 MB (2 GB). If you plan to store an image in the ZENworks partition, such as to enable the device to be restored to a certain state without connecting to the network, you might want to specify a larger size on this parameter. Example: PARTITIONSIZE=500
netsetup	If you are using DHCP, keep this option enabled. If you are using a specific IP address, replace "dhcp" with "1" and uncomment and configure the other three IP address lines (HostIP, NETMASK, and GATEWAY). Example: netsetup=dhcp
HostIP	The IP address used by a device to communicate on the network when you boot the device from the imaging boot media, if a static IP address is needed. Example: HostIP=137.65.95.126 If you want DHCP to be used, leave this and the next two parameters commented out.
NETMASK	Specifies the subnet mask to be used by the device, if the device is using a static IP address. Example: NETMASK=255.255.252.0 If DHCP is being used, leave this parameter commented out.

Parameter	Specifies
GATEWAY	Specifies the IP address of the gateway (router) to be used by the device, if the device is using a static IP address. Example: GATEWAY=137.65.95.254 If DHCP is being used, leave this parameter commented out.
NAMESERVER	Specifies the list of DNS name servers, by IP address, to use for resolving DNS domain names used on this device. Use a space to separate entries. Example: NAMESERVER=123.45.6.7 123.45.6.9 If DHCP is being used, leave this parameter commented out.
DOMAIN	Specifies the list of DNS domain suffixes to be used to identify connections used by this device. Use a space to separate entries. For example: DOMAIN=example.novell.com example.xyz.org If DHCP is being used, leave this parameter commented out.
PROXYADDR	Specifies the IP address or full DNS name of the imaging (proxy) server to connect to when you boot a device from the imaging boot media in auto-imaging mode. Examples: PROXYADDR=137.65.95.127 PROXYADDR=imaging.xyz.com This parameter is used to set the PROXYADDR environment variable in Linux when the device is booted from an imaging boot media (other than PXE). The ZENworks Imaging Engine then reads this variable to determine which server to contact if it is running in automatic mode. Whether it is running in automatic or manual mode, the ZENworks Imaging Engine attempts to log the imaging results to the server specified in this variable. **IMPORTANT:** This parameter is set automatically when booting PXE and normally should not be specified in `/srv/tftp/boot/settings.txt`, which is the copy of settings.txt that is used by PXE.
/bin/setleds -D +num < /dev/tty1	Turns on NUMLOCK upon booting.
export PS1="\`pwd \#"	Configures the string used by the bash shell. You can change the string by editing the text after the = symbol. The ' character is not a single quote mark, but is from the ~ key.
export IMGCMD	Use to alter the behavior of automated imaging. If this variable is defined as a script (or a series of commands), then that script (or those commands) are executed instead of the usual `img auto` command (see `/bin/imaging.s`).

Parameter	Specifies
export ENTERPISE_NAME=*name*	This feature is not supported in Novell ZENworks Linux Management - Dell Edition.
	This should be a valid Enterprise Name for an AMT device, such as entZENworks. It allows imaging utilities to access image-safe data in AMT NVRAM when AMT devices are disconnected from the ZENworks Management Zone.
	If you do not use this parameter for disconnected AMT devices, the imaging utilities might not be able to keep the image-safe data up to date.
netdevice=eth0	Selects a specific network adapter. If necessary, replace `eth0` with the correct interface.
noshell=1	Suppresses a secondary terminal program from displaying.

C.6 Imaging Boot Parameter for PCMCIA Cards

When performing imaging work using CDs or DVDs, some computers (particularly laptops) with PCMCIA cards can hang during the boot process. By default, ZENworks Linux Management allows the loading of a PCMCIA driver when a device boots for imaging work. Although loading this driver does not normally cause problems, you can use a command line parameter to prevent it from loading.

To prevent the PCMCIA card manager from starting, enter the following at the bash prompt when booting from a CD or DVD:

```
manual NoPCMCIA=1
```

C.7 Imaging Server

The imaging server is a software component of the Linux Management server. It enables imaging clients to connect with the network to receive imaging services, including:

- Storage or retrieval of an image on a server
- Automatic imaging based on settings created in the ZENworks Control Center
- Logging of the results of an imaging operation
- A multicast imaging session

Use the imaging server software to do the following:

- Section C.7.1, "Initiating the Imaging Processes," on page 491
- Section C.7.2, "Viewing Information About Imaging Requests," on page 500
- Section C.7.3, "Starting a Manual Multicast Session," on page 500

C.7.1 Initiating the Imaging Processes

An imaging server daemon is initiated by running the script at the Linux terminal program command line, which in turn calls the executable and uses the configuration set in the corresponding `.conf`

file. Because the scripts do not normally accept parameters, but only arguments (such as start), you can configure parameters in their corresponding .conf files.

The following Linux daemons run the imaging server processes:

- "novell-pbserv" on page 492
- "novell-proxydhcp" on page 493
- "novell-tftp" on page 495
- "novell-zmgprebootpolicy" on page 497
- "Understanding Script Arguments" on page 499

novell-pbserv

The novell-pbserv daemon provides imaging services to devices.

This daemon is started automatically when installing ZENworks Linux Management, or when rebooting the server.

- "Understanding the novell-pbserv Components" on page 492
- "Configuring novell-pbserv" on page 492

Understanding the novell-pbserv Components

To initiate the novell-pbserv daemon, enter the following command on the Linux command line in a terminal program:

```
/etc/init.d/novell-pbserv
```

Table C-5 lists the arguments for this command, the executable it starts, and the configuration file it uses:

Table C-5 *Novell-pbserv Command Details*

Script Arguments:	start, stop, restart, force-reload, status, showpid (for descriptions of these arguments, see "Understanding Script Arguments" on page 499)
Executable:	/opt/novell/zenworks/preboot/bin/novell-pbservd
Configuration File:	/etc/opt/novell/zenworks/preboot/novell-pbserv.conf

Configuring novell-pbserv

The novell-pbserv configuration file (novell-pbserv.conf), contains the following parameters:

Table C-6 *Novell-pbserv Parameters*

Parameter	Description
EnableLogging=YES	If YES, a log file is created for debug messages. This is the default.
	If NO, no log file is created for debug messages.
	The `novell-pbserv.log` file is created in the `/var/opt/novell/log/zenworks/preboot` directory.
IPAddress=	The IP address to be used by imaging for all communications. If nothing is entered, novell-pbserv attempts to detect an IP address.
	Can be used in a clustering environment to specify the IP address of the virtual server.
	Can also be used in a multiple-NIC environment to bind the imaging server to a specific IP address.
	By default, this is commented out.
LIBRARY_NAME=	Full path of the library to be loaded by the ZENWorks Imaging Service. If the library name is not specified, then by default `libzenimgweb.so` is loaded from the `/opt/novell/zenworks/preboot/lib` directory.
	By default, this is commented out.

novell-proxydhcp

The novell-proxydhcp daemon provides PXE devices with the information that they require to be able to connect to the ZENworks Preboot Services system.

This daemon is not started automatically when installing ZENworks Linux Management.

- ◆ "Understanding the novell-proxydhcp Components" on page 493
- ◆ "Configuring novell-proxydhcp" on page 494

Understanding the novell-proxydhcp Components

To initiate the novell-proxydhcp daemon, enter the following command listed for Script Location on the Linux command line in a terminal program:

`/etc/init.d/novell-proxydhcp`

Table C-7 lists the arguments for this command, the executable it starts, and the configuration file it uses:

Table C-7 *Novell-proxydhcp Command Details*

Script Arguments:	start, stop, restart, force-reload, status, showpid, install (for descriptions of these arguments, see "Understanding Script Arguments" on page 499)
Executable:	`/opt/novell/bin/novell-proxydhcpd`
Configuration File:	`/etc/opt/novell/novell-proxydhcp.conf`

Configuring novell-proxydhcp

The novell-proxydhcp configuration file (`novell-proxydhcp.conf`), contains the following parameters:

Table C-8   *Novell-proxydhcp Parameters*

Parameter	Description
LocalDHCPFlag = 0	Indicates whether the DHCP server for this subnet resides on the same server as novell-proxydhcp. 0 (the default) means novell-proxydhcp is not running on the same server as the DHCP service. 1 means they are running on the same server. The Proxy DHCP server needs to behave slightly differently if it is loaded on the same server as the DHCP service.
LocalInterface = 10.0.0.1	Indicates the IP address to be used by the Proxy DHCP server. This setting is intended only for use on servers with multiple LAN interfaces. The IP address must be valid on the server. By default, this parameter is commented out.
NovellPolicyEngine = 10.0.0.1	The IP address of the server where a Novell Preboot policy engine is running. Most often, this is a ZENworks imaging daemon. If no value is specified, the Proxy DHCP assumes that the daemon is running on the same server. By default, this parameter is commented out.
NBPx86 = nvlnbp.sys	The name of the boot file this service will suggest for all x86 computers, such as `nvlnbp.sys`.
MenuTimeout = 2	The number of seconds the F8 menu is displayed before automatically choosing the first option, which is always this server and its default NBP. The default is 2 seconds.
ProxyLogLevel = 2	The value assigned here determines which events are entered in `novell-proxydhcp.log`. Specifying a high level in an active system can quickly fill the log. Valid values are: 0, 1, 2, 3, and 4. The default is 2. Each message from the Proxy DHCP server is assigned a priority level. If *ProxyLogLevel* is set to a value equal to or greater than a message's priority level, that message is entered in `novell-proxydhcp.log`. All other messages are ignored. Priority meaning: **0:** Critical information. Service start, stop, and critical events are logged. **1:** Warning information. Additionally, warning information is logged. **2:** Transaction information. All completed client transactions are logged. **3:** Request information. All client requests and Proxy DHCP requests received are logged, including ignored requests. If a request is ignored, the reason for ignoring it is also logged. **4:** Debugging information. All DHCP packets received and accepted are decoded and logged.

Parameter	Description
ProxyLogFile = /var/ opt/novell/log/novell-proxydhcp.log	The file where all log file entries are placed. It is located at `/var/opt/ novell/log/novell-proxydhcp.log`.
	By default, this parameter is commented out.
ProxyLogFileSize = 15	The size of the *ProxyLogFile* file is controlled by the value of *ProxyLogFileSize*, where 15 is the default (in MB).
	When the log file exceeds the *ProxyLogFileSize* value, it is deleted and restarted.

Parameters that are not commented out, but contain no values, are given a default value.

The novell-proxydhcp daemon is compliant with the following RFCs:

RFC 2131 - Dynamic Host Configuration Protocol

RFC 2132 - DHCP Options and BOOTP Vendor Extensions

The novell-proxydhcp daemon is compliant with the Preboot eXecution Environment (PXE) Specification v2.1 industry specification, published by Intel.

novell-tftp

The novell-tftp daemon provides TFTP services to imaging clients.

This daemon is started automatically when installing ZENworks Linux Management, or when rebooting the server.

- "Understanding the novell-tftp Components" on page 495
- "Configuring novell-tftp" on page 495

Understanding the novell-tftp Components

To initiate the novell-tftp daemon, enter the following command (listed under Script Location) on the Linux command line in a terminal program:

`/etc/init.d/novell-tftp`

Table C-9 lists the arguments for this command, the executable it starts, and the configuration file it uses:

Table C-9 *Novell-tftp Command Details*

Script Arguments:	start, stop, restart, force-reload, status, showpid (for descriptions of these arguments, see "Understanding Script Arguments" on page 499)
Executable:	`/opt/novell/bin/novell-tftpd`
Configuration File:	`/etc/opt/novell/novell-tftp.conf`

Configuring novell-tftp

The novell-tftp configuration file (`novell-tftp.conf`), contains the following parameters for the Novell TFTP server:

Table C-10   *Novell-tftp Parameters*

Parameter	Description
TFTPInterface = 10.0.0.1	The IP address that is used for all TFTP communications. If a value is not given here, the service tries to detect one.
	This value is most useful for multihomed servers.
	By default, this parameter is commented out.
TransferBlockSize = 1428	This value determines the size of the data block used by the TFTP server to transmit and receive data to and from a client. Valid values are between 512 and 4428.
	For ethernet networks, this value should be 1428.
	For token ring networks, this value can be 4428, but only if you are sure there are no ethernet segments; otherwise, use 1428.
	Older TFTP clients might be restricted to 512 bytes, the original transfer block size before the adoption of RFC 2348. The Novell TFTP server is compatible with these clients.
	By default, this parameter is commented out.
TimeoutInterval = 1	This is the amount of time (in seconds) that the TFTP server waits for a client to acknowledge before resending a packet. However, because the TFTP server uses an adaptive algorithm to calculate the actual timeout interval, this value is only used as an initial value. It may increase or decrease, depending on the performance of the network.
	This value is only a default. It can be changed at the request of a client. See RFC 2349.
	Valid values are 1 through 60.
	By default, this parameter is commented out.
Linux -- TFTPDirectory = /srv/tftp	*TFTPDirectory* is the directory where the TFTP server can store and retrieve files. All paths submitted to the TFTP server by clients are assumed to be relative to this directory.
	Because TFTP has no security, it is suggested that you not place files with sensitive information in this directory, and that you place a space quota on it.
	The TFTP server does not load if this directory does not exist.
	By default, this parameter is commented out.
TFTPAllowWrites = 1	This variable tells the TFTP server whether to allow users to place new files on the server. Setting this variable to 0 makes the TFTP server more secure by not allowing users to place new files on the server. The other option is 1 (the default), which allows users to place new files on the server.

Parameter	Description
TFTPLogLevel = 2	The value assigned here determines which events are entered in `novell-tftp.log`. Specifying a high level in an active system can quickly fill the log. Valid values are: 0, 1, 2, 3, and 4. The default is 2. Each message from the TFTP server is assigned a priority level. If *TFTPLogLevel* is set to a value equal to or greater than a message's priority level, that message is entered in `novell-tftp.log`. All other messages are ignored. Priority meaning: **0:** Critical information. Service start, stop, and critical events are logged. **1:** Warning information. Only failed client transactions are logged. **2:** Transaction information. All completed client transactions are logged. **3:** Request information. All client requests and TFTP options are logged. **4:** Debugging information. All server events, including each packet received, are logged. By default, this parameter is commented out.
TFTPLogFile = /var/opt/novell/log/novell-tftp.log	The file where all log file entries are placed. By default, this parameter is commented out.
TFTPLogFileSize = 15	The size of the log file is controlled by the value of *TFTPLogFileSize*, where 15 is the default (in MB). When the log file exceeds the *TFTPLogFileSize* value, it is deleted and restarted. By default, this parameter is commented out.

Parameters that are not commented out, but contain no values, are given a default value.

The novell-tftp daemon is compliant with the following RFCs:

    RFC 1350 -- THE TFTP PROTOCOL (REVISION2)
    RFC 2347 - TFTP Option Extension
    RFC 2348 - TFTP Blocksize Option
    RFC 2349 - TFTP Timeout Interval and Transfer Size Options

novell-zmgprebootpolicy

The novell-zmgprebootpolicy daemon allows PXE devices to query the ZENworks Linux Management system for work to do and for Preboot Menu policies.

This daemon is started automatically when installing ZENworks Linux Management, or when rebooting the server.

- ◆ "Understanding the novell-zmgprebootpolicy Components" on page 498
- ◆ "Configuring novell-zmgprebootpolicy" on page 498

Understanding the novell-zmgprebootpolicy Components

To initiate the novell-zmgprebootpolicy daemon, enter the following command (listed under Script Location) on the Linux command line in a terminal program:

`/etc/init.d/novell-zmgprebootpolicy`

Table C-11 lists the arguments for this command, the executable it starts, and the configuration file it uses:

Table C-11   Novell-zmgprebootpolicy Command Details

Script Arguments:	start, stop, restart, force-reload, status, showpid (for descriptions of these arguments, see "Understanding Script Arguments" on page 499)
Executable:	`/opt/novell/zenworks/preboot/bin/novell-zmgprebootpolicyd`
Configuration File:	`/etc/opt/novell/zenworks/preboot/novell-zmgprebootpolicy.conf`

Configuring novell-zmgprebootpolicy

The novell-zmgprebootpolicy configuration file (`novell-zmgprebootpolicy.conf`), contains the following parameters:

Table C-12   Novell-zmgprebootpolicy Parameters

Parameter	Description
LocalInterface = 10.0.0.1	The IP address that is used by the Policy server. This setting is intended only for use on servers with multiple LAN interfaces. The address must be valid on the server. By default, this parameter is commented out.
PolicyLogLevel = 1	The value assigned here determines which events are entered in `novell-zenprebootpolicy.log`. Specifying a high level in an active system can quickly fill the log. Valid values are: 0, 1, 2, 3, and 4. The default is 2. Each message from the novell-zmgprebootpolicy server is assigned a priority level. If *PolicyLogLevel* is set to a value equal to or greater than a message's priority level, that message is entered in `novell-zenprebootpolicy.log`. All other messages are ignored. Priority meaning: **0:** Critical information. Service start, stop, and critical events are logged. **1:** Warning information. Only failed client transactions are logged. **2:** Transaction information. All completed client transactions are logged. **3:** Request information. All client requests are logged. **4:** Debugging information. All server events, including each packet received, are logged. By default, this parameter is commented out.

Parameter	Description
PolicyLogFile = /var/opt/novell/log/zenworks/preboot/novell-zenprebootpolicy.log	The file where all log file entries are placed. By default, this parameter is commented out.
PolicyLogFileSize = 15	The size of the log file is controlled by the value of *PolicyLogFileSize*, where 15 is the default (in MB). When the log file exceeds the *PolicyLogFileSize* value, it is deleted and restarted.
PrebootServer = 10.0.0.5	This field contains the address of the imaging server that should be used to resolve policies. By default, this parameter is commented out.
EnableAMTSupport = Yes	This feature is not currently supported in Novell ZENworks Linux Management - Dell Edition. This field enables or disables support for Intel's AMT technology. By default, this support is disabled by commenting out the parameter.

Parameters that are not commented out, but contain no values, are given a default value.

Understanding Script Arguments

The following arguments are available for each of the Preboot Services daemons described above:

Table C-13 *Script Arguments*

Argument	Function
start	Starts the daemon. Because novell-proxydhcp is optional, use this argument to start this daemon. However, this daemon does not automatically start when the server reboots. (See install below.)
start setjustimagedflag	For novell-zislnx only, it sets the Just Imaged flag so that a device can be imaged using its existing Image Safe Data.
stop	Stops the daemon.
restart	Stops and restarts the daemon if it is already running.
force-reload	Causes the daemon's configuration file to be reloaded.
status	Displays the current status of the daemon. For example, if you enter `/etc/inid.d/novell-pbserv status`, information similar to the following is returned: `Novell ZENworks Imaging Service running`

Argument	Function

showpid	Displays the daemon's process ID.
	
	For example, if you enter `/etc/inid.d/novell-pbserv showpid`, information similar to the following is returned:
	
	`Novell ZENworks Imaging Service running`
	`10211`

install	For novell-proxydhcp only, causes the daemon to be automatically loaded when the server boots.

C.7.2 Viewing Information About Imaging Requests

After the imaging server has started, you can view information about the status and results of the imaging requests that it has received from imaging clients. A statistical summary of these requests is shown on the server's terminal program command line. The statistics shown on this screen are explained below. All statistics are reset to zero if you restart the imaging server.

Table C-14 *Imaging Request Statistics*

Statistic	Specifies
Update Requests	The number of imaging requests of any kind that have been received by the imaging server since it was started. This includes requests that failed, were denied, or were referred to other imaging servers (see Client Referrals below). Information about each of these requests, such as the source, type, date/time, and results, is logged on the imaging server.
Images Sent	The number of images that the imaging server has sent to imaging clients since the imaging server was started. This includes only images that were retrieved from this imaging server. See Client Referrals below for more information.
Images Received	The number of new images that have been received and stored on the imaging server since it was started. This includes images that were received through client referrals (see Client Referrals below).
Client Referrals	The number of client requests that have been referred (redirected) by the imaging server to other imaging servers since this imaging server was started. Such referrals are made only when the client is running in auto-imaging mode and the imaging server determines that the image to be created or retrieved is on a different imaging server. **IMPORTANT:** If a client is running in manual imaging mode and it requests to store or retrieve an image on a different imaging server, the request is denied and an error is returned to the client. Referrals are currently supported only when the client is running in auto-imaging mode.

C.7.3 Starting a Manual Multicast Session

At the bash prompt, you can start a manual multicast session, see any sessions in progress, and delete sessions. For more information, see "Initiating a Multicast Session from Each Client" on page 396.

ZENworks Imaging Engine Commands

D

After booting a device from an imaging boot media, you can use the img command at the Linux bash prompt or the ZENworks® Imaging Engine menu to do any of the following:

- ◆ Take an image of the device's hard disks
- ◆ Put down an image on the device's hard disks
- ◆ View or manipulate the device's hard disk partitions
- ◆ View the device's hardware configuration or image-safe data
- ◆ Display a menu from which you can also perform all of these tasks

The ZENworks Imaging Engine is installed to the /bin directory on the imaging boot device. If the imaging boot device is a diskette, a CD, or DVD, the /bin directory is actually archived in the root file, which is expanded during the imaging boot process. If the imaging boot method is Preboot Services, the ZENworks Imaging Engine is downloaded to the device when booting.

Because the ZENworks Imaging Engine is a Linux application, the command syntax is case sensitive. The overall syntax is:

`img [mode]`

where *mode* is any of the modes described in the following sections:

Each mode can be abbreviated to the first letter of its name. For example, `img dump` can be abbreviated as `img d`.

To access the ZENworks Imaging Engine menu and perform all of these tasks, enter `img` with no parameters.

D.1 Help Mode (img help)

Use Help mode to get information about the img command if you don't have this documentation available.

To use the Help mode:

1 Do the one of following:

- Enter:

    ```
    img [help [mode]]
    ```

    where *mode* is the mode whose command syntax you want help with.

    Examples:

Example	Explanation
img help	Displays a short description of each mode.
img help m	Displays information on how to use the Make mode.
img help p	Displays information on how to use the Partition mode.

- Enter `img` to display the ZENworks Imaging Engine menu, click *Help*, then select a mode name.

D.2 Automatic Mode (img auto)

Use automatic mode to image the device automatically, based on any applicable Preboot Services default settings. The ZENworks Imaging Engine runs in this mode if you let the imaging boot process proceed without interruption, or if you type the command below at the Linux prompt.

To use the automatic mode, do any of the following at the bash prompt:

- Enter:

    ```
    img auto
    ```

- To display the ZENworks Imaging Engine menu, enter:

    ```
    img
    ```

    and on the menu bar, click *Imaging*, then click *Query for work*.

- To display the ZENworks Imaging Engine menu, enter:

    ```
    img
    ```

    then click *F9 Query for work* on the task bar.

- To display the ZENworks Imaging Engine menu, enter:

    ```
    img
    ```

    then press *F9*.

In this mode, the ZENworks Imaging Engine queries the imaging server specified in the PROXYADDR environment variable for any work to do. The imaging server checks the relevant Preboot Services default settings to determine what imaging tasks should be performed (if any), such as taking or putting down an image. It then instructs the ZENworks Imaging Engine to perform those tasks. If any tasks involve storing or retrieving images on other imaging servers, the imaging server refers the ZENworks Imaging Engine to those servers to complete those tasks. After the ZENworks Imaging Engine has completed its work, it communicates the results to the original imaging server, and the results are logged on that server.

For information on configuring the settings that control what happens in this mode, see Section 26.4, "Configuring Preboot Services Defaults," on page 320.

D.3 Make Mode (img make)

Use the Make mode to take an image of the device and store it in a specified location. Normally, all partitions on the local hard disks are included in the image, but there are some exceptions noted in Table D-1 on page 503.

You can take an image of a device using either the bash prompt or using the ZENworks Imaging Engine menu. For step-by-step instructions, see "Manually Taking an Image of a Device" on page 374. You can also use the Make Locally mode to take an image of the device and store it in a partition on a local hard disk. For step-by-step instructions, see Section 27.4.3, "Setting Up Disconnected Imaging Operations," on page 383.

The image size corresponds to about half the size of the data in all of the device's partitions, except that the ZENworks partition and Compaq or Dell configuration partitions are always excluded. Devices with logical volumes (LVMs) are not supported for imaging.

The syntax of this mode depends on whether you store the image locally or on an imaging (proxy) server.

The following sections contain additional information:

- Section D.3.1, "Make Locally (img makel)," on page 503
- Section D.3.2, "Make to Proxy (img makep)," on page 504

D.3.1 Make Locally (img makel)

Using the bash prompt, the following example explains the syntax and available parameters that you can use with the `makel` "make locally" parameter:

```
img makel[pNumber] filepath [comp=comp level] [xpartition]
```

Commands

Table D-1 *makel Commands*

Parameter	Specifies
makel[*pNumber*]	The partition number (as displayed by `img dump`) of the local partition for where to store the image. It must be a primary partition. This partition is excluded from the image that is created. If you omit the partition number from this parameter, the image is stored in the local ZENworks partition.
filepath	The image filename, including a `.zmg` extension (case sensitive) and the complete path from the root of the partition. The directories in the path must exist. If the file already exists, it is overwritten. However, you are prompted to verify whether to overwrite.

Parameter	Specifies
[comp=*comp level*]	*comp level* is the amount of compression used when creating the image. Specify any number from 0-9. 0 means no compression. 1 is the same as *Optimize for Speed* and is used by default if you do not specify this parameter. 6 is the same as *Balanced*. 9 is the same as *Optimize for Space*.
x*partition*	The partition number (as displayed by `img dump`) of a local partition to exclude from the image. You can repeat this parameter as needed to exclude multiple partitions. If you omit this parameter, all partitions are included in the image except the one where the image is stored.

Examples

Table D-2 *makel Examples*

Example	Explanation
`img makel8 /imgs/dellnt.zmg`	Takes an image of all partitions except the one in slot 8 and saves the image to `imgs/dellnt.zmg` in the partition in slot 8. (Assumes that slot 8 contains a primary partition.)
`img makel /imgs/dellnt.zmg`	Takes an image of all partitions and saves it to `imgs/dellnt.zmg` in the ZENworks partition. (Assumes that the partitions have been installed.)
`img makel /imgs/dellnt.zmg x2 x3`	Takes an image of all partitions except those in slots 2 and 3 and saves the image to `imgs/dellnt.zmg` in the ZENworks partition. (Assumes that the partitions have been installed.)

D.3.2 Make to Proxy (img makep)

Using the bash prompt, the following example explains the syntax and available parameters that you can use with the `makep` "make to proxy" parameter:

```
img makep address filepath [comp=comp level] [xpartition]
```

Commands

Table D-3 *makep Commands*

Parameter	Specifies
address	The IP address or DNS name of the imaging server to store the image on.

Parameter	Specifies
filepath	The image filename, including a `.zmg` extension (case sensitive) and the complete path in UNC style. The directories in the path must exist. If the file already exists, the imaging server won't overwrite it unless you enable this behavior in the ZENworks Control Center. If no folders are specified in the path, the image is created at the root of the volume or drive where the ZENworks Linux Management imaging server software is installed.
	IMPORTANT: Because Linux doesn't recognize backslashes, you must use forward slashes in the UNC path, or enclose the entire path in quotes.
[comp=*comp level*]	*comp level* is the amount of compression used when creating the image. Specify any number from 0-9. 0 means no compression. 1 is the same as *Optimize for Speed* and is used by default if you do not specify this parameter. 6 is the same as *Balanced*. 9 is the same as *Optimize for Space*.
x*partition*	The partition number (as displayed by `img dump`) of a local partition to exclude from the image. You can repeat this parameter as needed to exclude multiple partitions.
	If you omit this parameter, all partitions are included in the image.

Examples

Table D-4 *makep Examples*

Example	Explanation
`img makep 123.45.67.890 //xyz_server/sys/imgs/dellnt.zmg`	Takes an image of all partitions and saves it to `sys/imgs/dellnt.zmg` on `xyz_server`. (Assumes that 123.45.67.890 is the IP address of `xyz_server`.)
`img makep img.xyz.com //xyz_server/sys/imgs/dellnt.zmg x2 x3`	Takes an image of all partitions except those in slots 2 and 3 and saves the image to `sys/imgs/dellnt.zmg` on `xyz_server`. (Assumes that `img.xyz.com` is the DNS name of `xyz_server`.)

D.4 Restore Mode (img restore)

Use the Restore mode to retrieve an image from a specified location and put it down on a device.

You can restore an image of a device using either the bash prompt or using the ZENworks Imaging Engine menu. For step-by-step instructions, see "Manually Taking an Image of a Device" on page 374. You can also use the Restore mode to restore an image from a partition on a local hard disk. For step-by-step instructions, see Section 27.4.3, "Setting Up Disconnected Imaging Operations," on page 383.

Normally, if the image to be put down is a base image (one created previously by the ZENworks Imaging Engine), all existing partitions except the ZENworks partition and Dell or Compaq configuration partitions are removed from all local hard disks before the new image is put down. When the image is put down, the sizes of the original partitions from which the image was taken are preserved, if possible. If there is insufficient space, the last partition is shrunk to fit, unless this

would result in data loss, in which case the ZENworks Imaging Engine denies the requested operation. If there is extra space left after all partitions in the image have been restored to their original sizes, that space is left unpartitioned.

If the image to be put down is an add-on image, or if it's a base image and you specify the a*partition*:p*partition* parameter, none of the existing physical partitions are removed. Instead, the appropriate partitions are merely updated with the files from the image, overwriting any existing file of the same name and location.

Restoring add-on images over 4 GB in size is not supported by Linux Management imaging.

The syntax of this mode depends on whether you will retrieve the image from a local device or from an imaging (proxy) server, as explained in the subsections below:

- Section D.4.1, "Restore from Local (img restorel)," on page 506
- Section D.4.2, "Restore from Proxy (img restorep)," on page 508

D.4.1 Restore from Local (img restorel)

Use the Restore from Local mode to retrieve an image from a local device and put it down on the device. For more information, see Section 27.4.3, "Setting Up Disconnected Imaging Operations," on page 383.

Using the bash prompt, the following example explains the syntax and available parameters that you can use with the `restorel` "restore from local" parameter:

`img restorel[`*pNumber*`] `*filepath*` [s`*fileset*`] [a`*partition*`:p`*partition*`]`

Commands

Table D-5 *restorel Commands*

Parameter	Specifies
restorel[*pNumber*]	The partition number (as displayed by `img dump`) of the local partition to retrieve the image from. It must be a primary partition. This partition is not changed by the imaging operation.
	If you omit the partition number from this parameter, the image is retrieved from the local ZENworks partition.
filepath	The filename of the image to retrieve, including the `.zmg` extension (case-sensitive) and the complete path from the root of the partition.
s*fileset*	The number of the image file set to put down. Valid values are 1 through 10. For information on creating file sets of an image, see Section 25.5.3, "Creating, Installing, and Restoring Standard Images," on page 290.
	If you omit this parameter, file set 1 is used.

Parameter	Specifies
apartition:p*partition*	A mapping between a partition in the image archive (*apartition*) and a target physical partition on the local machine (p*partition*). Use this parameter to selectively restore a specific part of the image to a specific local partition.
	IMPORTANT: If you use this parameter, none of the existing local partitions are removed, and only the target local partition is updated. The update process does not remove any existing files; however, any existing files of the same names are overwritten. If you want to remove all existing files from the target partition before updating it, first use the Partition Mode (img part) to delete and recreate the partition.
	For *apartition*, use the partition number displayed for the source partition in the Image Explorer (imgexp.exe) utility. For p*partition*, use the partition number displayed by img dump for the target partition. The target partition must be a partition of a supported file system. You can repeat this parameter as needed to request multiple selective restorations in a single operation. In doing so, you can apply multiple parts of the image to a single local partition, but you can't apply the same part of an image to multiple local partitions in a single operation.

Examples

Table D-6 *restorel Examples*

Example	Explanation
img restorel8 /imgs/dellnt.zmg	Removes all existing local partitions except the one in slot 8, retrieves the image from imgs/dellnt.zmg in slot 8, and puts down the partitions and contents of that image on the available local writable devices (assuming there is sufficient local space and that slot 8 contains a primary partition).
img restorel /imgs/dellnt.zmg	Removes all existing local partitions, retrieves the image from imgs/dellnt.zmg in the ZENworks partition, and puts down the partitions and contents of that image on the available local writable devices (assuming there is sufficient space).
img restorel /imgs/dellnt.zmg s2	Removes all existing local partitions, retrieves the image from imgs/dellnt.zmg in the ZENworks partition, and puts down the partitions and contents of file set 2 of that image on the available local writable devices (assuming there is sufficient space).
img restorel /imgs/dellnt.zmg a2:p1 a3:p1	Retrieves the image from imgs/dellnt.zmg in the ZENworks partition, updates local partition 1 with the data from partitions 2 and 3 of that image, and leaves the other local partitions unchanged (assuming there is sufficient space in local partition 1).

D.4.2 Restore from Proxy (img restorep)

Use the Restore from Proxy mode to retrieve an image from an imaging (proxy) server and put it down on the device. For more information, see "Manually Putting an Image on a Device" on page 379.

Using the bash prompt, the following example explains the syntax and available parameters that you can use with the `restorep` "restore from proxy" parameter:

```
img restorep address filepath [sfileset] [apartition:ppartition]
```

Commands

Table D-7 *restorep Commands*

Parameter	Specifies
address	The IP address or DNS name of the imaging server to retrieve the image from.
filepath	The filename of the image to retrieve, including the `.zmg` extension (case-sensitive) and the complete path in UNC style. **IMPORTANT:** Because Linux doesn't recognize backslashes, you must use forward slashes in the UNC path or enclose the entire path in quotes.
s*fileset*	The number of the image file set to put down. Valid values are 1 through 10. For information on creating file sets of an image, see Section 25.5.3, "Creating, Installing, and Restoring Standard Images," on page 290. If you omit this parameter, file set 1 is used.
a*partition*:p*partition*	A mapping between a partition in the image archive (a*partition*) and a target physical partition on the local machine (p*partition*). Use this parameter to selectively restore a specific part of the image to a specific local partition. **IMPORTANT:** If you use this parameter, none of the existing local partitions are removed, and only the target local partition is updated. The update process does not remove any existing files or overwrite any existing files of the same names if they are newer. If you want to remove all existing files from the target partition before updating it, first use the Partition Mode (img part) to delete and recreate the partition. For a*partition*, use the partition number displayed for the source partition in the Image Explorer (imgexp.exe) utility. For p*partition*, use the partition number displayed by `img dump` for the target partition. The target partition must be a partition of a supported file system. You can repeat this parameter as needed to request multiple selective restorations in a single operation. In doing so, you can apply multiple parts of the image to a single local partition, but you can't apply the same part of an image to multiple local partitions in a single operation.

Examples

Table D-8 *restorep Examples*

Example	Explanation
`img restorep 137.65.95.127 // xyz_server/sys/imgs/dellnt.zmg`	Removes all existing local partitions, retrieves the image from `sys/imgs/dellnt.zmg` on `xyz_server`, and puts down the partitions and contents of that image on the available local writable devices (assuming there is sufficient local space and that 137.65.95.127 is the IP address of `xyz_server`).
`img restorep img.xyz.com // xyz_server/sys/imgs/dellnt.zmg s2`	Removes all existing local partitions, retrieves the image from `sys/imgs/dellnt.zmg` on `xyz_server`, and puts down the partitions and contents of file set 2 of that image on the available local writable devices (assuming there is sufficient local space and that `img.xyz.com` is the DNS name of `xyz_server`).
`img restorep img.xyz.com // xyz_server/sys/imgs/dellnt.zmg a2:p1`	Retrieves the image from `sys/imgs/dellnt.zmg` on `xyz_server`, updates local partition 1 with the data from partition 2 of that image, and leaves the other local partitions unchanged (assuming there is sufficient space in local partition 1 and that `img.xyz.com` is the DNS name of `xyz_server`).

D.5 Session (Multicast) Mode (img session)

Use the Session (Multicast) mode to take an image of one device and put it down on multiple other devices simultaneously over the network in a single operation.

IMPORTANT: For multicasting to work properly, the routers and switches on the network must have multicast features configured. Otherwise, multicast packets might not be routed properly.

For multicasting to work, each participating device must boot from an imaging boot media and run the ZENworks Imaging Engine in this mode, as explained below. The device from which the image is taken is called the *master*, and the devices that receive the image are called *participants*.

You can start the multicast session from the imaging server. If you start the session this way, you specify an image file for multicasting rather than a device as the session master. Otherwise, if you start the session from a client device, you can specify one of the session participants as the session master. In that case, an image of the session master's hard drive is sent to the session participants. For more information, see .

Using the bash prompt, the following example explains the syntax and available parameters that you can use with the `session` parameter:

```
img session name [master|client] [clients=count [t=minutes]]
```

Commands

Table D-9 *session Commands*

Parameter	Specifies
name	The name of the multicast session. Each device joining the session uses the same value for this parameter.
	IMPORTANT: The name must be unique among concurrent multicast sessions. It is hashed by the ZENworks Imaging Engine to produce a Class D IP address for the multicast session. To facilitate troubleshooting (wire sniffing), all Linux Management imaging multicast addresses start with 231. For example, the session name `mcast01` can produce the multicast address 231.139.79.72.
master|client	Specifies that this device is the session master or a session client.
	If you omit this parameter, the ZENworks Imaging Engine waits for the user at the master device to press `m` to designate that device as the master, or it waits for another device to be declared master for the imaging session to be started from the imaging server by selecting *Manually start multicast*, providing the required information, then selecting *Yes*.
clients=*count*	The number of participating devices that must register with the master before imaging begins. This option only applies to session masters.
	If you omit this parameter, the ZENworks Imaging Engine waits for the user at the master device to press `g`. After imaging has begun, any participating devices attempting to register are denied.
time=*minutes*	The number of minutes the master device waits for the next participant to register before starting the imaging process without reaching *count* registered participants. This option only applies to session masters.
	If you omit this parameter, the imaging process does not start until *count* is reached, or the user at the master device presses `g`. After that, any participants attempting to register are denied and queued for the next multicast session.

Examples

Table D-10 *session Examples*

Example	Explanation
`img session mcast01`	Starts a multicast session named `mcast01`. Each successive device that issues this same command before the imaging begins joins the session. Imaging doesn't start until one of the users presses `m` to designate himself as master and presses `g` to start the imaging, or the imaging session is started from the imaging server by selecting *Manually start multicast*, providing the required information, then selecting *Yes*.

Example	Explanation
`img session mcast01 m`	Starts a multicast session named `mcast01` and designates this device as the master. Each successive device that issues `img session mcast01` before the imaging begins joins the session as a participant. Imaging doesn't start until the master user presses `g`.
`img session mcast01 master clients=5`	Starts a multicast session named `mcast01`. Each successive device that issues `img session mcast01` before the imaging begins joins the session. Imaging doesn't start until one of the users presses `m` to designate himself as master, or until the imaging session is started from the imaging server by selecting *Manually Start Multicast*, providing the required information, then selecting *Yes*. Five other devices must also register as participants before the session begins.
`img session mcast01 master clients=5 time=20`	Starts a multicast session named `mcast01`. Each successive device that issues `img session mcast01` before the imaging begins joins the session. Imaging doesn't start until one of the users presses `m` to designate himself as master, or until the imaging session is started from the imaging server by selecting *Manually Start Multicast*, providing the required information, then selecting *Yes*. Either five other devices must register as participants or more than 20 minutes must elapse between any consecutive participant registrations, whichever occurs first, and then the session begins.

D.6 Partition Mode (img part)

Use the Partition mode to activate (make bootable), add, or delete a partition on the device.

You can activate, add, or delete a partition using either ZENworks Imaging Engine menu or the bash prompt.

The Partition mode can be used in two ways:

D.6.1 Using the ZENworks Imaging Engine Menu

1 Enter `img` to display the ZENworks Imaging Engine menu, then click *Partitioning*.

2 Click *Modify partitions*, then click an option:

Active: Select a partition that you want to activate (make bootable), then click *Active*.

Add: Opens the Create New Partition window. Click a partition type, partition size, and cluster size, then click *OK*.

Delete: Select a partition, then click *Delete*.

D.6.2 Using the Bash Prompt

1 From the bash prompt, enter:

`img p`*operation*

where *operation* is one of the following:

Operation	Action
pc*pNumber type* [*size*] [cluster=*clusterSize*]	Creates a new partition, where: • *pNumber* is the number of the partition slot (as displayed by `img dump`) in which to create the partition • *type* is a keyword, a partition name, Extended, or a numerical value for the partition type, for example 0x0C (hexadecimal) or 11 (decimal) If you are creating an extended partition, you can create a logical drive inside of the extended partition. (See the next table for an example.) • *size* is a valid size for the partition type in MB or a percentage If you omit this parameter, the largest valid size for the partition type is used, given the available unpartitioned space on the drive. If you give a percentage, include the % symbol; otherwise, the value is considered the size in MB. The new partition is recognizable by other operating systems, but must be formatted or have a base image restored to it before you can store files in it.
pd*pNumber*	Deletes the partition from slot number *pNumber*. Use `img dump` to get the slot number.
pd-all	Deletes all non-protected partitions.
pa*pNumber*	Activates (make bootable) the partition in slot number *pNumber*. Use `img dump` to get the slot number.

The following are examples:

Example	Explanation
`img pc1 ext2`	Creates the ext2 partition in slot 1 using all of the available unpartitioned space on the drive.
`img pc5 reiser 5671`	Creates a Reiser partition in slot 5 using 5,671 MB on the drive.
`img pd3`	Deletes the partition from slot 3.
`img pc2 extended 2500` `img pc2 reiser 500`	Creates an extended partition with a 2500 ext2 logical drive and a 500 MB Reiser logical drive.

D.7 ZENworks Partition Mode (img zenPartition)

Use the ZENPartition mode to enable, disable, or remove the installed ZENworks partition.

1 Do one of the following:
   ◆ From the bash prompt, enter the following:

```
img zenPartition operation
```

where *operation* is enable, disable, or remove.

* Enter `img` to display the ZENworks Imaging Engine menu, click *Partitioning*, then click one of the following:

Disable ZENworks partition

Enable ZENworks partition

Remove ZENworks partition

2 Enter `lilo.s` to make this change effective.

IMPORTANT: If you remove an installed ZENworks partition, you must immediately restore a base image with a valid non-LILO MBR (Master Boot Record). If you do not, the device cannot boot properly.

D.8 Dump Mode (img dump)

Use the Dump mode to view information about the hard drives and partitions on the device.

1 Do one of the following:

* Enter `img` to display the ZENworks Imaging Engine menu, click *System information*, then click *Drive information*.

* Enter the following:

```
img dump [geo]
```

where:

Parameter	Action
dump	Lists the existing partitions on all local hard drives. For each partition, the type, size, and slot number of the partition are given. The ZENworks partition and Dell or Compaq configuration partitions are not listed.
geo	Displays additional information about the geometry (cylinders, heads, and sectors) and capacity of each hard drive.

Examples:

Example	Explanation
img dump	Lists the current partitions on all local writable devices.
img dump geo	Lists all hard drives, their geometry and capacity, and the current partitions on the writable devices.

D.9 Information Mode (img info)

Use the Information mode to view the following:

* The data currently stored in the image-safe area on the device

This data is saved by the Novell ZENworks Linux Management Imaging Agent (novell-zislnx) during each device's session to ensure that it can be restored after the device is reimaged. If the device is new and doesn't have an operating system yet, an initial set of data is supplied from the default configuration for the ZENworks Management Zone, such as IP addresses.

 * Information about the hardware devices on the device

   This information is detected during the imaging boot process. If the ZENworks Imaging Engine runs in auto-imaging mode, this information is sent to the imaging server to help determine which image to put on the device, if necessary.

 * Name of the base image that was last put down on the device

To use the Information mode:

1 Enter `img` to display the ZENworks Imaging Engine menu, click *System information*, then click *Image-safe data* or *Detected hardware*. (See Table D-11 for details.)

or

Enter the following from the bash prompt:

```
img info [zisd]
```

Commands

Table D-11   *Information Mode Parameters*

Menu item or parameter	Action

System Information > Detected Hardware
 	
or
 	
info *(from the bash prompt)*
	Lists the detected hardware devices on the device, including:
	
	  • CPU chipset
	  • BIOS asset tag
	  • BIOS serial number
	  • Video adapter
	  • Network adapter
	  • MAC address
	  • Sound card
	  • Hard drive controller
	  • Hard disk capacity
	  • Detected RAM
	  • Boot media

System Information > Image Safe Data *or* img info zisd *(from the bash prompt)*	Lists the data currently stored in the image-safe area on the device. The items that comprise this data are listed in Section C.3, "Image-Safe Data Viewer and Editor (zisview and zisedit)," on page 483. In addition to the image-safe data, the last base image that was put down on the device is also listed.

Examples

Table D-12 *Examples*

Example	Explanation
`img info`	Lists the detected hardware devices on the device.
`img info zisd`	Lists the Linux Management image-safe data currently stored on the device and the last base image that was put down.

Updating ZENworks Imaging Resource Files

E
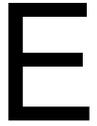
In Novell® ZENworks® 7 Linux Management, you can manually update ZENworks imaging resource files.

The following sections provide concepts on how the boot process works with ZENworks imaging, and instructions for updating imaging resource files:

E.1 The Linux Distribution for Imaging

ZENworks Imaging uses a small Linux distribution on the client device to perform imaging operations. The distribution shipping with ZENworks 7 is based on the SUSE® installation system, where SUSE Linux or SUSE Linux Enterprise Server (SLES) boot to a small distribution to perform a YaST installation. ZENworks Imaging uses the same installation system found in SLES, but instead of starting a YaST installation, it starts a ZENworks Imaging session.

In ZENworks 6.5 SP1 and earlier, Linux kernel 2.4.x is used in the customized distribution. In ZENworks 6.5 SP2, the kernel is updated to 2.6 and is a SLES-based distribution.

Using a stable Linux distribution based on SLES gives customers a distribution with the broadest range of stable drivers available. The hardware industry is continually introducing new and updated network and disk drivers, so it's not always possible to provide the latest drivers in its software releases.

This section covers how to update Linux drivers using the new distribution. It deals with the imaging resource files that are based on the SLES distribution and ZENworks Preboot Services processing.

E.2 Understanding Device Boot Processes in a ZENworks Imaging Environment

The following provides a high-level overview of a Linux boot process and how ZENworks 7 imaging affects it:

1. A boot loader program loads the Linux kernel and initrd (initial RAM drive) into memory.

   The SLES-based imaging distribution uses `isolinux` as the boot loader for imaging CDs, a modified `pxelinux` for booting using PXE, or `linld.com` when using a single diskette with the CD. If you have a ZENworks partition installed, it uses the lilo program to boot alternately between the ZENworks partition and the installed operating system.

   Following are the filenames and paths:

Files	When booting from a CD	When booting from PXE
Preboot Loader	`isolinux`	`linld.com`
Linux Kernel Name	`/boot/loader/linux`	`/srv/tftp/boot/linux`
Initrd Filename	`/boot/loader/initrd`	`/srv/tftp/boot/initrd`

2. The Linux kernel starts running, does some device driver setup, then mounts the `initrd` file system.

   Regardless of which boot loader method is used, the main purpose is to set up the `initrd` file as a RAM drive, load the Linux kernel into memory, then turn control over to it with an indication to the Linux kernel of where to find `initrd`.

3. The Linux kernel turns control over to linuxrc, for performing initial hardware detection. When finished, control is returned to the Linux kernel.

4. The Linux kernel starts a background process (`/sbin/init`).

   After control is passed to the linuxrc program, control is never returned to the Linux kernel or passed on to the init process.

For more information on `linuxrc` and `zenworks.s`, review the following sections:

E.2.1 linuxrc

When control is turned over to linuxrc, there are several processes it performs to get the system ready for the imaging process. Linuxrc is initially configured from the `/linuxrc.config` file, which is located in the `initrd` file system. Additional configuration information for linuxrc can be placed in the `/info` file (located in the `initrd` file system), but ZENworks does not normally use this information.

Linuxrc also loads a `root` file system, which is combined with the `initrd` file system that is set up by the boot loader. The `root` file system is located on an imaging CD as the file `/boot/root`. For PXE booting, the `root` file system is stored on the ZENworks imaging server as `/srv/tftp/boot/root`.

Linuxrc attempts to locate and load the `settings.txt` file, either on the root of the imaging CD, or on the ZENworks imaging server in the `/srv/tftp` directory. From `settings.txt`, linuxrc reads and processes any parameters that pertain to itself, then copies `settings.txt` to the root (`/`) of the file system.

Linuxrc then also attempts to locate and load a file named `driverupdate`. It is usually located in the same directory as `root`. This file is used to update drivers and other files in the imaging distribution.

The `driverupdate` file is based on standard SUSE technology during a PXE boot. Because the network must be operating normally in order to obtain `driverupdate`, this file cannot update drivers for the active network device. However, other files and drivers can be updated by using the `driverupdate` file. For more information, see Section E.4.3, "Using the Driverupdate File Method," on page 523.

E.2.2 zenworks.s

A normal SUSE installation for SUSE Linux or SLES boots to a small distribution to perform a YaST installation. ZENworks Imaging boots with the same installation system, but instead of starting a YaST installation, it starts the ZENworks Imaging process. Control is turned over to the ZENworks script `/bin/zenworks.s`, which is the main script file for ZENworks imaging processing. The script performs a certain number of setup tasks, then gives control to the appropriate script for the selected imaging process. For more information on the imaging process, see Chapter 25, "Understanding Preboot Services in ZENworks Linux Management," on page 273.

One of the setup tasks is to apply any update files. When booting from a CD, `zenworks.s` copies the `/addfiles` directory structure to the Linux file system. For more information, see Section E.4.1, "Adding Files to an Imaging Boot CD," on page 521.

E.3 Understanding ZENworks Partitions and Command Line Parameters

The following sections provide an understanding of the ZENworks partition and imaging commands that are used when updating Linux drivers:

- Section E.3.1, "The ZENworks Partition," on page 519
- Section E.3.2, "Command Line Parameters and Variables," on page 520

E.3.1 The ZENworks Partition

The ZENworks partition is used to store the files required to load Linux into RAM, making the result similar to using a CD or PXE boot method. The ZENworks partition has a similar boot media layout as an imaging CD. It has a minimum size of 150 MB.

The files stored on the ZENworks partition are `/boot/loader/linux`, `/boot/loader/initrd`, and `/boot/root`, which are the same directories as on the imaging CD. In ZENworks 7, the boot loader continues to be lilo, which loads Linux as described under Section E.2, "Understanding Device Boot Processes in a ZENworks Imaging Environment," on page 518. The `driverupdate` and `settings.txt` files are searched for and loaded from the ZENworks partition.

If you need to modify the Linux files, you must modify the initrd or root file sets the same way as you would for other boot methods. For information, see Section E.4.2, "Adding Files to the Initrd or Root File Systems," on page 522.

E.3.2 Command Line Parameters and Variables

There are four types of command line parameters that can be used with the ZENworks imaging process. They are entered manually on the command line when booting from a CD or they can be placed in the `isolinux.cfg` file located in the `/boot/loader` directory. The commands are also located in the `*.cfg` files for PXE and are located in the `/srv/tftp` directory on the ZENworks imaging server.

- **Kernel parameters:** The valid parameters for the Linux kernel are found in the `/Documentation/kernel-parameters.txt` file that is installed with the kernel source.

  Some devices have a faulty BIOS, where you must turn off ACPI processing for the kernel to load properly. To do this, use the kernel parameter acpi=off. For more information, see Novell Support (http://www.novell.com/support/search.do?cmd=displayKC&docType=kc&externalId=10099330&sliceId=&dialogID=1284337&stateId=1%200%20548668).

- **Linuxrc parameters:** These parameters affect the way linuxrc detects hardware or sets hardware settings. They are described briefly in the `/usr/share/doc/packages/linuxrc/linuxrc.html` file in a Linux system.

  Linuxrc parameters can be found in the `/linuxrc.config` or `/info` files that reside in the `initrd` file system. Some parameters can be placed in the `settings.txt` file that is located on the root of the imaging CD or ZENworks partition, or in the `/srv/tftp/boot` file for PXE booting.

  Parameters that can be placed in the `settings.txt` file (the easiest file to edit) are limited. During PXE booting, parameters that affect the network are not processed from `settings.txt`, because by the time linuxrc loads the `settings.txt` file, the network is already set up. Network settings can be placed in the `settings.txt` file when booting from an imaging CD, because it is loaded early enough in the process to take effect.

- **ZENworks variables:** Some environment variables affect the way imaging performs. They can be configured in any file, but should normally be configured in the `settings.txt` file.

  If you add variables to the `settings.txt` file that were not originally defined there, you must export the variable. For example, in the settings.txt file, enter:

  `export IMGCMD="myscript"`

  A list of all image engine or script variables is listed under Section E.7, "Variables and Parameters," on page 528.

- **Other variables:** Environment variables that you might want in your script can be added in the same manner as described for the ZENworks variables.

E.4 Modifying ZENworks Imaging Resource Files

From time to time you might want to modify an imaging distribution by adding your own files. These can be additional programs, scripts, data files, or updated Linux drivers.

You can use the following methods to update imaging resource files:

- The easiest method is to edit the `settings.txt` file, which is located on the root of the imaging CD or in `/srv/tftp/boot` on the ZENworks imaging server for PXE booting.

- Where you are using a ZENworks partition, you can boot to the manual or maintenance mode, mount the ZENworks partition, then copy the modified `settings.txt` and the files in `initrd` or `root` to the mounted ZENworks partition.

- Another easy method is to edit the `.cfg` files located in `/srv/tftp` on the ZENworks imaging server for PXE booting.

- You can modify files in the `initrd` or `root` file systems, but you need a Linux environment for performing the modification process. Files required during the initial setup (during linuxrc processing time), such as LAN drivers, must be placed in the `initrd` file system. Other files that are not needed until the `zenworks.s` script file takes control can be placed in the `root` file system (for example, an imaging script), or you can use the `driverupdate` file.

  This method is discussed in this section.

The following sections provide various methods for modifying imaging resource files:

E.4.1 Adding Files to an Imaging Boot CD

If you have files to add to an imaging boot CD so they can be available for use when you get to the actual imaging process (such as scripts, but normally not driver modules), you can copy the files to the `/addfiles` directory on the imaging CD. This is an easy way to insert your script or other files into the distribution without modifying the `initrd` or `root` file systems. However, these files are not available during the boot and module loading phases.

Files should be placed below the `/addfiles` directory on the imaging boot CD in their proper directory names. They are then available in this directory structure during the imaging process.

An example of how you can add files:

1 If you want to execute your own script instead of the normal imaging process, create a script file named `myscript.s` and place it on the boot CD. For example, `/addfiles/bin/myscript.s`.

IMPORTANT: The script file must have proper LF line terminators that Linux requires, not the DOS CR and LF end-of-line characters. In other words, you cannot use `Notepad.exe` to create the script; you must use a text editor compatible with Linux or UNIX, such as TextPad.

---

2 To place the following line in the `settings.txt` file, enter:

`export IMGCMD=/bin/myscript.s`

When imaging is run, it runs `/bin/myscript.s` instead of using the normal `img auto` command.

E.4.2 Adding Files to the Initrd or Root File Systems

This is the preferred method for updating imaging resource files, and must be performed in a Linux environment.

Before performing the procedure given below, make sure you have created backup copies of any files you plan to change, specifically the `/srv/tftp/boot/initrd` file. If you want to change the files on an imaging CD, you need a program such as winiso, or some other process for extracting and replacing the file in the `bootcd.iso` image file.

IMPORTANT: When updating or adding files and Linux drivers in the `initrd` or `root` file systems, document the changes you make. When you receive updated resource files from Novell, they do not contain your customized changes. If the kernel version has changed with the newer resource files from Novell, previously added drivers must be updated either by obtaining a new version from the manufacturer or recompiling the driver using the correct Linux kernel version source.

You can use the following procedure for the `root` file system by simply replacing "initrd" with "root" in the steps. However, Linux drivers should always be placed in the `initrd` file system, not the `root` file system.

To add files to the `root` file system, you can also use the `driverupdate` file method described in

To modify the `initrd` or `root` file system:

1 Using a Linux device, create a working directory and change to that directory.

2 To copy initrd from the PXE server or the boot CD to the new working directory:

  ◆ For PXE, copy `\tftp\boot\initrd` to the Linux workstation's working directory.

  ◆ For the CD, extract `initrd` from the `\boot\loader` directory on the boot CD, then copy the extracted `initrd` to the Linux workstation's working directory.

3 To rename `initrd` to `initrd.gz`, enter:

```
cp initrd initrd.gz
```

4 To unzip the `initrd.gz` file, enter:

```
gunzip initrd.gz
```

5 To create another working directory for use as a mount point in the subsequent steps, enter:

```
mkdir work
```

6 To mount the `initrd` file system to the `/work` directory, enter:

```
mount -o loop initrd work
```

7 To copy your files or updated driver to the mounted `initrd` file system, enter:

```
cp /your_path/module.ko work/lib/modules/2.6.5-override-default/
initrd
```

where *your_path* is the path to the `module.ko` file and module is the name of the module.

Other files to be included in the `initrd` file system should be copied to the appropriate directory.

8 To unmount the `initrd` file system, enter:

```
umount work
```

9 To zip the new `initrd` file, enter:

```
gzip -v9c initrd > initrd.gz
```

10 To rename `initrd.gz` back to `initrd`, enter:

```
cp initrd.gz initrd
```

11 To copy the file back:

- ◆ For PXE, copy the updated `initrd` file to the `\tftp\boot` directory on the PXE server.

- ◆ For the CD, copy the updated `initrd` file to the `\boot\loader` directory on the boot CD.

E.4.3 Using the Driverupdate File Method

Another way to customize the imaging distribution is to utilize the driver update mechanism that is built into all SUSE distributions. This entails modifying a file named `driverupdate` that is located in the `/srv/tftp/boot` directory on your imaging server or on the root (`/`) of an imaging boot CD.

This method is a little less intrusive than modifying the `initrd` or `root` file systems. You simply create an additional file that is incorporated into the imaging operating system during boot time.

There are three types of driver update operations that can be performed:

- ◆ Install the kernel modules or hardware drivers
- ◆ Install files and execute a script
- ◆ Simply place files into the operating system

This section describes the second method. For more information on the other methods, see "Tech Talk #3 - Spittin' Image" (http://www.novell.com/connectionmagazine/2005/11/tech_talk_3.html) in the *Novell Connection Magazine*.

The example in this section takes the program "tree" that is not currently available in the imaging distribution and installs it at boot time.

The driver update mechanism seeks the `driverupdate` file, which contains a directory structure that mimics the directory structure in the operating system after a device has booted with the ZENworks distribution. If it is present, linuxrc downloads it during booting and incorporates it into the operating system dynamically.

The `driverupdate` file is a file system file that can be of any file system type, such as EXT3 or REISER. For simplicity, we'll use the CRAMFS file system in our example.

To place the tree program into the `driverupdate` file:

1 Create a working directory on your imaging server, such as `/work`.

2 (Conditional) If you are using the driverupdate file, download the `driverupdate.tgz` file into the `/work` directory, then untar it by entering:

```
mkdir work
cd work
wget http://www.novell.com/connectionmagazine/2005/11/download/
driverupdate.tgz
tar -xzvf driverupdate.tgz
```

The `driverupdate.tgz` file contains the same directory structure as is created in Step 3.

3 (Conditional) If you are manually creating the directories, create the following directory structure under the `/work` directory:

```
`-- dirstruct
    `-- linux
        `-- suse
            |-- i386-9.2
                |-- dud.config
                |-- inst-sys
                    `-- lib
                    `-- bin
                |-- adddir.s
```

The contents of the `dud.config` file should contain lines similar to those listed below. You should maintain the keywords by supplying your own data. However, you can use the listed values:

```
UpdateName:     ZENworks 7 Patch 1
UpdateID:       a37f92556e4dd99e
UpdatePriority: 100
```

The `adddir.s` file should be an executable script that contains the following lines:

```
echo "Processing: adddir.s" > /dev/tty3 2>&1
# driver update: add files to inst-sys
for i in /update/[0-9]*/inst-sys ; do
    [ -d "$i" ] && adddir "$i" /
done
# driver update: run update.pre scripts
for i in /update/[0?9]*/install/update.pre ; do
    echo "Processing: $i" > /dev/tty3 2>&1
    [ -x "$i" ] && "$i"
done
```

4 To copy the tree program into the `/bin` directory, enter:

```
cp /usr/bin/tree dirstruct/linux/suse/i386-9.2/inst-sys/bin/
```

5 To create the CRAMFS file, enter:

```
mkfs.cramfs dirstruct/ driverupdate
```

6 To copy the `driverupdate` file into `/srv/tftp/boot`, enter:

```
cp driverupdate /srv/tftp/boot
```

7 Add the following lines to the end of the `/srv/tftp/boot/settings.txt` file:

```
# SUSE driver update
for i in /update/[0?9]*/install/adddir.s ; do
    [ -x "$i" ] && "$i"
    rm $i
done
```

This causes the `adddir.s` script to run, which creates soft links to all of the new files being copied.

These lines might already be present in the `settings.txt` file.

8 Reboot the PXE-enabled device.

You should see the text "ZENworks 7 Patch 1" at the bash prompt after the operating system has booted.

9 Execute the tree program.

All of the files you put into the `driverupdate` file are now located under the `/update` directory in the operating system after booting. Then, the `adddir.s` script (or the code that you added to the `settings.txt` file in Step 7) creates soft links under the `root` file system that point to the corresponding files under the `/update` directory structure. You can verify this by running:

```
/# which tree
/bin/tree
/# ll /bin/tree
lrwxrwxrwx  1 root root 29 Aug 31 21:45 /bin/tree -> /update/000/inst-
sys/bin/tree
```

If you want to simply include a new hardware driver or kernel module in the imaging operating system, an easier process might be to copy the `.ko` file into the `/dirstruct/linux/suse/ i386-9.2/modules/` directory. Then, the imaging operating system automatically loads any `.ko` files that are in this directory.

E.5 Adding or Updating LAN Drivers

As LAN card manufacturers develop and release new LAN adapters, they usually release new or updated drivers as well. Sometimes the new LAN card functions properly with an earlier driver, and sometimes the earlier driver does not recognize the new LAN card and refuses to load. Occasionally, the older driver does load, but the LAN card exhibits serious performance problems. To obtain the full performance capabilities of a new LAN card, you should use the new driver.

The following sections explain how to obtain or compile drivers:

- Section E.5.1, "Obtaining Drivers," on page 525
- Section E.5.2, "Building Drivers," on page 525

If you need to load your drivers with specific parameters, see Section E.5.3, "Loading Drivers with Parameters," on page 527.

E.5.1 Obtaining Drivers

New LAN drivers should be obtained from the manufacturer. Most LAN card manufacturers have drivers available for free downloading from their Web site. Some drivers are available from www.scyld.com/network, and the source to the Broadcom BCM5700 driver can be downloaded from http://www.broadcom.com/drivers/downloaddrivers.php.

If a manufacturer has a binary driver compiled specifically for the kernel version used by ZENworks, you can obtain the driver and use one of the update methods to add the driver. ZENworks 7 is based on SLES 9 SP2, kernel version 2.6.5-7.191. If the driver is not for this specific version, you must obtain the source and compile it for this version. For more information, see Section E.5.2, "Building Drivers," on page 525.

E.5.2 Building Drivers

Nearly all Linux drivers are distributed in source code form and need to be compiled before they can be used. Follow the manufacturer's instructions included with the new driver to build the driver module. Many drivers can be built in such a way that they are built into the kernel itself; however, we recommend that LAN drivers be built as external kernel modules.

When building your LAN drivers, make sure that your build machine uses the same kernel as the imaging environment. If you have a LAN driver that doesn't load in your imaging environment, it usually means that you have a mismatch between your build environment and the imaging environment.

You can find the current kernel version of your Linux environment using the following command:
`uname -r`

However, you might need to modify the results from the `uname` command to get your kernel versions to match. For more information, see Section E.6, "Using Uname," on page 527.

To build your drivers:

- "Obtaining the Linux Source Code Tree" on page 526
- "Compiling the Module" on page 527

Obtaining the Linux Source Code Tree

To compile a module, you need the Linux source code tree that contains the configuration matching the ZENworks kernel. To obtain the necessary source code, see TID 10091818 in the Novell Support Knowledgebase (http://support.novell.com/search/kb_index.jsp).

To use the Linux source code tree:

1 Unzip the file and install the source code tree in the `/usr/src` directory.

For example, the tar file creates the following directories:
```
/usr/src/linux-2.6.5-7.191
/usr/src/linux-2.6.5-7.191-obj
```

2 Obtain the proper configuration file from one of the following locations:

- A running ZENworks imaging distribution file (`/proc/config`.gz)
- TID 10091818 in the Novell Support Knowledgebase (http://support.novell.com/search/kb_index.jsp).

3 Copy this configuration file to the directory created in Step 1.

For example, `/usr/src/linux-2.6.5-7.191.`

4 To create a link to the source tree:

4a

To change to the `/usr/src` directory, enter:
`cd /usr/src`

4b If there is a Linux soft link in the directory, delete it.

4c Create the Linux soft link, such as:
`ln -s linux-2.6.5-7.191 linux`

You now have the Linux kernel source tree and soft link ready for compiling the module. Continue with "Compiling the Module" on page 527.

Compiling the Module

To manually compile the module:

1 Install the source.

Follow the manufacturer's instructions to install the source.

Normally, the module source is in a directory under `/usr/src`. Module source files usually come in the form of a gzipped tar file (`.tar.gz` or `.tgz`). The file might also be a bzipped file (`.bz2`).

2 To compile the source:

2a Change directories to the source.

2b If you modified `uname` to change to the proper kernel version, issue a `make` command.

3 When you have your module compiled for ZENworks, take the generated `.ko` module file (make sure you select the proper module name and not a work `.ko` file) and install it by using the driver update method or placing it in the `initrd` file system.

E.5.3 Loading Drivers with Parameters

If there is a module that you want to load during the linuxrc processing time, and if linuxrc does not recognize that it needs to be loaded or you want to specify the load parameters, you can enter a line in the `linuxrc.config` or `/info` file. This file then needs to be updated in the `initrd` file system.

You might need to load a LAN driver module with specific parameters. You can do this with a line like:

```
insmod="moduleName parm=xxx"
```

This type of line is most commonly used to load a LAN driver with specific parameters, such as full duplex or specific speed.

E.6 Using Uname

The `uname` command enables you to find the current kernel version of your Linux environment. However, you might need to modify the results from the `uname` command to get your kernel versions to match.

The following steps modify the `uname` command to provide the value you need:

1 To obtain your current kernel version, enter:

```
uname -r
```

Write down the version number so you can use it in Step 4. This example uses version 2.6.13-15-smp from a SLES 9 SP2 installation.

2 To create a new directory, enter:

```
mkdir /bin/orig
```

3 To move the `uname` binary to the `/bin/orig` directory that you just created, enter:

```
mv /bin/uname /bin/orig/uname
```

4 Use a Linux editor (such as vi) to create the `/bin/uname` file that contains the following lines:

```
#!/bin/sh
#uname
if [ $KRNLVERSION"a" = "a" ] ; then
     if [ $(/bin/orig/uname -r) = "2.6.13-15-smp" ] ; then
          export KRNLVERSION=2.6.13-15-smp
     else
          export KRNLVERSION=2.4.31
     fi
fi
if [ $1"a" = "-ra" ] ; then
     echo $KRNLVERSION
else
     /bin/orig/uname $*
fi
```

> **IMPORTANT:** Replace the strings "2.6.13-15-smp" with the version you found in Step 1.

5 To make the new `uname` command script executable, enter:

```
chmod +x /bin/uname
```

6 Enter the following to cause the `uname -r` command to return a specific version, such as when compiling a module:

```
export KRNLVERSION="2.6.5-7.191"
```

7 Following the manufacturer's directions, compile the module using the appropriate `make` command.

8 Reset `uname` so that it returns actual values:

```
unset KRNLVERSION
```

E.7 Variables and Parameters

The following sections describe the variables and parameters used in updating resource files:

E.7.1 Imaging Script Variables

The following environment variables are used in imaging scripts and must not be modified:

Table E-1 *Imaging Script Variables*

Variable	Definition
ACTIVEPARTITION	Device of the active OS partition.
CDBOOT	YES = Booted from a CD.
DISABLEZEN	1 = Disable the ZENworks partition.
ENABLEZEN	1 = Re-enable the ZENworks partition.

Variable	Definition
ZENDEVICE	Device name of the ZENworks partition.
ZENPARTBOOT	YES = Booted from ZENworks partition.

The following environment variables can be modified or set in the `settings.txt` file:

Table E-2 *Environment Variables*

Variable	Definition
HDPARM	NO = Do not set hdparm parameters.
IMGCMD	Imaging command to run instead of the `img a` command.
MANUALREBOOT	YES = Do not automatically reboot.
PARTITIONSIZE	Size in MB to create the ZENworks partition.
PROXYADDR	IP/DNS address of the Imaging server.
PROMPT	Go to the bash prompt after imaging is complete.

E.7.2 Linuxrc Parameters Specified in Settings.txt

Table E-3 *Linuxrc Parameters*

Variable	Definition
netsetup	dhcp = Use DHCP.
	1 = Static IP.
HostIP	Static IP address to use.
NetMask	Network mask.
Gateway	Network gateway.
HostName	Host name to assign.
Nameserver	DNS name server.
Domain	Domain suffix.
NetDevice	ethx = Define which network device to configure.

E.7.3 Image Engine Variables

Table E-4 *Image Engine Variables*

Variable	Definition
DEVELOPER_LOG	"A" creates a verbose `imglog` debug file.

Variable	Definition
ZENIMGLOG	"A" creates a less verbose `imglog` debug file.
ZEN_IGNORE_GEO_MISMATCH	Ignore geometry device mismatches when restoring raw image formats.
NOABORTBUTTON	If defined, do not display the Abort button during imaging.

E.8 Troubleshooting Linux Driver Problems

E.8.1 Troubleshooting During the Boot Process

While booting ZENworks imaging, there are several things that you can do to help troubleshoot if there is a problem:

- Press Esc to see the kernel messages. Usually, messages are shown for failures.
- Screen 3 (press Alt+F3) is used to show the progress of the linuxrc process. It lists progress results, what linuxrc is doing, which modules are loaded, and so on.
- Screen 4 (press Alt+F4) is used to show output from the modules during the linuxrc process.
- Screens 1 (press Alt+F1), 3, and 4 can be used to help determine which part of the process is failing or causing a problem.
- Screens 3 and 4 indicate which drivers are loaded.
- If a drive is loaded properly but fails in some way, view screen 4 to see if there is an outdated driver.

If the boot process fails, the first command line parameter to use is acpi=off.

E.8.2 Troubleshooting at the Bash Prompt

When the bash prompt is displayed, there are a few tools that you can use to gather information about the hardware:

- **hwinfo:** This utility is used by linuxrc to load hardware. You can use `hwinfo -pci` to determine exactly what hardware was recognized.

  Pipe to "less," because `hwinfo` can create a lot of output. For example, `hwinfo -pci | less`.

  If you need to contact Novell Support for help, you should capture the output from `hwinfo -pci` to a file for their use. You can gather the most information with this command:
  `hwinfo -pci -log /logfilename`

  where *logfilename* is the name of the file that you should send.

  You can then mount a device, such as a Thumb drive or other USB device, and save the output file for later use. You might also be able to use FTP to save the file where it can be available.

- **ethtool:** This is a valuable tool (contained in a ZENworks distribution) that can be used to change the configuration on most Ethernet network devices.

Upgrading the Dell DTK

<div style="text-align: right; font-size: 3em;">F</div>

Novell® ZENworks® 7 Linux Management - Dell Edition ships with the latest available version of the Dell OpenManage Deployment Toolkit (DTK). The DTK is integrated with ZENworks. However, when new versions of the DTK become available, you can upgrade to a newer version.

To obtain the current Dell DTK build from Dell:

1 Make sure the novell-proxydhcp daemon is running on a server in your network.

This service must be available so that the device's PXE can access files from the ZENworks server, such as the Preboot Services Menu file and the Dell DTK (Maintenance Mode) files.

2 Boot a Dell device that is PXE-enabled and press the Ctrl-Alt keys during booting.

This boots the device into the Preboot Services Menu. These keys must be pressed by the time a string starting with "Novell ..." is displayed during the boot process.

3 Select the Dell DTK (Maintenance Mode) option.

4 When the bash (#) prompt is displayed, view the content of the `/BUILD` file.

This shows the version of the Dell DTK currently used by ZENworks.

5 Make a note of the build version of your current DTK.

6 Obtain the latest Linux-based Dell DTK from `www.dell.com`.

This is normally available as an ISO file.

7 Create a CD using the downloaded ISO file.

8 Insert the newly created CD into the drive of any Linux workstation or server.

9 To update the Linux resource files:

  9a Using a file manager or a terminal on the server or workstation, copy the `SA.1` and `SA.2` files from the `/isolinux` directory on the CD to the `/srv/tftp/dell-dtk` directory on your ZENworks server.

    This replaces files of the same name.

    You might need drives mapped to your servers from the device where you inserted the CD. Otherwise, you need to place the CD into the drive of each device in order to copy the files locally to replace them.

  9b Repeat Step 9a for each ZENworks server to replace these files on them.

10 If necessary, to modify the configuration files:

  10a Mount the Dell DTK CD's ISO file, then open `/isolinux/isolinux.cfg` using any editor, text-viewing command, or utility.

  10b Look for the `ramdisk_size` parameter and note its assigned value.

    This parameter occurs many times in the file, but each instance is generally set to the same value. If different values are assigned in different instances of the parameter, use the value displayed in the default configuration section. For example, the file might contain `default 1` on a line, indicating the default section's name is "1," which is displayed immediately after "label" on a line, such as `label 1` under which the `ramdisk_size` parameter is listed.

10c On the ZENworks server, open the `/srv/tftp/dell-dtk.cfg` and `/srv/tftp/dell-dtk_maint.cfg` files and modify all `ramdisk_size` parameter values to be the same as what you obtained in step .

11 To ensure that the Dell DTK has been properly updated on the ZENworks server, reboot the device into PXE and select the Dell DTK (Maintenance Mode) option.

12 To verify that the build number has increased from the number you noted in , when the bash (#) prompt is displayed, view the content of the `/BUILD` file:

If the build number has not incremented, either it really isn't an update, or the copy operations failed.

Supported Ethernet Cards

G

Novell® ZENworks® Linux Management provides the Ethernet card drivers contained in the Linux kernel (2.6) that ships with ZENworks 7.

To determine which Linux kernel you are using, enter `uname -r` at the bash prompt.

If your device or laptop computer uses a different card that is not supported, you must supply your own Ethernet driver.

Accessing IP Addresses for Devices Running Dual NICs

H

To automatically obtain an IP address for a device running dual NICs, you can modify the `/srv/tftp/z_auto.cfg` or the `/srv/tftp/z_maint.cfg` file (or both) by adding the following line at the end of the *Append* command:

```
netdevice=eth0
```

This eliminates the need to select a NIC's IP address manually.

Establishing SSH Tunneling

If you are using Remote Management over a network that is not secure, the data between the Remote Management Viewer running on the management console and the Remote Management Agent on the managed device is unencrypted and could be viewed by someone with access to the intervening network. You should tunnel your Remote Management sessions through a secure channel such as SSH.

I.1 SSH Tunneling between a Linux Management Console and a Linux Managed Device

If you are using Linux, SSH clients and servers are freely available on the internet. The SSH client and server RPMs can be downloaded from the OpenSSH site. (http://www.openssh.com).

I.1.1 Basic Use

SSH provides you with a "Secure SHell" to the remote device. All traffic is encrypted between the two devices using public key encryption techniques, making it very difficult for anyone else to spy on it. When SSH is installed, you could connect to a managed device from elsewhere simply by running the SSH client. For example, if you want to connect to a managed device called "work." you use the following command:

```
ssh work
```

You are then prompted for the password of your account on the managed device and you are logged in, just like a telnet session, but safer. You can also request that it listens on a particular port on your local management console and forwards that down the secure connection to a port on a managed device at the other end. To do this, use the following command:

```
ssh -L x:work:y work
```

This starts an SSH connection to a device named "work" and also listen on port x on the local management console, and forwards any connections there to port y on "work."

Remote Management uses two ports on the managed device. By default, the Remote Control service listens on port 5950 and the Remote Login service listens on port 5951. If you want to enable SSH tunneling for Remote Control, you need to forward Remote Management data from a port on your local management console to 5950 of managed device.

Similarly, you should forward data to 5951 if you want to tunnel Remote Login:

- If you are running Remote Control service on "work" on 5950 and you want a secure connection to it from your local management console, you can start the SSH session using:
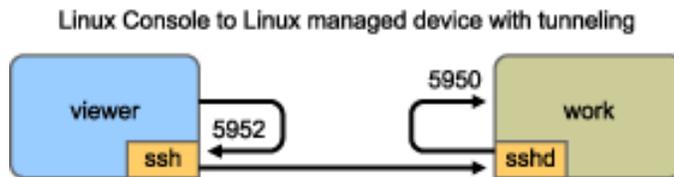
```
ssh -L 5952:work:5950 work
```

- Any connections to port 5952 on your local management console would actually connect to 5950 on "work," so instead of running a vnc viewer as:

```
vncviewer work:50
```

run it as follows:

```
vncviewer localhost:52
```

Figure I-1   *Linux Console to Linux Managed Device with Tunneling*

Linux Console to Linux managed device with tunneling

NOTE: If you are using the Linux VNC viewer to connect via SSH, when the viewer connects to a server on the local management console, by default it uses VNC's pixel encoding because this generally gives better performance for local access. If this server is actually an SSHD redirecting the data for another workstation, you can override this using the -tight option to the viewer or you can send a lot more data over the network.

I.2 SSH Tunneling between a Windows Management Console and a Linux Managed Device

SSH clients are also available for Windows, Macintosh, and other platforms, but if you want servers on these platforms you might need to use a commercial version or route your connection via a Linux device.

There are several scenarios for using SSH tunneling between a Windows management console and a Linux managed device. For the sake of simplicity, the following procedure uses a scenario in which you are using a Windows laptop "viewer" in a non-secure Wide Area Network to remotely control your Linux managed device "work" installed inside your secure Local Area Network. Another Linux device called "gateway" is in your secure local area network and runs the SSH daemon. The following steps explain how you can use the PuTTY SSH client to configure an SSH tunnel so that the Remote Management data is encrypted when it travels between "viewer" and "gateway" and is then forwarded to "work" inside the secure network.

NOTE: The PuTTY SSH client is available at the PuTTY site (http://www.chiark.greenend.org.uk/~sgtatham/putty). If you are using other SSH client software, use the appropriate commands for that software.

1 Enter the following command in the shell prompt:

```
putty -L 5952:work:5950 gateway
```

The first argument is the local forwarding option, which says that the local fake port 5952 should be created and connected to the genuine port work:5950. The second argument is the
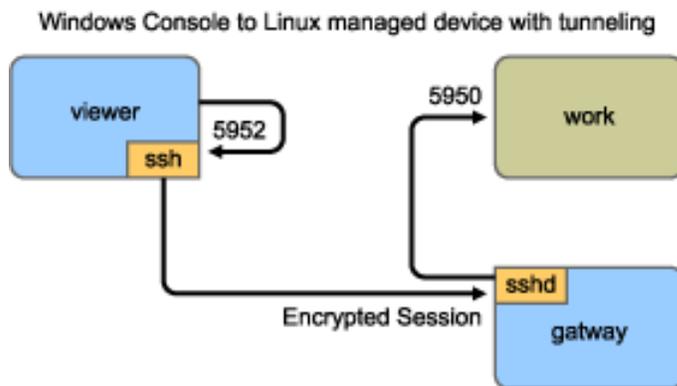
main non-option parameter to SSH, which tells it to connect to the device that runs the SSH daemon.

2 In the PuTTY Security Alert dialog box, verify that the key matches with that of the "gateway" device, then click *Yes*.

3 To establish the SSH tunnel between "viewer" and "gateway," you need to require authentication to "gateway." Specify a valid username and password of the "gateway" device in the PuTTY dialog box, then click *Yes*.

4 Any connections to port 5952 on your local management console would actually connect to 5950 on "work," so instead of running a vnc viewer as

```
vncviewer work:50
```

run it as follows

```
vncviewer localhost:52
```

Windows Console to Linux managed device with tunneling

NOTE: If you are using the Linux VNC viewer to connect via SSH, when the viewer connects to a server on the local management console, by default it uses VNC's pixel encoding because this generally gives better performance for local access. If this server is actually an SSHD redirecting the data for another workstation, you can override this using the -tight option to the viewer or you can send more data over the network.

I.3 Compression

SSH can also compress the data. This is particularly useful if the link between your management console and the managed device is a slow one, such as a modem, but even on a faster network it can be helpful, because encryption takes a certain amount of time and can slow the link down a little. To add simple compression, use the -C option. For more control, set it up in your SSH configuration files. To see how much your data is being compressed, use the -v option.

Limited Support for SLES10 and SLED10 Managed Devices

<div style="text-align: right">**J**</div>

ZENworks® Linux Management - Dell Edition offers limited support for SUSE® Linux Enterprise Server 10 (SLES10) and SUSE Linux Enterprise Desktop (SLED10) managed devices.

Using ZENworks Linux Management - Dell Edition, you can use the zlmmirror utility to mirror updates to your ZENworks Linux server and then use catalogs to distribute the updates to your SLES10 and SLED10 managed devices. At this time, you cannot use ZENworks Linux Management to apply policies, distribute bundles, apply images, inventory managed devices, remotely control devices, and so forth. These capabilities will be added in an upcoming release.

For detailed information and step-by-step information about obtaining and distributing updates to SLES10 and SLED10 managed devices, see the following Technical Information Documents (TIDs):

- TID 3480922 in the Novell Support Knowledgebase (http://www.novell.com/support/supportcentral/supportcentral.do?id=m1).
- TID 3860200 in the Novell Support Knowledgebase (http://www.novell.com/support/supportcentral/supportcentral.do?id=m1).
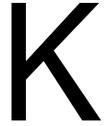
NOTE: Ensure that you select the Search by TID ID box before clicking Search.

For more information about using the zlmmirror utility, see Chapter 22, "Mirroring Software," on page 253. For more information about using catalogs, see Chapter 19, "Using Catalogs," on page 229.

License Agreement for libacl and libgconf

<div style="text-align: right; font-size: 3em;">K</div>

The following is the license agreement for the libacl and libgconf library that is used in the ZENworks® 7 Linux Management Policy Handler/Enforcer software:

K.1 Library GNU Public License

This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you."

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification.")

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

- You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

  You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

- You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

  - The modified work must itself be a software library.

  - You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

- ◆ You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

- ◆ If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

  (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

  These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

  Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

  In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

- ◆ You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

  Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

  This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

- ◆ You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

  If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

- ◆ A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library." Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library." The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

- As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

  - Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library," as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

  - Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

  - Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

  - If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

  - Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

  - For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a

special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

- It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

- You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

  - Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

  - Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

- You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

- You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

- Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

- If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

  If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

  It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting

the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

◆ If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

◆ The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version," you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

◆ If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

◆ NO WARRANTY

◆ BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

◆ IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Documentation Updates

L
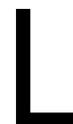
This section contains information on documentation content changes that have been made in the *Administration Guide* after the initial release of Novell® ZENworks® 7 Linux Management - Dell Edition. The information will help you to keep current on updates to the documentation.

All changes that are noted in this section were also made in the documentation. The documentation is provided on the Web in two formats: HTML and PDF. The HTML and PDF documentation are both kept up-to-date with the documentation changes listed in this section.

The documentation update information is grouped according to the date the changes were published. Within a dated section, the changes are alphabetically listed by the names of the main table of contents sections for ZENworks 7 Linux Management - Dell Edition.

If you need to know whether a copy of the PDF documentation you are using is the most recent, the PDF document contains the date it was published on the front title page or in the Legal Notices section immediately following the title page.

The documentation was updated on the following dates:

- Section L.1, "January 17, 2007," on page 551
- Section L.2, "September 29, 2006," on page 551
- Section L.3, "June 29, 2006," on page 552

L.1 January 17, 2007

Updates were made to the following sections:

Location	Change
Section 5.3.2, "Uninstalling a Secondary ZENworks Server Using zlm-config," on page 47	Added entire section.

L.2 September 29, 2006

Updates were made to the following sections:

Location	Change
Appendix J, "Limited Support for SLES10 and SLED10 Managed Devices," on page 543	Added information describing limited support for SUSE Linux Enterprise Server 10 (SLES10) and SUSE Linux Enterprise Desktop 10 (SLED10) managed devices.

L.3 June 29, 2006

Updates were made to the following sections:

Location	Change
Section 22.5, "Deploying Red Hat Network Updates," on page 262	Replaced the text in this section with newer information.