

ZENworks 11 SP3

Test Scenarios for *Imaging*

This document contains test scenarios for ZENworks 11 SP3 Beta.

Purpose of the Test Scenarios

The purpose of these exercises is to help you get familiar with some of the new features added in the *Imaging* component of ZENworks 11 SP3.

Assumptions

- You have followed the instructions for installing ZENworks 11 SP3 by using the *ZENworks 11 SP3 guide* (<http://www.novell.com/documentation/zenworks113>).

Test Scenarios

1. [Downloading and Uploading the Tuxera Driver through ZENworks Control Center](#)
2. [Verifying the Tuxera Driver Status and File Changes After Upload](#)
3. [Creating ZENworks Partition on a UEFI Device](#)
4. [Creating Partitions on a UEFI Device through the Imaging Engine GUI](#)
5. [Creating Partitions on a UEFI Device through the Imaging Engine Command Line](#)
6. [Manually Taking and Restoring an Image of a UEFI Device](#)
7. [Taking and Restoring an Image of a UEFI Device Automatically](#)
8. [Using Third-Party Imaging for Windows 8 UEFI Devices](#)
9. [Writing Scripts that Can be Executed on a Third-Party Distro](#)
10. [Bootting the Third-Party Distro Based on CPU Architecture](#)

Test Scenario #1: Downloading and Uploading the Tuxera Driver through ZENworks Control Center

Objective

This scenario will enable you to download the Tuxera driver zip file and then upload the same through ZENworks Control Center.

Procedure

1. Log in to ZENworks Control Center.
2. Click *Configuration > Device Management > Preboot Services > Tuxera High Performance NTFS Driver Integration Settings*.
3. Download the latest high performance NTFS driver zip file from the hyperlink provided and save it on your local system.
4. Upload the driver to the server by browsing through the file system.
5. Click *Apply*.
6. Click *Ok* to exit the Preboot Services page.
7. After a few minutes, in the Preboot Services page, click *Status* to check the upload status.

Expected Results

- The Tuxera driver is downloaded successfully.
- The Tuxera driver is uploaded to the server from the local system through the ZENworks Control Center.
- The Tuxera driver is replicated to all Secondary Servers and Imaging Satellite servers.

Logs

If you are unable to successfully perform the procedure, send us the following files:

- *loader-messages.log*
- *zcc.log*

Test Scenario #2: Verifying the Tuxera Driver Status and File Changes After Upload

Objective

This scenario will enable you to verify the status of the Tuxera driver and the file changes after a successful upload.

Procedure

1. Log in to ZENworks Control Center.
2. Click *Configuration > Device Management > Preboot Services > Tuxera High Performance NTFS Driver Integration Settings*.
3. Click *Status* to verify the upload status. If the status is Available, check for the expected results.

Expected Results

- The tftp folder on the server contains the *HighPerfDriver.conf* file.
- Under the *tftp/11.3.0* folder, two folders called *x86* and *x86_64* are created. These contain 32-bit and 64-bit *tntfs.ko* files respectively.
- The new *bootcd_tntfs.iso* file is created on the server.

Logs

If you are unable to successfully perform the procedure, send us the following file:

- *loader-messages.log*

Test Scenario #3: Creating ZENworks Partition on a UEFI Device

Objective

This scenario will enable you to create a ZENworks partition on a UEFI device.

Procedure

1. Boot a UEFI device with the *bootcd.iso* or *bootcd_tntfs.iso* file.
2. From the menu, select *Install or Update ZEN partition*.

Expected Results

A ZENworks partition is created and all existing data on the device is erased.

Test Scenario #4: Creating Partitions on a UEFI Device through the Imaging Engine GUI

Objective

This scenario will enable you to create different types of partitions through the Imaging Engine GUI on a UEFI device.

Procedure

1. Boot a UEFI device into the Imaging distro either through the Preboot Execution Environment (PXE), or by using the boot cd, or ZEN partition.
2. Run the `img` command to launch the Imaging Engine.
3. Press the F8 key to launch the Partition wizard.
4. To create a new partition, select an empty slot on the disk, then click *Add*.
5. In the Create New Partition wizard, select the *Partition Type* and provide the size of the partition.
6. On an empty hard disk, you can also select the *Partition scheme* as *MBR* or *GPT*. If the disk has pre-existing partitions, the same scheme will be used for the new partition.
 - If you select MBR, a partition with the required type and size is created in the selected slot.
 - If you select GPT, the next screen provides an option to select the Partition Type GUID.

You can create the following GUID types:

- MBD – Microsoft Basic Data partition
- ESP – EFI System Partition
- WRE – Windows Recovery Environment
- MRP – Microsoft Reserved Partition.

Expected Results

The new partition is created in the selected empty slot on the disk.

Logs

If you are unable to successfully perform the procedure, send us following file:

- *imglog*

Also provide details about the type of firmware interface of the device (UEFI or BIOS) and the type of hard disk (Basic, Advanced Format, or Solid State drives).

Test Scenario #5: Creating Partitions on a UEFI Device through the Imaging Engine Command Line

Objective

This scenario will enable you to create different kinds of partitions through the Imaging Engine command line on a UEFI device.

Procedure

1. Boot a UEFI device into the Imaging distro either through PXE, or by using the boot cd, or the ZEN partition.
2. On the hash prompt, type the following command to create partitions:

```
img -pc <pnumber> -type=<ptype> -guid=<guidtype>  
-size=<size>
```

pnumber: Partition number of a free slot that is available in the disk.

ptype: Partition type that can contain values such as FAT32, NTFS.

guidtype: GUID type that can contain values such as MBD, ESP, WRE and MRP.

Note: If this parameter is ignored, the Imaging engine assumes that an MBR partition is being created.

size: Size of the partition to be created either in MB or as a percentage of the entire disk.

Logs

If you are unable to successfully perform the procedure, send us the following file:

- *imglog*

Also provide details about the type of firmware interface of the device (UEFI or BIOS) and the type of hard disk (Basic, Advanced Format, or Solid State drives).

Test Scenario #6: Manually Taking and Restoring an Image of a UEFI Device

Objective

This scenario will enable you to manually take and restore an image of a UEFI device.

Procedure

To take an image of a system manually:

1. On the device whose image needs to be taken run the `chkdsk` command and reboot the device into the operating system.
2. Boot the system into the ZENworks Maintenance mode either through PXE, or by using the `bootcd_tntfs.iso` file, `bootcd.iso` file, or the ZEN partition.
3. Run the `export DEVELOPER_LOG=a` command, then run the `img` command.
4. Press the F5 key to take an image.
5. On the Make Image page, select *Server*, then click *Next*.
6. Provide the file name of the image to be archived.

To restore an image:

1. Boot the system into the ZENworks Maintenance mode either through PXE, or by using the `bootcd_tntfs.iso` file, `bootcd.iso` file, or by using the ZEN partition.
2. Run the `export DEVELOPER_LOG=a` command, then run the `img` command.
3. Press the F6 key to restore the image.
4. On the Restore Image page, select *Server*, then click *Next*.
5. Provide the file name of the archived image.
6. Click *Next* to start the restoration process.

Expected Results

- The image is restored successfully.
- The ISD is updated.
- After reboot, the device boots into the operating system.

Logs

If you are unable to successfully perform the procedure, send us the following files and the output of the `dmesg` command:

- *sidChange.log*
- *imglog*
- *novell-pbserv.log*

Test Scenario #7: Taking and Restoring an Image of a UEFI Device Automatically

Objective

This scenario will enable you to take and restore an image of a UEFI device automatically.

Procedure

To take an image of a system automatically:

1. On the machine whose image needs to be taken run the `chkdsk` command and reboot the device into the operating system.
2. Install ZENworks agent on the device and reboot to register the device to the ZENworks zone.
3. In the command prompt, run the `zac fsg -d` command.
4. In ZENworks Control Center, navigate to the *Devices* page. Browse under the *Servers* or *Workstation* folder to locate the device.
5. Select the device, click *Action*, then select *Take an Image* from the drop-down list.
6. In the wizard that appears, provide the Type of image to be taken, the Server path and the Image name with proper extension.
7. Click *Next*, then click *Finish* to close the wizard.
8. Reboot the agent through PXE.

To restore an image:

1. In ZENworks Control Center, create a preboot bundle of the image that needs to be restored by navigating to *Bundles > New > Bundle > Preboot bundle > <image type>*.
2. Provide a name for the bundle.
3. Browse the server and select the existing image from the server.
4. Click *Next*, then *Finish* to complete the bundle creation.
5. Navigate to *Configuration > Device Management > Preboot Services > Device Imaging Work Assignment*.
6. Create a hardware rule by specifying the bundle to be restored, hardware parameter, and a value that matches the device on which you need to restore the image. For example, MAC address is equal to 00:0c:29:xx:xx:xx.
7. Click *Apply*, then click *Ok*.

8. Boot the system on which the image needs to be restored through PXE.

Expected Results

- The Take Image operation is successful and the image is stored on the server.
- The image is restored successfully.
- The ISD is updated.
- After reboot, the device boots into the operating system.

Logs

If you are unable to successfully perform the procedure, send us the following files:

- *novell-pbserv.log*
- *imglog*
- *sidChanger.log*

Test Scenario #8: Using Third-Party Imaging for Windows 8 UEFI Devices

Objective

This scenario will enable you to use third-party Imaging on Windows 8 UEFI devices.

Procedure

To enable third-party imaging on Windows 8 UEFI devices, you need to upload the 64-bit *winpe.wim* and *imagex.exe* files into the ZENworks server. To upload these files, Windows Assessment and Deployment Kit (WADK) must be installed on the device from where you access the ZENworks Control Center.

To upload the files:

1. Log in to ZENworks Control Center.
2. Click *Configuration > Device Management > Preboot Services > Third Party Imaging Settings > 64 bit Upload Settings*.
3. In *Upload WinPE Base Distribution*, click the Browse icon.
4. Browse through the file system and select the appropriate *winpe.wim* file.
5. In *Upload ImageX files to support WIM imaging*, click the Browse icon.
6. Browse through the file system and select the appropriate *imagex.exe* file.
7. Click *Apply*, then click *Ok*.
8. You can start the third-party imaging process when the status of the *wim* and *imagex* files is Available.

Expected Results

Files are successfully uploaded to enable third-party imaging of UEFI devices.

Logs

If you are unable to successfully perform the procedure, send us the following files:

- *novell-loader-messages.log*
- *zcc.log*

Test Scenario #9: Writing Scripts that Can be Executed on a Third-Party Distro

Objective

This scenario will enable you to write scripts that can be executed on a third-party distro.

Procedure

The following five types of scripts that can be executed on a third-party distro are supported by ZENworks:

- Batch script
- JScript
- VB script
- Windows script file
- HTML application

To create a Third-party Scripted Imaging bundle:

1. Log in to ZENworks Control Center.
2. Click *Bundles > New > Bundle > Preboot Bundle > Third party script*.
3. Provide a name for the bundle.
4. Select the type of script from the drop-down list and write the required script in the text box provided.
5. Click *Next*, then click *Finish* to complete the bundle creation.

Expected Results

- The bundle is created successfully.
- The bundle is applied to any device and executed on it.
- The bundle is manually executed by booting a device into the third-party distro through PXE.
- The script gets downloaded on the device with the file name as given:

`WinScript <file number>.<ext>`

Note: The extension of the file name depends on the type of script selected.

Logs

If you are unable to successfully perform the procedure, send us the following files:

- *worktododebug.txt*
- *novell-loader-messages.log*
- *zcc.log*

Test Scenario #10: Booting the Third-Party Distro Based on CPU Architecture

Objective

This scenario will enable you to boot the third-party distro on device based on its CPU architecture.

Procedure

ZENworks supports the booting of a third-party distro on a device based on its CPU architecture. To enable this, you need to upload the 32-bit `winpe.wim` either from Windows Automated Installation Kit (WAIK) or from Windows Assessment and Deployment Kit (WADK) and the 64-bit `winpe.wim` from Windows Assessment and Deployment Kit (WADK).

To upload the 32-bit files, WAIK or WADK must be installed on the device from where you access ZENworks Control Center. To upload 64-bit files, WADK must be installed on the device from where you access ZENworks Control Center.

To upload the file:

1. Log in to ZENworks Control Center.
2. Click *Configuration > Device Management > Preboot Services > Third Party Imaging Settings > 64 bit Upload Settings*.
3. In *Upload WinPE Base Distribution*, click the Browse icon.
4. Browse through the file system and select the appropriate `winpe.wim` file.
5. Click *Apply*, then click *Ok*.

Follow a similar procedure for a 32-bit `winpe.wim` upload.

Expected Results

When the status of the files is Available in the ZENworks Control Center, booting up a device into the third-party distro loads the corresponding distro.

Logs

If you are unable to successfully perform the procedure, send us the following files:

- `novell-loader-messages.log`
- `zcc.log`