

Network Image Deployment

V0.5.7

Documentation for Database, Server and Client Setup

© 2016 David Maiolo

Contents

Deployment Basics _____	1
Generating Deployment DVD _____	4
Preparing an Image For Capture _____	5
Capturing (Uploading) an Image _____	6
Verifying or Modifying Capture _____	8
Deploying an Image _____	9
Image Notes _____	11

“Deployment is based on ImageX, a command-line tool that enables corporations to capture, to modify, and to apply file-based disk images for rapid deployment.”

NID Technology Basics

Deployment Technology

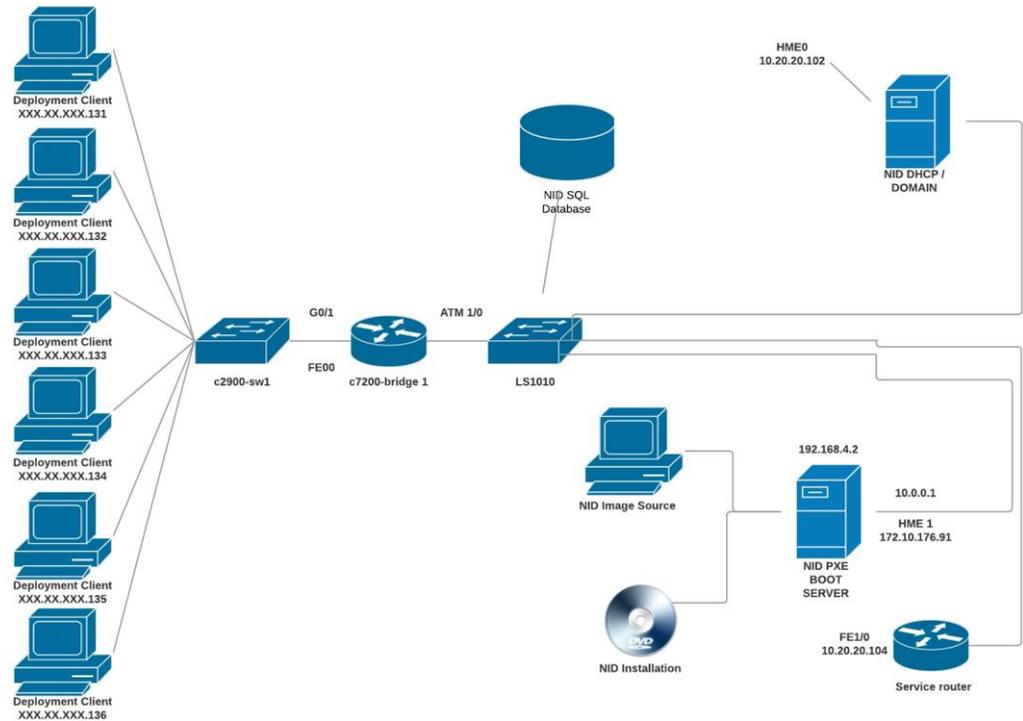
The NID system was designed and programmed by David Maiolo with Microsoft and third party deployment technologies used as the compatibility model of the system.

This Network Image Deployment (NID) System is designed to capture, maintain and deploy computer images for a corporate or educational environment with up to 15,000 computers. Offline image mounting and updating with hardware independence natively supported allows one central image for each major departmental structure, reducing the costs associated with traditional imaging platforms.

Deployment is based on ImageX, a command-line tool that enables corporations to capture, to modify, and to apply file-based disk images for rapid deployment. ImageX works with Windows image (.wim) files for copying to a network, or it can work with other technologies that use .wim images, such as Windows Setup, Windows Deployment Services (Windows DS), and the System Management Server (SMS) Operating System Feature Deployment Pack.

CISCO ISG DIAGRAM

Network Image Deployment (NID) 0.5.7 by David Maiolo



Method

Images are captured and deployed using custom scripting tailored for a corporate or educational environment for image deployment. The deployment system can be configured to operate over PXE. However, direct DVD boot is also available for smaller systems and for the quickest deployment strategy.

Support Technology

The code for the deployment system itself is contained within a self-booting ISO, which operates in a Windows PE environment. The hardware used to deploy the system images can be completely self-contained with one server acting as the image store or can be scaled to be contained on two primary servers:

- NID DHCP / DOMAIN
- NID IMAGE STORE / PXE BOOT SERVER
- NIS SQL SERVER (Optional)

Looking Ahead

The image format, the .WIM, is compatible with Microsoft's foreseeable deployment structure. This format can be integrated into a large scale WDS style system, or kept in a standalone deployment as used currently.

Generating Deployment DVD

The scripts used to generate the deployment environment are designed to generate a bootable ISO. The generation of the ISO is based off mounting techniques and scripts that are editable before the ISO is compiled. The generation scripts and associated files are contained at:

/CreateDeploymentDVD

Customize .ISO

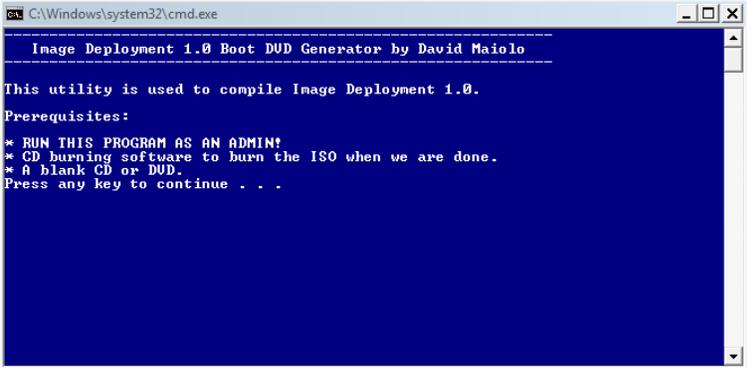
The various scripts and tools can be customized before the ISO is generated. Most notably, you will find the server drive mapping information and passwords contained in:

- *./Support_Files\Scriptsandtools\Support_Scripts/2_capture.cmd*
- *./Support_Files\Scriptsandtools\Support_Scripts/3_deploy.cmd*

Compile a DVD .ISO

To compile the very first bootable DVD, you must be working from a Windows workstation environment. To begin the compilation process, assure you are a local administrator on the machine and have sufficient rights your intended deployment server. Then, execute

CreateDisc.cmd



```
C:\Windows\system32\cmd.exe
Image Deployment 1.0 Boot DVD Generator by David Maiolo
-----
This utility is used to compile Image Deployment 1.0.
Prerequisites:
* RUN THIS PROGRAM AS AN ADMIN!
* CD burning software to burn the ISO when we are done.
* A blank CD or DVD.
Press any key to continue . . .
```

Figure 1: DVD Boot Generator

Follow the prompts to generate the disc. Pay special note to the Unique ID that is assigned to this disc, as you will need to enter to continue compiling the ISO. Once completed, the ISO will be placed on your desktop. Use a sufficient application to burn to a blank DVD or CD.

Preparing an Image for Capture

Preparing a computer to be captured (or uploaded) is a vital step for proper deployment within the system. This is the process of “*syspreping*” the image. Utilize the deployment DVD that you generated and launch it within Windows from the computer intended to be captured. Note the DVD has two separate partitions. The first available to a Windows machine for the preparation tools, and the second as the bootable environment itself, with a separate set of capture and deployment tools. We will be utilizing the first partition. The preparation menu is intended to load as within Autorun.ini. In other words, it should launch automatically.

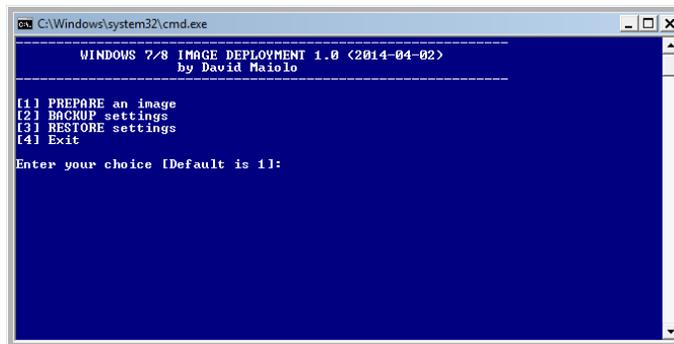


Figure 2: Prepare an image for capture (upload)

This tool will sysprep the image, utilizing the unattend.xml file that is contained on the deployment DVD. Follow the steps to sysprep and prepare the image. This process will remove machine specific settings to prepare for a generic deployment, such as network settings, drivers, unique identifiers (SID's), etc. Note that group policy objects assigned to specific local users will break as these users will be given unique identifiers.

Eventually the computer will shut down. It is very important you do not turn this machine back on under any circumstances without intending to run the Capture scripts as the next process, which will be outlined next.



Capturing (Uploading) an Image

Once the image has been prepared, it now needs to be uploaded to one of the two available servers you assigned during setup. Currently, the .ISO has been compiled to map the following two servers to place the image.

- Primary
- Backup

Booting to Capture Environment

After the system has been prepared, and the machine is off, you will utilize the same disc to boot into the deployment or capture environment. Turn on the machine and assure the DVD is booted. If the hard drive boots before the DVD, the entire preparation process must be repeated. Usually on Dell systems, hitting F12 will allow you the option to choose your boot device.

Capturing an Image

The deployment system will launch under a customized Windows 10 based PE environment.

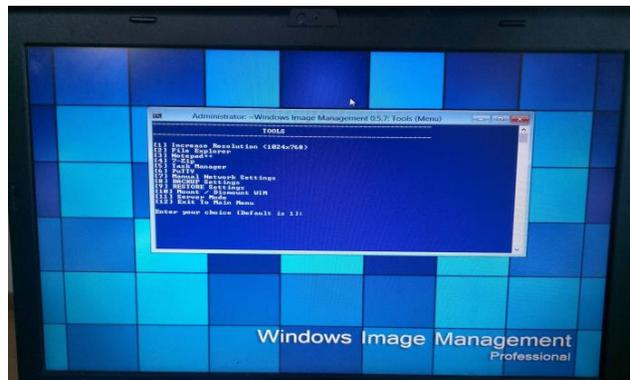


Figure 3: Capturing an Image

Choose option [2] **CAPTURE an Image** to begin the capture process. Because we need write access to the servers noted above, it is required that you enter in the relevant passwords as these shares are mapped. These passwords will be contained within a second, more secure document.

During the capture process, you will be asked which server to upload to. You will also choose your capture drive. This is typically your C: drive under windows. It is very important to note

Capturing an Image.

It is very important that the preparation system is not booted prior to capture.

that this drive may have been mounted as the D: drive under this deployment system. Make certain you capture the correct drive when prompted. You may need to change to a different drive. A sample directly listing of the drive about to be captured will be displayed before you continue. Do not worry if the drive is not mounted as C:, etc. as the ImageX process is file based, and drive structure will not be stored in the image (but file structure will).

Verifying or Modifying Capture

Prior to deploying an image, you may want to verify or modify your captured image. .WIM images are stored much like a zip file and using special tools, can be modified from a separate system without actually deploying the image. This is useful for smaller updates within the image, such as file deletion, etc. Note you will ONLY have access to the file system, and will not be able to modify things without a thorough understanding of Windows directory structure. If you wish to make larger changes to the image that require you to be using the image, it is recommended you deploy the image to a machine (next step), make your changes, then prepare the image again and capture it again. This would be similar to the Symantec Ghost type of image modification prior to technologies such as .WIM.

Mounting an Image

Although there are tools provided into the environment to mount and unmounts images automatically, it is preferred you understand and run these tools directly to have a thorough understanding of what you are doing. From a SEPARATE Windows 7 workstation (not logged into the workstation to be modified), run the following command:

```
Dism /Mount-Image /ImageFile:\\Server\Share\imagename.wim /index:1  
/MountDir:C:\mount
```

Where \\Server\Share is the path to the image, imagename is the Image Name and C:\mount is your desired location on your current workstation to work on the image's files.

Unmounting an Image

Once you have made your desired changes, you can unmount the image using the following command:

```
Dism /Commit-Image /MountDir:C:\Mount
```

Again, where C:\Mount is the desired location you chose on your current workstation to work on the image's files.

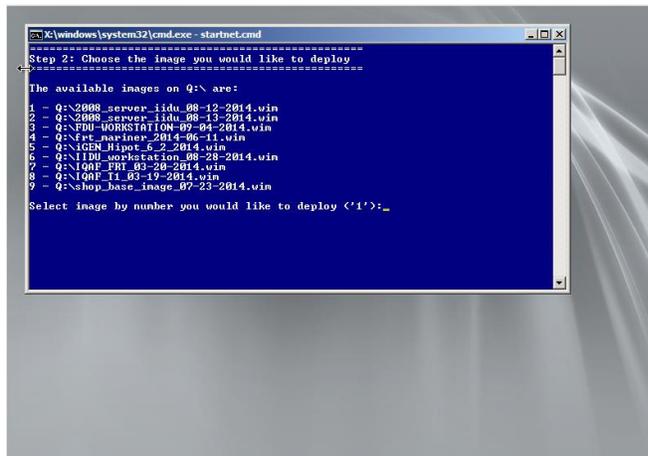


Figure 5: Choosing an Image For Deployment

Once an image has been chosen, the drive will be formatted and partitioned according to the settings contained within the **3_Deploy.cmd** script, which generates an on-the-fly diskpart script. This file can be modified as a onetime occurrence with the soft X: file system or prior to compiling the Deployment DVD.

Once the image has been deployed, the deployment will automatically attempt to generate the necessary boot files and partition utilizing this line of code:

```
c:\Windows\System32\bcdboot c:\Windows /l en-US
```

If for any reason you get a NTLDR error, you will want to investigate that this line is applicable to the type of environment you are deploying (Windows XP, 7, 8, etc.) as they all require their own separate boot environments. Currently, the system is configured for Windows 7.

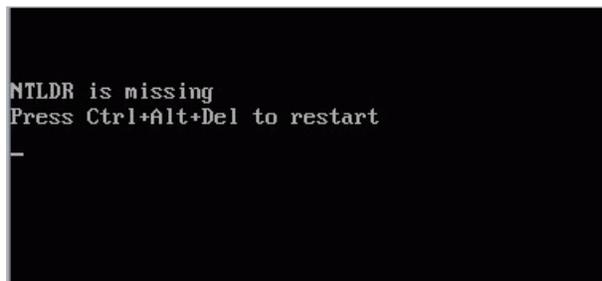


Figure 6: NTLDR errors are typically related to failed Boot Environments.

Image Notes

The image's that are created also contain their own scripting to perform various tasks. Most notably, you will see a series of scripts on the main image captured that are designed to setup the image for particular environment that it will be used in.

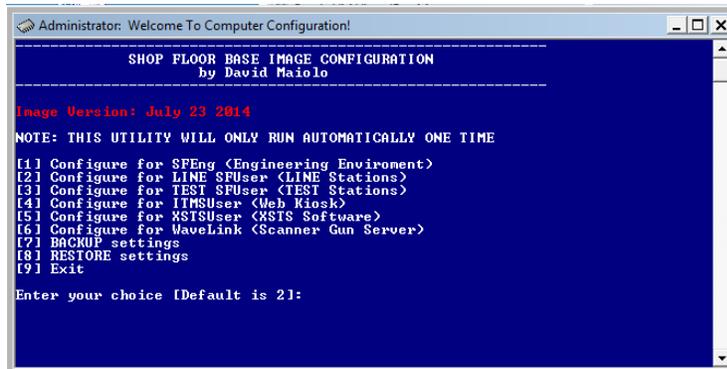


Figure 7: Image Configuration Scripts

You are welcome to utilize this environment to configure and modify the image, or go the route of customizing on your own. The most notable occurrences within these scripts are to regenerate the proper Group Policy ID's for the SFUser account now that Sysprep has broken the original link. Logic such as this is used to determine the new SFUser SID:

- ```
for /f "delims= " %%a in ("wmic path win32_useraccount where name='SFUser' get sid")
do (
 if not "%%a"=="SID" (
 set sfusersid=%%a
 goto :loop_end
)
)
```

Whichever method you implement, assure your Group Policy Objects retain the proper SID's before leaving a customer with a deployment. These GPO linked SIDs will be found within:

*C:\Windows\System32\GroupPolicyUsers*