

Mac OS X Panther Imaging Do's and Don'ts

Chris Wiesemann

University of Oregon
5249 University of Oregon
Eugene, OR 97403
1 (541) 346 2082

chris@uoregon.edu

ABSTRACT

In fall of 2003, our department decided to deploy OS X on our 100 or so Apple lab computers. Apple provided a free training session on its OS X Server Network Image Utility, and we jumped at the opportunity. A month of testing left us frustrated with the Network Image Utility and the NetInstall procedure; they simply did not work as advertised.

Apple's OS X Server imaging tools work fine in certain environments, or with post-flight modifications, but we found them to be unusable in our environment.

After consultation with Apple engineers and with colleagues from other schools, we settled on Mike Bombich's Carbon Copy Cloner and NetRestore applications in conjunction with Apple Software Restore and Mac OS X Server's NetBoot feature.

This paper and presentation discuss our experience with Apple's imaging tools as well as third-party imaging software. We detail the advantages, disadvantages, and recommended use of each program. We also discuss pairing the Apple Remote Desktop software with scripts and tools such as ARD Companion and NetRestore to create an easily managed and deployed OS X lab environment.

Categories and Subject Descriptors

K.3 Computers and Education

General Terms

Management, Documentation, Performance, Design, Reliability, Security, Human Factors, Standardization, Languages,

Keywords

NetRestore, NetRestoreHelper, Radmind, Blast Image Config, Casper, NetInstall, NetBoot, Carbon Copy Cloner, Apple Software Restore, Mac OS X, labs, imaging.

1.1 Introduction

Having Mac OS X in public computer labs makes everyone's lives so much easier. It's a magnificent step forward. Some people, however, have been leery of this step, holding on to their Classic (Mac OS 9) roots. With this paper, I hope to aid people in

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGUCCS'04, October 10-13, 2004, Baltimore, Maryland, USA.

Copyright 2004 ACM 1-58113-869-5/04/0010...\$5.00.

the process of installing, configuring, and deploying OS X in their computer labs.

1.2 Overview

There are many options for deployment and maintenance of OS X.

No one has yet written a good how-to guide for imaging Mac OS X 10.3 (Panther) using Apple's tools: Network Image Utility, NetBoot, and NetInstall. The obvious problem with Apple's system is how their tools create and copy disk images: their NetInstall system uses *discutil*, which does file copy duplication. Their Apple Software Restore (ASR) system, which is used by NetRestore, does block copy duplication. File-copy, as the name implies, merely copies file by file. This method is good in small situations, or in an environment where you do not need to preserve creation date, modification date, and permissions. Block copy copies sections of the file system. Block copying is faster and more exact than file copies.

The following is a list of some of the other tools that are currently available for deploying Mac OS X disk images and maintaining them once they are installed:

1.2.1 Radmind

Radmind is a suite of UNIX command line tools and a server to administer file systems on multiple UNIX computers. Written by people at the University of Michigan, it is offered under a BSD-style license. It is a great tool to use in maintaining and updating computers. There is a large learning curve with this software, but after it is set up, those who use it swear by it. Radmind works by taking a snapshot of the files on a computer. It then creates a loadset of all the necessary files. Your Radmind server keeps track of the changes to your loadset. If you change the loadset, when the other machines check in with the server, they automatically download these changes. This system allows easy deployment of applications, settings, or updates to all of your public computers.

More information on Radmind can be found at <http://rsug.itd.umich.edu/software/radmind/>

1.2.2 PSU Blast Image Config

PSU Blast Image Config is a solution to restore and configure Mac OS computers running OS X 10.2.2 or higher. It is written by Justin Elliott of Penn State and is offered as freeware. It is very similar to Mike Bombich's NetRestore system (see below), but offers a couple more features. With PSU BIC you can configure most computer settings. It allows you to set Date/Time, Network, Open Firmware, and utilize pre and post imaging scripts. It relies on the *ncutil* program (which is not installed on a normal OS X computer) to modify network settings. One perk of this program is

the inclusion of an elapsed time display. This software only allows you to deploy an image; you must create it beforehand with other tools. It also supports distribution over http.

More information on PSU Blast Image Config can be found at <http://clc.its.psu.edu/Labs/Mac/Resources/>

1.2.3 Casper

Casper is a commercial solution to deployment. Offered by JAMF software, it offers a high level of customization during installation. Casper allows you to create, modify and distribute images over the network. As far as I am aware, however, it does not support block copy modes yet, which make it a slower option.

More information on Casper can be found at <http://www.jamfsoftware.com/casper.html>

1.2.4 Apple System Imaging

Network Image Utility is Apple's attempt to offer automated disk imaging solutions. For those doing basic deployments, it is great. A basic deployment is a minimal installation of OS X. It comes with Mac OS X Server, supports packages (see apple.com for details on packages), and is well integrated with Server Admin. Even though it does not support block copy modes, I found it to be very convenient for doing fast network OS X installs unattended. As you may know, swapping CDs when doing an OS X install can be frustrating. The major flaw with NetInstall, as of the version distributed with Mac OS X Server 10.3.3, appears when you attempt to image third party fonts; they appear on the destination as 0KB. This lack of third party fonts caused the Adobe suite of applications, as well as Quark products, to be unable to function. Some people have found that network font folders have helped them to avoid this difficulty. If your computers use network logins, using network font folders appears to be painless. Apple offers documentation on System Imaging on their website.

http://www.apple.com/server/pdfs/System_Image_Admin.pdf

1.2.5 NetRestore

NetRestore and NetRestore Helper are a graphical front end to Apple's command line Apple System Restore (asr) tools. Mike Bombich provides what I consider the easiest solution to imaging. It works in conjunction with Apple's Netboot service to provide block copy (ASR) imaging. His website also offers other useful software, such as Carbon Copy Cloner (CCC) which some people use to create their images. Mike Bombich, however, advises against using CCC to create your images. There are also forums available on his website with the answers to almost every common imaging question. For those who don't know, Mike Bombich is an Apple System Engineer.

More information on NetRestore can be found at <http://www.bombich.com/software/netrestore.html>

1.2.6 Other Options

There are many other programs and methods to perform imaging, I tried to hit on the most common in the academic environment. However, I will give mention to them.

Back in 10.2, we handled our lab deployment with Carbon Copy Cloner (CCC). Using a single firewire hard drive, we were able to clone the perfect image onto the firewire drive, and then back to other machines. This option is useful in small environments, or places where networking is less than 100Mbit.

Apple Software Restore is the underlying technology to several of the above referenced tools. It is possible however to directly use it in at a command line (i.e. terminal window) to create or restore images.

NetBoot is also an option. If you have good networking (100Mbit or better), this may be preferred for managing multiple images in a lab. You can configure an unlimited number of images in Apple OS X Server. Net booting on older machines may be difficult. You should remember to have current firmware on all old machines. I was able to get everything with a G4 processor to Netboot. G3 based machines require the older version of Netboot, which you can enable, but is not enabled by default. Instructions for this can be found here: <http://docs.info.apple.com/article.html?artnum=107902>

After experimenting with PSU BIC, NetInstall, CCC, NetBoot and NetRestore, we decided to use a combination of CCC and NetRestore to deploy our disk images. We chose those tools because they worked well and were easy to work with. Since the initial deployment of our computer labs to OS X, we have switched to solely using NetRestore and NetRestore Helper.

1.3 Creating Images

1.3.1 Server Configuration

The heart of every good NetRestore system is a Mac OS X Server. What hardware you use is entirely up to you. I cannot however, stress enough the advantage of using an XServe. Those who feed their XServe with a gigabit Ethernet connection will also have an advantage. The imaging distribution process is highly network intensive. Using gigabit to distribute out will speed the process of downloading for your clients. However, gigabit alone will not speed you up if the hard drives are a limiting factor. The XServe has the ability to do raid, which will speed up access, and overall imaging speed. In our environment, we do not use gigabit or raid drives, and our imaging times are much slower than others. I compared our speeds to the same setup on our campus with gigabit feed and striping on the XServe, and the latter are consistently 20%-40% faster in deployment.

NetRestore uses the Apple NetBoot service. If you don't have this service enabled, you need to turn it on. You can do this in the Server Admin utility. NetBoot requires NFS or AFP, but not DHCP. If your Mac OS X Server has multiple drives, you may also want to tell the NetBoot service which drive your NetBoot images will be stored. Remember, this is not yet where we are putting the NetRestore images; only the master image will reside here. NetBoot also allows filtering, and if you so choose, you would change this in Server Admin. NetBoot filtering allows you to restrict access to your server, by ethernet address. This could allow you to specify exactly which machines can NetBoot, which if you are concerned about licensed software could be a helpful idea.

1.3.2 Creating a User and Group for Imaging

One generally accepted method to restrict access to your images is to create a special imaging user that has read-only access to the directory in which you store your images. You can do this in Workgroup Manager. Create the Image group, then click on the Groups tab, and then click on the New Record button to create a new group. I recommend naming the group imaging; click on the Save button to continue. Now create the imaging user. Choose the Users tab, and then use the New Record button to create a new

user. I recommend naming the user `imaging`. Choose a password that you won't mind giving out to your staff and student employees. Then click the Group tab for the user record and set the user's primary group to the `imaging` group.

1.3.3 Creating a Directory and Share for Imaging

In the Finder, create a folder named `Images` on the root of one of the server's hard drive. Should it be the same drive as the System files, or a different one, or does it matter, you ask. For the best imaging speeds and results I would keep it on your fastest drive, or a raid set if at all possible (as discussed above). In Workgroup Manager, click on the Sharing icon in the toolbar. Then click on the "All" tab and navigate to the folder you created for your disk image. Now, check on the "Share this item and its contents" box. Set the owner to the admin user you will use to upload disk images. Set the group to the `Imaging` group. Set the owner privileges to read/write. Set the group privileges to read-only. Set the Everyone privileges to none. Stop and restart the Apple File Protocol (AFP) service on the server and your directory will be ready.

Now that your server is set up and ready to handle NetRestore, download the software from bombich.com. I recommend putting NetRestore in the `Images` folder you created above. Now open NetRestore Helper. Choose Create NetInstall Set. This will allow you to set up the NetInstall, which will help you deploy your images later. This is a basic image that runs NetRestore. It will never be installed or used for anything but NetBooting into the NetRestore program.

Once the creation of your master image is done, NetRestore will prompt you to change the preferences. This is where you will set up configurations for your different images. If you are following along, you can't set up configurations at this time, because we haven't created any images yet. However, you can come back and add configurations later. To do this, go to the Edit Configurations option in the NetRestoreHelper menu. I warn you here, however, that the delete feature doesn't work right. You should remove the default entries first. With the above server setup, the image would be over AFP with user imaging, password image, and sharename images. Then type in the name of the ASR scanned file in source box. Configurations are not mandatory, and you can configure NetRestore's preferences every time you use it. Using configurations will enhance your ability to automate the imaging process.

1.3.4 Creating a Successful Image

Creating your image is a task unique to each need. What follows are generic hints and directions to this process.

A frequently asked question when making images is: do I need to create a separate image for each different model of computer? Under Windows 2000 and higher, the answer is yes (usually). For Mac OS X, the same image works on all hardware, at least for me (so far). I keep very few images, for simplicity.

I recommend keeping a basic image available, as shown below, in addition to any configurations. The nice thing is that since the release of Mac OS X 10.3, the same image will work on any Mac with a G3, G4, or G5 processor. You should always remember to create your images on the newest hardware. Preferably this computer would also have all the options (including Airport). You will probably end up creating more images to handle the situation where a lab has some specialized software. This is the case in our Music school, which has "class" images, which have expensive

sound editing software, and "lab" images, which only have simple software such as Microsoft Office.

Configure the perfect installation. When your master is ready, we need to create an image file to use with NetRestore. Use NetRestore Helper's Create Master Image tab to create an ASR ready image file. Name the image files appropriately. Below is my setup of three images; `basic.dmg`, `faculty.dmg`, and `lab_keyed_image.dmg`)

- `Basic.dmg`: This image contains the latest version of Mac OS X with all software updates installed and a few basic configuration tweaks that are common to all of my images.
- `Faculty.dmg`: This is `basic.dmg` with MS Office, e-mail software, printers, fonts, and Antivirus.
- `Lab_keyed_image.dmg`: The Full Monty. It contains two dozen software programs, lab printers, lots of configuration tweaks, file permission changes, automated cleanup, ARD Client, and is over 9GB compressed.

Copy the image files to the `images` shared folder on your server.

1.3.5 Restore User (Manually Initiated)

One trick of imaging is a hidden "restore" account that lab managers or teachers can use if the generic "student" account breaks. My lab computers are only using one restricted student user. Your situation may be different, but the idea behind the hidden "restore" account is the same.

To do this, create an admin level user named `restore` with a userid of less than 500. This will hide the user from the login screen. Create the perfect "student" account and make sure it is configured exactly as desired with aliases and such. Copy the `/Users/student` folder to `/Library/management/student/`, with the management folder owned by the `restore` user. Then configure the `restore` user to have a login hook (read about this on macosxlabs.org) that runs a script to replace the student user home folder. It erases the `/Users/student` folder and replaces it entirely with `/Library/management/student/`. The script then adjusts permissions and logs out. These scripts are also available on www.bombich.com. The other possibility, which is based on the same concept, is the Automated restore process, which I will explain in a little more detail below.

1.3.6 Restore User (Automatically Initiated)

This trick is commonly discussed and implemented, however I will credit Mike Bombich (who has a great explanation available online). The above process is very useful in a lab with low use and large staff maintenance abilities. If that doesn't suit your situation, you may want to have an automatic cleaning cycle. If this is the case, login hooks and a restore script work very well.

Two scripts are available on www.bombich.com for automated user restoration. `Refresh-default-homedir-savetmp.sh` is my recommendation, especially if your lab users occasionally leave important files on your computers. The script is configurable to allow you to set username, adminname, homedir location, and template location. Thus, you can store a perfect copy of your user's home directory (e.g. `/Users/student`) in a secure place (`/etc/Management/student`). The script will allow you to specify a place to save data. One option is `/tmp/`; this directory's contents are deleted upon the first reboot, however. I recommend saving backups in a place such as your Management folder. A simple cron task can be configured to delete these archived folders at any interval you wish.

1.3.7 Other Tips and Tricks

There's a preference setting for Finder window 'force column view.' This is the only enforceable display setting. Window location, size, fonts, and colors are all adjustable by individual computer users unless you remove the System Preference for those settings.

You may notice when deploying your image to computers that some of your programs have "first run" screens. There is a script you can use which will fix this. The `byhost.sh` available on macosx-labs.org and bombich.com will adjust all of the files on a computer to reference the correct host information.

You may also be interested in protecting preferences (even more than the accounts section of System Preferences allows you to). You can change what System Preferences panels users see by changing file locations. The locations are as follows.

- `/system/library/preferencepanes/`
Default location of preference panes.
- `/users/username/library/preferencepanes/`
Only 'username' will see these.
- `/library/preferencepanes/`
Global panes (e.g. Norton installs here)

1.4 Deploying Images

1.4.1 Deploying the perfect image

The following is a step-by-step guide to deploying. This guide to imaging will assume that you have followed the above steps to prepare the server, master image, and clients.

After creating images, you need to configure NetRestore. Open NetRestore, click on the NetRestore menu, and then choose Configurations. This will open the Configuration Manager, and allow you to configure your images. As per above, create configurations for your images, using the AFP sharepoint. If you only create one configuration, it is also possible to configure NetRestore to be completely hands free. When you have finished configuring, choose save. It is also a good idea to configure some of the preferences. Via the NetRestore menu again, select Preferences. Most likely, you should check the "Restart when finished" and "Set target as boot disk" options. If you are interested in higher machine security, you may be interested in setting the Open Firmware password, which can be found in the "Post-restore actions" entry in the NetRestore menu. After saving your preferences, NetRestore is ready to go. But your preferences are in the wrong location. One of NetRestores' cool new features is the ability to retrieve remote preferences. You can either copy the current preferences to the NetRestore master image (mount the `dmg` and place the settings file in the NetRestore folder), or you can use your server to host it. In my setup, hosting the preferences in an http accessible folder works best, because I am in attendance when we do imaging. For those doing full automation, this may not be a useful suggestion.

Your server at this point should have the NetRestore master image selected in Server Admin. For imaging, please make sure the NetRestore master image is the default for NetBoot. If your NetBoot is not configured, see above. NetBoot the client to the NetRestore master image; Restart the machine and hold down the `N` key.

The machine should boot up to NetRestore. In NetRestore, you should see your configurations. You may want to do other post

restore actions at this point, such as computer naming. This is possible via the "Post-restore actions" section of the NetRestore menu. If you do not have classic installed, do not check "Set Classic Name"; this will cause NetRestore to error and not reboot after imaging is complete. Choose your target (drive). Click the lock to authenticate and then click restore. Your image will be deployed on the machine.

Should NetBoot fail, here are a few suggestions:

- Check your NetBoot server, which has logs for NetBoot in Server Admin.
- Attempt to NetBoot another image, to confirm if your NetRestore master image functions.
- Verify that the local computer has a hard drive, and said hard drive does not have any errors. I have seen many reported cases of "kernel panic on NetBoot" which have been solved by formatting the HD and retrying.
- Verify that the machine has the latest firmware.
- Verify that you are attempting to boot off the primary network card.
- Verify the memory in the client machine; this may sound redundant, but it stumped me for a few hours in one case.

But if all else fails, start over. The process to start over does not take nearly as long as beating your head on a watermelon.

1.4.2 Automatic Computer Naming

Our Apple System Engineer explained how they Apple's consultants were doing computer naming. They are using 'ARD Companion' (<http://www.solidsun.com/ardcompanion/>) to reference a list of names and an identifying value. This was very useful they found because they move computers between schools, and to rename, they merely edit the main list, and the computer self updates. Several people online, such as seen on the system-imaging@lists.apple.com mailing list, have written curl scripts to accomplish this same task.

1.4.3 Imaging over subnets

Netbooting over subnets is difficult. For those who are not running DHCP from the computer that will be hosting the NetBoot service, you may be in for a surprise. This is easily avoided, however. There are two solutions to Netbooting over subnets. One option is a handy tool created by our good friend Mike Bombich, called, not surprisingly, NetBoot Across Subnets. This tool does not work in all environments, but it is a good place to start. The optimal solution requires modifying your routers. You need to configure your router so that broadcast traffic is passed on to your server. If the router is configured as a DHCP relay agent, it should take broadcast DHCP messages in the port the clients are on and repackage them as unicast to the DHCP server (ask your local network administrator what these words mean). By adding the NetBoot server in to the list of "DHCP servers" on the routers relay list, the NetBoot server should also receive the unicasted message and respond correctly. Those curious about the details of this process should read the man page for `bootpd`.

1.5 Conclusion

Imaging can be a fun, quick, and easy process. When I set out to deploy Apple's System Imaging tools, it was three weeks of frustration. Fortunately, I was able to come up with the above

system, after some help from an Apple Systems Engineer or two, several email lists, and bombich.com. After a successful deployment, I was truly amazed at the ease with which students used Mac OS X.

Keep in mind that imaging and deploying with NetRestore is only the starting place in maintaining Mac OS X systems. A maintenance plan for deploying applications, updates, and other is also a necessity. For this, I look towards Radmind, while surviving with Apple Remote Desktop. For those who do not use it, you should consider it. It allows you to have remote screen observation and control, perform inventory and monitoring, and most importantly, install packages remotely. ARD's ability to install packages remotely allows you to distribute Apple Software Updates, new programs, and just about anything. This allows you to be able to update your labs without leaving your office. As I mentioned above, Radmind is also a great system for performing maintenance and updates to your computers.

1.6 Resources

<http://www.bombich.com>

<http://www.macosxlabs.org>

<http://www.info.apple.com>

<http://lists.apple.com/mailman/listinfo/system-imaging>

<https://mailman.rice.edu/mailman/listinfo/radmind>

<http://rsug.itd.umich.edu/software/radmind/>

<http://clc.its.psu.edu/Labs/Mac/Resources/>

<http://www.jamfsoftware.com/casper.html>

http://www.apple.com/server/pdfs/System_Image_Admin.pdf

<http://www.bombich.com/software/netrestore.html>

<http://www.solidsun.com/ardcompanion/>

<http://docs.info.apple.com/article.html?artnum=107902>