Reimage Every Day, Patch Ever Time: A Framework for Maintenance Free University Computer Laboratories

Brett J. L. Landry

Department of Management
University of New Orleans
2000 Lakeshore Drive
New Orleans, LA 70148
Tel: 504-280-3226
E-mail: Brett.Landry@uno.edu

M. Scott Koger

University Computing and Communications
University of New Orleans
2000 Lakeshore Drive
New Orleans, LA 70148
Tel: 504-280-1058
E-mail: Scott.Koger@uno.edu

ABSTRACT

Personal Computers (PC) are difficult to manage in university labs because they are individual machines with individual problems. This paper sets out to provide a procedure to successfully and consistently reimage and patch PCs so they are in a known good state. This procedure also provides an almost zero maintenance effort on the part of lab administrators. Lastly, different patch management scenarios are discussed.

INTRODUCTION

One of the biggest problems with Personal Computers (PCs) in an open use lab environments is managing them as individual items. There have been numerous solutions for managing these devices, but there are always exceptions and loopholes. In September of 2003, university networks around the country were inoperable due to attacks from viruses and worms. The fact that universities were attacked was not new; it was the speed and ferociousness with which it happened that was amazing. On infected networks, an unprotected PC could be attacked and compromised in as little as 30-45 seconds. The reason for the mass infection was that the majority of workstations on campuses were un-patched, did not have adequate passwords, and did not run local virus protection with current pattern files. This paper provides a framework for a secure academic computing reimaging and patching process. Under this process, PCs are reimaged every day and patched every time at reboot. To explain the process, it is best to provide a background on how PC management has been done over the last ten years, discussing models that worked and ones that did not. This discussion begins with Boot Prom Image PCs from the early 1990's.

Boot Prom Image PCs

In the early 1990's it was possible to build a bulletproof PC. A boot prom was installed on the network interface card (NIC) that enabled the PC to boot from a read only boot image across the network. The network server contained the image and stored all user files. This environment worked for both DOS and Windows 3.X clients. Basically, the methodology was simple. A PC was powered on, and sent a request across the network requesting an image. A network file server responded with the corresponding image file and the machine booted. Since the image file was read only, a pristine, virus free client was provided each time the machine was restarted (Landry, 1995). If the machine had a hard drive for temporary storage, it was quick formatted at the next boot. The hard drive also allowed students some flexibility to make changes to their local configuration, complete assignments, and experiment with the PC. The good news was that any changes made by the user were temporary and a reboot reset all configuration changes.

Although hardware was not as dependable in the early 1990s as it is today, maintaining hardware in this timeframe was much easier. The reason was simple; if a PC had a problem, the lab operator would reboot the PC. If the PC received network boot prom issues, it was the NIC or the network media. This was more common in the early 1990s than today mainly because of Thinnet coaxial networks. If the image loaded and there was a problem, it was a hardware issue, because the software was common to all PCs, and could not be changed by the end user. When the PC needed to be replaced, a tech would disconnect the old PC, add the new one to the corresponding tables, plug a new one in and everything worked. Literally a five minute fix and the hardware problem could be fixed off site offering minimum downtime to the end user.

In 1992, with the advent of the World Wide Web, PCs at universities in labs and on administrative desktops had a legitimate need to have more than 16 colors and sound cards. For Windows 3.X Operating Systems (OS) this meant that different drivers had to be enabled for different video cards and sounds cards. It is important to note that this was before the time of plug and play enabled OS. Although a nuisance, LAN systems programmers at Mississippi State University got around this issue by using common UNIX tools such as SED and AWK that were ported to DOS to dynamically rewrite the system.ini and win.ini files at boot up to the appropriate file based on the MAC address for that station (Landry, Burrell, & Griffin, 1996).

Windows 95 Registry

When Windows 95 was unveiled at the 1995 Comdex show in Atlanta, GA., it was quite clear all that was going to change. The advent of the registry meant that third party tools could not as easily modify settings. Worse yet, the size and complexity of Windows 95 did not lend it itself to the boot prom methodology. Windows 95 was only the beginning. All future Windows releases including 98, ME, NT 3.51, 4.0, 2000 and XP all integrate the registry not only for windows settings but for all application settings also. The *Microsoft Computer Dictionary*, Fifth Edition, defines the registry as (Microsoft, 2004b):

> A central hierarchical database used in Microsoft Windows 9x, Windows CE, Windows NT, and Windows 2000 used to store information necessary to configure the system for one or more users, applications and hardware devices. The Registry contains information that Windows continually

> references during operation, such as profiles for each user, the applications installed on the computer and the types of documents that each can create, property sheet settings for folders and application icons, what hardware exists on the system, and the ports that are being used. The Registry replaces most of the text-based .INI files used in Windows 3.X and MS-DOS configuration files, such as the Autoexec.bat and Config.sys. Although the Registry is common to several Windows operating systems, there are some differences among them.

RESTRICTIVE APPROACHES TO WINDOWS MANAGEMENT

A number of tools have been introduced to help manage and restrict access on local PCs. This restricts the user from installing new applications and making unwanted changes to the desktop. Depending on the level of security, it may also keep the user from making changes to the registry. These restrictions can be enforced by Windows policy manger, via directory tools such as Microsoft's Windows 2000 Change and Configuration Management services or Novell's ZENWorks for Desktops. Both of these solutions require that workstations be initially connected to the network to be managed. For Microsoft Windows clients (NT 4.0, Windows 2000 and Windows XP) that participate in an Active Directory (ADS) domain, a security template is pushed to the local machines registry, defining levels of privilege both on the local machine and on the network for the user account that is used to log on to the workstation. Each user account is a member of a predefined group policy object, which defines both local and network levels of privilege. In an ADS environment, Windows 2000 and XP clients check with the domain controllers at a pre-defined interval plus or minus some randomly generated number of minutes for changes to security settings or changes to group policy. The default interval is 90 minutes.

Both the Microsoft and Novell solutions restrict what applications a user can run and can be used to remove the local user's ability to install new software. This restriction quite often means that users do not have the privileges to install a service and security patches on their own. As a result, although the user cannot install new applications, the PCs become infected because they are not patched and vulnerable to worms and viruses on the network.

Windows Profiles

One of the problems that routinely plague computer laboratories is the issue of profiles. Profiles save individual user settings. They are most commonly local to an individual PC. Microsoft defines local profiles as:

> A local user profile is created the first time that a user logs on to a computer. The profile is stored on the computer's local hard drive. Changes made to the local user profile are specific to the user and to the computer on which the changes are made. (Microsoft, 2004a)

The problem with local profiles is that there is a maximum number of local profiles that can be stored on a PC. When this maximum is reached, administrators must either script or manually delete local profiles. Another problem, is that once these local profiles are created, making

changes on the PC may not carry over to all users. Therefore, administrators have to make sure that icon and folder changes are available to all profiles. Unfortunately, in practice, this step has been quite often forgotten.

Infected PCs

In practice, solutions based on restrictive user policies alone do not eliminate infection of individual computers with malicious code. Increasingly, viruses and worms are exploiting system level privileges to install their malware, requiring a defense in depth approach to desktop security. (Short for ***mal****icious soft****ware***, software designed specifically to damage or disrupt a system, such as a virus or a Trojan horse (Webopedia, 2004).

Once a PC is infected it can either be manually cleaned, formatted and reinstalled, or reimaged. Reimaging is the process of applying a saved image of the hard drive and restoring all data to the PC. St. Saver points out that of the hundreds of infected PCs on college campuses few were formatted and reinstalled. Most of the compromised PCs were only patched, potentially leaving backdoors for future infections and attacks (2003).

There is no particular way to ensure that infected PCs are really cleaned, as remnants can still exist even after the most through inspection by an experienced technician. The most that can reasonably be expected is that no obvious additions or deletions have taken place. The technician can look for unfamiliar applications, newly added local user accounts, or unusual services running in the background for example. Without an enterprise wide standard desktop environment and a current method to authenticate that the contents of the hard drive have not been altered there is no real way to ensure that any dropped files or "tool kits" haven't been left behind, leaving the system vulnerable to future attacks. With the unique requirements of academic computing environments, a standard desktop environment is unlikely at best, and with the constant stream of updates to individual software products and new virus template files constantly being delivered to anti-virus clients via the web, as well as patches and security updates for operating systems, an accurate listing of what should be on a given hard drive would be impractical if not impossible to maintain.

Reimaging is the process of taking a saved copy of the hard drive and restoring the continents. It can be considered a point in time backup of the OS, applications, registry, and files. Two of the most widely used applications for reimaging are Ghost and Power Quest Drive Image, commonly referred to as PQDI, which are both owned by Symantec as of September 2004.

WIRELESS LAPTOP PROJECT

In the spring of 2002, the Department of Management at University of New Orleans (UNO) purchased 40 wireless laptops for use in Information Technology (IT) classes. One of the requirements for the project was that students have full control of the laptops and be able to install applications and make changes to complete IT assignments. This requirement arose because it was found that locked down Windows PCs were so restrictive that students could not complete in class assignments. A second requirement stated that the laptops uses as little maintenance and support time as possible since they were being supported by IT faculty in their

spare time. The goal was to develop a system that would be as dependable and low maintenance as the old prom PCs from ten years before.

To support these goals, the department bought Powerquest Deploy Center to provide a means to reimage the PCs from the local hard drive. This involved installing Windows 98, and then installing Windows 2000. Using the built in Windows boot manager, the user could use the default option and boot Windows 2000 or choose the rebuild option. The rebuild option restored the PC back to the original settings in 15 minutes. It should be noted that the 15 minute restore time was based upon 1Ghz laptops with average speed hard drive. This time will be less on faster desktops or could be more with larger images. Powerquest was scripted to automatically delete the active partition, select free space, and restore the image. While the process used a standard image for all PCs, Microsoft's Sysprep utility was scripted so that the PC was restored with the correct unique workstation name and settings. Additionally, the laptops contained a hidden image that contained the image file, so the laptop could be reimaged at any time of the day without any concern for what it might be doing to network traffic. This also ensured that the PCs could be reimaged even when the network was unavailable.

This meant that if anyone had a problem with a PC, they could choose the second option in the boot menu and in 15 minutes the PC was restored. Any changes, deletions, viruses, worms, new applications, or P2P sharing applications and media files were not restored. While this worked very well for this project, it did not catch on across the university for a number of reasons. The limitation of having Windows 98 installed first meant that computer administrators would have to rebuild their PC images after installing the older version of Windows. Because Windows 98 does not support the NTFS file system, and Windows 2000 has to be installed to the first partition, the entire hard drive was formatted FAT 32 which is an older and less reliable and secure file structure. Another issue was that since the default boot partition was being reimaged if the PC was turned off or lost power during the reimage, it could not boot. Additionally, there was no way to update the image on the laptop, so someone had to visit all 40 laptops and copy the image to the hidden partition.

A second approach was sought in the summer of 2004 to address these issues and to upgrade to Windows XP. Using Powerquest and Sysprep, a solution was developed that matched all of these needs. The solution was to first install Windows 98 and then Windows XP. Then the Windows structures for 98 and XP were deleted so that the PC booted into command line mode. The boot.ini was modified to point to an XP install on the second partition. An image was created and named smalboot and is a 1.5 meg image. (The actual partition size varies bases on cluster size, but restores to approximately 50 megabytes). The drive was then formatted and a standard XP image with all of the applications was installed as initially created. This image was copied to the hidden partition. The smalboot image was restored first, then the XP image.

Because XP was installed without the smalboot Windows 98 partition, it loads itself as C:\ and the Windows 98 partition as another drive letter. Since this drive is a system partition, Windows Disk Manager cannot remove the drive letter, so Partition Magic was used to make this partition hidden. Hiding this partition will keep students from altering the contents of that drive. The image partition is best to be hidden as a mapped folder in windows so updates can be copied to

the PC. When the laptop reboots to be reimaged, the XP image is hidden because Windows 98 cannot see NTFS partitions.

The process was then automated so that every night the laptops can be reimaged by scheduling a change in the boot.ini and rebooting the PC. The laptop also automatically checks for a new image and if one is available, it downloads it to the hidden partition. A CRC algorithm then verifies the completeness and validity of the image file. If the machine loses power during this process, when it is restarted the smalboot image is loaded via the boot.ini to reimage the PC. The boot.ini file will cause the PC to be reimaged until a successful reimage takes place no matter how many times power is lost. Once a successful reimage takes place, then the boot.ini is rewritten for a normal XP boot. To clean up the process, you can modify the msdos.sys file on the smalboot partition to include the line "logo=0" so the Windows 98 splash screen does not load.

The question may be asked, why not install a FAT32 partition as drive C:\ and XP on drive D:\ and not go through the process of creating special partitions and hiding them. Having a C: and a D: drive still introduces the vulnerabilities of having a FAT32 partition. Every default script and installer looks for Windows on C:\, so users would always have to be cognitive of changing this and looking for files in 2 places. This is not a realistic solution. The last issue is that Microsoft Sysprep that is needed to rename the workstation only works on C:\. Having derived this solution, we determined that there was one additional component not covered; the issue of patching the workstation. The notion of patching every time is simple; the methodology varies upon the individual computing environment.

PATCH MANAGEMENT

Path management is a fact that cannot be ignored today. The worm outbreaks of 2003 taught the IT community that infected PCs are not an individual problem. They are a corporate problem when they choke and congest local networks to the point that nothing can travel on the network. The best approach depends on the infrastructure in place. The discussion here is not intended to be an all inclusive list but simply suggestions alternatives.

Microsoft Environments

When Microsoft's System Update Server (SUS) is in place each domain member computer can be scheduled to check an update server for new security patches and updates. When present, the system will download them across the local network, rather than every PC downloading the patches over the Internet from Microsoft's update site. Not only does this provide for much greater download speeds of the updates, it also allows local administrators to control which updates are released to the domain member computers. This has become especially important in the Windows XP SP2 case. If there are known compatibility issues with a particular update, a particular group of users, or the community at large, local administrators can block its deployment until a work around or fix can be found.

If the PC is booting for the first time after a reimage, it will check for product updates as part of the security template check when it initially connects to the domain controllers if its machine

account is a member of an Organizational Unit (OU) that has been assigned to a System Update Server. SUS is currently provided as a free supplemental product for Windows 2000 and 2003 servers. This process uses a passive "pull" approach. Administrators must wait for the individual computers to request updates from the SUS server. A proactive approach is available with System Management Server (SMS). This product allows administrators to actively poll domain clients, determine their patch level, and "push" updates that are missing. This system can even remotely force reboots to complete installations if necessary. There are also third party applications available for remote change and configuration management.

Novell Environments

For universities with Netware infrastructures, ZENWorks for desktops can be used to administer patches, install new applications, and even remove applications. ZENWorks leverages the Novell Directory Structure (NDS) so usage is limited to directory administrators. Unlike SUS, ZENWorks handles all remote change and configuration management, so additional third party applications are not needed.

Non Directory Environments

There are times when SUS, SMS, and ZENWorks may not be realistic solutions. This may be the case because there are not ADS or NDS infrastructures in place. It may also be a limitation because departmental administrators may not be directory administrators and control cannot be granted due to technical, political, or skill set issues. It may be due to the fact that the directory structured is not partitioned into the appropriate OUs to allow the granularity needed. In this case, the recommendation is to develop your own solution. This solution can be as simple as having the Windows scheduler copy all patch files to the local hard drive and have the PC call a batch file using the Microsoft Qchain patch files. All patches needed would be installed upon reimage and only missing files on every boot. Even this manual approach is better that the traditional lab patching methods which is basically nothing at all or "sneakernet" during university holidays.

SUMMARY

Lab administration is a difficult and time intensive task. Unfortunately, with wireless laptops, universities now have mobile labs. When a class is planned to se these technologies and they are not available, the entire class is lost. Therefore, methods and procedures must be deployed to ensure that PCs are in a known good state. They are not sharing and spreading viruses and worms. These laptops need to provide a stable environment. It is the authors' hope that by deploying scheduled reimaging and regular patching that PCs in student labs can do this.

REFERENCES

Landry, B. J. L. (1995). An Overview of Classroom Networking. In *Mississippi Educational Computing Association (MECA) 1995*.

Landry, B. J. L., Burrell, J. G., & Griffin, T. A. (6-30-1996). LVH+ (Unpublished work).

Microsoft (2004a). About User Profiles. http://msdn.microsoft.com/library/default.asp?url=/library/en-us/policy/policy/about_user_profiles.asp [On-line].

Microsoft (2004b). Description of the Microsoft Windows registry. http://support.microsoft.com/default.aspx?scid=kb;en-us;256986&Product=winxp [On-line].

St.Saver, J. (2003). Picking at a Virus-Ridden Corpse, Part II. Syllabus [On-line]. Available: http://syllabus.com/news_issue.asp?id=153&IssueDate=9/25/2003

Webopedia (2004). malware. http://www.webopedia.com/TERM/M/malware.html [On-line].