MacOS 10.4.x Client in Lab Environments with ASURITE

Performance Issues

When I installed MacOS 10.4 my old lab computers got really slow, how can I fix that?

Buy new computers ☺

Or you can disable CPU intensive applications that lab users don't need.

Note: Some applications may rely on the following applications. Use these instructions at your discretion on a test machine.

Killing Dashboard

Start the **Terminal** application located in **/Application/Utilities**
Type the following:
sudo defaults write com.apple.dashboard mcx-disable –boolean YES

If you happen to change your mind and need Dashboard later, then type this:
sudo defaults write com.apple.dashboard mcx-disable –boolean NO

Killing Spotlight

To kill Spotlight indexing, which most labs do not need do this:
Start the **Terminal** application located in **/Application/Utilities**
Type the following:
sudo mdutil –i off /
sudo mdutil –E /

To disable Spotlight edit the **/etc/hostconfig** and change **SPOTLIGHT=-YES-** to **SPOTLIGHT=-NO-**
Spotlight sometimes indexes other users files, which can be a bad thing! Kind of like Google Desktop for Windows.

Security

Preventing users from logging in via single user mode (a.k.a. holding down the CMD and S key on bootup).

Edit the file **/etc/ttys**
Start the **Terminal** application located in **/Application/Utilities**
Type the following:
cd /etc
sudo cp ttys ttys.backup
sudo pico ttys
Change **secure** to **insecure**
Save and exit pico
Next we will enable root so you have a working password in Single User mode.
Type the following:
cd /etc
cp master.password master.password.backup

sudo pico master.passwd
delete the asterisk following the word "**root**"
Open a second terminal window and type in the following command:
openssl passwd –salt <*xx*> <*password*>

Replace <*xx*> with two random characters and <*password*> with the root password. Make sure that no unauthorized people are viewing the screen.

Copy the random generated hash from your second terminal window to the first window replacing the asterisks you deleted in a previous step.
Save and exit.

Creating Login Warning Banners:

Start the **Terminal** application located in **/Application/Utilities**
Type the following:
plutil –convert xml1 /Library/Preferences/com.apple.loginwindow.plist
sudo pico /Library/Preferences/com.apple.loginwindow.plist
After the first <dict> tag, type in the following on a new line
<key>LoginwindowText</key>
On the next new line, type in the following
<string>*This is my funky login banner to scare and annoy users.*
</string>
Replace "This is my funky login banner to scare and annoy users." with your own banner message.
plutil –convert binary1 /Library/Preferences/com.apple.loginwindow.plist
Save and exit.

Here is a nice banner for ASU:
WARNING! You are accessing a computer protected by federal and state law and ASU policies. By using this system you agree to comply with these laws and policies, including ACD 125 (Computer, Internet, and Electronic Communications Policy) and you consent to system monitoring for law enforcement, administrative, and other purposes. Unauthorized uses of this computer system may subject you to criminal prosecution, civil liability and university sanctions.

Creating a remote login banner message for FTP users

Even if you have disabled remote login, you may wish to add a banner message for legal reasons in case of an intrusion.

Start the **Terminal** application located in **/Application/Utilities**
Type the following:
sudo pico /etc/motd
Type in your banner message, save and exit.

Creating a remote login banner message specifically for SSH users

Start the **Terminal** application located in **/Application/Utilities**
Type the following:
sudo touch /etc/login_display
sudo pico /etc/login_display
type in your message
sudo pico /etc/sshd_config
Replace the line **#Banner /some/path** with **Banner /etc/login_display**

After you reboot, the changes will take effect.

Disabling iTunes:

Open **/Applications/System Preferences/Sharing/Firewall**

You may wish to disable iTunes Music Sharing for legal reasons by disabling the icon and blocking port 3689 in the firewall.

Additional Login Customizations:

When MacOS 10.4x starts up, the "Starting Mac OS X…" text comes up. This can be changed in
/System/Library/CoreServices/Resources/English.lproj/SystemStarterUI.strings
Change
"System_Starting_Message" = "Starting Mac OS X…";
to
"System_Starting_Message" = "*Whatever you want***";**

Disabling Bluetooth

Some systems purchased may have come with built-in Bluetooth capabilities. In a lab environment, this is not needed and may be a potential security risk.

Through the GUI interface go to **/System/Library/Extensions**
Drag the following files to the trash
IOBluetoothFamily.kext
IOBluetoothHIDriver.kext

Disabling Wireless
Some systems purchased may have come with built-in Wireless capabilities. In a lab environment, this is may not be needed and may be a potential security risk.

Through the GUI interface go to **/System/Library/Extensions**
Drag the following files to the trash
AppleAirPort.kext
AppleAirPort2.kext
AppleAirPortFW.kext

Disable Core Dumps

If MacOS software crashes, a core dump may occur. The dump may contain user security information. I recommend disabling the dump by doing the following.
Start the Terminal application located in /Application/Utilities
Type the following:
sudo pico /etc/sysctl.conf
add the following to the file
kern.coredump=0
Save and Exit.

Securing the Apple Firmware and startup
Apple has an excellent article at:

http://docs.info.apple.com/article.html?artnum=106482

Printing History
MacOS keeps a history of print jobs in the queue. This is typically accessible by going to a web browser on the local machine and typing http://127.0.0.1:631 or http://localhost:631 from there the user can navigate to see their printing history.

If a hacker should gain access to the system, they may be able to retrieve the recent print jobs on the system. To stop this from happening type in the following commands:

sudo pico /private/etc/cups/cupsd.conf
change the following line from
#PreserveJobHistory Yes
to
PreserveJobHistory No
Save and Exit.

User Configuration

Disabling Displayed Usernames and securing the workstation

Go to **System Preferences\Accounts**
Click on **Login Options**
Uncheck **Automatically login as:**
Check **Name and password**
Uncheck **Show password hints**
Uncheck **Enable fast user switching**

Adding a password to the Screensaver:
Go to **System Preferences\Desktop and Screensaver**
Select an appropriate time limit for the Screensaver to take effect.
Go to **System Preferences\Security**
Check **Require password to wake this computer from sleep or screen saver**

Login Hooks:

Apple has an excellent article on login hooks at:
http://docs.info.apple.com/article.html?artnum=301446

Also check out Bombich's site at: http://www.bombich.com/software/lwm.html
He has written an excellent Login Manager program.

If you upgraded from 10.3 to 10.4 and your login scripts stopped working then you may have to do something like this:
Start the **Terminal** application located in **/Application/Utilities**
Type the following:
sudo defaults write com.apple.loginwindow LoginHook *</script/path>*
sudo defaults write com.apple.loginwindow LogoutHook *</script/path>*

Setting Up a Shared and Configurable Profile for Active Directory Users

We are assuming that you are already in Active Directory
Create a standard account that will be shared (for this exercise we will call the shared account Dude)
Start the **Terminal** application located in **/Application/Utilities**
Type the following:
sudo dsconfigad –mobile disable –mobileconfirm disable –localhome enable -nogroups
sudo dsconfigad –staticmap "dsAttrTypeStandard:UniqueID" 502
sudo dsconfigad –staticmap "dsAttrTypeStandard:GeneratedID" 502
sudo dsconfigad –staticmap "dsAttrTypeStandard:PrimaryGroupID" 502
sudo dsconfigad –staticmap "dsAttrTypeStandard:NFSHomeDirectory" Users/Dude
Note: This refers to the users home directory on the local machine.
sudo dsconfigad –staticmap "dsAttrTypeStandard:HomeDirectory" Users/Dude
Note: This refers to the user's home directory on a remote server. This will override the Active Directory home directory setup by Enterprise Admins if the path is the same as the NFSHomeDirectory. To go to a specific server, replace Users/Dude with something like this:
<homeDir><url>cifs://192.168.1.1/Freds_stuff</url><path></path> </homedir>

Reboot and login with your Active Directory Account. You should be logged into **/Users/***Dude*. If you are logged into an account where your home directory is not "*Dude*", then there has been a misconfiguration. Log back in as an administrator and delete your Active Directory account.

If you are creating an image, then unbind from Active Directory. After deploying the image you can rename the computer and rebind. All of the previous settings will be retained.

Killing Keychain Data

Apple Keychains store private passwords and data. This could be very bad on a shared profile. Legal issues could ensue from a variety of scenarios. The solution is to kill the data when the user logs off.

First create a directory where you want to store the script. I suggest putting it in the /Library folder.
Example:
Start the **Terminal** application located in **/Application/Utilities**
Type the following:
sudo mkdir /Library/*mystuff*
Note: *mystuff* is the name of the directory you made up.
cd /Library/*mystuff*
pico LogoutScript.sh
Type in the following information to kill keychains
#!/bin/csh
#Keychain Killer
/usr/bin/chflags nouchg /Users/*username***/Library/Keychains/login.keychain**
/bin/rm –R /Users/*username***/Library/Keychains/login.keychain**
exit 0
Save and Exit.
Note: *username* is the name of the account you wish to kill the keychains in.
Now we want to make the script executable.
sudo chmod a+x /Library/*mystuff***/LogoutScript.sh**
Next we want to enable the script.

sudo defaults write com.apple.loginwindow LogoutHook /Library/*mystuff***/LogoutScript.sh**

Preventing .DS_Store files from being written to the servers

Start the **Terminal** application located in **/Application/Utilities**
Type the following:
sudo defaults write com.apple.desktopservices DSDontWriteNetworkStores true
Note: This keeps Windows System Administrators happy. The DS_Store tells the Apple Explorer how to present the files to the users (Thumbnails, positions, list, details, etc, etc)

See http://docs.info.apple.com/article.html?artnum=301711 for more info.

Backing up a User Profile

Backup through GUI

Click on **System Preferences**
Click **Accounts**
Click on the account to backup
Click on the **–** (minus sign)
*** Do not click on Delete Immediately ***
Click **OK**

The users home folder has been turned into a disk image located at **/Users/Deleted Users/**
Note: This file can be transferred to another computer or used as a backup.
Simply double click on the file with the users name and the .dmg file extensions to open it up.

In order to restore the account, you must do the following:

Double click and open up **/Users/Deleted Users/***username*
Create a folder in **/Users** with the *username*
Copy the all the file from the disk image on your desktop to **/Users/***username*
Unmount the disk image on your desktop by dragging it to the trash
Click on **System Preferences**
Click **Accounts**
Create an account with *username*
Click **OK** when the dialog box comes up saying the user account already exists.

OR

Backup through Terminal

Profiles sometimes become corrupted. It is a good idea to backup the user profile.
Login as a different user with administrative access
Start the **Terminal** application located in **/Application/Utilities**
Type the following:
sudo /usr/bin/ditto –rsrcFork /Users/*username* **/private/var/root/***username*
Note: *username* is the name of the account you wish to backup

If you wish to restore the account you backed up at a future date due to corruption or other reasons, then you must do the following.

sudo /usr/bin/chflags –R nouchg /Users/*username***/***
sudo /usr/bin/chflags –R nouchg /Users/*username***/.??***
Note: The step unlocked the files.

sudo /bin/rm –R /Users/*username***/***
sudo /bin/rm –R /Users/*username***/.??***
Note: The step deletes the files. Be extremely careful when typing the rm –R command. Incorrectly typing this command can have devastating consequences!!!

sudo /usr/bin/ditto –rsrcFork /private/var/root/*username* **/Users/***username*
sudo /usr/sbin/chown –R *username***:** *username* **/Users/***username*
Note: As with everything in this document, you can script it through AppleScript or UNIX .sh.

Things I've found:

If you're using Carbon Copy Cloner or Net Restore, then don't forget to propagate all the necessary networking information in the Network settings. Things get a bit freaky if you don't!!

Instructions by Jason Wulf
(Don't ask me questions, I just wrote up what I found out)
"I know nothing!!" – (Schultz-- Hogans Heroes)