Mac OS X Security

A Brief Look At The Dark Side

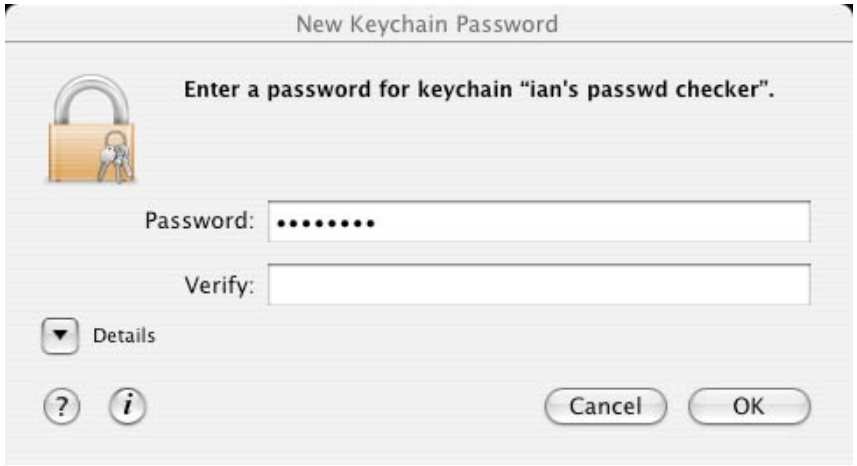
Ian Kaufman March 2005

We've Been Hacked! Or have we?

- Recently, 3 machines were compromised
- How did we find out? IRC traffic caught going to the machines
- No evidence of root compromise detected
- Same account/password across all 3 machines via Netinfo Database - check out the CPP document about securing Netinfo! http://www.lbl.gov/ITSD/Security/systems/mac_guidelines.html
- This was not an OS X specific problem!
- The password was guessed, was not a "good" password

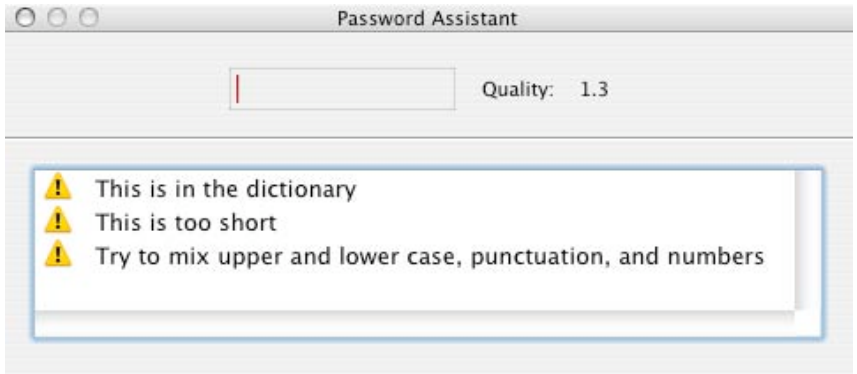
Passwords – How Strong Are They?

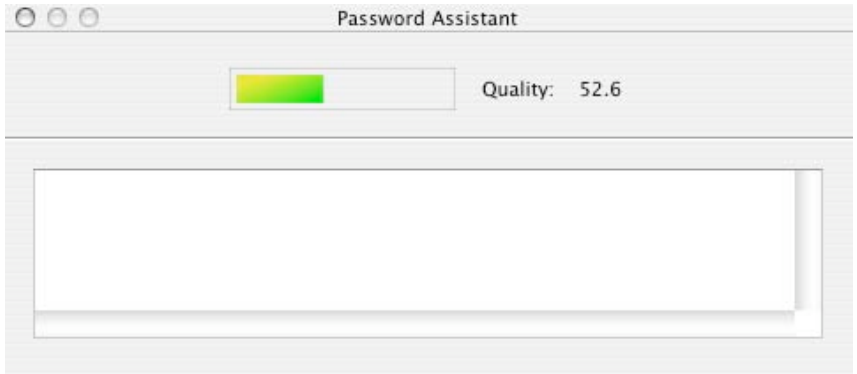
- Fortunately, OS X has a built in password checker – the Keychain!
- Create a new Keychain, and in the password dialog box, click the “i” button



Password Checking part II

- A dialog box will come up showing how weak/strong your password is, and make suggestions on how to strengthen it





HFS+ Security Problems

- HFS+ stores info in multiple forks
- Non-Carbonized OS 9 apps use a data fork (which contains the executable or binary data) and a resource fork (icons, dialogs, sound)
- OS X is based on UNIX which only uses single forked files – data only
- Modern OS X apps dump the resource fork and use either a .rsrc file (Carbon) or store the resources as separate files (Cocoa)

HFS+ vs. UNIX

- On a UFS volume, OS X stores any resource fork as a separate file prefixed by a “._Fork” or “..namedfork”
- When viewed at in the command line, it appears as a subdirectory called /rsrc, but are invisible to “ls” unless specifically targeted
- As a result of all of this, server daemons that open file streams can be fooled into opening the respective file resource and/or file forks, opening up the underlying source code of the server side documents to remote users

HFS+ Security Fixes

- Apple released a security patch for Apache 1.3.29 to fix this
- Implemented a mod_rewrite rule to httpd.conf:

```
<Files “rsrc”>
Order allow,deny
Deny from all
Satisfy All
</Files>

<DirectoryMatch “.*\.\.namedfork”>
Order allow,deny
Deny from all
Satisfy All
</DirectoryMatch>
```

More HFS+ fixes

- 4D (WebSTAR Web Server V) is also vulnerable, you can get instructions on how to secure the server at http://www.4d.com/products/hfs_sec.html
- Any service of this type might be vulnerable, so if you run a dedicated webserver – use UFS

Anti-Virus Software: Yes or No

- Currently, there are no known Mac OS X viruses in the wild (yet!)
- This most likely will change as OS X rises in popularity and deployment
- Windows viruses can be transferred in attachments, some macros can travel cross-platform

Anti-Virus Software – cont'd

- It's free from the lab and has little overhead
- Might be a DOE/OA requirement in the future?
- Bottom line – Why not?
- Better safe than sorry ☺

FileVault – the good

- FileVault has strong encryption – AES 128 bit
- Encrypts and decrypts on the fly without you noticing
- If you have a lot of info you want guarded, this is a good idea
- If your laptop gets stolen, your data is pretty much secured

FileVault – the bad!

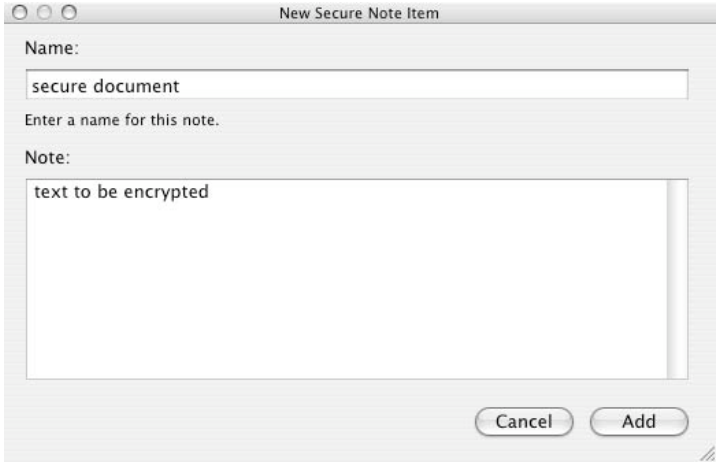
- If you have limited RAM and/or deal with a lot of CPU intensive tasks, the performance hit becomes noticeable
- Don’t lose your key/password - no way to decrypt the files! The only way to decrypt a user’s files if s/he loses the password is the Master Password.
- Some backup apps do not deal with FileVault well – the smallest of changes can cause the entire image to be backed up
- Tricky to ssh into FileVault protected account or if you use File Sharing and the account is not already logged in at the console. All that exists is an encrypted sparseimage.

FileVault – the options

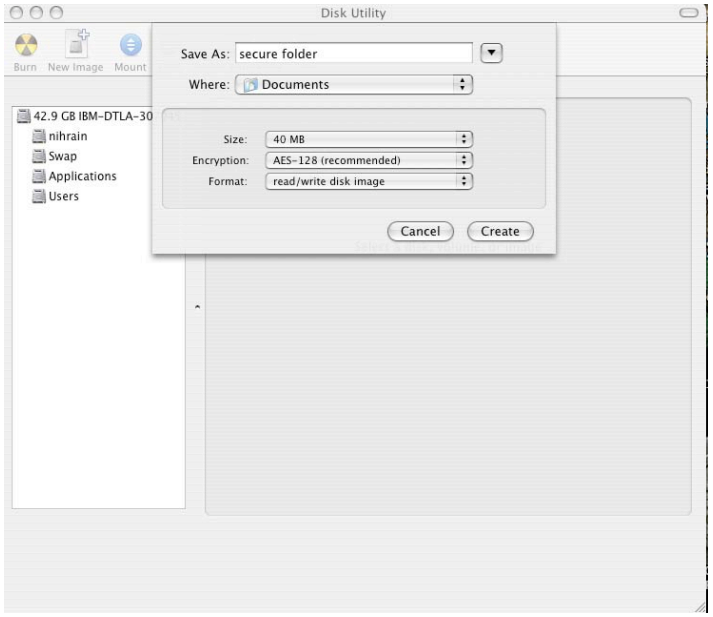
- For most users, this is overkill (and potentially risky)
- Cannot guarantee the sanctity of data that resided on the disk prior to enabling FileVault – any data that was deleted may still be resident
- One solution – encrypt files as needed with PGP or GnuPG
- Another built in solution is to use the Keychain

Keychain Notes and Encrypted Disk Images

- Keychain can let you write encrypted notes – whole text documents can be encrypted this way



- Or keep important items in a single file/directory, and create your own encrypted disk image



Spyware – Is it on my system?

- Finding spyware in open source code is like looking for a needle in a haystack
- Most spyware will probably be found in Library > StartupItems, Library > Scripts, Library > Extensions at both the system level and in your homedir
- Regularly do process accounting – use OS X's Activity Monitor, write/find a shell or perl script or find some nice GUI approach

Spyware – con’td

- Tools are out there to help detect spyware that may be already installed on your system
- Intego’s NetBarrier and Allume’s (originally Aladdin) Internet Cleanup can see suspicious outgoing activity. Internet Cleanup has bad reviews though
- Little Snitch (shareware) – http://www.obdev.at/products/littlesnitch
  note, the Opener malware/OS X Trojan Horse specifically disables Little Snitch

Firewalls

- Mac OS X uses IP Firewall (ipfw)
- Not exactly the easiest one to write rules for
- OS X's GUI interface is very limited – and only deals with TCP connections, not UDP
- Xupport 2.3 ipfw GUI
  http://www.computer-support.ch/Xupport/
- BrickHouse 1.2b12 – ipfw GUI (shareware)
  http://personalpages.tds.net/~brianhill/brickhouse.html
  the latest version is found at http://www.versiontracker.com
- sunShield 1.5 – ipfw GUI (freeware)
  http://www.sunProtectingFactory.com/sunShield

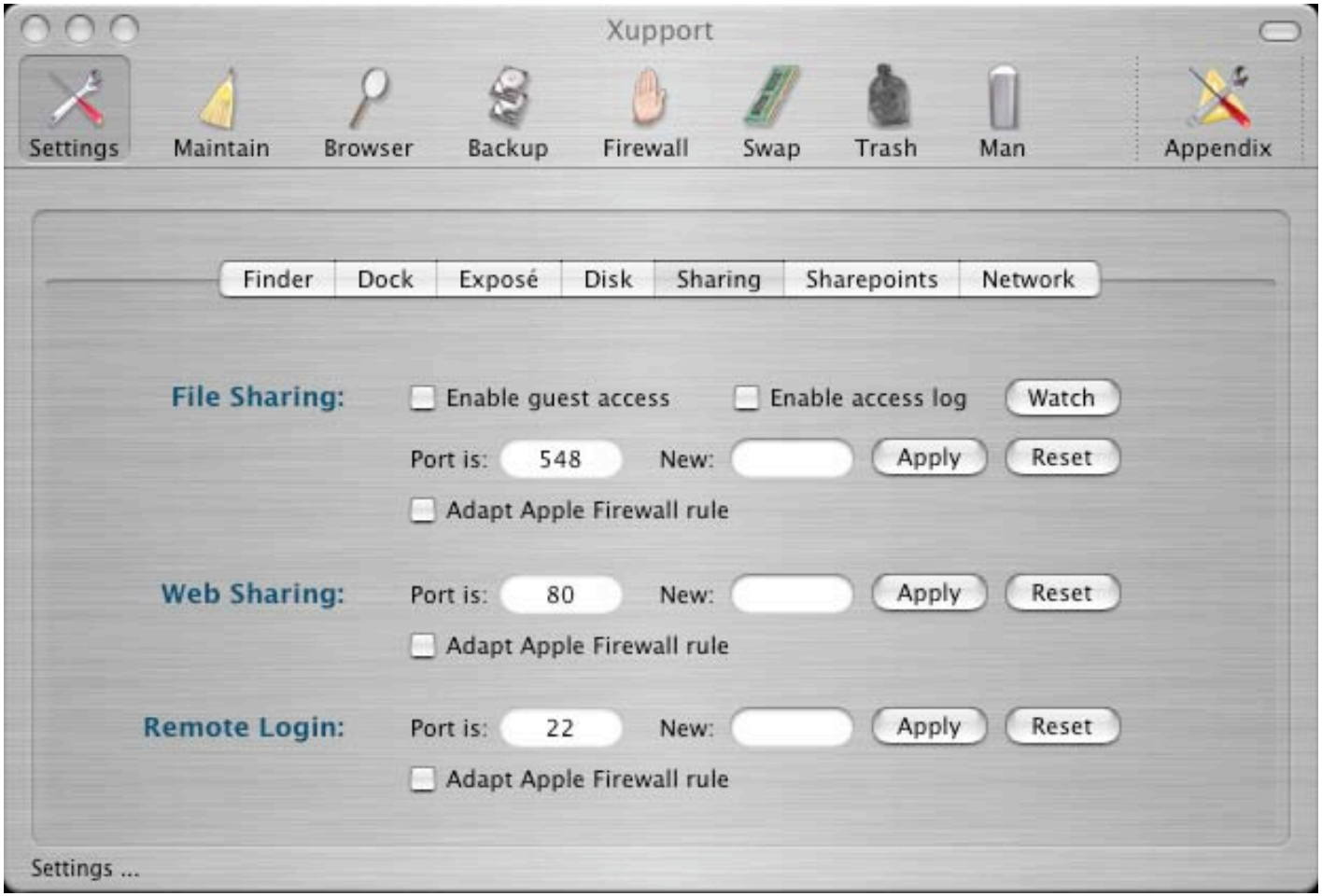
Firewalls – cont'd

- FirewalkX – standalone (shareware)
  http://www.pliris-soft.com/products/firewalkx/index.html
- IPNetRouterX 1.0.4 – standalone
  http://www.sustworks.com/site/prod_ipnrx_overview.html
- Look up or find out what port numbers you might actually use – block things you have no need for, restrict things the world should not have access to

More Firewalls

- For a list of Apple specific ports: http://docs.info.apple.com/article.html?artnum=106439
- Xupport lets you easily modify Apple's built in firewall, and can get more advanced – it can even deal with UDP ports. Plus, it has a list of known Apple and known IETF ports and examples built in!

Xupport Screenshots - Settings



Xupport Screenshot - Simple

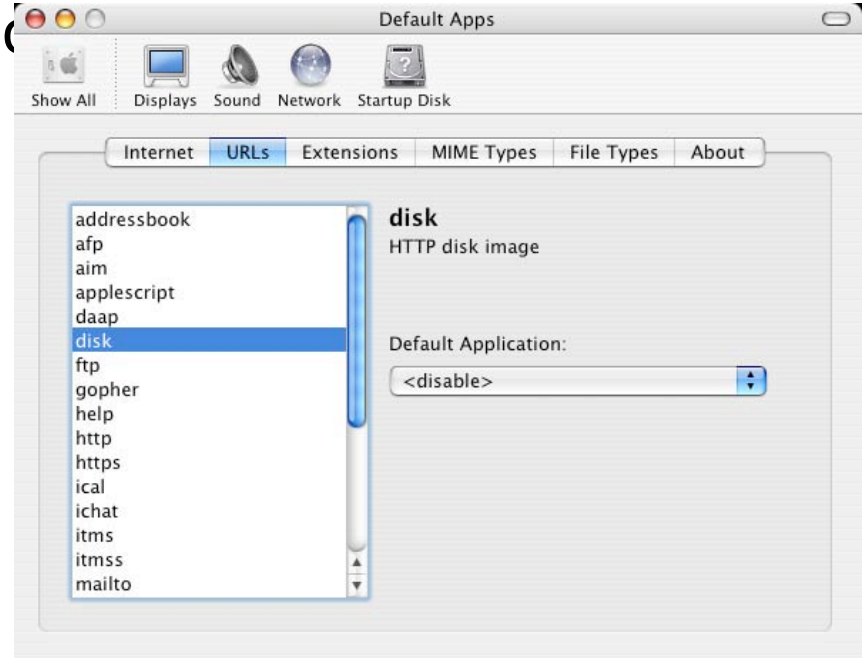
Xupport Screenshot - Examples

Uniform Resource Identifier (URI)

- Not just OS X, but not fun either
- Crackers can set up web pages that can mount a disk image and then uses the ‘help’ protocol to trick the Help Viewer into executing a script from the disk image
- By default, disk images will automatically be mounted – embedded code runs with whatever privileges the logged in user has
- Apple released a patch for Help Viewer, but it doesn’t entirely fix the problem

URI Solution

- Get Rubicode's RCDefaultApp
  http://www.rubicode.com/Software/RCDefaultApp
- Not only will it let you redefine how some URIs are handled by default, but it also gives you a friendly one stop GUI to perform filetype associati



Conclusion and Questions

- Remember, OS X is UNIX/BSD based – and heavily populated with Open Source software – any vulnerabilities that affect them can very well affect OS X
- In the immortal words of Sgt. Phil Esterhaus (the late Michael Conrad) from *Hill Street Blues*:
  "Let's be careful out there."

Sources and Links

- Toporek, Chuck, etc., *Mac OS X Panther In A Nutshell*, O'Reilly, June 2004
- McElhearn, Kirk, "Protecting Data in Panther", Macworld June 2004
- Anbinder, Mark H. etc, "Mac Security: Fact and Fiction", Macworld March 2005
- CapMac Forums "Mac and Spyware surveillance", http:/capmac.org/phpbb2/viewtopic.php?t=2131

Sources and Links con’td

- Lavigne, Dru “BSD Firewalls: IPFW Rulesets”, http://www.onlamp.com/lpt/a/831
- Gruber, John “Disabling Unsafe URI Handlers With RCDefaultApp”, http://daringfireball.net/2004/05/unsafe_uri_handlers
- NetSec Security Operations Center http://www.net-security.org/vuln.php?id=4032
- De Kermadec, Francois “A Security Primer for Mac OS X”, http://macdevcenter.com/pub/a/mac/2004/02/20/security.html

Special Thanks

- Special thanks to Dan Cheng and Marilyn Saarni for their topic suggestions
- Thanks to Gene Schultz and Jim Mellander for their support
- Thanks to the LBNL-MUG for keeping the topics hot
- And thanks to Tom DeBoni for his gracious lending of his Powerbook