

File Security, Keychains, Encryption, and More with Mac OS X (10.3+)

A Whitepaper and Guide for Teachers

by John Hendron
Instructional Technologist
johnhendron.net

April 4, 2005

When Apple chose to adopt the NeXTStep operating system for its own next-generation operating system in December of 1996, they not only got the vision of today's CEO at Apple, co-founder Steve Jobs, but also the stability and security of FreeBSD, a UNIX operating system. While this stability and security is currently being enjoyed by the millions of folks who now use versions of Mac OS X on the desktop in variations known by their cat nicknames (Jaguar and Panther, and soon, Tiger), this in no-way implies that Mac OS X super-secure out of the box, or infallible to hardware or software problems. "Super-secure" obviously isn't an industry standard or label, but what I mean by the label is something that relates to the security we might imagine in television shows like "24", "Alias," or the popular movies with Tom Cruise, "Mission Impossible." Beyond the attractiveness of having a secure desktop environment as portrayed in movies and television, Mac OS X has the potential to be a world-class, high-security environment.

This document does not replace others that can be found online, such as as the report (http://www.nsa.gov/snac/os/applemac/osx_client_final_v.1.pdf) by the U.S. Government's NSA on how to truly secure Mac OS X. Instead, it details how to gain higher levels of security with relation to files, e-mail, and more for users in public education who may find having access to a "super-secure" environment attractive as one form of their professional development with relation to technology.

This document will perhaps be of more use to these same individuals as they embark at home with online purchasing and using their personal computers for maintaining household records and finances.

I will cover the following topics in this whitepaper:

- Overview and Best Practices
- the Mac OS X Keychain
- Secure Disk Images
- Burning CDs
- USB Flash-Memory Keys
- Encrypted E-mail
- File Vault

Overview and Best Practices

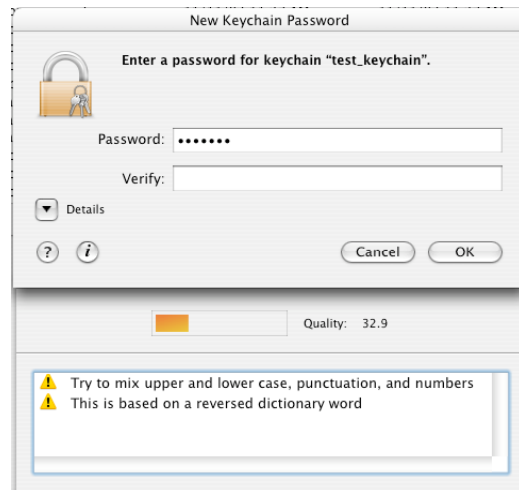
Access to a machine running OS X requires authentication. Whenever system-level changes are to be made (installation of new software, for example) an administrator password is required. This model stems from OS X's UNIX "core", a version of FreeBSD with a Mach-based system kernel. Apple calls this open-source system Darwin. This is more than we need to know, but if you read elsewhere for information relating to OS X security, you're likely to come across these terms.

Authentication is a way to protect information on the computer—obviously, a good password is important. You should always set OS X to boot into the login panel, and not "auto login" for you, when using a laptop. Also choose the login option to type in both a username and password, not a user list. All these changes are made in **System Preferences > Accounts**.

Apple proffers the following guidelines on choosing a good password:

- Use a different password for each resource you need to protect. For example, use different passwords to get your email and to log in to a network file server.
- Use a mix of uppercase and lowercase letters, numbers, punctuation, and symbols.
- Don't use any information that's easy to guess, such as your address, birth date, or child's name.
- Don't share your passwords with anyone.

Mac OS X also includes a password assistant, that analyzes passwords you create in the **Keychain Access** application. It can help you develop a good, strong password.



The passwords teachers use in Goochland are not only login passwords, but also administrator passwords. Choose wisely.

This website (<http://boredzo.fourx.org/passwordtester/>) also offers a free downloadable application (**Password Tester**) that uses the Mac OS X (10.3, as of this writing) password assistant without having to go into **Keychain Access**.

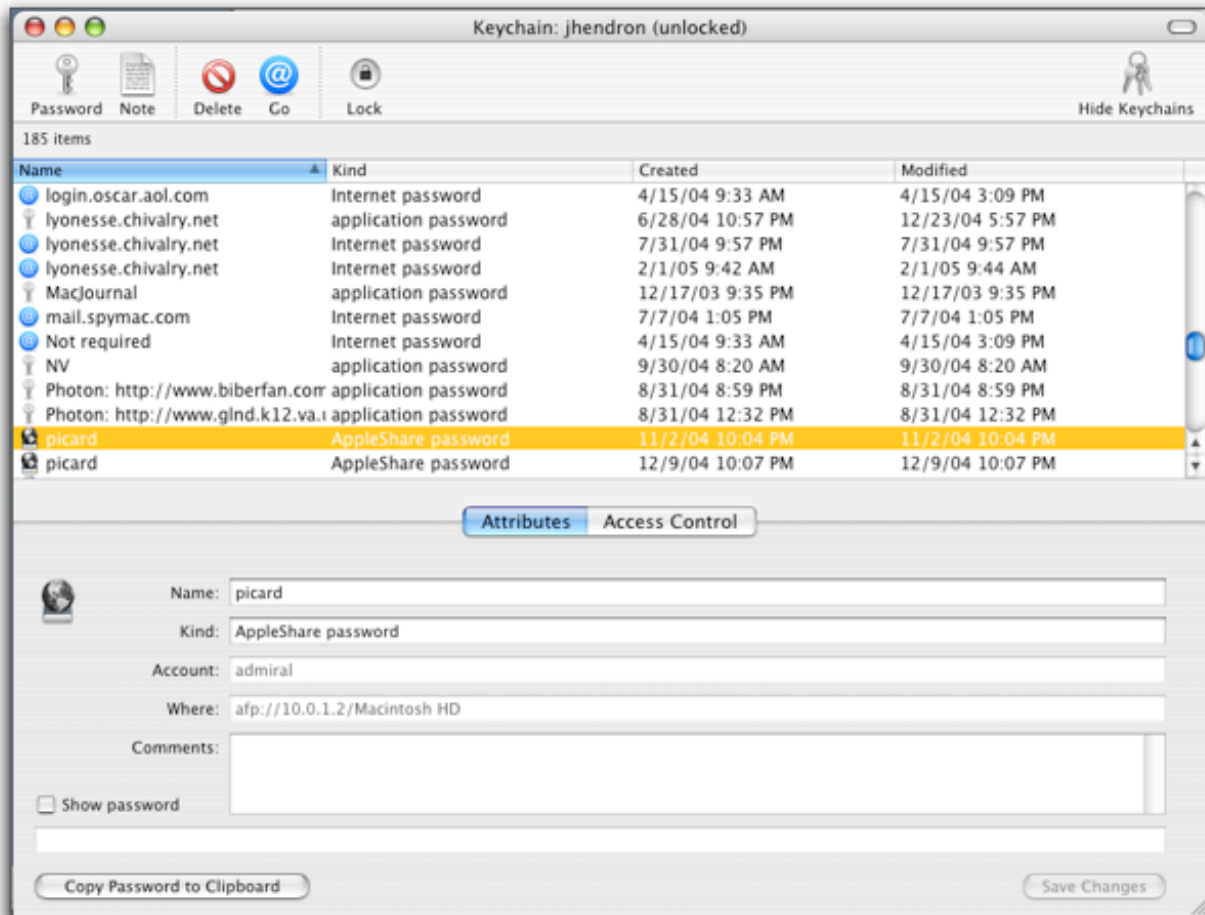
Princeton University (<http://helpdesk.princeton.edu/kb/display.plx?ID=2762>) also has some good information on passwords. One caveat with OS X is that your Administrator password can be changed with access to a bootable CD-ROM for the installation of OS X system software. We have taken measures beyond the scope of this whitepaper to prevent this. Even so, no system is 100% secure. With physical access to a machine, data can always be compromised. But choosing good passwords are a deterrent and helps keep out folks who don't know what they're doing. Most don't. Later on, we'll cover measures to help protect data that does matter from folks who do know how to bypass and compromise the security of your laptop.

Another concern nowadays, especially with the advent of publicly-available wireless Internet access (WiFi) is network security. How do you know that the information you're sending or receiving over a network isn't being monitored? Does it matter? If it does, there are ways to protect yourself. One way is by using encrypted e-mail. Another, more complex method is the use of something called a SSH tunnel. Corporations use VPNs. Yes, we're creeping into techie/geek territory again. I think the solutions offered in this paper are reasonable, easy solutions that most folks can handle.

Now that we've chosen a good, quality password for our account, we'll talk about Keychains. The keychain is an Mac OS X technology that keeps track of passwords, authentication to servers, and even information for filling-in forms in Safari. With "one" password you can "unlock" access to many. As you will see, the keychain can also store private, encrypted notes, such as web passwords, credit card numbers, etc. You won't necessarily use these on a school computer, but knowing how this service works is important for times you may wish to use it.

Keychains and Keychain Access

Keychains are managed by the application **Keychain Access**, located in /Applications/Utilities/Keychain Access. As you use your computer, you might notice a prodigious list of entries in your keychain. I currently have 185 items.



The entry highlighted is a password for connecting to my home computer. By setting up the Finder to remember my home server's address in the keychain, future attempts to login only require one key press.

Poke around your own keychain and take a look around. You may need to "unlock" the keychain with your login password before you can view the contents of the keychain entries.

Some experts suggest setting up more than one keychain. While I think this is unnecessary for our profession, it does have advantages.

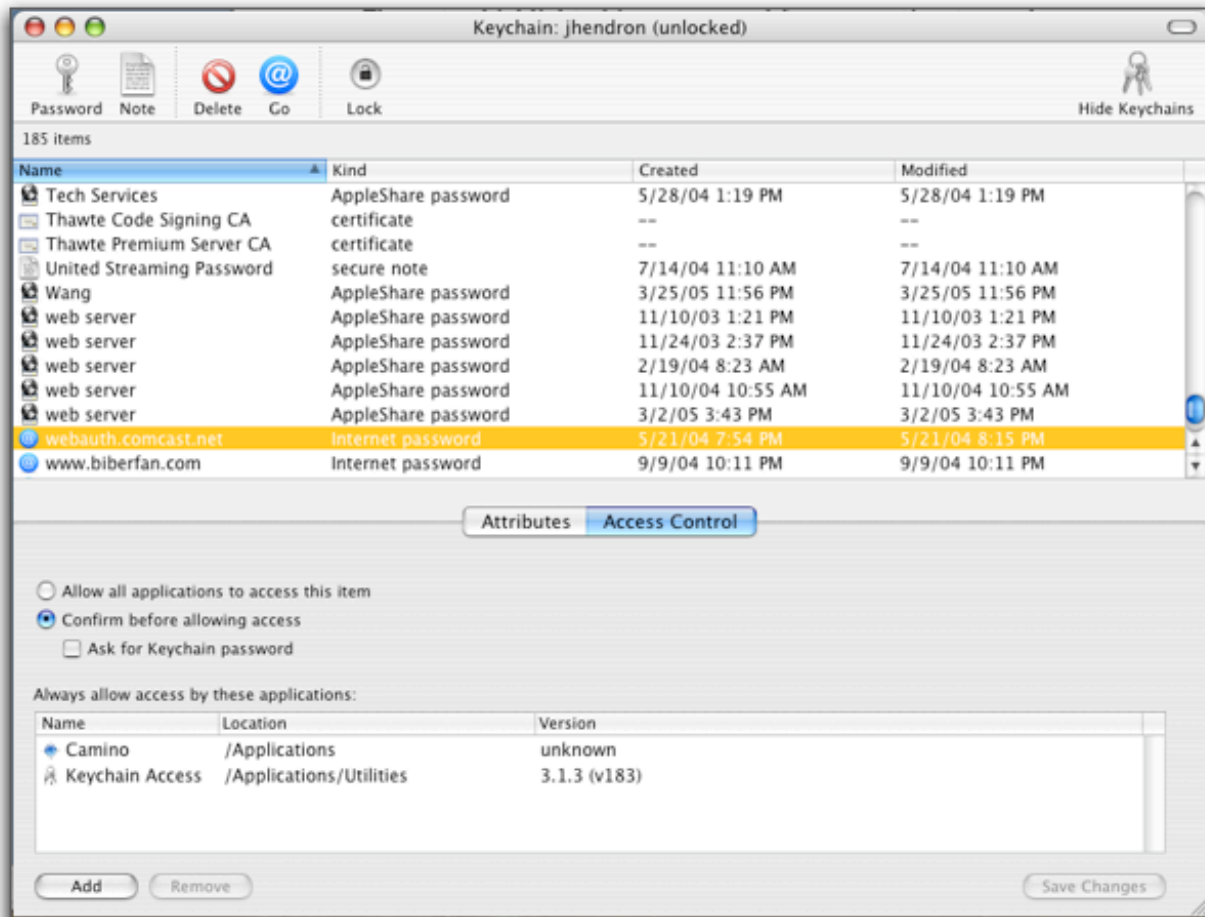
Keychain Access allows you to manage not only your main keychain (usually named the same as your login shortname), but also any other number of keychains you want to create. I could, in theory, have a keychain for personal notes (SS #, home e-mail passwords, etc.) and have another for servers at work. All keychains are, however, just files. I can carry those files off the computer.

Before we create a new keychain, take a look at some of the options available to keychain entries:



If I need to see a password, I can click on the check-box within **Keychain Access** to “show password.” The next thing I need is to authenticate—to get permission to do so. And then what? I now have access to “Access Control.” This panel allows me to say which applications can have access to my information, and if I need them to ask nicely, or just get at what they need.

In the example that follows, the Camino web browser used my keychain to store a password for getting into my Comcast e-mail through the web. Safari will do the same.



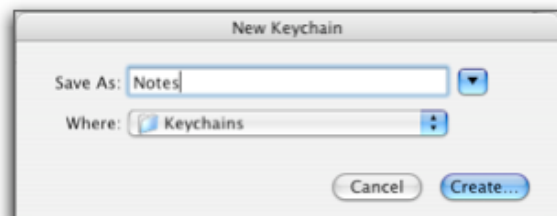
Clicking on the “Go” button in the toolbar, above, will take me to that website. I can also Delete this entry, create a secure note within this keychain, or, create a new password for my keychain. The “Hide Keychains” button will show and hide the drawer for accessing more than one keychain.

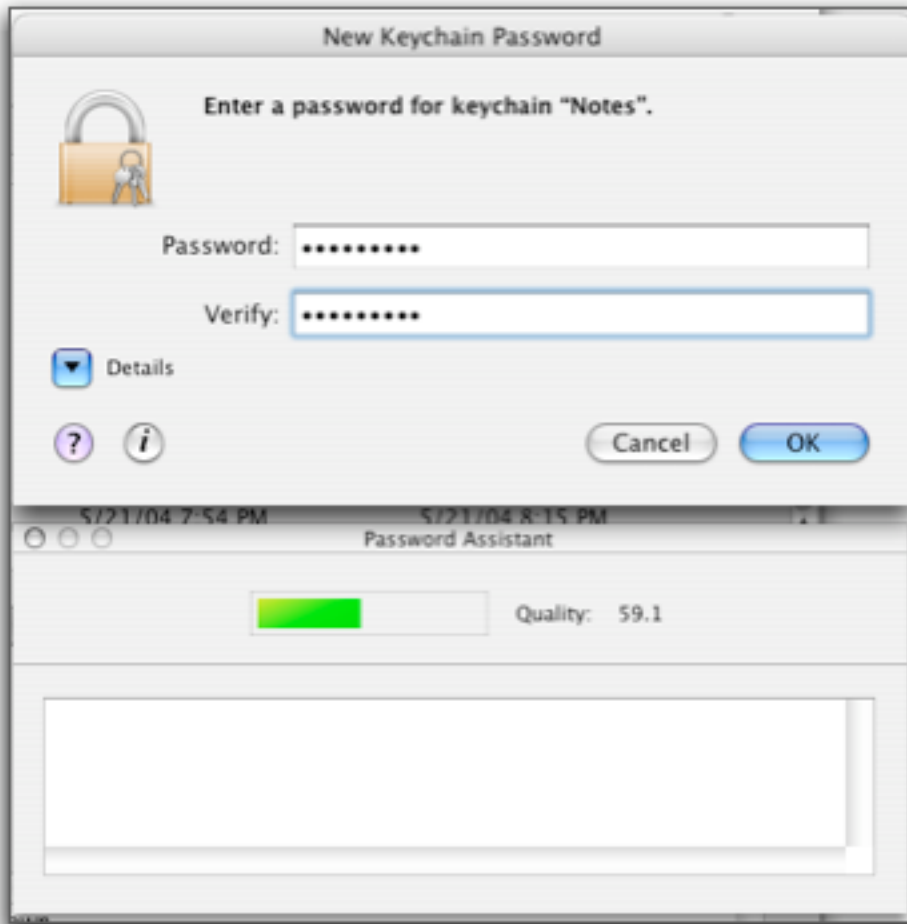
Let’s start by creating a new keychain, and then creating a secure note for it.

1. Choose New Keychain from the File menu.

2. Name the new keychain “Notes.”

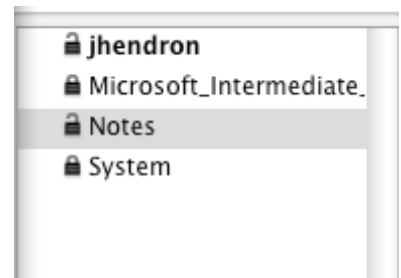
3. Choose a password for this keychain. It should be a new, secure password. Click on the “i” button in the dialog box to bring up the password assistant.



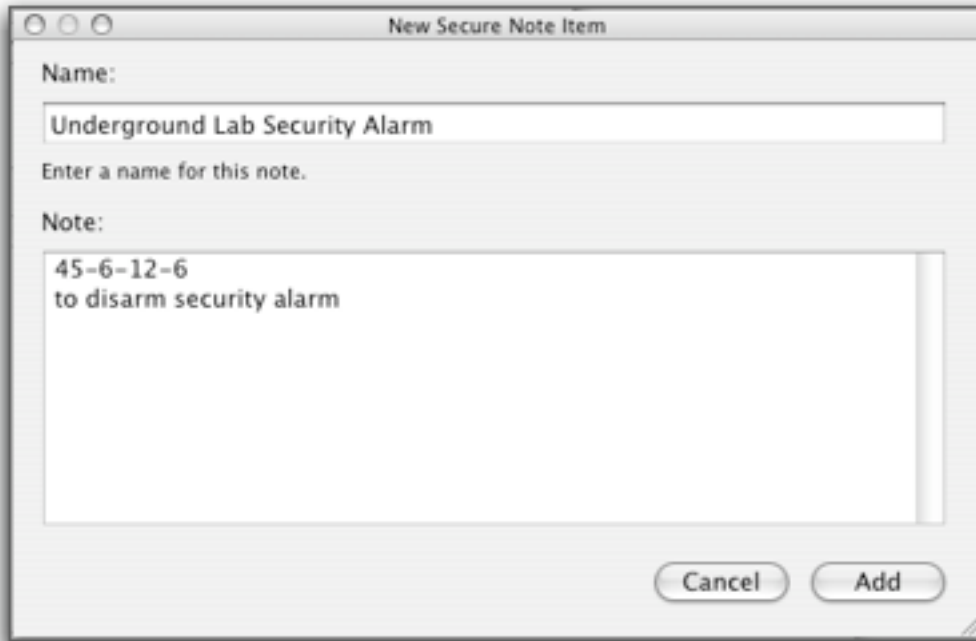


I chose a good password, but not a great one. Aim for “40” or higher. Obviously, a higher number will result in higher security. This should be obvious, but if you forget this password, anything you store in the keychain is not recoverable.

4. In the keychain drawer, choose your Notes keychain. Notice how jhendron is my main, system keychain—and is marked in bold. This is where new keychain items will go, by default.
5. In Notes, create a new note. Put in whatever you like, below is an example.



6. Next, take a look at the options for this entry in the main **Keychain Access** window.



When choosing to “Copy Note to Clipboard” (so I can paste the information, say, into an e-mail message), I am again confronted with the option to authenticate. This time, I need to supply my keychain password, not my system/login password:



I’ve also displayed here the “details.” This keychain file, located in my ~/Library folder, can be moved! I can copy it to another Macintosh, or I

can place it on a USB Flash drive. The information contained within is secure. The file is “encrypted”* with my keychain password.

Why do I mention the USB Flash drive? If you store your main keychain (or any other number) on a removable drive, your Macintosh is secure when you remove the thumb drive. In other words, you can keep all of your authentication information on the drive, and not on your laptop, proper. The USB device becomes a “key” to accessing your server shares, encrypted notes, and Internet passwords.

More information on general OS X security and Keychains can be found at the following websites, including keychain locking and adding a password to regain access to your machine after the screen saver has begun:

- <http://itinfo.mit.edu/article.php?id=6938>
- <http://www.macworld.com/2004/10/secrets/workingmac/>
- <http://www.apple.com/macosx/features/security/>
- <http://www.hmug.org/Pres/Keychain/index.php?7>

When backing up data, it’s a good idea to backup the keychains you create. I should also warn you, if you choose to store your main keychain on a USB drive, you ought to have a backup (say, on CD-ROM) because USB drives do fail. Lose the USB drive, and access to passwords entrusted to your keychain are gone!

Keychains can also develop bad habits (corruption). Visit this site to learn more, and how to download a free utility that “fixes” broken keychains: <http://8help.osu.edu/1215.html>

* I say “encrypted” because the titles of notes, and what passwords you are storing are available as “plain text” in the Keychain file. The actual passwords and note contents, are not.

Disk Images

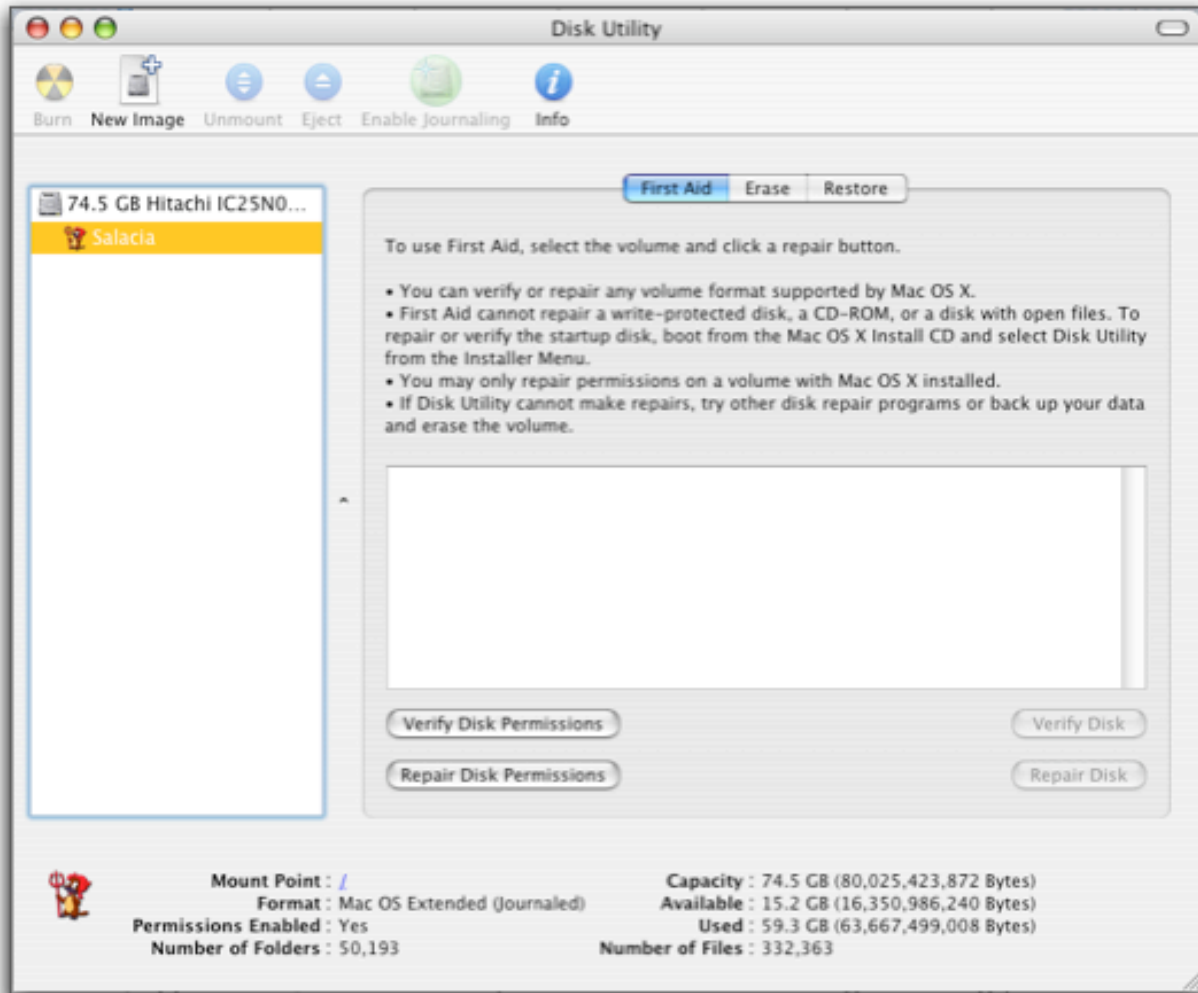
Disk images are “files” in OS X that are perceived by the file system to be “disks.” Everything in UNIX is a “file”—folders, documents, and applications. It’s not such an extension, then, that a physical, or virtual disk to the operating system can be treated, too, like a file.

There are both graphical (GUI) and command-line (CLI) ways to create disk images in OS X. Disk images are encountered (most often) when you are downloading new software. The files typically end in the file extension .dmg (disk image). Double clicking a disk image file is like “inserting” a floppy—it appears on the desktop, and files can either be written to, or copied from, the disk image. The disk image (what appears as a disk, not the .dmg file) can also be “ejected” and removed from the operating system’s /Volumes directory.

There are many reasons and needs for using disk images, but one in particular is to have a place to store confidential documents. Disk images can live on the hard disk on your laptop, but also on Flash drives, and removable hard disks. If I need to send confidential files to myself at home, I can wrap these up in a disk image, send the image, and worry less about network and Internet security. That is, if I protect the disk image.

OS X allows you to apply 128-bit encryption to a disk image by using the **Disk Utility**. Found like **Keychain Access** in your /Applications/Utilities folder, **Disk Utility** combines both disk image handling (in OS X 10.3, 10.4) and also hard disk repair utility.

In the screenshot below, I am looking at repairing permissions on my boot disk on my laptop. This is among steps one can take when OS X’s operation becomes quirky. Notice, however, the buttons for creating a “New Image” and “Burn.” Disk images can also be created as “CD” or “DVD” master files, when you want to burn several of the same data CDs or DVDs.



To create an encrypted, secure disk image, follow the following steps:

1. Click “New Image” in the toolbar.
2. Choose a place to store the image, and provide a name for the image in the sheet/dialog box.
3. Choose encryption, the size you need. I chose “sparse disk image” because these types of images can shrink/grow as you add and remove files.

Currently, only AES-128 bit compression is supported. All of these steps can also be done using the CLI. Using `hdiutil` is beyond the scope of this whitepaper.

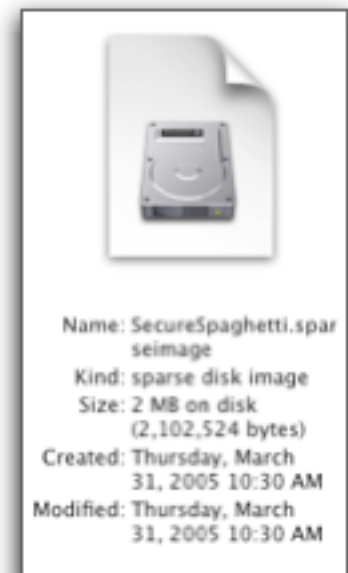


4. The next step is choosing a password for your disk image. You also have the option of storing this password in your keychain. The choice is up to you, but remember—if your keychain password is compromised, then your secure disk image password is also compromised.

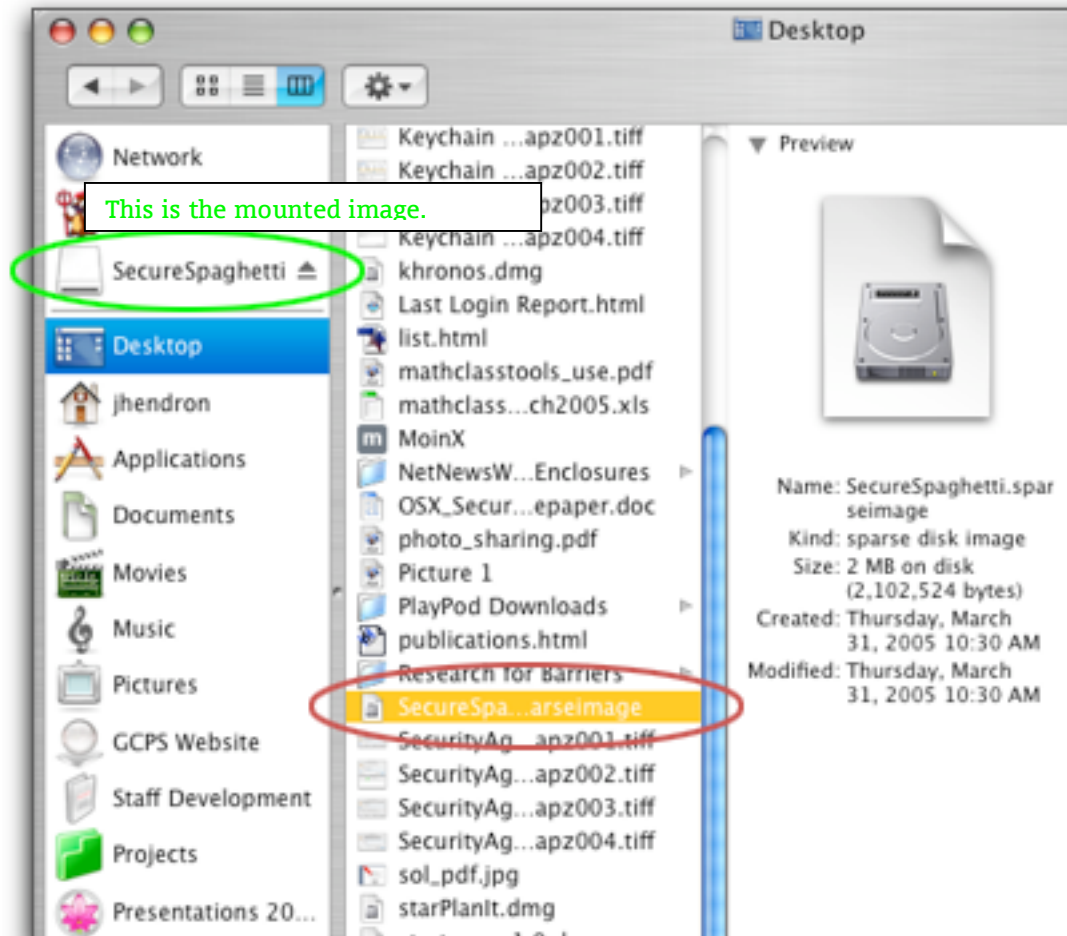


You will also notice that the extension for a sparseimage-style disk image is “.sparseimage” rather than “.dmg.”

Checking in the **Finder**, the image file is not 10 MB! Why? I chose a sparse image. I’m saving space with this format.



5. The final step is to leave **Drive Utility**, and copy files to your new disk image, that **Drive Utility** mounted for you. When you are done, and ready for “encryption,” simply un-mount (drag to trash) your mounted image. The data copied over is now secure.



A note on “unmounting” your disk image:

Make sure the “image” you are dragging to the trash is not the image file (marked here in red). This will throw your image away! Instead, drag the picture of the drive from your desktop to the trash, or simply use the “eject” icon next to the mounted image (circled in green) to unmount your encrypted, sparse disk image.

Now that you have a secure disk image, you can transport the image—and the files within—securely. One method is by e-mail, if the image is small. Other methods are to use file sharing to a server, Apple’s .Mac service, or by burning the image file to CD-ROM.

Burning Discs in Mac OS X

There are many methods for creating a read-only CD-ROM in Mac OS X. One series of methods, which create specialized discs, is to use the built-in support for burning within various applications. Three examples include:

- iTunes,
- iPhoto, and
- iDVD

The method we demonstrate here is great for burning data to a CD, in much the same way you'd do with a Zip disk, floppy, or other removable storage, such as magneto-optical or SyQuest.

1. First, ensure that your Mac can burn CDs (check the **System Profiler** application in /Applications/Utilities). If it cannot, connect a Firewire-based CD-burner drive.
2. Insert blank media into the drive. I recommend 80-minute, 700 MB media from Imation.
3. When the CD is read, it should appear in the Finder as a CD-R disk, labeled "Untitled." Drag any files to this disk you want to store. You cannot drag-over more than 650 MB of files, however. To get the full, 700 MB of storage possible, you need to use the CLI, again, outside the purview of this guide.
4. Title the CD-R anything you like by clicking on the title on the Desktop, and rename the CD. Since OS X, by default, creates PC-Mac cross-compatible discs, you may be limited to 31 characters in the filename. Mac-only discs, with longer filenames, can only be created by the CLI, or by using third-party software, such as Roxio **Toast**.
5. You can also drag over the disk image file (in our case, a .sparediskimage file) to the CD-R. This is a great way to back-up, or transfer your secured files. Make sure you don't copy the files themselves off a mounted image—this will copy the files, but without any protection.
6. When you are ready to "burn" the disk, and commit to storing the files permanently on the media, click the burn symbol (it appears to be a warning for nuclear doom) next to the CD-R in the Finder's sidebar, or click on the CD icon and choose "Burn Disc" from the File menu.

Burning and verifying a CD-R disc takes time... the length of time will depend on the size of files you have chosen to copy, and the speed of your CD-R drive.

If you chose to burn a disk image to CD, it will require mounting each time you insert the CD into a new machine. The disk image file will likely only work on Mac OS X machines. It may appear on/in a PC, but the mechanism for unlocking the disk image is handled by the Mac.

Users wishing to trade files between platforms in a secure state are best served by third party software, such as **PGP Desktop** (<http://www.pgp.com/>) from the PGP Corporation. PGP offers more encryption options other than AES at 128-bit.

If you have stored the password for the disk image in your keychain, you will either need the keychain mounted on another Macintosh, or remember the password for your disk image.



USB Flash Memory Keys/Thumb Drives

Another popular method for carrying around data, or passing data between computers are inexpensive flash-memory drives that plug into computers via USB or Firewire.

These drives are a great security risk, if your data is important. Consider using a disk image, encrypted through **Disk Utility**, if your data is sensitive. By storing the encryption password in your keychain, it's a rather seamless process to access the data on your own machine. Should the drive fall into the wrong hands, however, the data and drive are useless. You can also store both a secure image and non-secure files on the same drive.

The types of data worth securing, obviously, include:

- Any student data
 - Grades
 - Student Names
 - Student Work
 - E-mail with parents
 - Progress Reports
- Keychains
- Personal data

There are some advanced users who use the USB drive to store their main keychain, and in some cases, if large enough (2-4 GB) their entire Home folder. The last section of this report will detail the use of FileVault, a part of OS X that doesn't require such extraordinary means.

Some well-known brands of Flash-memory drives include:

- SanDisk
- Lexar
- Sony

Some vendors market "secure" versions of their drives. I don't recommend them (they usually cost extra) because the functionality for security already exists as part of OS X, and much of the software for these products is Windows-only.

Encrypted E-mail

If you're interested in the history of encryption, and need for network security, I highly recommend the easy-read by Steven Levy entitled *Crypto*, available in paperback. He provides an excellent account into the creation of public key authentication, including Zimmerman's Pretty Good Privacy (PGP).

I haven't explained what encryption is, but in very basic terms, encrypting is a way to scramble a file (application, document, disk image, etc.) or data (e-mail, virtual memory stored on hard disk) that makes the file, or data contained within, useless. The mathematics involved allow someone with a key to re-assemble the data into the meaningful version we began with. By encrypting data over a network, our information (such as credit card numbers) is useless to anyone sniffing the network as our information is en route.

Personally, I don't send too many e-mails with confidential data, but I also don't like the fact that anyone can look at my data, en route, or on a server, without my permission. Doing so may be against the law, but nevertheless, if I want or demand privacy standard e-mail won't cut it.

Step-in something called public-key encryption. Genius stuff, really. I can create on my computer a set of keys: one private, one public. These two keys relate to one another mathematically, but not in such a way that is obvious or discernable. Using my private key, I scramble (read: encrypt) my message to you. But I also use your public key: a set of characters that you share with anyone. Using the private code for my message, with the public part of yours, only the two of us can read the information. Anyone else on the Internet or our local networks who sniffs-out our message is lost: they may have your "half" of the code (your public key), but they don't have my private key. When you return the message, it is encrypted using your private key, and my public key.

There are a couple of different solutions for using **Mail** in Mac OS X with public/private key encryption. I personally use the pay-for version of PGP Corporation's PGP product at home. On my laptop for work, I have used their free version. It relatively easy to use.

First, download the PGP software:

<http://www.pgp.com/downloads/freeware/index.html>

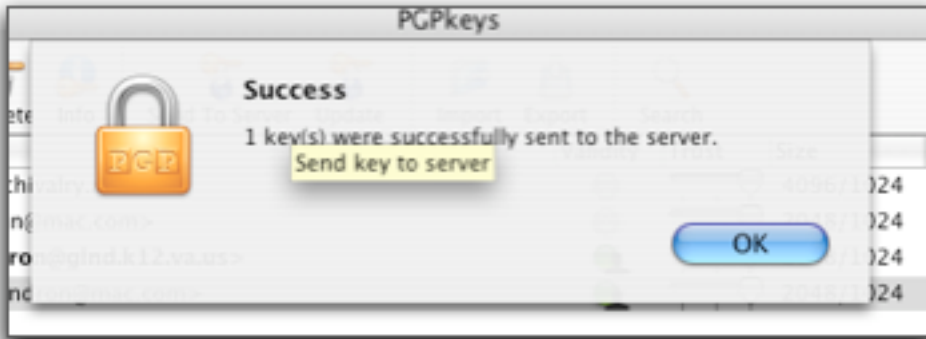
The current version for Macintosh is version 8.1.



Next, we will generate our keys. After installation, run the PGP application in your /Applications folder.

1. Go to Window > PGPKeys
2. You will click "New Key."
3. The assistant will load. Follow the directions to generate your key pair. As with your OS X password, and keychains, a good-quality password must be used.
4. Once the process is complete, you can send your public key to a key server so others can "find you" for sending encrypted e-mail messages.





You should also know that the keys generated with this program can be used in other implementations of PGP and on other machines. I highly recommend backing-up your private key to a CD-ROM, and keeping it in a safe place. Should your laptop fail, your key will be gone, and any e-mail encrypted for you cannot be accessed without your keypair.

The “Export” option in PGPKeys can be used to backup your keys. Never (never, ever) share your private key with others, or leave it vulnerable. When doing so, the security model for public-key authentication fails.

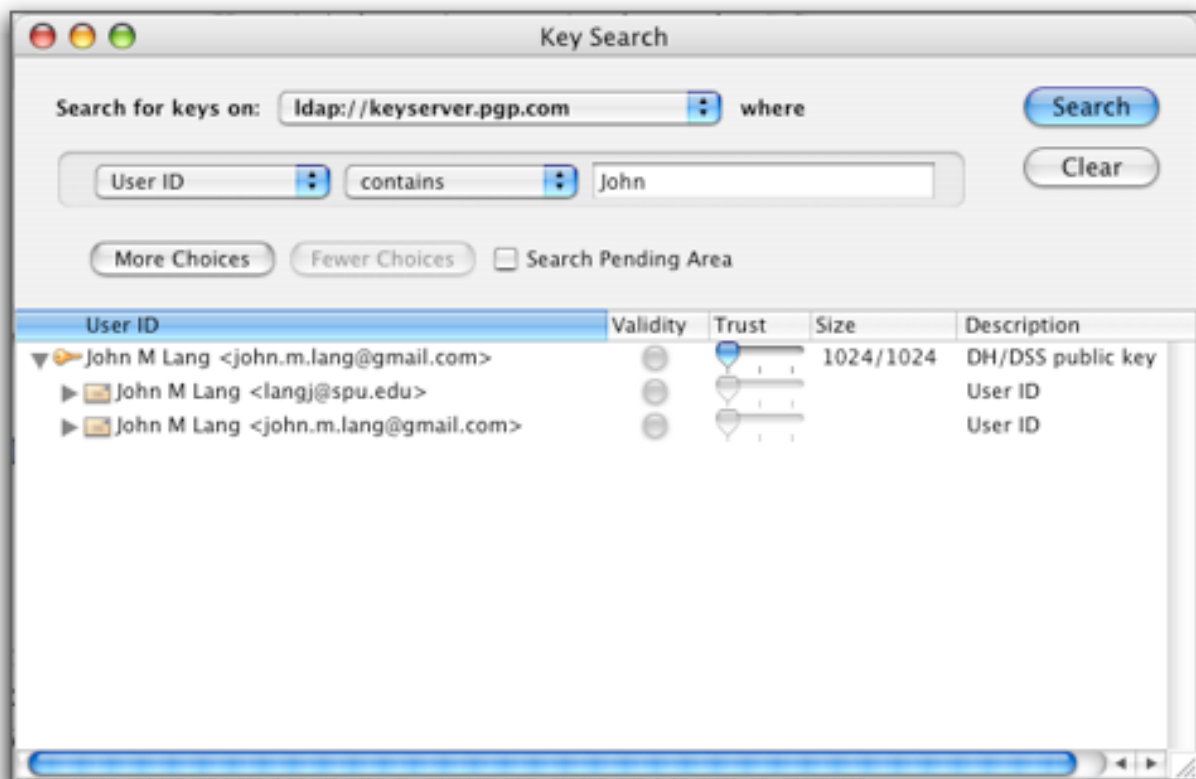
I exported by public key (all it is, really, is a text file), and here it is:

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: PGP 8.1 - not licensed for commercial use: www.pgp.com

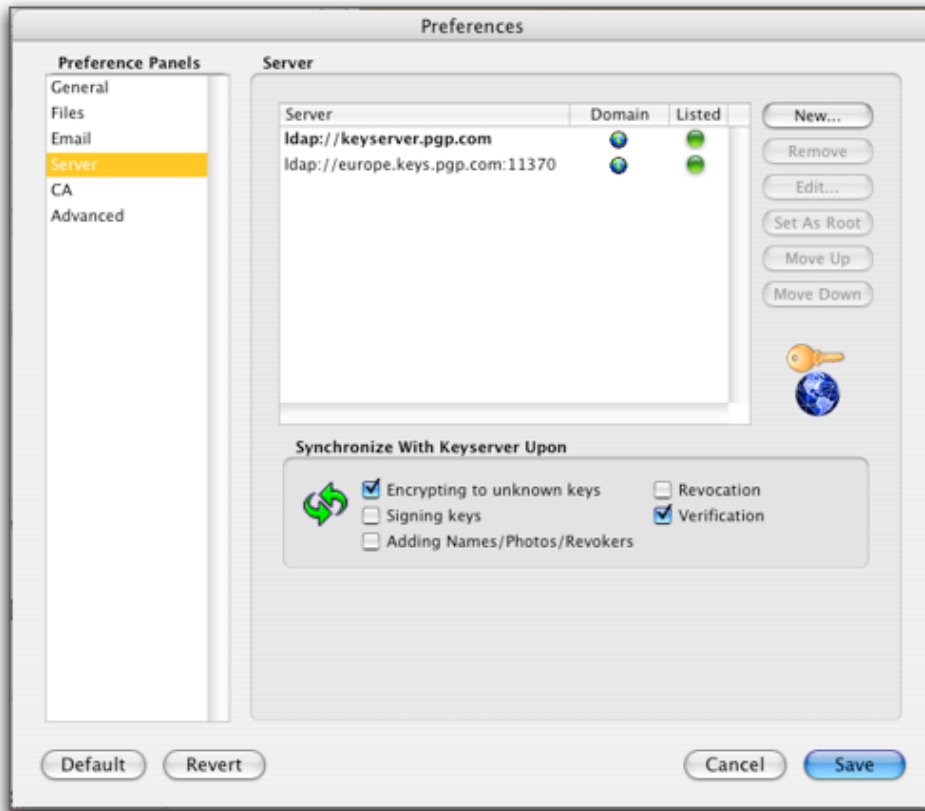
mQGiBEJMJzgrBADlS8NWjGnFyzfCKkAS9GXphOpXlNvdMc8u0sdsOovwRu6dIro9
nVbQA/FWJyOR5ed65PvJ4oJhtr6jURjntzXxseWfaoolpC/FFx+BvBzn8S1jFk4Q
WUtTNG3zuVFz3FCdIkt4jTaVMvwyCra7pGlulU5UhU1VesogonRcMZBNwCg/xI0
RGjo8lowSH+OwHDqrJUREOkD/2w+LDmBt1BNAovfYB9RGqyotfEbrTPBwIUnJ+Gs
TsdvZhyiVneKxp/tkCweV5wVnlddzc7o0AeYidzOU+zyEfWvDbHu73X8EB/Twen6
HGI2TdheiqIXISv5ctk+0tCH6saTt4Eu7ga5ythLvJHK2rwdjXUu11f8C3XLoqKR
DcaLBACwdf+VWP2nvlP6JieiX+ey6VS8Vu5IUH3A0KWWyPdDkVFF9QXpwLdy+9u7
wrg9szrGgQWuQNhwdDfozqSOE4thRPOq89TSwdumnoZX9wyOswA8sUva6Hix6ee2
TzFEYybJkZIXFXP2lwbw5+n2YVo8G0+/xxi67Evx7732SI8slbQISm9obiBIZW5k
cm9uIDxqb2huaGvuZHJvbkbBtYWMuY29tPokAXQQQEIAHQUCQkwnOAcLCQgHAWIK
AhkBBrSDAAAAABR4BAAAAAAAJEALGT6Bax6hXVZgAoN3CQbPC/ZgxPO49a3F1BN4g
fAFGAKDmf2s64ww5deItSiKtZDZglUki7bkCDQRCtCc4EAgA9kJXtwh/CBdyorrW
qULzBej5UxE5T7bxbrrLLOCdaAadWoxTpj0BV89AHxstDqZSt90xkhkn4DIO9ZekX
1KHTUPj1WV/cdlJPPT2N286Z4VeSWc39uK50T8X8dryDxUcwYc58yWb/Ffm7/ZFe
xwGq0luejaCjcrUGvC/RgBYK+X0iPlYTknbzSC0neSRBzZrM2w4DUUdD3yIsxx8
Wy209vPJI8BD8KVbGI2OulWMuF040zt9fBdXQ6MdGGzeMyEstSr/POGxKUAYEY18
hKcKctaGxAMZyAcpeSqVDmWn6vQC1CbAkbtCD1mpF1Bn5x8vYLLhkmquixsNV
6TILowACAgf/WkuZJxdzOe6TlUefRmOWbz0VKYohW5n/Vv1bpz3SxWEzRvHqFWzr
iXSKMrUERxsMaCtjvespEhl5KI9dItfbFcDAbcfafuPsm1Fz9RAfsT1Sx1HD849R
ZXKA0YmQZnTv5ByISBHibgOpq4KMGcO145w7LbdW+ftEk9UH3AgPx/eZjHX50HGL
bxqWMFd5nVVTSGp0rYTSpb2/mlhMA6kxJ36abwMhbojfw+p673Vrbv5w1j8WEhj
5zSmLDZzsUXJZbyqArwrkUiNzPLXnIzuOj4UTEUG67itRLmwCSR8vrRZoFwaA5sn
T+6UDlNo/KABKw4cQUs3umb+tBEC438Wo4kATAQYEQIADAUCQkwnOAuBDAaaaaak
CRACxk+gWseoV5Y8AKCTYX9dQZpywbfbk600q0p70/mrGbwCfXzxFOBx+anPEJQcY
2y6Won2Wrt0=
=5ZGT
-----END PGP PUBLIC KEY BLOCK-----
```

To send me encrypted e-mails, you either need to import that into your PGPKeys window (either by copy-pasting that block of text into a file, or by getting my little text file), so that you can encode your message to me. For each person you wish to send e-mail to, you need their public key! Their keys are managed by the PGPKeys window, just like your own.

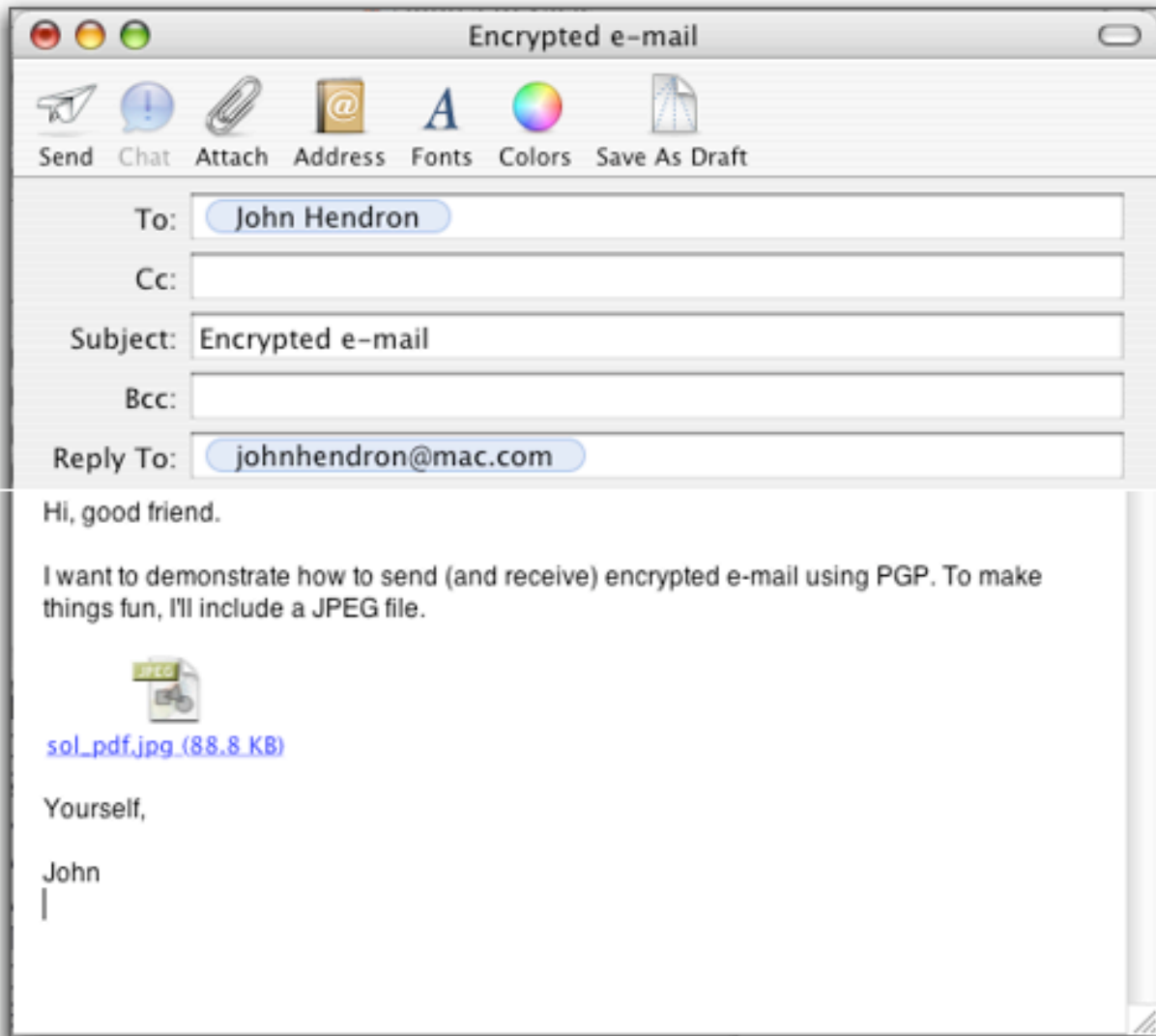
Using a key server, then, makes a lot of sense. I can request keys from the server without having to bother my associates. When creating this brief, I had some trouble with keys I sent to PGP's default keyserver. I couldn't yet find my own key:



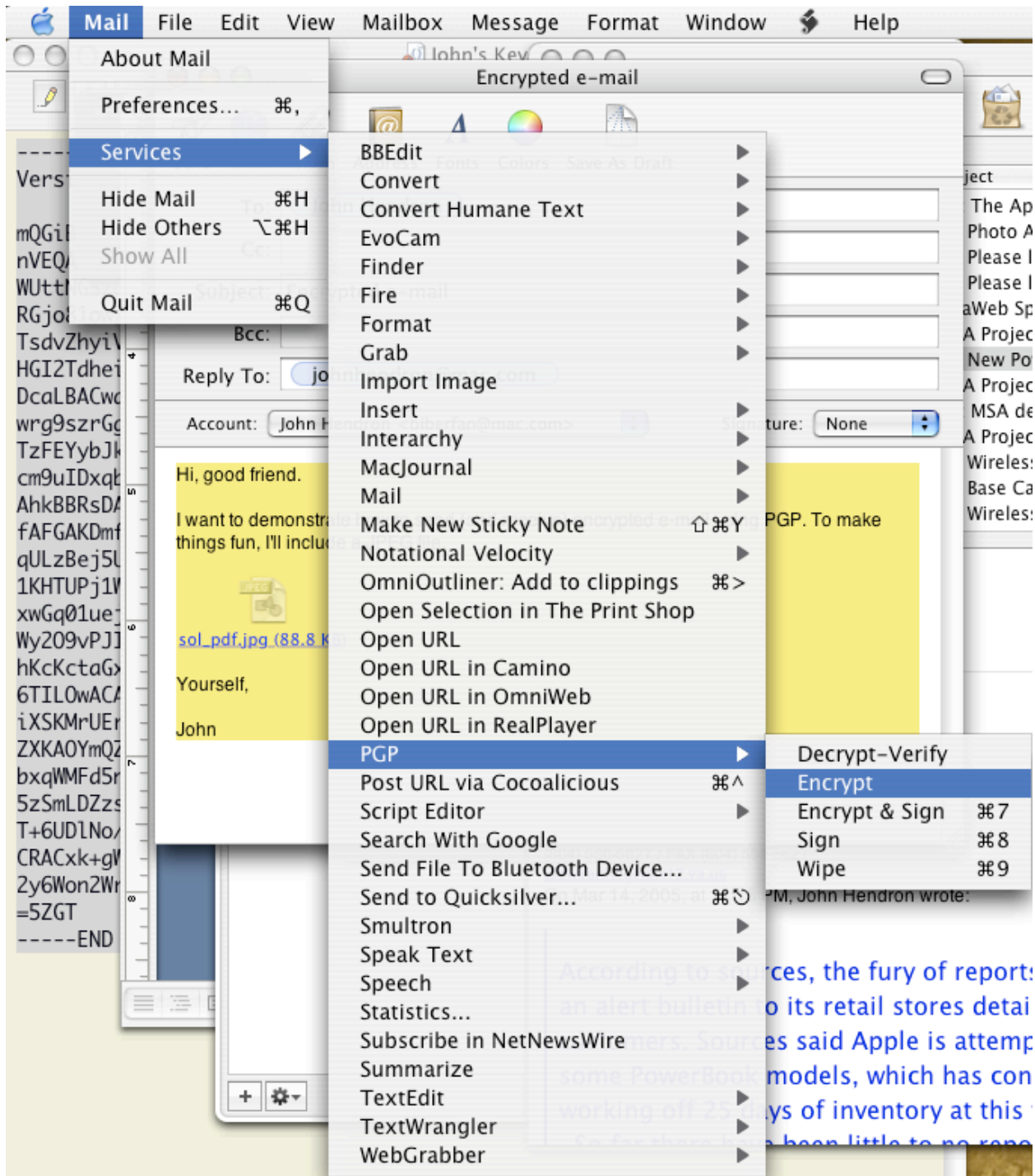
But you can add more keyserver (if you know their URIs) through preferences:



So, to start the e-mail process once my keys have been created, I go into **Mail**, and compose my letter. PGP runs as an OS X Service, so that's how we'll use it. If I am using my paid version of **PGP Desktop** at home, there's actually a PGP button that appears within **Mail**, which admittedly, is kind of nice. In this instance, I'm sending myself an e-mail from my personal account (johnhendron@mac.com) to my school account (jhendron@glnd.k12.va.us). Both keys are already defined in my PGPKeys window in **PGP**.



Next, highlight the body of your message, and access the **Services** menu in OS X; choose PGP, Encrypt.



For me, PGP didn't like my JPEG file. I took it out. PGP will call-up a dialog asking for the recipient. Choose the recipient(s) whom the message is for—this is using their public key. If you don't see names, go back to PGP and add those public keys to your list.

This is what PGP generated, next:

-----BEGIN PGP MESSAGE-----

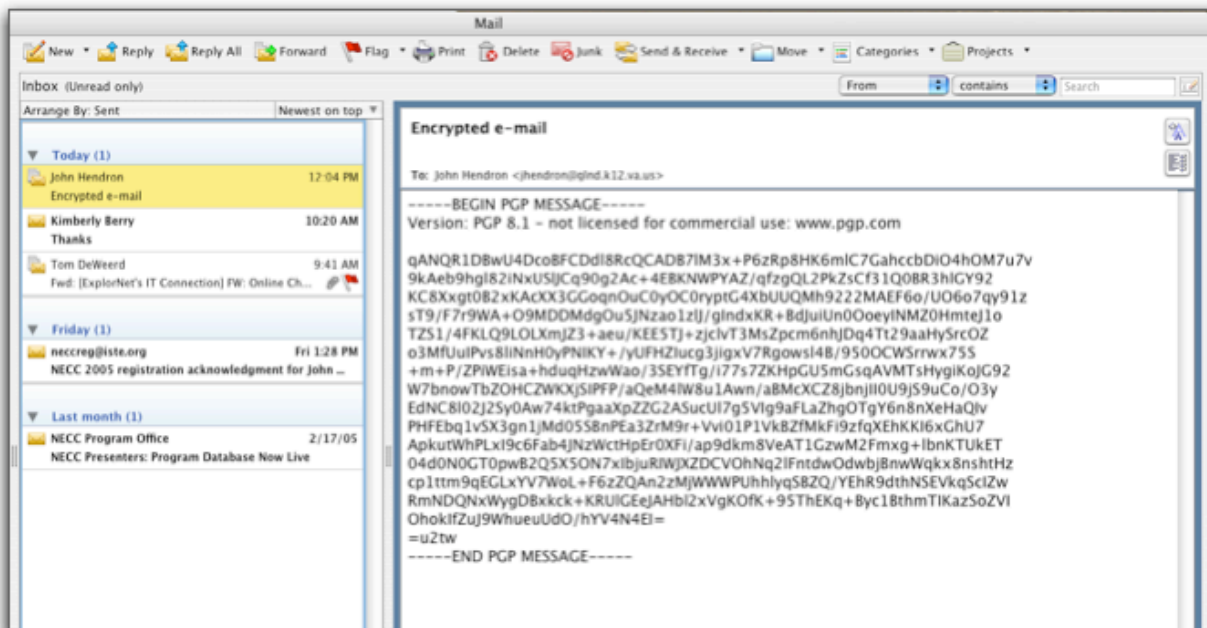
Version: PGP 8.1 - not licensed for commercial use: www.pgp.com

```
qANQR1DBwU4DcoBFCDDl8RcQCADB7lM3x+P6zRp8HK6mIC7GahccbDiO4hOM7u7v
9kAeb9hg182iNnXUSlJCq90g2Ac+4EBKNWPYAZ/qfzqQL2PkZsCf3lQ0BR3h1GY92
KC8Xxgt0B2xKACXX3GGoqnOuC0yOC0ryptG4XbUUQMh9222MAEF6o/UO6o7qy91z
sT9/F7r9WA+O9MDDMdG0u5JNzao1zlj/gIndxKR+BdJuiUn0OoeyINMZOHmteJlO
TZS1/4FKLQ9LOLXmJZ3+aeu/KEE5TJ+zjclvT3MsZpcm6nhJDq4Tt29aaHySrcOZ
o3MfUuIPvs8liNnH0yPNIKY+/yUFHziucg3jigxV7Rgows14B/950OCWSrrwx75S
+m+P/ZPiWEisa+hduqHzwWao/3SEYfTg/i77s7ZKHpGU5mGsqAVMTsHygiKoJG92
W7bnowTbZOHcZWKXjSIPFP/aQeM4lW8u1Awn/aBMcXCZ8jbnjII0U9jS9uCo/O3y
EdNC8l02J2Sy0Aw74ktPgaaXpZG2ASucU17g5VIg9aFLaZhgoTgyY6n8nXeHaQIv
PHFEbqlvSX3gn1jMd05SBnPEa3ZrM9r+Vvi01P1VkBZfMkFi9zfQXhKKI6xGhU7
ApkutWhPLxI9c6Fab4JNzWctHpEr0XFi/ap9dkm8VeAT1GzwM2Fmxg+1bnKTUkET
04d0N0GT0pwB2Q5X5ON7xIbjuRIWJXZDCVOhNq2lFntdwOdwbjBnwWqkx8nshtHz
cp1ttm9qEGLxYV7WoL+F6zZQAn2zMjWWPUhhlyqSBZQ/YEhR9dthNSEVkgScIZw
RmNDQNXWygDBxkck+KRUIGEeJAHbl2xVgKOfK+95ThEKq+Byc1BthmTIKazSoZVI
OhokIfZuJ9WhueuUdO/hYV4N4EI=
=u2tw
```

-----END PGP MESSAGE-----

This nonsense is now ready for sending!

Since I use Entourage for e-mail at work, look what appears in my Entourage box:



Like in **Mail**, I highlight the message, go to the **Services** menu, and this time choose *Decrypt & Verify*. This does two things: it verifies who the message came from, using a digital signature, and also unscrambles the message. You can also choose, using **PGP**, just to digitally sign messages without encryption. To decrypt, you may need to provide your PGP private key password.

PGP is not something you may use on a day-to-day basis. The procedure used for encrypting and decrypting messages is enough of a hassle to encourage folks to buy the commercial version. Yet, it still works with the free version, and when you need to use it, it's there for the using.

The more advanced user may wish to explore **OpenPGP** solutions for Mac OS X. The following pages might be of help:

- http://fiatlux.zeitform.info/en/instructions/pgp_macosx.html
- <http://macgpg.sourceforge.net/>
- [http://www.gnupg.org/\(en\)/related_software/frontends.html](http://www.gnupg.org/(en)/related_software/frontends.html)

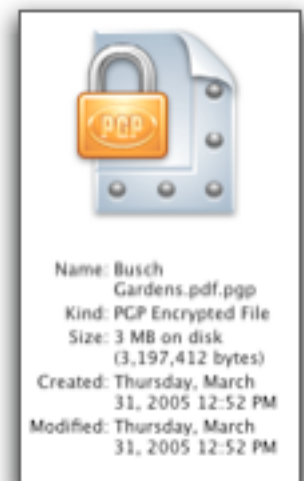
The **PGP Desktop** application and other command-line versions of PGP (OpenPGP) allow you to encrypt individual files (as opposed to full disk images), securely wipe disks, and create PGP-key enabled disk images.

You can use the **PGP** application that comes with the free version of **PGPmail** to encrypt individual files in Mac OS X using a keypair. You must list a recipient when doing so. If it is for your own personal use, choose yourself from the list by double-clicking your name and associated e-mail address.

1. In **PGPmail**, click on Encrypt.
2. Locate the file you wish to encrypt.
3. Choose the recipient.

To open (decrypt) the file, **PGP** must be on your system. Double-click the file.

Enter your private key passphrase to unlock the file. PGP does not destroy or eliminate the original file. It also does not replace the PGP version of the file upon unlocking (decrypting) it. The file you encrypt can be embedded into a mail message to send encrypted attachments.



FileVault (Mac OS X 10.3.1, 10.4)

FileVault is Apple's answer to providing an easy solution for mobile (laptop) users who care about security. FileVault adds AES/128-bit encryption to your laptop in a pretty transparent way. It automatically creates a sparse disk image out of your home folder, and decrypts it each time you login. The only way you can really tell a difference with a FileVault-enabled user account is by the icon in the Finder: instead of a home folder, you'll see a home with a padlock on it.

Apple's website on FileVault gives a good overview of the technology and level of security at play (<http://www.apple.com/macosx/features/filevault/>). To quote them, on the strength of AES security:

AES gives you 3.4×10^{38} possible 128-bit keys. In comparison, the Digital Encryption Standard (DES) keys are a mere 56 bits long, which means there are approximately 7.2×10^{16} possible DES keys. Thus, there are on the order of 10^{21} times more possible AES 128-bit keys than DES 56-bit keys. Assuming that one could build a machine that could recover a DES key in a second, it would take that machine approximately 149 trillion years to crack a 128-bit AES key.

This also means that if you forget the passcode you set for FileVault, everything on your computer (read: everything for the user account created with FileVault, not necessarily Applications, other user accounts, etc.) is worthless to you. If you know your passwords, however, it's a nice level of protection.

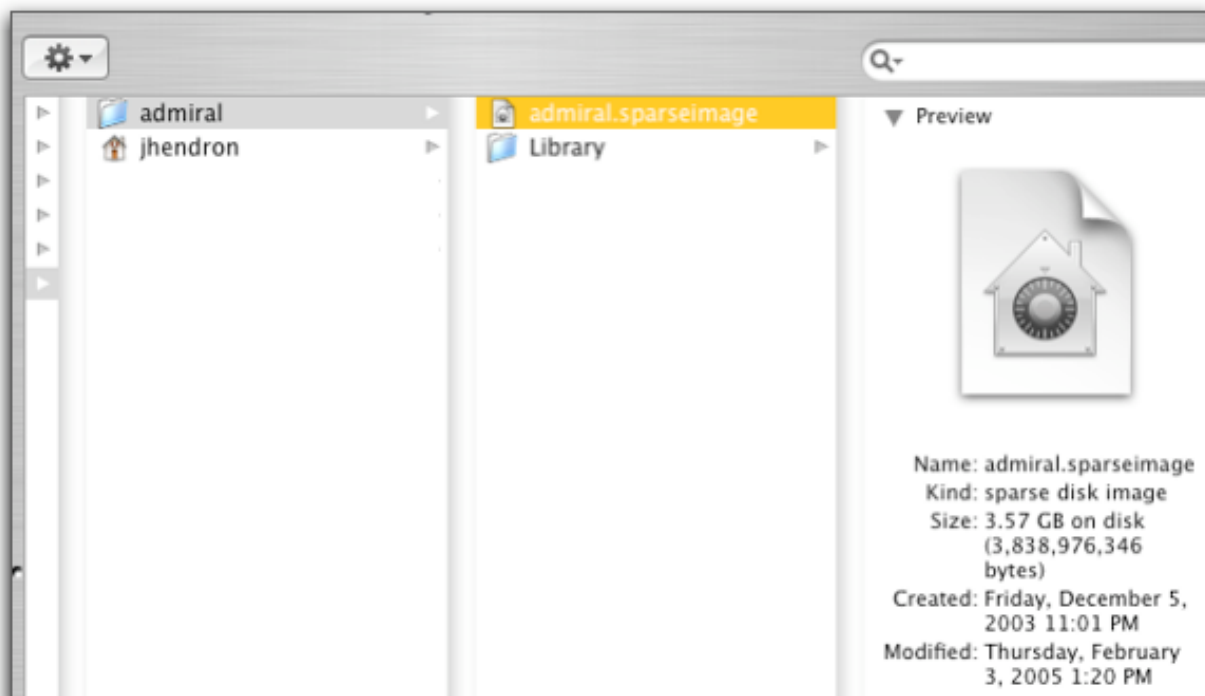
Apple's added an extra level of protection for FileVault. Check out the area within Accounts from System Preferences, where we enable FileVault—you are asked to set up two passwords.



One password is automatic—it's the already established password for your user account. The second, a

“master password” will be applied to every account using FileVault on the computer. This master password can be used to open an encrypted home folder for any user who creates a FileVault-enabled account.

On a school machine, we will require you to let us set this password. If you are using Mac OS X on a home machine, choose a strong password that you keep in a safe place (not on the machine itself). If you forget both the master password and the user account password, say goodbye to any data belonging to that account.



This is what setup of a FileVault user looks like on my laptop. *jhendron* is my normal user account. The *admiral* account is one I set up using FileVault. To other users on the computer (or a thief), all they will see of your home folder is a sparse image file. You can of course open the disk image from another account, but you’ll need to know the passwords to do so. File permissions in OS X also restrict non-admin users from removing the image, as well.

Limitations

FileVault does put a slight performance hit on your system, when used. Documents saved and opened from within the account are encrypted and decrypted on the fly. There have also been accounts of

incompatibility with some software applications and FileVault. Prime examples include software that require fast read/write access to the drive, such as iMovie.

One website recommends a complete backup before turning-on FileVault:

<http://www.macattorney.com/panther.html#Anchor-FileVault-22658>

If you want to see what working with totally-encrypted data is like, create a new account on your system, and enable FileVault. That's exactly what I did (and why there are two accounts).

I personally don't use FileVault for day-to-day tasks, but I am considering its use when I adopt Tiger (as of this writing, Tiger has not yet been released by Apple). If I do switch for full-time use, I will still have to come up with another solution when doing video editing (FinalCut Pro), music composition (GarageBand), etc.

In the end, FileVault is a pretty easy, automatic way to lock-down the data on your computer and provide security. For teachers, it's not a bad idea to use if you do (and you most likely do) have student data on your computer. I have used my FileVault account since 10.3.2, without problems (there did exist one in 10.3.0). For the user who wants security without fuss, it's one great solution. For someone like myself, who has concerns about security and doesn't mind spot-managing these issues, some of the other solutions outlined in this report may work best.

Conclusion

As we move well-into the information age, security and privacy will only continue to be important issues for users of information technology. OS X provides strong support on many fronts for access and data security.

One issue that has not been discussed at length is network security. Our discussion has reached solutions that encapsulate different data files (e-mail, disk images, and individual documents), but sitting on a public network opens all traffic to vulnerabilities.

If you manage in your home a wireless network, or take your laptop to conduct personal or school business outside of your school building, please be aware of the security risks when using other networks. Turn file and web sharing off when they are not required. For “airtight” security when working in un-trusted environments, use of a VPN or SSH tunnel is required. If you’re setting up a wireless network in the home, be sure and investigate all the features of your router—and the different levels of protection afforded through WEP and WPA. These issues are another chapter, outside the purview of this guide.

If some of the topics covered in this guide are new to your vocabulary, I hope they were helpful. Operating with security in place gives us peace of mind and helps protect the value and privacy of the important work we do.



John Hendron

johnhendron@gmail.com

This work is protected by a Creative Commons Attribution-NonCommercial-ShareAlike 2.0 license. Details can be found online at: <http://creativecommons.org/licenses/by-nc-sa/2.0/legalcode>