File Encryption

Steven M. Bellovin

https://www.cs.columbia.edu/~smb

Why Encrypt Files?

- Theft of files
- Theft of media
- Theft of computer
- Cloud storage? I.e. Someone else's computer

Issues with File Encryption

- Suppose we want to use crypto to protect files. Now what?
- What to encrypt?
- Where should keys be stored?
- What is the tradeoff between availability and confidentiality?

Bad Reasons and Good

- Is there a flaw in the operating system's protection mechanisms? Why can't the OS keep bad guys from the file?
- You don't trust the system administrator? Can the sysadmin steal the decryption key?

✓ The files are on a laptop, which might be stolen

✓ The files are on removable media (CD, flash drive, etc.)

✓ Avoid concerns when discarding drives

✓ Cloud-based file system?

Limitations of File or Disk Encryption

- Doesn't protect against on-machine threats, including malware
- How are keys stored or entered?
- What happens if you lose the key?

How Do You Enter a Key?

- Type it in to a window?
  - Malware can sniff the keystrokes
- External keypad
  - Where would you put one on a laptop?
  - Not much room on a flash disk…
  - Can or will users type long-enough keys?
- Your host's key store?
  - What if your host is compromised or seized?
- External smart card or equivalent? TPM?

Lost Keys

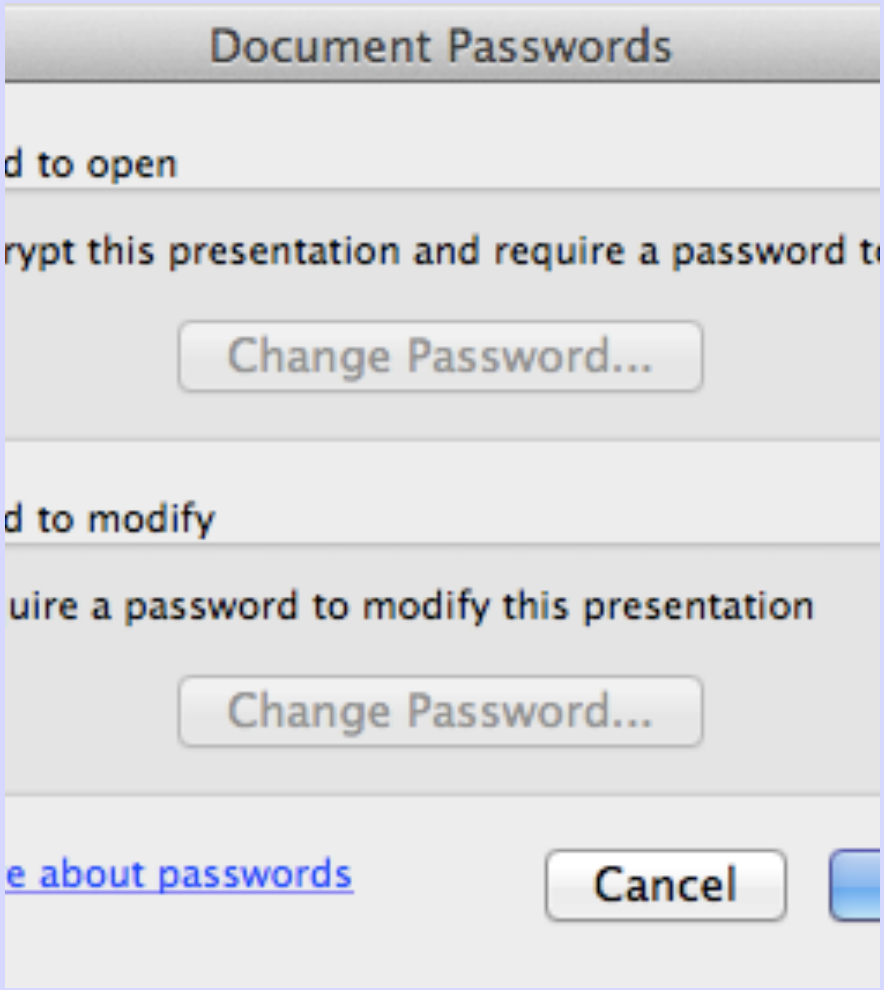
- If you lose the key (i.e., forget the passphrase), you lose access to the encrypted files
- For communication encryption, you can restart the session
- This is "data at rest" (also known as "object encryption"); there's no negotiation of a key, and no way to restart
- You *must* have some form of key backup or key recovery
  - For corporate use, there is often an administrator key

Encryption Options

- File at a time, perhaps manually
- A file system tree
- A disk or disk image
- Disk hardware

File at a Time

- Very fine-grained encryption: protect only what needs protecting
- Easy to use different keys for different data
- But…
  - Must *remember* to encrypt the files
  - Not all applications have built-in support
  - Easy to forget your key



Encrypted Directory

- Encrypt any subtree
  - Best choice on Linux
- File sizes show through; length of file names show, also
  - The equivalent of traffic analysis?
- Advantages
  - Easy to do fine-grained keying
  - Doesn't waste space in disk images

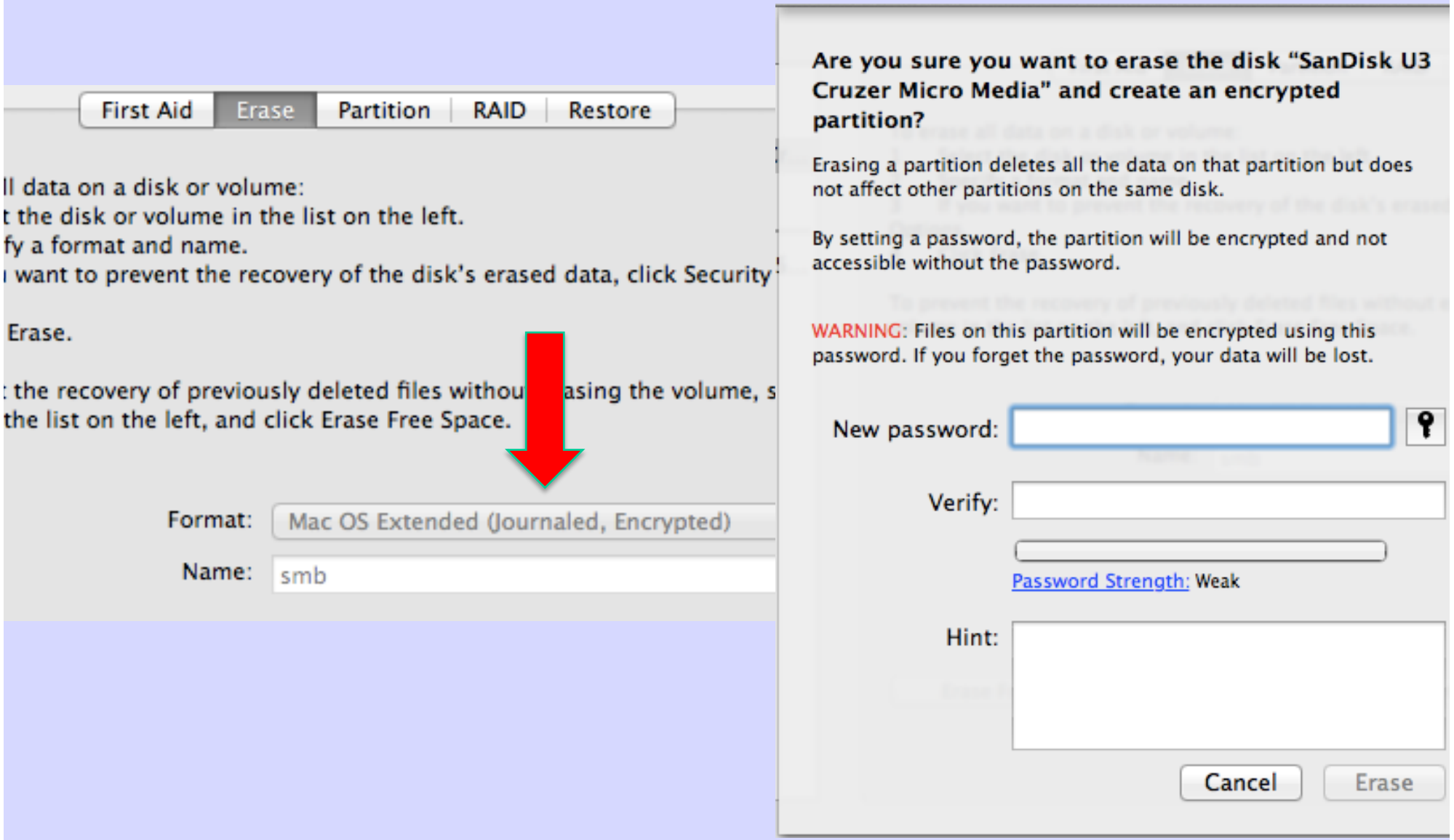
Encrypted Disk or Disk Image

- The most popular today
- No need for encryption options in every program
- No need for the user to remember to encrypt
- Hides file sizes and other metadata
- But: only one key per partition; all users share that key

Hardware Options

- Some flash drives and hard drives do encryption in hardware
  - Even for desktops, eliminates need to erase disk before discarding
- Check your vendor carefully; some have done it wrong: http://www.zdnet.com/blog/hardware/encryption-busted-on-nist-certified-kingston-sandisk-and-verbatim-usb-flash-drives/6655
- Actual disk key is usually randomly generated; user-supplied pass phrase is used to encrypt the key

Encrypted Disks: Mac OS

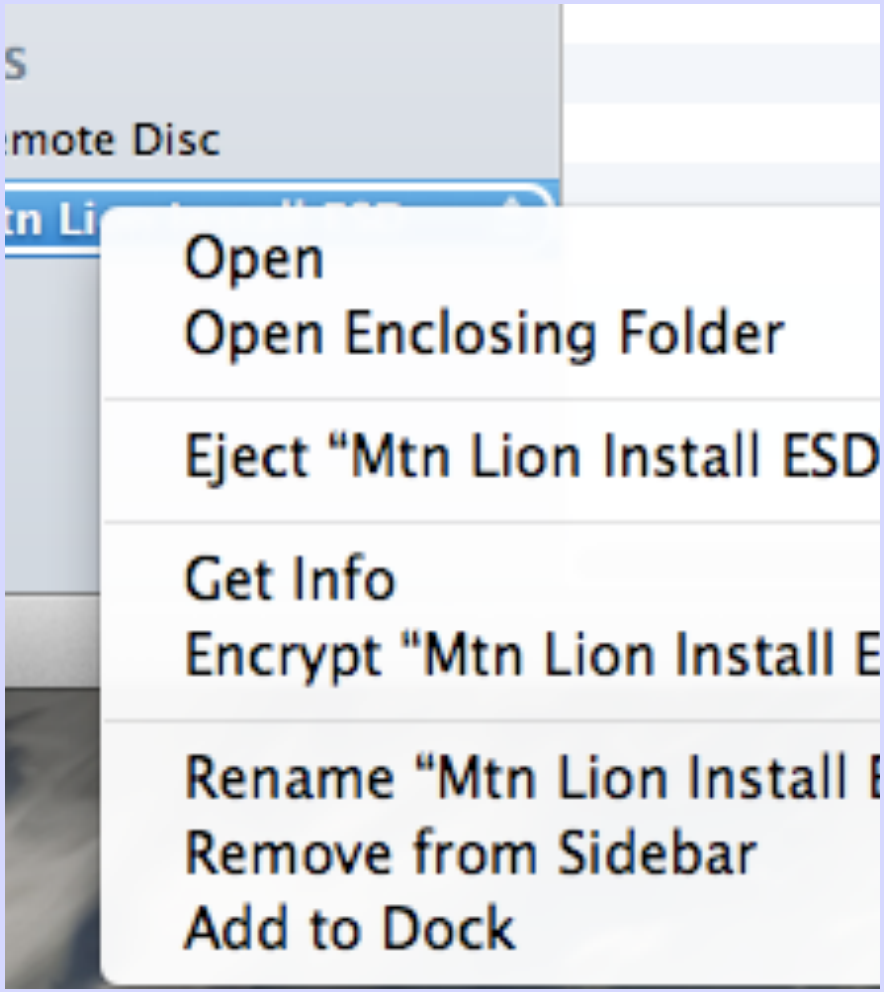


Encrypting a Disk on Existing Mac OS

When you erase the drive, select "Encrypted"

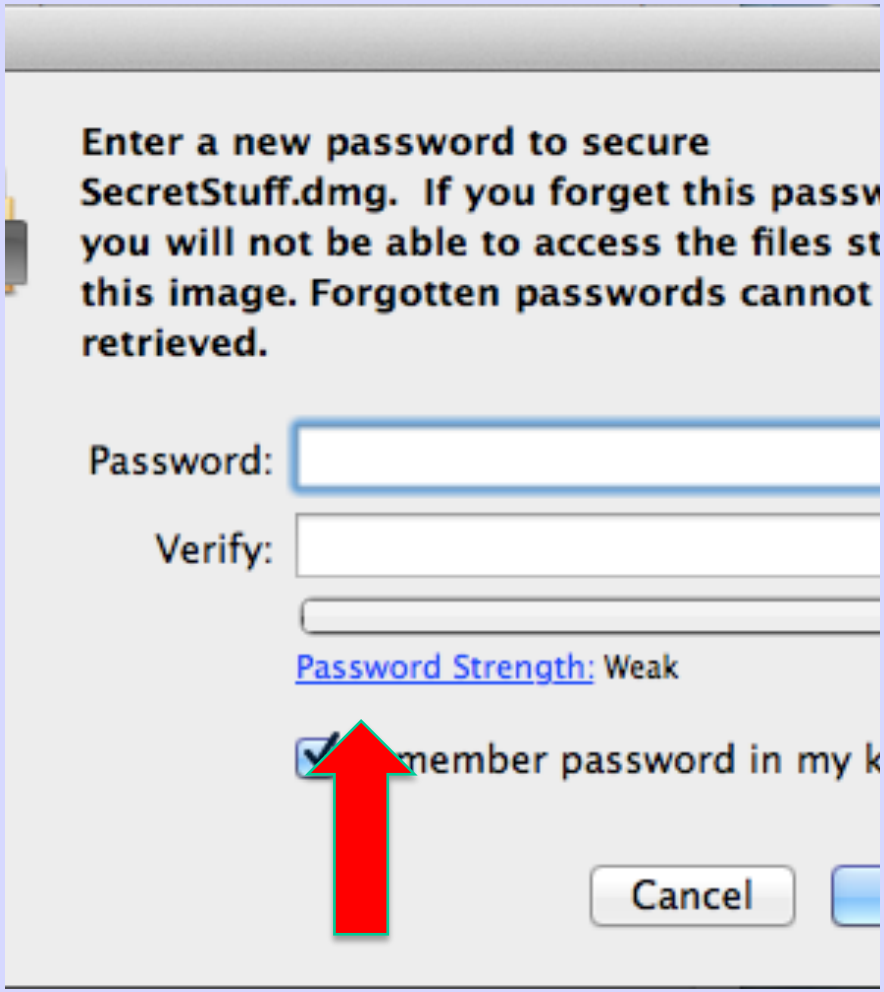
To encrypt an existing disk (Mountain Lion), go to finder

For Lion, use "diskutil"



Encrypting a Mac Disk Image

- You can encrypt disk images, too
- Note the option to store the newly-created key in your keychain
- What is your threat model? This is a good choice for cloud-resident images



Encrypted Disk Images on Linux

```
# Installation
$ sudo apt-get install encfs fuse-utils
$ sudo modprobe fuse

# Add yourself to the group
$ sudo adduser <your username> fuse

# Create the directories
$ mkdir ~/ciphertext ~/plaintext

# Create it or mount it
$ encfs ~/ciphertext ~/plaintext

# To unmount
$ fusermount -u ~/plaintext
```

Bitlocker on Windows

You need TPM to encrypt your boot drive