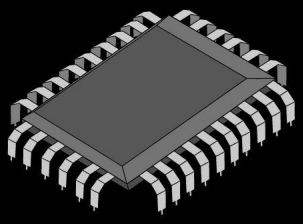
NMAAHC Workshop
Day 2–2021

FORENSIC IMAGING TOOLS

→

SOFTWARE
HARDWARE



NMAAHC Workshop,
Caroline Gil and Eddy Colloton
2021

OVERVIEW HARDWARE
OVERVIEW
OVERVIEW SOFTWARE

Writeblockers
ESD bags
Enclosures
FRED
TX1

Bitcurator
FTK Imager
libewf

Hardware – write blockers

Connections:

- Forensic card reader (Compact flash card, Memory stick card, MicroSD, etc)
- USB 3.0
- FireWire
- PCIe
- SATA/IDE
- SAS

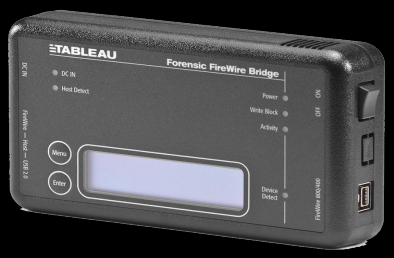


Tableau Forensic USB Bridge USB 3.0

NMAAHC Workshop,
Caroline Gil and Eddy Colloton
2021

Looking at hardware, here you can see a variety of forensic bridges, also known as write-blockers. As the name alludes to, write blockers prevent writes on storage media. You mount your storage device with read-only access to prevent compromising the integrity of the data. Essential when establishing a chain of custody. Write blockers are employed to allow the safe transfer of media from a carrier to your workstation without compromising or altering the integrity of the data.

Write blockers are often connection-type specific, requiring either multiple blockers for different connection types (such as PATA/IDE, SATA, USB, and FireWire), or “all in one” units such as the FRED (Forensic Recovery of Evidence Device), which are often appealing to archives with a variety of born-digital volumes (Prael and Wickner 2015). As manufacturers introduce SSDs as standard equipment in new computers, conservators will also need the ability to write block when they create disk images from these drives. There are currently several different interfaces for SSDs and, manufacturers such as Tableau require purchasing a SATA or PCIe write blocker along with a specific adapter for each SSD type, which unfortunately requires an additional investment.

The USB 3.0 or USB 2.0 are the hardware that are used most frequently. They are often employed the acquisition of digital materials from USB data carriers. Other connector types are available as well, for ex. DI UltraBlock Forensic Card Reader used for compact flash cards, microsd, etc

Write blockers are important tools when creating disk images from artwork-related hard drives and flash drives. Mounting a drive without one can inadvertently change the data stored on that drive. Especially in cases in which bit-level file integrity, or metadata such as “date modified” fields are determinative to a drive’s value, the effects of such a change can be catastrophic. Regardless, ensuring that original information on a storage device is not changed, either through automated manipulation or human error, should be considered best practice in media conservation.

Electronic components, such as a HD, are vulnerable to Electrostatic Discharge. ESD occurs when static electricity suddenly flows from one charged object to another. This can occur, for example, when a person charged with static electricity touches exposed components, such as a hard drive or a RAM chip. ESD exposure may permanently damage the component.”

One of the more common used bags is a low charging Pink Poly bag. These bags are made from a tinted polyethylene material with an antistatic coating that can wear away.

Metallized Shielding bags are constructed from a metalized polyester film and a low charging polyethylene laminate. This provides the bags with a shielding layer protecting the ESD sensitive components within the bag from possible ESD event damage. The low charging inner layer and outer layer of the bag prevent tribocharging (contact electrification) from occurring, minimizing the build up of ESD charges when handling components.

Anti-static wrist straps are recommended when working with electrical

components such as computer hard-drives, specifically if taking them out for imaging. The wrist strap is usually worn on the nondominant hand (the left wrist for a right-handed person). It is connected to ground through a coiled retractable cable and 1 megohm resistor, which allows high-voltage charges to leak through but prevents a shock hazard when working with low-voltage parts.

Typically when disk imaging computers, you would have to remove the HD.

It is often straightforward to open the computer's case and physically extract the hard drive.However, certain computer models make this task more challenging than others. Particularly with Apple Macintosh computers, disassembling the computer's case and removing the hard drive may be extremely difficult and time-consuming. In addition, a museum may not have the proper write-blocking equipment on hand to connect with some of Apple's proprietary interfaces for solid state drives (SSDs). Macs have a feature called Target Disk Mode that allows for forensically sound disk imaging without physically removing the hard drive from the computer's case.

Physical removal of the hard drive from a computer's case incurs some risks—pin connections on the drive could be bent or broken, or delicate cable connectors could be damaged. While outside the computer's casing, the hard drive could sustain physical damage through bumps, shocks, or electrostatic discharge (ESD).

Hard drive enclosures

Different connection types:

- SCSI
- SAS
- Fibre
- eSATA
- USB
- FireWire

NMAAHC Workshop,
Caroline Gil and Eddy Colloton
2021

A disk enclosure is a casing designed to hold and power disk drives while providing a mechanism to allow them to communicate to one or more separate computers.
Drive enclosures provide power to the drives therein and convert the data sent across their native data bus into a format usable by an external connection on the computer to which it is connected.

Different connection types;

PCIe, SATA and SAS drives are typically hot swappable, which means that it can be replaced without stopping, shutting down or rebooting a system.

Enterprise enclosures is a term that refers to larger physical chassis, such as the ones you see on NAS (network attached storage) and SAN (Storage Area Network)

Hardware – FRED

NMAAHC Workshop,
Caroline Gil and Eddy Colloton
2021

FREDs (Forensic Recovery of Evidence Device) are popular in libraries and archives. They're basically just a computer that is purpose built for disk imaging and digital forensics specifically. These systems are purpose built with top-quality, technology optimized for Digital Forensics and eDiscovery work. FREDs are built with the processing power and storage needed to run modern industry standard software applications. FRED systems also include a number of exclusive system components such as the Digital Intelligence UltraBay 4d Write Blocker and the Digital Intelligence Ventilated Imaging Shelf.
These machines are shipped from Digital Intelligence (New Berlin, WI) and include:

- System restore media - bootable Blu-ray disk containing restore environment and factory configured operating system images
- Symantec Ghost software
- Tableau Imager (TIM)
- FTK
- Forensic toolbox containing drive adapters and power / signal cables (SAS, SATA, IDE, microSATA, SATA LIF, MacBook Air, Blade SSD)
- PCIe SSD drive adapters (PCIe SSD m.2 NVMe, 2013 or newer MacBook Pro SSD, and server class PCIe SSD)

- Security screwdriver set with assorted bits for opening enclosures

FRED is only equipped with 4GB of storage, so large disk images must be written to a larger storage area or you can configure your FRED to have larger storage capacity

Hardware – TX1

NMAAHC Workshop,
Caroline Gil and Eddy Colloton
2021

There are also stand alone forensic imagers like the Tx1. It's also super expensive. But it's a write blocker, disk imaging software, and a mini computer all in one. You just connect the volume you want to image to the device, connect the drive you want to write the disk image out to on the other end, and hit "go."

TX1 is built on a custom Linux kernel, making it lean and powerful
The TX1 can forensically image a broad range of media, including PCIe and 10Gb Ethernet devices, and supports up to two active forensic jobs at a time (simultaneous imaging). When imaging, TX1 outputs to raw .DD and .dmg formats, .e01 (compressed), or .ex01 (compressed), and features extensive file system support (ExFAT, NTFS, EXT4, FAT32, HFS+).

The TX1 can support two concurrent forensic jobs without sacrificing performance.

Software

ddrescue	CLI only	Open source	Data recovery, good for optical media
Guymager	CLI & GUI	Open source	In BitCurator
FTK Imager for Mac	CLI only	Closed source	DEPRECATED
FTK Imager for Windows	GUI	Closed source	Very popular
Tableau Forensic Imager	GUI	Closed source	Free, less popular

NMAAHC Workshop,
Caroline Gil and Eddy Colloton
2021

Software- these are the tools that actually make the disk image.

These are some of the more common disk imaging software you'll encounter. There's probably some personal bias as these are the ones I'm most familiar with as well.

dd_rescue, is an an advanced evolution of dd, a command line program that has been ported only for UNIX/Linux. The program uses a complex series of flags to allow the user to image or write data from and to raw image files. (dd is one of the oldest of the imaging tools, and produces raw image files. Extended into dcfldd)

guymager supports all relevant forensic formats (dd, ewf, aff). It is very user friendly and faster than known commercial imagers running under Windows. As it is based on libewf, it supports all the different subformats found in libewf.

The **Forensic Toolkit Imager** (**FTK Imager**) is a commercial forensic imaging software package distributed by AccessData.

FTK Imager supports storage of disk images in EnCase's or SMART's file format, as well as in raw (dd) format. With Isobuster technology built in, **FTK Imager** Images CD's to a ISO/CUE file combination. This also includes multi and open

session CDs. GUI version comes preinstalled on FRED workstations.

TIM supports E0*, dd, and .DMG output formats.

I've had way better support from the open source community than closed source proprietary help. I've been able to get in touch directly with the developer of Guymager several times, and he responds promptly and thoroughly, which has not been the case with the companies that provide support for FTK or Tableau.

Eddy C tested all 5 of these applications...

BitCurator environment

-Provides reliable acquisition of bitstreams from digital media
-scalable forensic analysis of disk images
-metadata acquisition from common file systems
-preservation metadata and finding aid metadata export
-data visualization for complex data sources
-redaction of private and sensitive information
-identification and removal of duplicate files

NMAAHC Workshop,
Caroline Gil and Eddy Colloton
2021

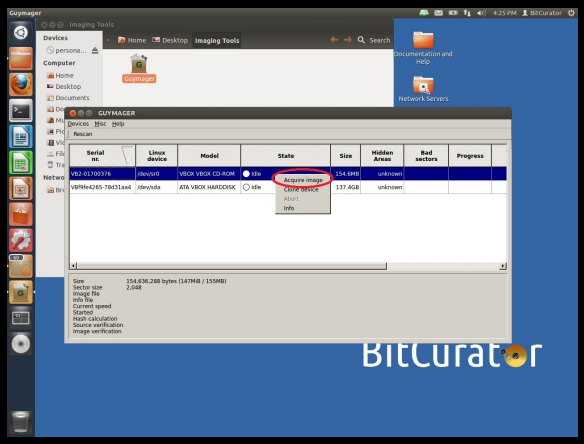
The BitCurator Environment is an Ubuntu-derived Linux stack of free and open-source digital forensics tools and software libraries. The environment is designed to help collecting institutions triage, acquire, describe, identify, and analyze born-digital materials, incorporating software and practices adopted from the digital forensics community with a low barrier to entry.
It can run on either a dedicated machine or through a virtual machine.

The environment groups tools and scripts under common steps in the digital forensics workflow: Forensic Disk Imaging, Forensic Processing and Identification of Potentially Sensitive Information, Data Triage, and Metadata Export. A number of tools allow users to view, browse, and analyze disk images (BitCurator Disk Image Access Tool, BitCurator Reporting Tool, fiwalk, bulk_extractor), identify and prioritize important information in or about disk images (ssdeep, sdhash, ClamAV, DFXML tools, FSlint), as well as additional scripts and tools that further support or enhance the described workflow steps (BitCurator Mounter, GHex, a hex editor/viewer, custom BitCurator tools and Nautilus scripts).

As an open-ended and modular environment, BitCurator provides a lot of space for contextual decision-making. Users can mount disk images and explore them manually and visually, and they can run a variety of reporting or analysis tools (such as bulk_extractor).

BitCurator environment– Guymager

- Easy user interface in different languages
- Runs under Linux
- Really fast, due to multi-threaded, pipelined design and multi-threaded data compression
- Makes full usage of multiprocessor machines
- Generates flat (dd), EWF (E01) and AFF images, supports disk cloning
- Open source, and free

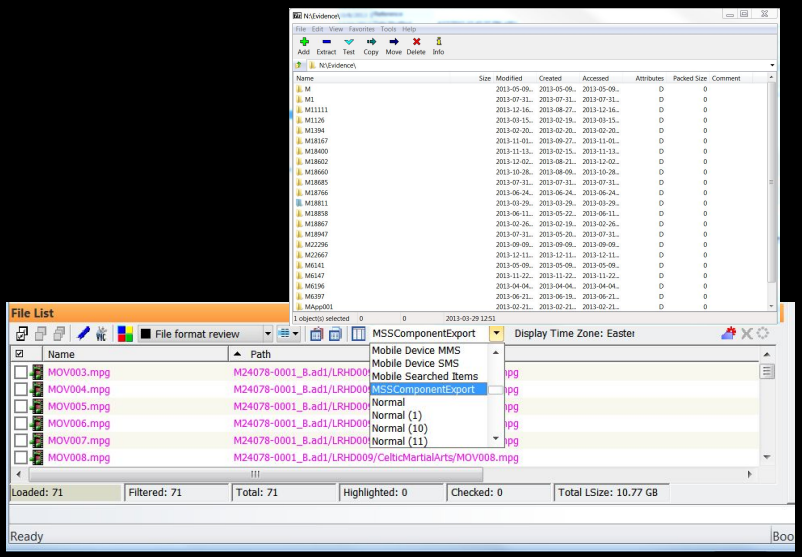


NMAAHC Workshop,
Caroline Gil and Eddy Colloton
2021

Guymager is an open source forensic imager. It focuses on user friendliness and high speed.The BitCurator environment includes Guymager, an open-source, graphical application for creating disk images. Guymager has support for raw dd images, EO1, and AFF image formats. The latter two image formats are commonly used in the digital forensics community and have the ability to incorporate metadata about the original media into the disk image itsel

It is one of the first forensic imaging tools to utilize mutli-threading for the imaging process.

FTK (Forensic Toolkit) and FTK Imager



NMAAHC Workshop,
Caroline Gil and Eddy Colloton
2021

Forensic ToolKit (FTK) is a proprietary computer forensics software made by AccessData designed to “help law enforcement officials, corporate security, and IT professionals access and evaluate the evidentiary value of files, folders, and computers,” therefore its features correspond with stages in the e-discovery process: identification, preservation, collection, processing, review, and production. Runs on Windows

Functionality includes creation of disk images (e.g., of hard drives, floppy, CD/DVD, portable media such as USB drives, and/or live data from any common electronic source); identification, analysis, and redaction of deleted files and objects (data carving), contact information, and other data specified by the user; filter searching to quickly locate specific item types and/or exclude specific data from review and reporting; as well as decryption. The filter feature allows users to search for keywords and specified data (e.g. email addresses, phone numbers) at a high level of granularity: predefined filters can target specific mobile and communication data such as email and addresses or signatures within them, phone history, calendar information, etc.

FTK presents high financial and technical barriers. The software costs a few thousand dollars without a support contract. Language and metadata further pose an obstacle.

Language used in the FTK manual and user interface are drawn directly from law enforcement, automatically creating a barrier of knowledge and

understanding related to terminology for users of the tool outside of those audiences. While features can be useful to those outside of the designated user group, use of the tool comes at an additional labor cost. Workflow steps and metadata fields must be mapped between FTK's terminology and those more prominently known in the archival and digital preservation communities (e.g., "evidence" and "case" vs. collection or accession, "custodian" vs. subjects, creator, or communities).

At NYPL used to conduct archival processing of electronic records

libewf

The libewf package contains the following tools:

- **ewfacquire**; which writes storage media data from devices and files to EWF files.
- **ewfacquirestream;** which writes data from stdin to EWF files.
- **ewfdebug**; experimental tool does nothing at the moment.
- **ewfexport**; which exports storage media data in EWF files to (split) RAW format or a specific version of EWF files.
- **ewfinfo**; which shows the metadata in EWF files.
- **ewfmount**; which FUSE mounts EWF files.
- **ewfrecover;** special variant of ewfexport to create a new set of EWF files from a corrupt set.
- **ewfverify**; which verifies the storage media data in EWF files.

Libewf is an open source tool created by Joachim Metz for working with EWF disk images. The library of tools can be run from the command line. The commands are described on the tool's github as they are on this s lids. I have primarily used ewfacquire, ewfexport, ewfmount, and ewfverify.

ewfacquire can be used to create ewf disk images from either raw disk images or a physical carrier. ewfverify can be used to ensure that an ewf disk image matches it's embedded fixity data..

How to install libewf on macos

1. Run: `brew install osxfuse`
2. Run: `brew edit libewf`
3. In the text editor, edit line 38 of libewf.rb to read “with-libfuse=yes”.
   a. Default is vim, use “`i`” to insert text, “`esc`” and then “`:x!`” to save & quit
4. Run: `brew install --build-from-source [path to libewf.rb]`

NMAAHC Workshop,
Caroline Gil and Eddy Colloton
2021

libewf comes pre-installed on BitCurator. This is likely the easiest way to use the tool, but it can, of course, be helpful to have libewf installed on your host operating system. If you would like to have libewf installed on macos, you can do so with homebrew.

I wouldn’t recommend this for right now, it’s possible this will take a second, but if this is something you’re interested in doing, just let me know and we can work on it together. I’ve just found that trouble shooting homebrew issues is not the most fun group activity.

You first need to install a dependency, osxfuse, AND you need to edit the homebrew libewf ruby script.

A thing I didn’t even know you could do until Ethan Gates of Yale University Libraries taught me how. So you run brew edit libewf, that opens a text editor. The default is vim. If you haven’t used vim before, you’re lucky, cause it’s terrible. But, you can find the part on line 38 that says with-libfuse=no, then change it to say with-libfuse=yes by entering the “insert mode” by pressing i. change the text, press escape, then finally save and quit by hitting the completely un-intuitive combination colon, x, and then exclamation point. When you exit vim it will show you the path to the file you just edited.

Finally, run brew install --build-from-source and the path to the ruby script you just edited.

How to install libewf on macos

Run ewfmount to confirm installation:

1. Run: `ewfmount -X allow_other [path to disk image] [path the mount point]`
   a. May need to open System preference - Security & Privacy, unlock with your password, select “allow” and restart
2. Ta da! In finder, there is now a OSXFUSE Volume mounted, containing a file named “ewf1”
3. Run: `rsync [path to osxfusevolume/ewf1] [path to new local folder]`
4. After rsync is complete, renamed the local ewf1 file to something more descriptive and change file extension to .iso
5. Mount the .iso and see all the files!

NMAAHC Workshop,
Caroline Gil and Eddy Colloton
2021

You can use ewfmount to ensure that both libewf and it’s underlying dependency osxfuse have installed correctly. ewfmount will create a mountpoint on your computer and extract a raw disk image from a provided ewf disk image. So, ewf disk image going into ewfmount, and a raw disk image comes out, in a directory that acts like a mounted volume.

You do this by calling out the extended options flag (-X)

Once you have the raw disk image in the mount volume, you can copy it out using rsync, as described above.

macos won’t recognize the copy of the raw disk image yet though, you’ll first need to change the file extension to something it will recognize, like .iso.Once, you’ve done that, you can mount the disk image by just double clicking on it.

How to use ewfexport

1. Run: `ewfexport [path to disk image]`
2. Follow the onscreen instructions.
   Simply pressing "`enter`" chooses the default response, listed in [brackets]
   a. For example:
      ```
      Information for export required, please provide the necessary
      input
      Export to format (raw, files, ewf, smart, ftk, encase1, encase2,
      encase3, encase4, encase5, encase6, encase7, encase7-v2, linen5,
      linen6, linen7, ewfx) [raw]:
      ```
3. After the process is complete, change file extension to .iso
4. Mount the .iso

NMAAHC Workshop,
Caroline Gil and Eddy Colloton
2021

Running ewfmount will confirm that all dependencies for libewf are employed correctly, but it is an inefficient way to export a raw disk image from an ewf disk image. The much faster way to do is to simply run ewfexport.

https://linux.die.net/man/1/ewfexport

In the activity, we'll show one last piece of software used to create disk images: Guymager.