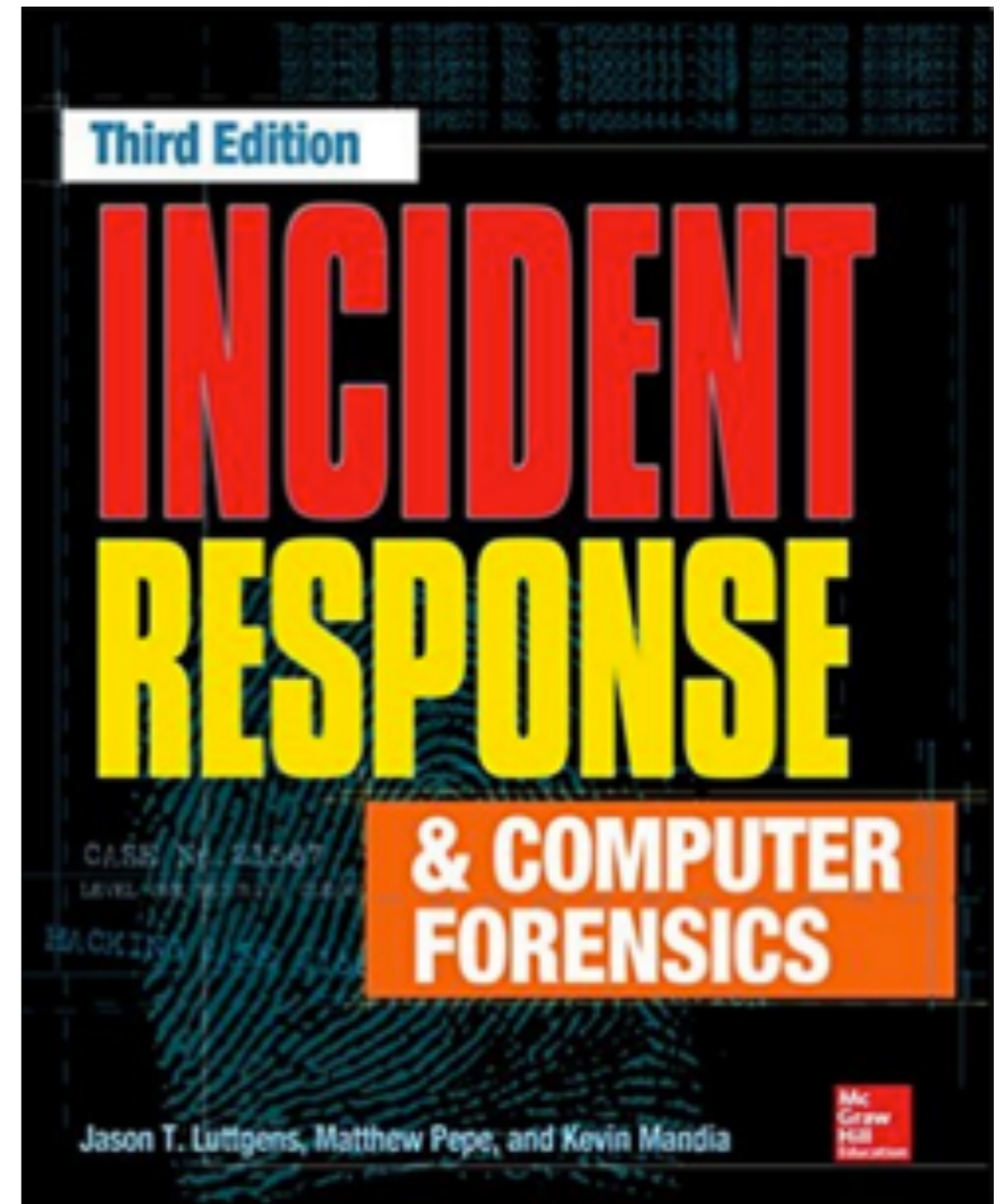


# CNIT 121: Computer Forensics



## 8 Forensic Duplication

# Types of Duplication

- **Simple duplication**
  - **Copy selected data; file, folder, partition...**
- **Forensic duplication**
  - **Every bit on the source is retained**
  - **Including deleted files**
  - **Goal: act as admissible evidence in court proceedings**

# Requirements

- The tool must have the ability to image or account for every bit of accessible data on the storage medium.
- The tool must create a forensic duplicate of the original storage medium.
- The tool must handle read errors in a robust and graceful manner. If the imaging operation fails after repeated attempts, the error is noted and the process continues. A placeholder may be put in the output file with the same dimensions as the portion of the input with errors. The contents of this placeholder must be detailed in the tool's documentation.

# Requirements

- The tool or the process must not make any changes to the original storage medium.
- The tool must generate results that are repeatable and verifiable by a third party.
- The tool must generate logs that detail the actions requested and any errors encountered.

# Every Bit?

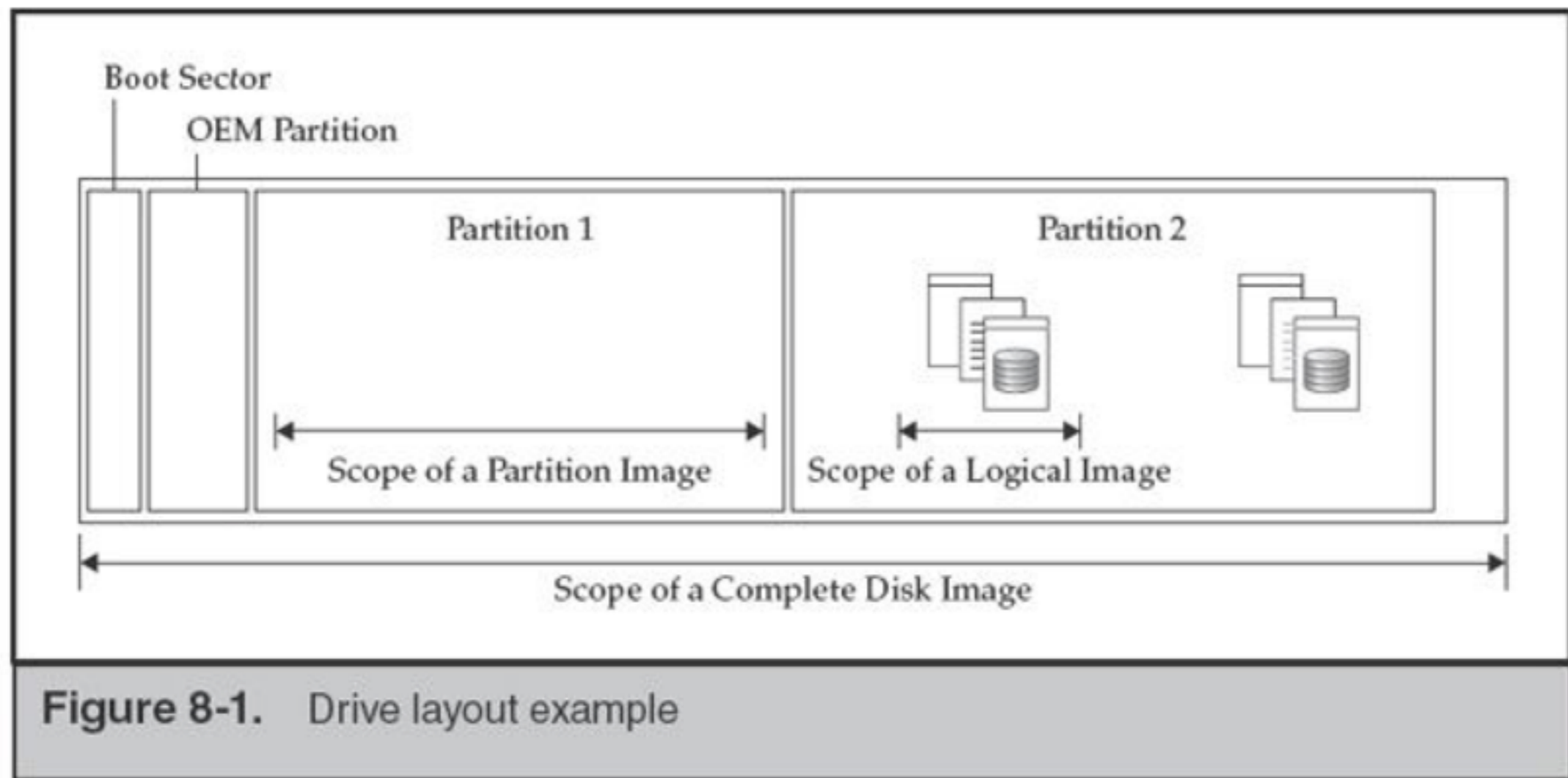
- **Some data on a hard disk or SSD isn't normally used to store user data**
  - **It contains firmware**
  - **"Host Protected Area" (HPA)**
  - **Not normally included in a forensic image**

# Forensic Image Formats

# Three Types of Forensic Images

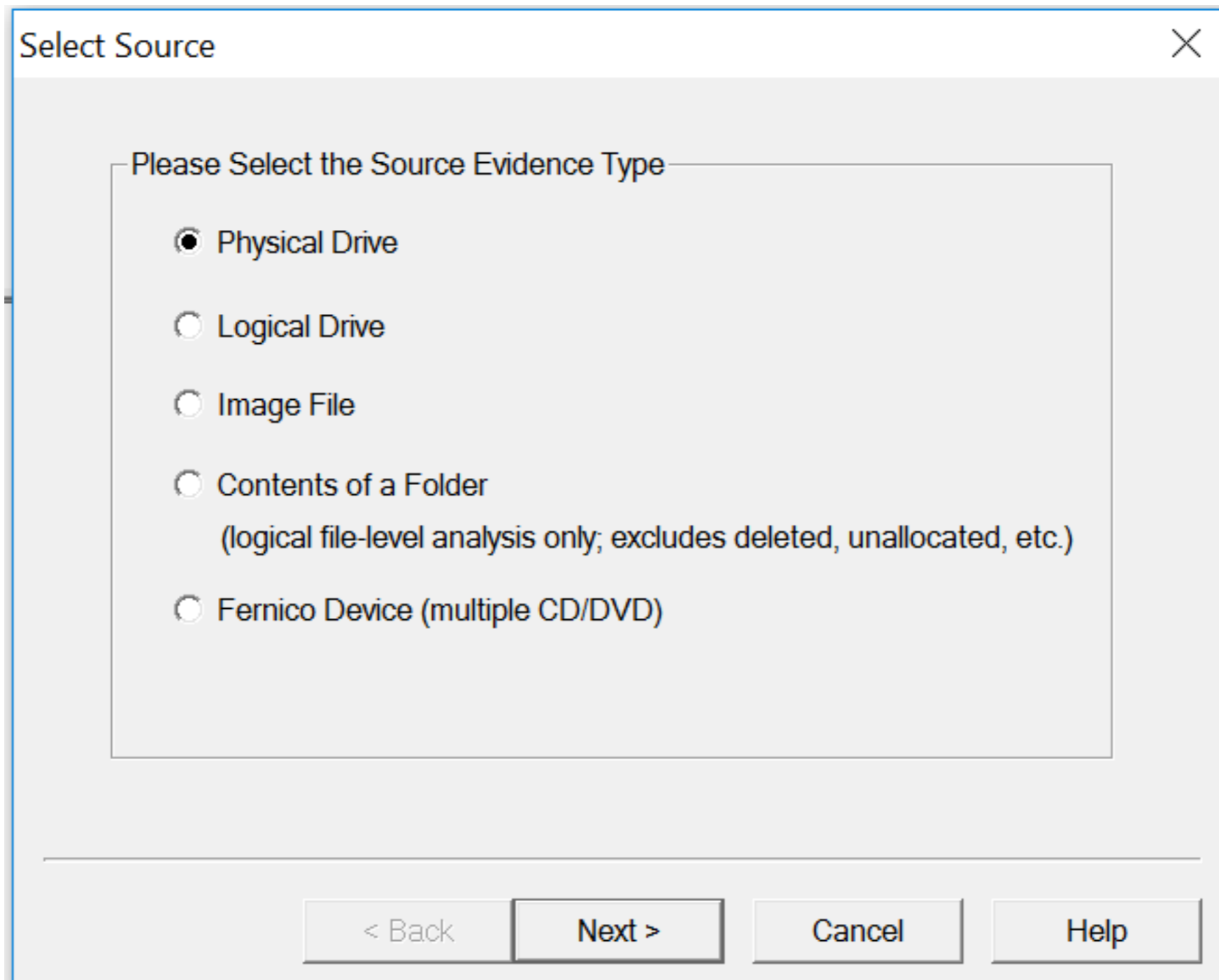
- **Complete disk**
- **Partition**
- **Logical**

# Complete Disk Image

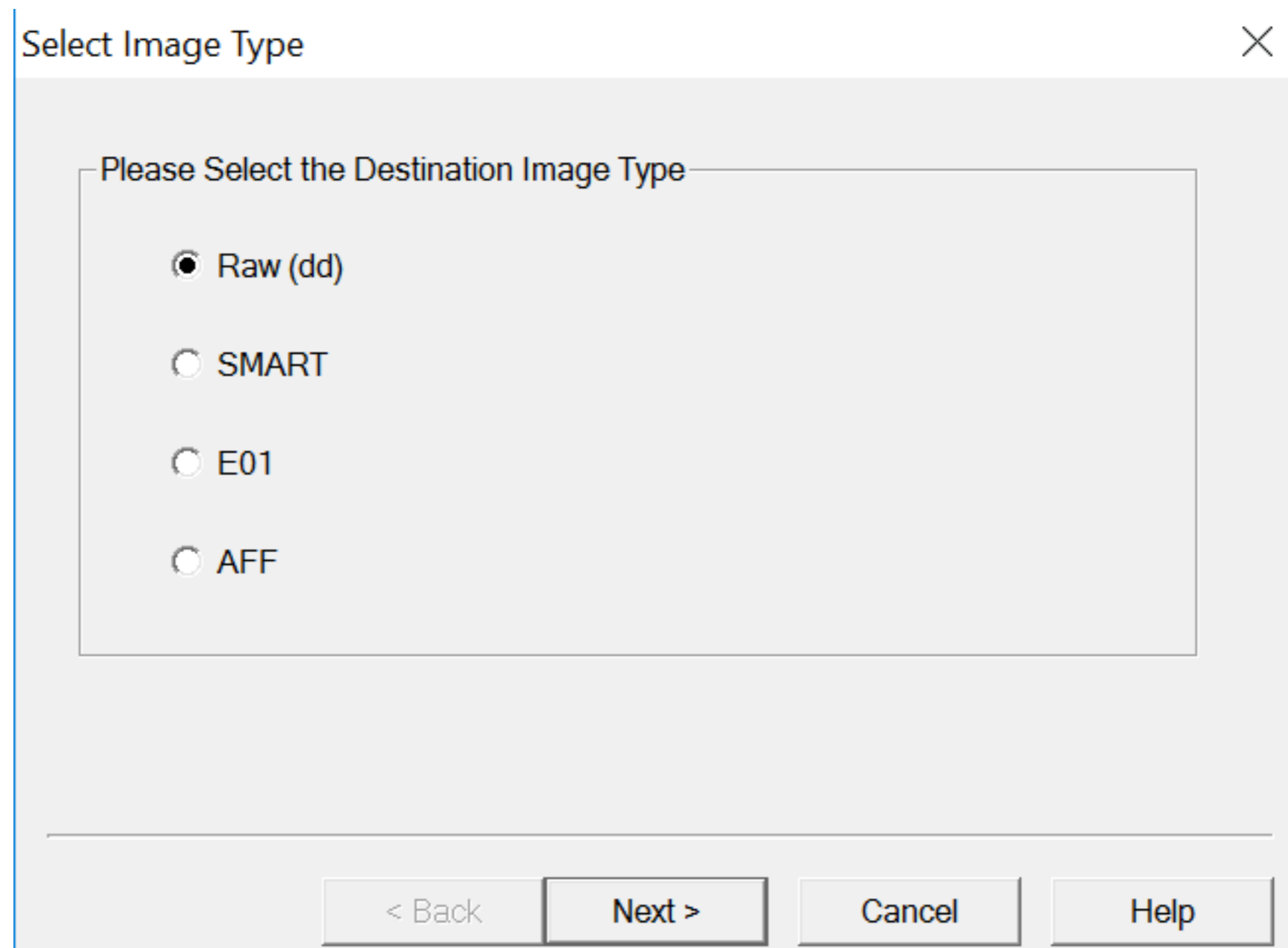




# Demo: FTK Imager



# Demo: FTK Imager



# Recovering Deleted Files

- **If a suspect attempts to hide data by**
  - **Deleting files or partitions**
  - **Reinstalling the OS**
  - **Reformatting**
- **Then a whole-drive image gives the best chance of recovering the missing data**

# HPA and DCO

- **Host Protected Area (HPA) and Device Configuration Overlay (DCO)**
- **A portion of the disk hidden from the computers's OS**
- **Used for boot and recovery utilities**
- **Rootkits can also hide here (link Ch 8a)**

# Three Data Types

- **Active data**
  - **Files and folders in use, in the directory**
- **Unallocated Space**
  - **Remnants of deleted files**
- **File slack**
  - **Fragments of data left at the end of other files**

# Partition Image

- **Not a common technique**
  - **May be required because of limited scope of authority, or an excessively large disk**
- **All allocation units from a partition**
- **Allows recovery of deleted files on that partition only**
  - **But not unpartitioned space, reserved areas, or other partitions**

# Logical Image

- **A simple copy of selected files or folders**
- **Active data only--no chance to recover deleted files**
- **If you are required to use a logical image, record the reason for later reference**

# When to Acquire a Logical Image

- **Court order only allows certain files to be collected**
- **Only one user's files from a shared storage device, such as a NAS (Network Attached Storage) or SAN (Storage Area Network)**
- **Files from a business-critical NAS or SAN that cannot be taken offline for duplication**
- **And you are not able to perform a live image**



# Acquiring Logical Images

- **You need to save file metadata**
  - **Creation times, permissions, etc.**
- **Also integrity hashes**
- **FTK Imager and EnCase can collect logical images**

# Non-Standard Data

- **System admin gives you a USB stick full of logs**
- **VM server admin hands over virtual machine files**
- **Network admin submits network capture files**
- **Document as much as you can and track the data the same way you track forensic images**

# Image Integrity

- **Hashes ensure that data is not changed after the time when the hash was computed**
  - **Also ensures that copies are accurate**
- **Drives with bad sectors give a different hash each time they are imaged**
- **Document that if it happens**

# Image Formats

- **AFF (Advanced Forensic Framework)**
  - **Used by AccessData's FTK and ASR Data's SMART**
- **Expert Witness Format (EWF)**
  - **Used by EnCase**
- **Both store MD5 or SHA1 hashes automatically**
- **Both are compressed formats & split data into several files; such as .E01, .E02, .E03, ...**

# DD Files

- **.dd files are exact copies of a drive**
  - **A 500 GB drive results in a 500 GB .dd file**
  - **No compression, no extra data like hash values**
- **dcfldd computes hashes also, and can optionally save them in a separate text file**

# Documentation

- **Evidence documentation must include integrity hashes**
- **Chain of custody**
- **Reports, other documents**

# Choosing a Format

- **All forensic image formats contain the same disk data, of course**
- **Each can be converted to the others, but it's a lengthy process**
- **Commercial Windows tools usually expect EWF files**
- **Open-source tools usually require .dd files**
- **For RAID and other multi-disk arrays, .dd files are best for advanced processing**

# Traditional Duplication



# Static Image

- **Hard drive only**
- **Computer has been powered off**
- **Image is made with a hardware disk duplicator**
- **Or by booting from a forensic LiveDVD**

# Hardware Write Blockers

- **Best way to ensure that the drive is not modified during image collection (image: Wikipedia)**

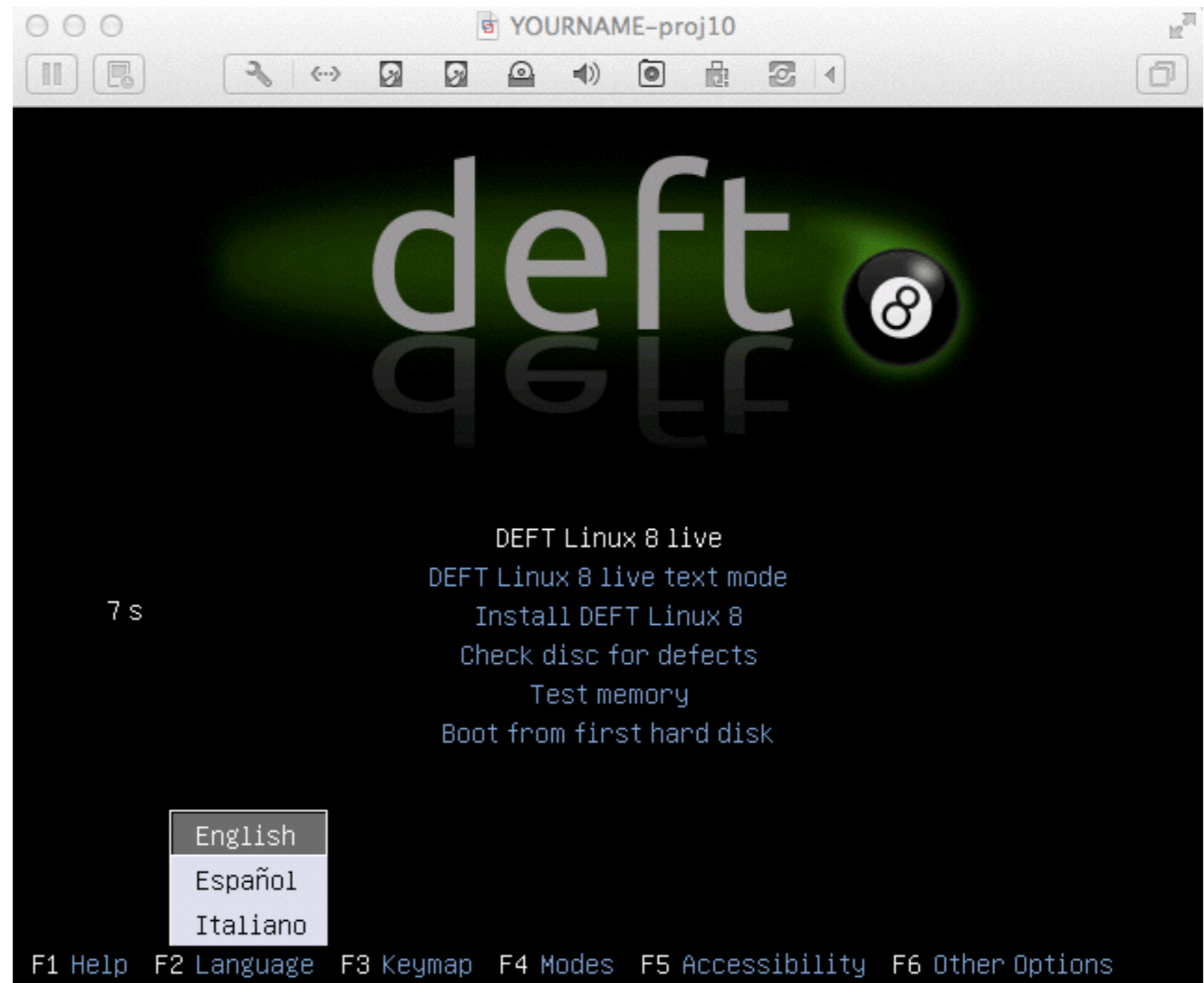


# Write-Blockers

- **Industry leaders are Tableau and WeibeTech**
- **They cost hundreds of dollars**

# Forensic LiveDVD

- **Boot disk**
- **Blocks writing with software**



# Image Creation Tools

- **Software tools: dc3dd, FTK Imager, EnCase**
- **Hardware disk duplicators**
  - **Expensive but convenient**

# Imaging Considerations

- Is my source media write protected?
- Will my examination environment attempt to perform any actions automatically, if I'm in a situation where a hardware write-protection device isn't feasible?
- Do I have sufficient space for the output files?
- How do I address the source media?
- What command-line options are required to get the expected output?

# dd, dcfldd, dc3dd

- **dd is included in Linux and Unix systems**
  - **It works, but doesn't create a hash value and doesn't provide user feedback**
- **dcfldd and dc3dd**
  - **Add the missing features to dd**
  - **From US DoD Computer Forensics Laboratory (DCFL) and Defense Cyber Crime Center (DC3)**

# Device Automounting

- **Every modern OS mounts disks automatically**
  - **And writes on them immediately**
  - **Changing timestamps, journal entries, etc.**
- **Hardware write-blockers are the best defense**
- **Forensic LiveDVDs block this process in software**



# EnCase

- **Several tools to create forensic images**
- **Directly in Windows with Encase Forensic**
- **Two command-line utilities**
  - **winen.exe or winacq.exe**
- **LinEn: Linux-based boot disk**
- **You must own EnCase to use them**

# Live System Duplication

# Live Imaging

- **Creating an image of media in a computer while it is running**
- **Not ideal; called a "smear"**
- **May be only option for**
  - **Business-critical systems**
  - **Encrypted drives**
- **Document what you did**

# Risks of Live Imaging

- **No write-blocker**
- **You are changing the system**
- **You might destroy evidence**
- **You might cause performance problems or even crash the system**
- **Don't install anything or save anything on the evidence system**
- **Run FTK Imager Lite from a network share or removable media**

# Apple Hardware

- **Components are integrated, hard to access**
- **Use strange connectors, like ZIF ribbon connector**
- **Reboot into Target Disk Mode**
  - **Makes the Mac act like a portable disk drive**
  - **Image it using Firewire or Thunderbolt connector**
  - **Tableau sells a FireWire write-blocker**






















# Central Storage Systems

- **RAID, SAN, NAS**
- **Not feasible to duplicate the entire original source, due to size and complexity**
  - **Sometimes using proprietary methods**
- **Determine where relevant data is, and make a logical copy of it**
- **Forensic tools like FTK can place the copy in a "container" with original metadata and a hash**
- **Live imaging might work best**

# Virtual Machines

- **Many servers are now virtualized**
- **Can simply copy VM files, including RAM**
- **Document the source and calculate a hash**

## Kali-Linux-2.0.0-vm-amd64

Name	^	Date Modified	Size
▶ caches		Sep 12, 2016, 12:44 PM	--
 Kali-Linux-2.0.0-vm-amd64-349e4818.vmem		Sep 12, 2016, 12:44 PM	2.15 GB
 Kali-Linux-2.0.0-vm-amd64-349e4818.vms		Sep 12, 2016, 12:44 PM	72.6 MB
 Kali-Linux-2.0.0-vm-amd64-s001.vmdk		Sep 12, 2016, 12:44 PM	3.88 GB
 Kali-Linux-2.0.0-vm-amd64-s002.vmdk		Sep 12, 2016, 12:44 PM	1.57 GB
 Kali-Linux-2.0.0-vm-amd64-s003.vmdk		Sep 12, 2016, 12:42 PM	1.38 GB
 Kali-Linux-2.0.0-vm-amd64-s004.vmdk		Sep 12, 2016, 12:44 PM	1.89 GB
 Kali-Linux-2.0.0-vm-amd64-s005.vmdk		Sep 12, 2016, 12:44 PM	485.8 MB
 Kali-Linux-2.0.0-vm-amd64-s006.vmdk		Sep 12, 2016, 12:44 PM	858.7 MB
 Kali-Linux-2.0.0-vm-amd64-s007.vmdk		Sep 12, 2016, 12:44 PM	1.51 GB
 Kali-Linux-2.0.0-vm-amd64-s008.vmdk		Sep 12, 2016, 12:44 PM	69.9 MB
 Kali-Linux-2.0.0-vm-amd64.nvram		Sep 12, 2016, 12:44 PM	9 KB
 Kali-Linux-2.0.0-vm-amd64.plist		Sep 12, 2016, 12:25 PM	588 bytes
 Kali-Linux-2.0.0-vm-amd64.vmdk		Sep 12, 2016, 12:09 PM	948 bytes
 Kali-Linux-2.0.0-vm-amd64.vmsd		Aug 10, 2015, 2:34 PM	Zero bytes
 Kali-Linux-2.0.0-vm-amd64.vmx		Sep 12, 2016, 12:44 PM	3 KB
 Kali-Linux-2.0.0-vm-amd64.vmx		Aug 10, 2015, 3:51 PM	383 bytes
 startMenu.plist		Sep 12, 2016, 12:44 PM	884 bytes
 vmware-0.log		Sep 10, 2016, 3:08 PM	307 KB
 vmware-1.log		Aug 23, 2016, 3:35 PM	309 KB
 vmware-2.log		Jun 28, 2016, 4:22 PM	311 KB
 vmware.log		Sep 12, 2016, 12:44 PM	1.1 MB