498.1
Evidence File Quick Wins and Scene Management

This page intentionally left blank.

FOR498.1 | Battlefield Forensics & Data Acquisition

Evidence File Quick Wins and Scene Management

Authors:
Eric Zimmerman – saericzimmerman@gmail.com
Kevin Ripa – kevin.ripa@gmail.com

http://twitter.com/ericrzimmerman
http://twitter.com/kevinripa

This page intentionally left blank.

FOR498 Battlefield Forensics Track Agenda

Section 1 Evidence File Quick Wins & Storage Interfaces
- Understanding data, file systems, evidence files, collection quick wins, scene management

Section 2 Evidence Acquisition and Collection
- Portable devices, data acquisition

Section 3 Quick Win Forensics
- Mounting evidence, triage, RAM, and live acquisition

Section 4 Non-Traditional and Cloud Acquisition
- Battlefield forensics, remote acquisition, multi-drive storage and network acquisition

Section 5 Apple Acquisition, Internet of Things, and Online Attribution
- IoT, Apple device acquisition, online identification

Section 6 Beyond the Forensic Tools: The Deeper Dive
- Data carving & rebuilding, data recovery

This page intentionally left blank.

FOR498.1: Evidence File Quick Wins & Scene Management Agenda

SIFT Introduction

1.1 Intro to Digital Forensic Acquisition

1.2 Understanding the Data

1.3 Scene Management & Evidence Acquisition

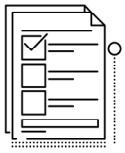
1.4 Device & Interface Identification

This page intentionally left blank.

SIFT Workstation

Windows 10 Enterprise installed in virtual machine

Includes dozens of forensic tools



Requires 64-bit computer and minimum of 16 GB of RAM

Preferred hypervisor is VMWare

The dataset being used for this course consists of Windows 10 Enterprise installed within a Virtual Machine (VM). This VM contains dozens of forensic and other tools to assist a user in their examination.

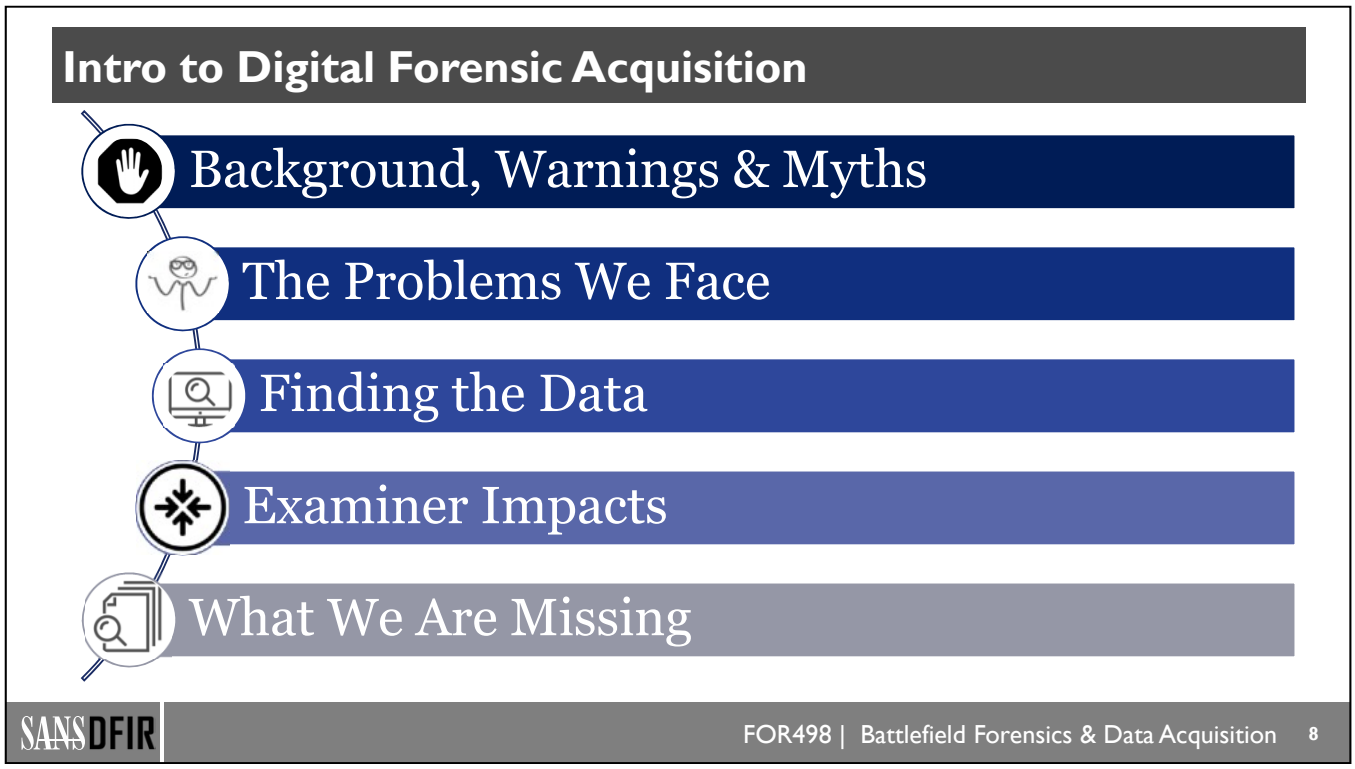
This course requires a computer using a 64-bit system, and a minimum of 16 GB of RAM.

The default VM hypervisor is VMWare. We recommend this for consistency. The student is welcome to use others (Fusion, Virtual Box, etc.), however it is expected that the student will know how to troubleshoot and manage those platforms themselves.

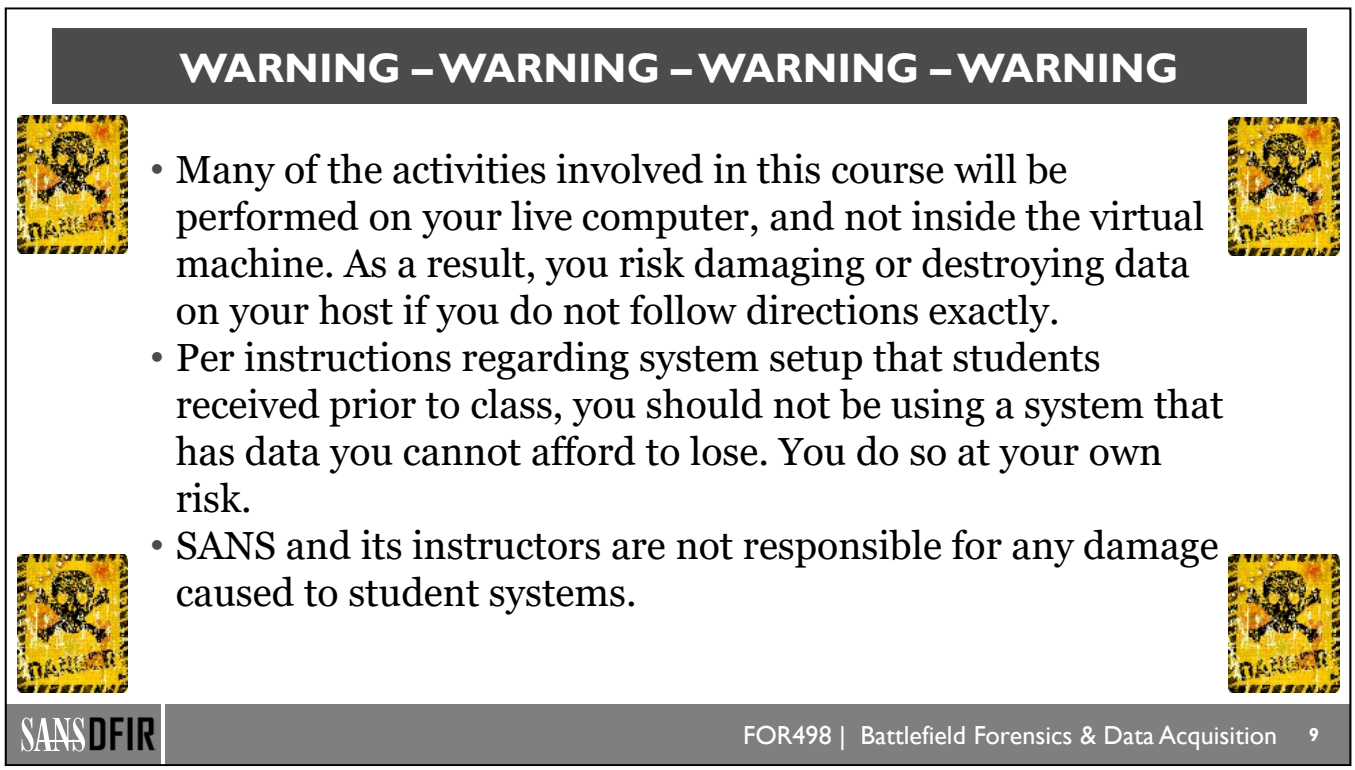
FOR498.1: Evidence File Quick Wins & Scene Management Agenda

- SIFT Introduction
- 1.1 Intro to Digital Forensic Acquisition
- 1.2 Understanding the Data
- 1.3 Scene Management & Evidence Acquisition
- 1.4 Device & Interface Identification

This page intentionally left blank.



This page intentionally left blank.



WARNING – WARNING – WARNING – WARNING

Many of the activities involved in this course will be performed on your live computer, and not inside the virtual machine. As a result, you risk damaging or destroying data on your host if you do not follow directions exactly.

Per instructions regarding system setup that students received prior to class, you should not be using a system that has data you cannot afford to lose. You do so at your own risk.

SANS and its instructors are not responsible for any damage caused to student systems.

Digital Forensics Definition

"The use of scientifically derived and proven methods toward the preservation, collection, validation, identification, analysis, interpretation, documentation and presentation of digital evidence derived from digital sources for the purpose of facilitating or furthering the reconstruction of events found to be criminal, or helping to anticipate unauthorized actions shown to be disruptive to planned operations."

There are various definitions on the Internet for the term "Digital Forensics", however one of the earliest ones came from a conference convened in Utica, New York in 2001 called the Digital Forensic Research Workshop.

"The use of scientifically derived and proven methods toward the preservation, collection, validation, identification, analysis, interpretation, documentation and presentation of digital evidence derived from digital sources for the purpose of facilitating or furthering the reconstruction of events found to be criminal, or helping to anticipate unauthorized actions shown to be disruptive to planned operations."

The workshop was designed to create "A Road Map For Digital Forensic Research" [1]. The document that emerged from this workshop should be required reading for all forensicators.

[1] A Road Map For Digital Forensic Research | `https://for498.com/frcsp`

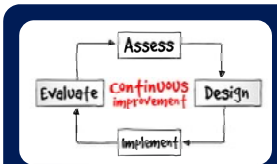
Battlefield Forensics

Data overload

- More and more data is being created in innumerably different ways
- Makes finding and collecting relevant data more difficult

Current challenges

- How to make sense of it all
- How to leverage collected data to the immediate benefit of a case



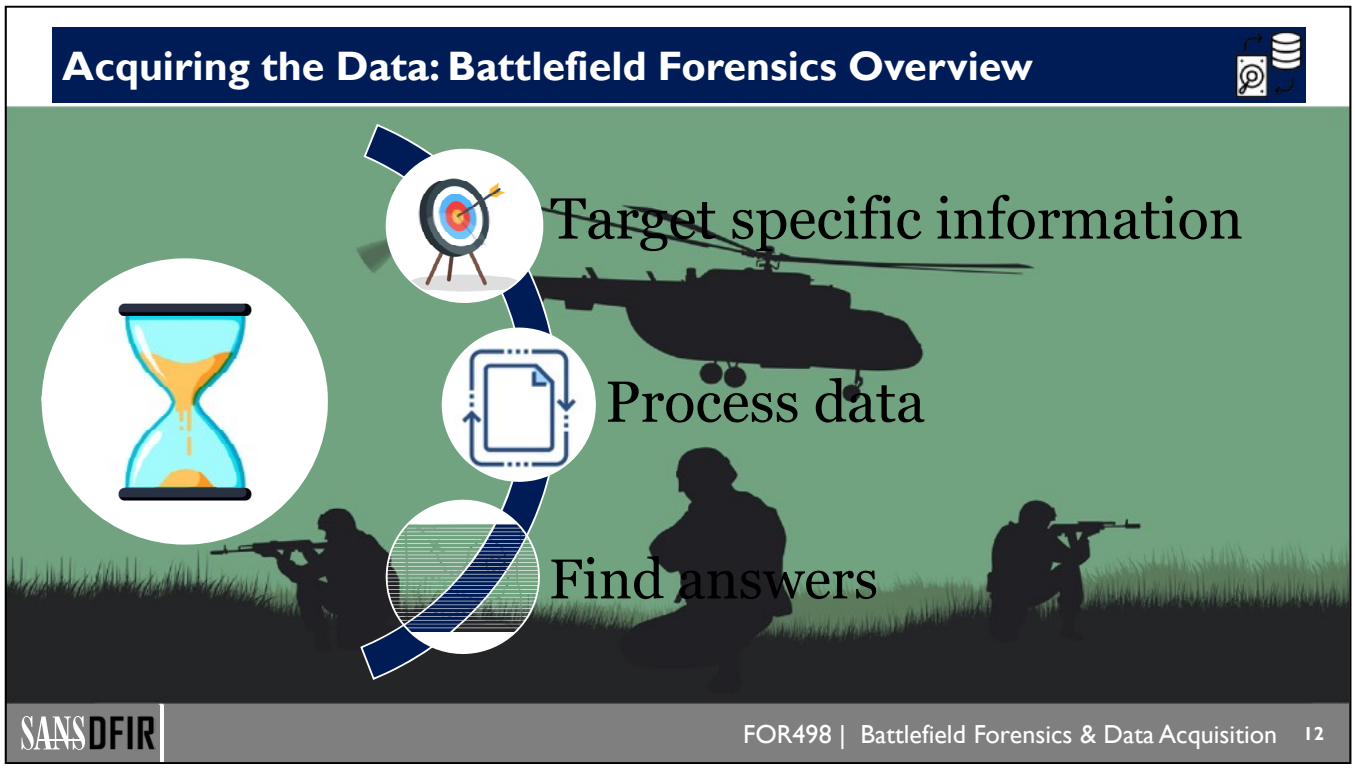
Replace, or refine

- This is not a replacement for traditional forensics
- Target the most useful data first
- Often done in conjunction with the acquisition process

The goal of a large part of this course is to change the way digital forensics is approached. Certainly we will have a need for traditional forensic collection and analysis for the foreseeable future, but the examiner must adapt their response process to the changes of the environment, as well as changes to response times.

The most important part of this process is being able to adjust our way of thinking. For too long, we have subscribed to the notion that digital forensics must be this "checklist-able" process for which there will be no deviation.

Welcome to FOR498 Battlefield Forensics & Data Acquisition, where we will challenge these very notions and many more, and assist the forensic examiner in making more informed decisions regarding the equipment and data sets we come into contact with every day.



Battlefield Forensics is the art and science of identifying and extracting actionable intelligence from a device in 90 minutes or less. But what does this mean, and more importantly, how can we conduct battlefield forensics in our own environment? The concept of battlefield forensics has in the past also been referred to as triage collection and serves much the same purpose as triaging of injured people, namely, to quickly identify the most important systems to prioritize by focusing attention on key artifacts and leveraging those artifacts to drive your case forward.

Battlefield forensics can be broken down into three major pieces: Targeting specific information, processing that information, and analyzing the output to find answers.

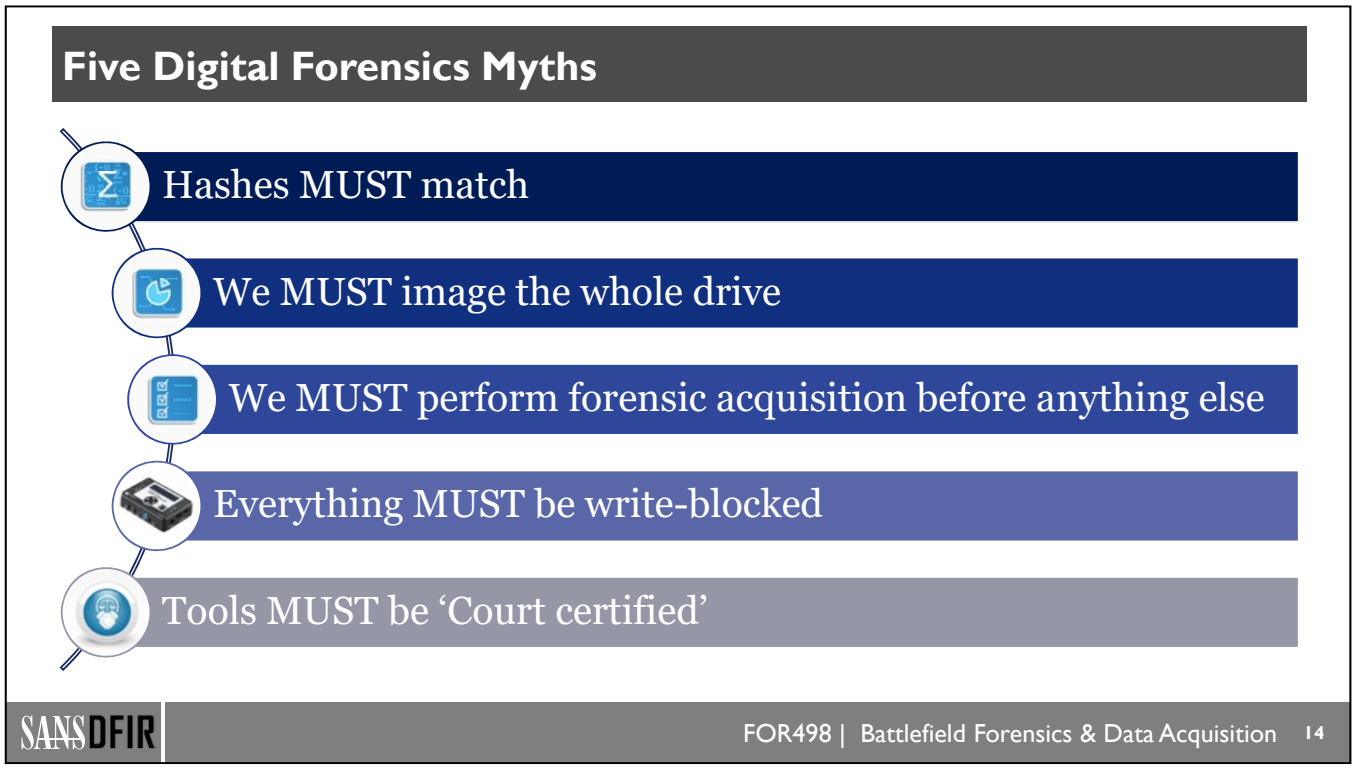
Target specific information
In most cases, less than one percent of the data on a typical device contains more than 95% of the information you will need quickly. Key forensic artifacts such as the MFT, prefetch, and Registry hives (to name just a few) contain a wealth of information amidst the sea of noise. By focusing in on specific files in specific types of cases, several things happen. One is a more focused work product that gets you toward the answers you need without a lot of distractions. The other is being able to quickly pivot and expand from an initial targeting of information to another artifact to gain additional information. By starting out with a narrower focus and expanding as necessary, information overload is reduced. The targeting phase is not limited to Windows artifacts. Any kind of file can be looked for and leveraged for intelligence, such as videos and images, Microsoft Office documents, and even proprietary database files.

Process data
Once artifacts and files are collected, they often need processing in order to make them usable by an investigator. While this is not always the case (reading Word documents for example), most of the useful forensic information is locked away in binary files where timestamps are encoded, strings are saved in Unicode, and other details are obfuscated. By processing the data using one or more parsers that know how to convert this binary data into a more human readable form, significant time is saved when working a case. Making this an automatic process speeds up the time between getting the data and leveraging it for intelligence.

Find answers

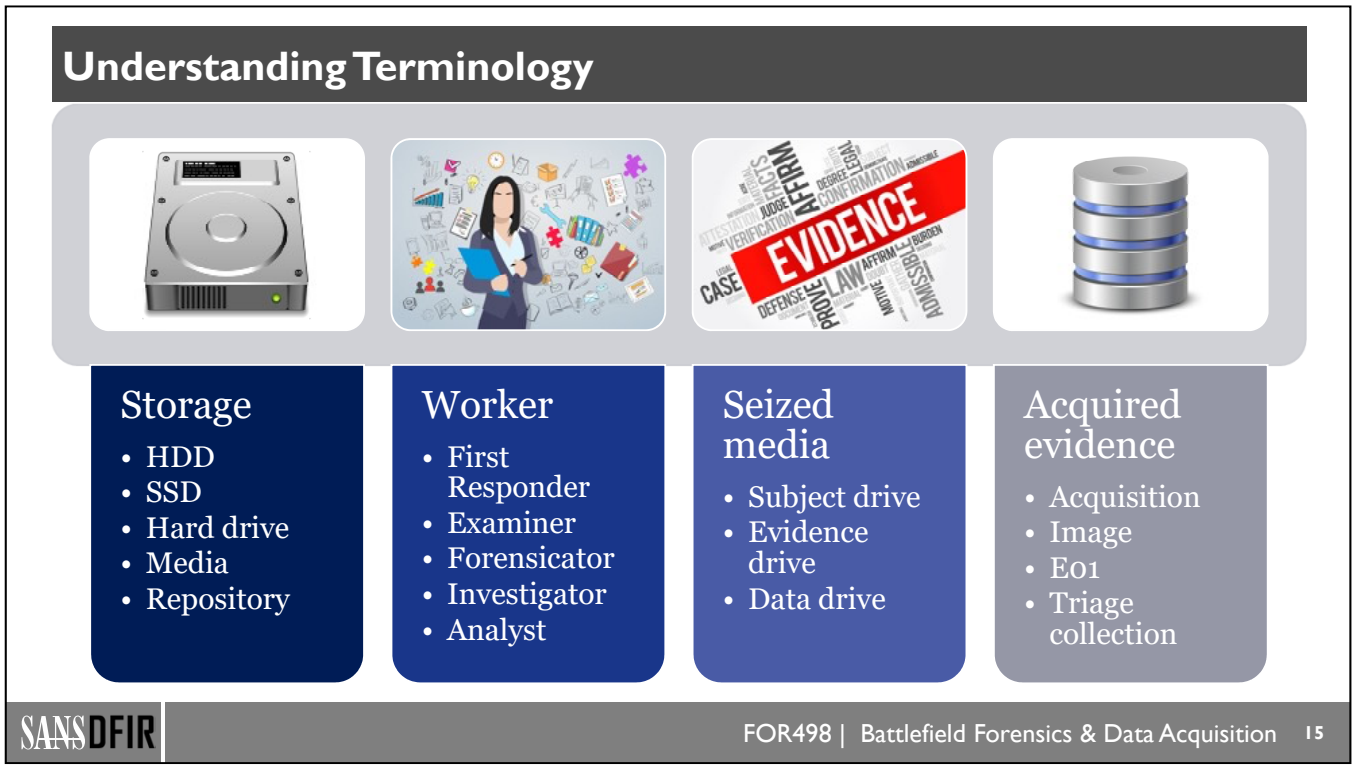
Once the data is processed, it can be analyzed. This can be a review of when files were created or deleted, when certain directories were visited, web site browser history, phone calls and text message history, and so on. The data being reviewed can many times lead to additional questions that may require additional files. In these situations, we go back to the first step and extract the newly sought-after information, which is then fed into the rest of the process.

By focusing in on and gathering the important stuff first, you gain the advantage of removing most of the noise from a case. By looking at a few hundred or fewer files vs. hundreds of thousands of files, your attention can be given to those areas where answers are often found.



"Hashes must match or our evidence is no good."
"We must image an entire hard drive or our process is flawed."
"We must perform a forensic acquisition before we can start our investigation."
"Everything we do must be write-blocked. We cannot interact with a live machine."
"Our tools must be 'court approved' or 'court certified'".

The above five often heard lines are fallacy. There are many reasons a hash (or digital fingerprint) may not match between acquisition and verification. There is nothing wrong with this, as long as the examiner can explain why they don't match, and why it does not affect the investigation. Only in very specific circumstances must we image entire hard drives today, and very seldom is the case where we must first image before we can start examining. Interacting with a live system is not the optimal way to conduct an investigation, but it is very necessary today, and the examiner must be armed with the appropriate tools and knowledge to perform in this environment. Back when dinosaurs roamed the Internet, it was dictated that every acquisition had to be write-blocked. While this is a sound rule to follow, it is not wrong to NOT write-block in certain circumstances. Cellular devices are a case in point. They are never write-blocked and cannot be. Lastly, we must get over this myth that digital forensics software or tools are somehow court approved or certified. THERE IS NO SUCH THING. This is a vendor driven misguidance designed to drive potential customers to a specific product.



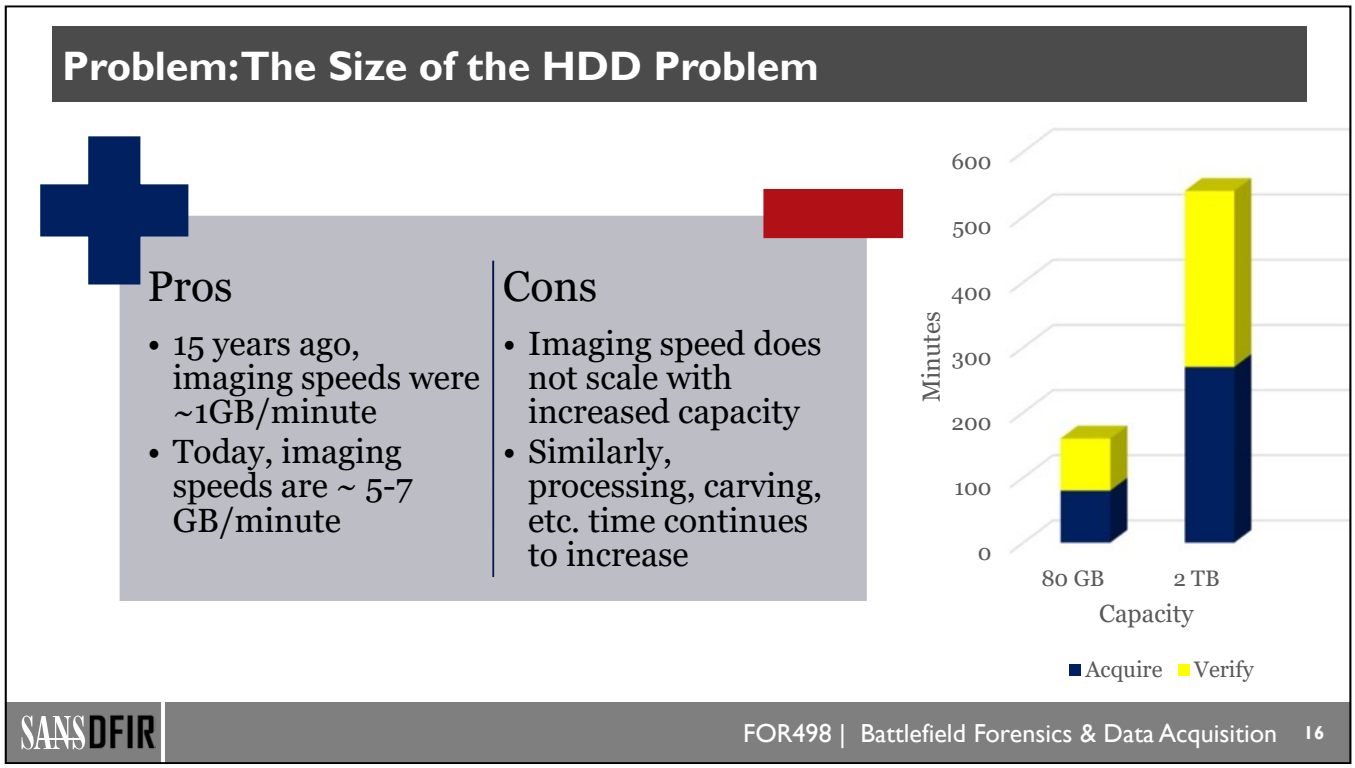
We have endeavored throughout the training manuals to provide a standardized approach to terminology. Still, there are times where we will use words interchangeably.

When we are talking about the forensic image, we may refer to it as simply an image. Forensic acquisition and forensic image are likewise, the same thing.

When we discuss the equipment or media that has been seized or is being examined, we may refer to it as the evidence drive or the subject drive. We try to avoid terms such as the suspect drive, as it can, in certain circles, be interpreted to infer a type of guilt or negative connotation.

As regards the person doing the work, whether it be acquisition or analysis, we may refer to them as the first responder, analyst, investigator, examiner, or forensicator. We use these words interchangeably, and do not suggest that one term indicates a higher degree of knowledge than another.

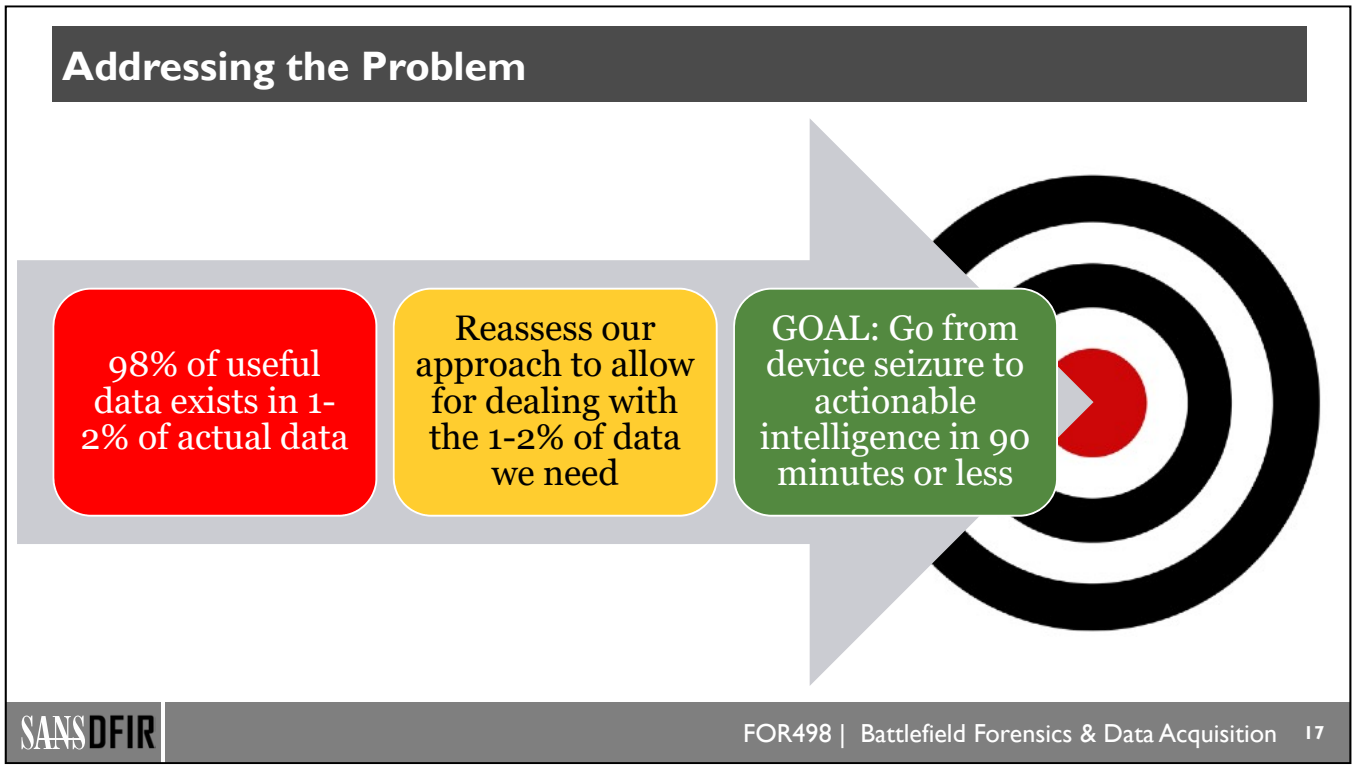
When we refer to where the data resides, we may call this media, storage media, hard drive, or data repository. We use these synonymously.



Not too long ago, examiners were collecting 80 GB hard drives, with collection speeds of 1 GB per minute. In those days, 80 minutes for collection and 80 minutes for verification were not barriers to rapid response. The first 80 GB hard drives were rolled out in 2000[1], and their use was routine through the next 5 years. Fast forward to today, and even though we can routinely acquire hard drives at a rate of 5 -7 GB per minute, this still means an acquisition plus verification time of about 9.5 hours at best, for a 2 TB hard drive. Now imagine arriving on site to find a 12 TB storage array! In the case of corporate environments, even larger storage pools may be encountered.

The argument can be made, and indeed frequently is, that imaging devices have exceeded the 25-30 GB per minute mark. While this is true, there are some caveats. These are certainly not everyday speeds with everyday devices. These are very specific situations involving specific types of solid state hard drives being imaged to other solid state hard drives, typically using multiple destination devices. Do not let marketing hype get in the way of your decision making.

[1] First 80 GB HDD ships | `https://for498.com/x9qs2`

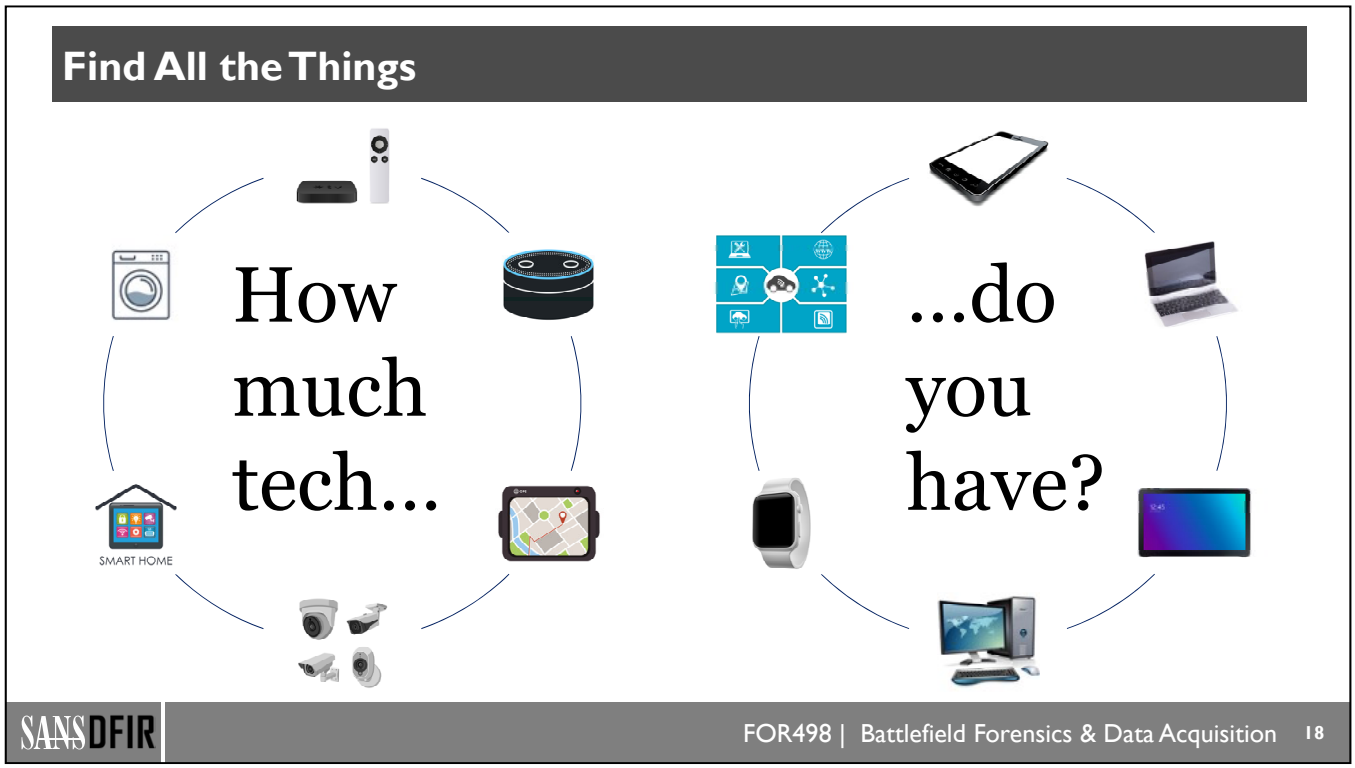


We must find new ways to be effective. It is a fact that in the vast majority of cases, 98% of necessary information exists within 1-2% of the data. This is where Battlefield Forensics takes shape. Your aim is to go from seizure of data to actionable intelligence in 90 minutes or less!

This is largely believed to be an unachievable goal; however you will be doing it by the end of the course. Certainly we are not suggesting that your investigation will be over in 90 minutes. But in many cases, early information is necessary to help move an investigation forward. Often there are facets of an investigation that are beyond the forensics realm, and these parts of the investigation rely on forensic findings.

For example, a case that involves theft of intellectual property. A company is losing hundreds of thousands, if not millions of dollars a day due to lost revenue from theft of IP. They need a court injunction immediately to stop the offenders. They don't need the entire investigation completed in a day, but they do need some kind of evidence to go into court with. Quickly extracting an offending email may be just enough to stop the loss, at least temporarily, while the rest of the investigation is completed.

In another example, law enforcement may be dealing with a missing persons case and cannot wait for a full examination of the victim's computer. If the next lead can be derived from a rapid parsing of Internet history, a life may be saved.



There is hardly an investigation today that is not touched in some way by an electronic device. The average person in the industrialized world has at least two devices. These being a cellular phone and a computer. Many people have more than that. In fact, it is not uncommon to see someone with a phone, a laptop, a desktop, a tablet, and work computer, and a watch that syncs to all of this!

From being able to remotely control the temperature in our homes, to being able to connect to our security cameras or control our entertainment and environment, technology (and the vast amounts of data collected as a result of it) can be found everywhere.

It was easy 10-15 years ago to simply seize a desktop computer and analyze the hard drive. Sadly today, many investigations still take that route, ignoring so much other data because examiners are unaware of other storage areas, or simply don't know how to process them.

The following list is by no means complete, but how many of these devices do you have in your household?

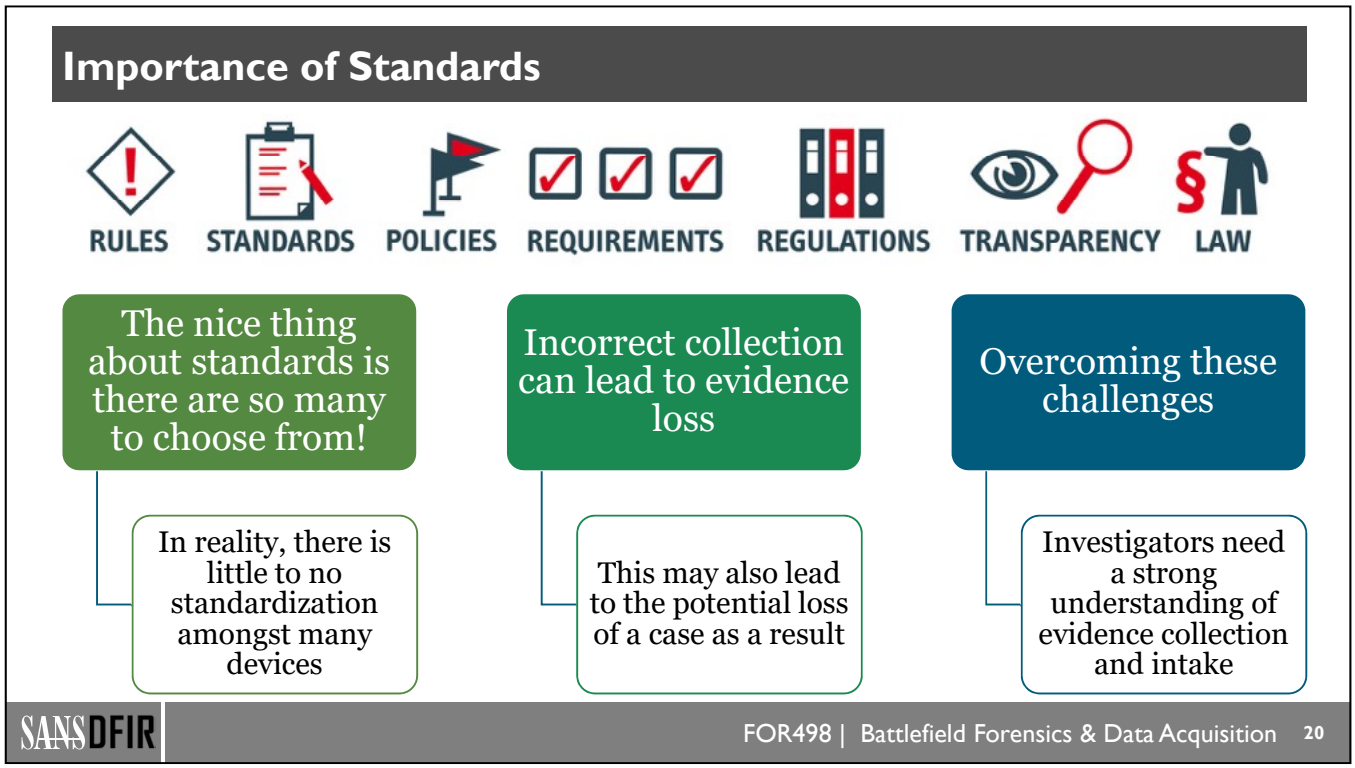
- Cellular telephone
- Laptop/tablet/desktop/USB storage
- Smart watch/iPod
- Apple TV/Chromecast
- Echo
- Vehicle infotainment system/GPS
- Appliances
- Security cameras
- Home automation

Beyond traditional devices, we must also consider the devices that connect the devices. Devices such as switches, routers, firewalls, Internet of Things (IoT), cloud storage, and social media apps are also involved in most every facet of most every investigation. In fact, we are exposed to tech in ways we don't even realize. We may think we are driving an automobile, but in reality, we are driving a computer with wheels.

We are very rapidly hurtling towards a day when the device in our hands is merely a connection mechanism to a storage repository somewhere else. In many cases, we are already there!

Incident response cannot ignore the devices that connect the devices. With the advances in computer security, the adversary is being forced to look for other places to hide. They are now hiding in devices that connect devices, because nobody is looking there. Yet each of these devices contains an operating system and at least some form of storage.

These devices are often ignored because the thought is that, for example, a switch is merely a "pass through" mechanism. However, what if the adversary has compromised the switch and the settings, and is altering the Address Resolution Protocol (ARP) table in order to create a "man in the middle" scenario? You are losing data, but since it is not happening from an actual computer, you will not find its existence.

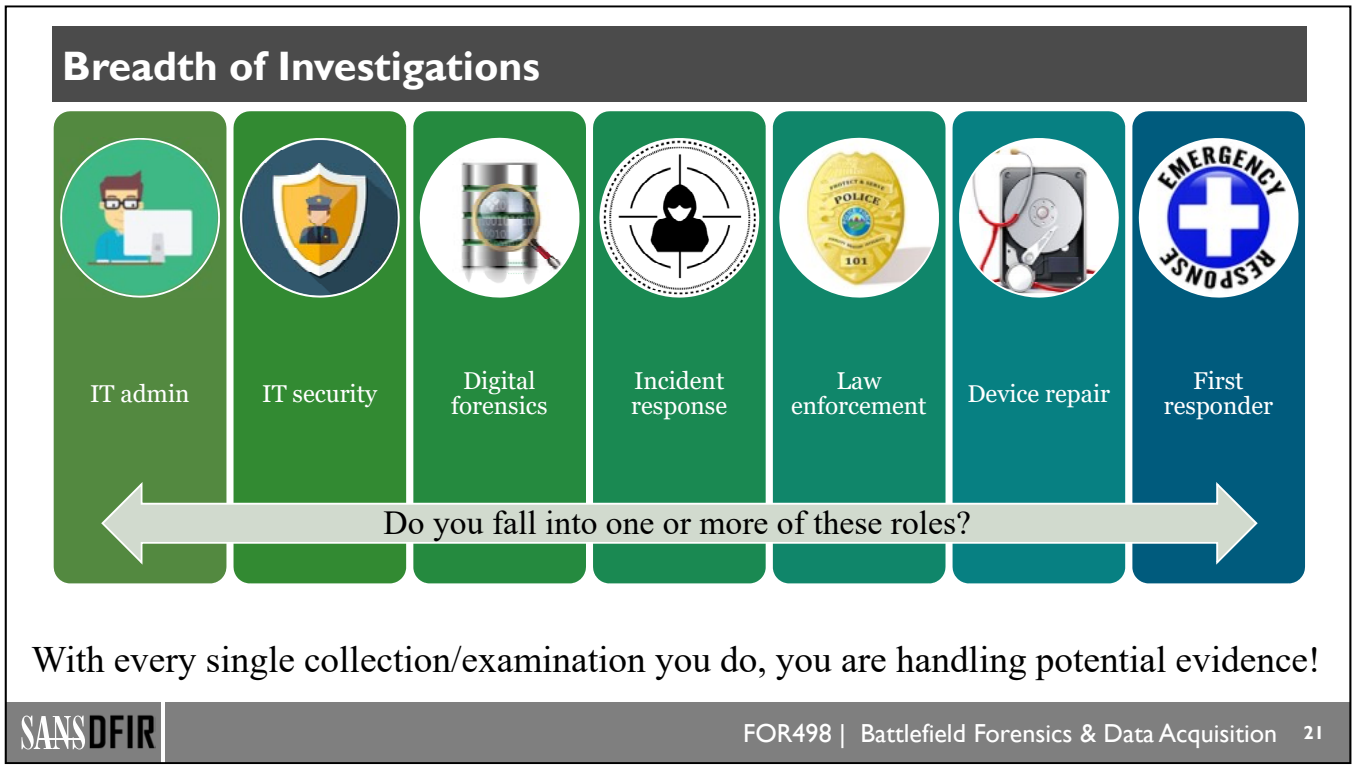


The difficulty for people that are tasked with investigating this tech, is that there cannot possibly be any standard acquisition process for every instance. What we CAN have though, is a strong understanding of the evidence intake and collection process, to ensure that if challenged, we are not losing valuable evidence simply because we did not collect it properly.

Often times in the past, we have seen examiners get away with using excuses such as, “Well we did our best”. This is unacceptable today. If a decision has been made to handle a piece of evidence (or even simply an artifact) in a certain way, the examiner must be able to explain the reasoning. “We didn’t know” is not adequate reasoning.

Today more than ever before, we have a vast repository of data being shared by the digital forensics community in unprecedented volumes and endless ways. In the early days, the difficulty was that there was a complete lack of information. Now it is the opposite. Our challenge is now finding the knowledge within the noise.

Imagine a scenario 20 years ago, where there was virtually no information about forensics. It was essentially being made up as we went along, based on what we believed to be best practice. The mistakes have been made for us by those that went before us. We can no longer use lack of information as an excuse.



Although this course is geared to most anyone that may be involved in any facet of investigation, collection, etc., do not think that just because you are not a police officer, that proper collection does not apply to you.

Look at the following list:

- IT Administration
- IT Security
- Digital Forensics
- Incident Response
- Law Enforcement
- First Responder
- Electronic Device Repair

Does your role fall within any of those categories? Chances are, if you are taking this class, the answer is yes. As such, every time you are collecting any data, or forensicating something, you are handling potential evidence that could find its way into a court room.

Given that you cannot go back and recreate things (or "unbake the cake"), you must ensure that collection and handling are performed properly from the very start of contact with potential evidence.

Importance of an Examiner's Role

- You are the difference between:
  - Someone losing their job or not
  - Someone losing their freedom or not
  - Someone losing their reputation or not
  - Someone losing their family or not

- The importance of your role cannot be overstated.

The importance of your work cannot be overstated. It is easy to become complacent and believe that today is just another day, but you must understand the potential impact of the findings of your work. The work done by anyone in the previous list can be the difference between someone losing their job, or keeping their job; or in extreme cases, the difference between going to prison or not. The stakes are potentially very high, and the examiner has a huge responsibility to their craft.

The significance of the outcome in an investigation cannot always be immediately noted at the beginning. Unfortunately, you cannot go back in time and re-collect the data. Given the potentially incredible importance of the outcome, you must absolutely get it right the first time. You must approach every investigation as though it was YOUR future hanging in the balance, because in many cases, these are the stakes.

Attorney Craig D. Ball, PC, has a fantastic resource for deeper diving into court related matters.[1]

[1] Craig Ball Website on forensics and the legal world | `https://for498.com/uvyt7`

Types of Witnesses

Fact witness

- Also referred to as "lay" witness
- Can only testify to what they saw/heard/did

Expert witness

- Can provide opinion or hypothesis based on unique and advanced knowledge
- In the United States, Daubert is the standard

The fact witness is an individual who is knowledgeable towards the facts of the case through a direct participation or observation of the intricacies involved. For example in a murder case, a fact witness would be an observer of the actual murder or an acquaintance of the individuals involved in the case. The fact witness simply delivers truthful statements regarding the character of those involved or an account of what they saw take place.

In contrast, an expert witness is an individual who holds a specialized knowledge in a particular educational field concerning the case. For example, an expert witness can be a doctor who is well-versed in a particular field of medicine. As an expert witness the individual will use his or her advanced knowledge of a particular subject to elucidate on a piece of information regarding the trial to facilitate an appropriate verdict. In the United States, the standard used to qualify someone as an expert is Daubert[1].

[1] Daubert standard - Wikiwand | `https://for498.com/d7jv2`

Opinion vs. Evidence

"A conclusion that isn't based on evidence is an opinion. It's okay to have opinions because they help drive the pursuit of evidence to fill gaps. It's not okay if you're unable to recognize when your conclusions are based entirely on opinion."

"Recognizing the rigor of your conclusions is a skill that builds on your sense of self-awareness and your ability to ask the right questions, consider alternative timelines, and retrieve and evaluate evidence. It's not trivial, but you can build this skill."

Chris Sanders

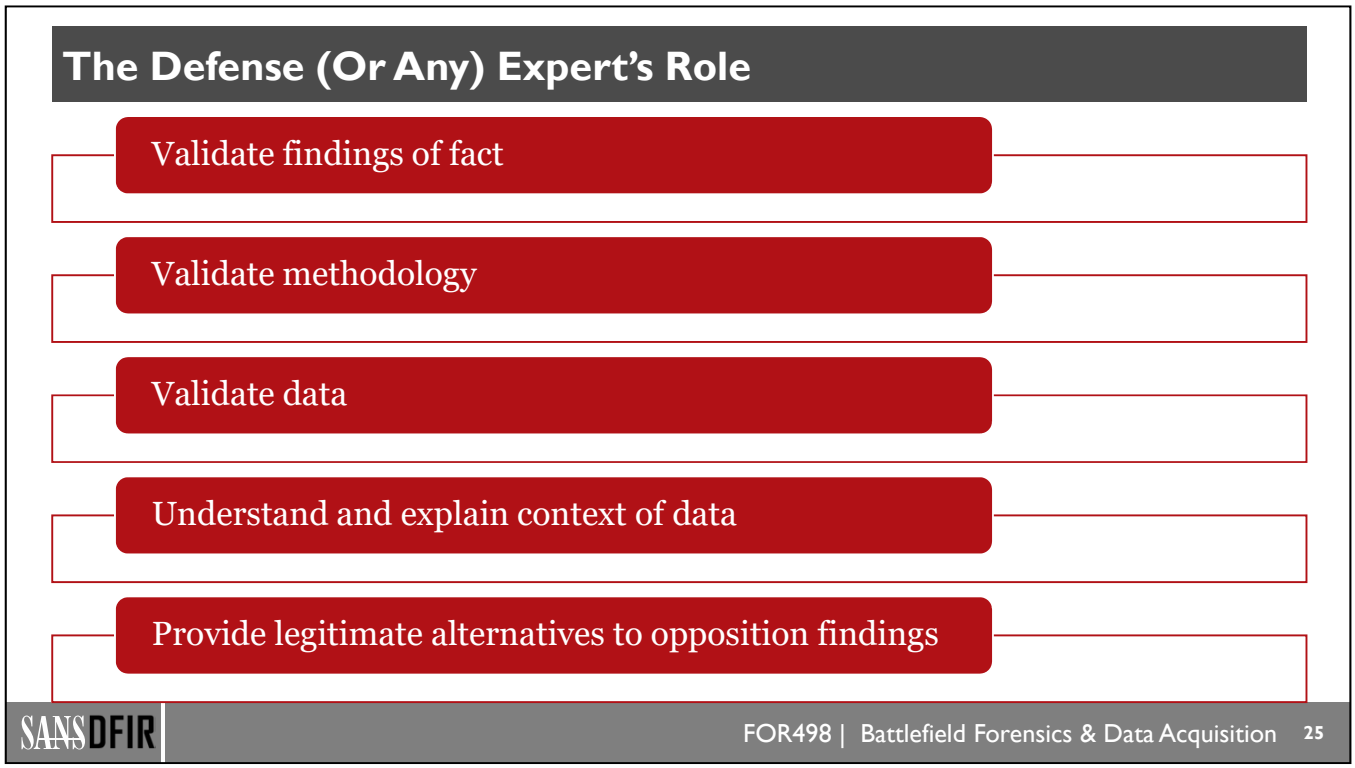
A conclusion that isn't based on evidence is an opinion. It's okay to have opinions because they help drive the pursuit of evidence to fill gaps. It's not okay if you're unable to recognize when your conclusions are based entirely on opinion.

Recognizing the rigor of your conclusions is a skill that builds on your sense of self-awareness and your ability to ask the right questions, consider alternative timelines, and retrieve and evaluate evidence. It's not trivial, but you can build this skill. [1]

Chris has also created an online course that helps with expanding the investigator mindset when it comes to computer and network investigations. [2]

- Syllabus
- Metacognition: How to Approach an Investigation
- Evidence: Planning Visibility with a Compromise in Mind
- Investigation Playbooks: How to Analyze IPs, Domains, and Files
- Open Source Intel: Understanding the Unknown
- Mise en Place: Mastering Your Environment with Any Toolset
- The Timeline: Tracking the Investigation Process
- The Curious Hunter: Finding Investigation Leads without Alerts
- Your Own Worst Enemy: Recognizing and Limiting Bias
- Reporting: Effective Communication of Breaches and False Alarms 10. Case Studies in Thinking Like an Analyst

[1] Chris Sanders – Information Security Analyst, Author, and Instructor | `https://for498.com/j4xn0`
[2] Investigation Theory Training | `https://for498.com/1d92n`



In many cases, both civil and criminal, the examiner's role is to educate the trier of fact. The trier of fact is the Judge, or the Jury, depending on the type of proceeding. It is the examiner's job to take complex ideas and processes and explain them in a way that is understandable. Even from a defense expert's perspective, they are not there for the movie-style GOTCHA moment (although some think they are).

Especially for (but not reserved to) defense experts, your function is not to win the case for your client. Your role is to provide the evidence in an understandable way that is non-judgmental and completely unbiased. Stick to the facts only, and let the lawyers act out the theater. There is no such thing as a question that you should be afraid of in court. Always tell the truth, never embellish, and never use the words "always" and "never". If opposing counsel ever asks a question for which the answer is unfavorable to your client, remember that for all intents and purposes, as a court accepted expert, your end client is the trier of fact.

Validate Findings Of Fact

Are the facts as outlined in the charging instrument/statement of claim (indictment or information) correct?

Did the Defendant/Respondent, on or about the time and date alleged, with the corresponding culpable mental state, commit the acts that would constitute the elements of the offense/tort?

Validate Methodology

Did the examiner utilize the methodology that would yield the most complete results?

Did the examiner utilize a methodology that can withstand scrutiny?

Did the examiner use acceptable industry standard practices?

If not, where did it stray, how did it stray and what effect did it have on the findings of facts?

Validate Data

Are the files alleged to be on the drive, actually on the drive?

If the files/data are actually on the drive, are they what the Prosecutor/Plaintiff claim them to be?

This should be the easiest to validate.

Understand And Explain Context Of The Data

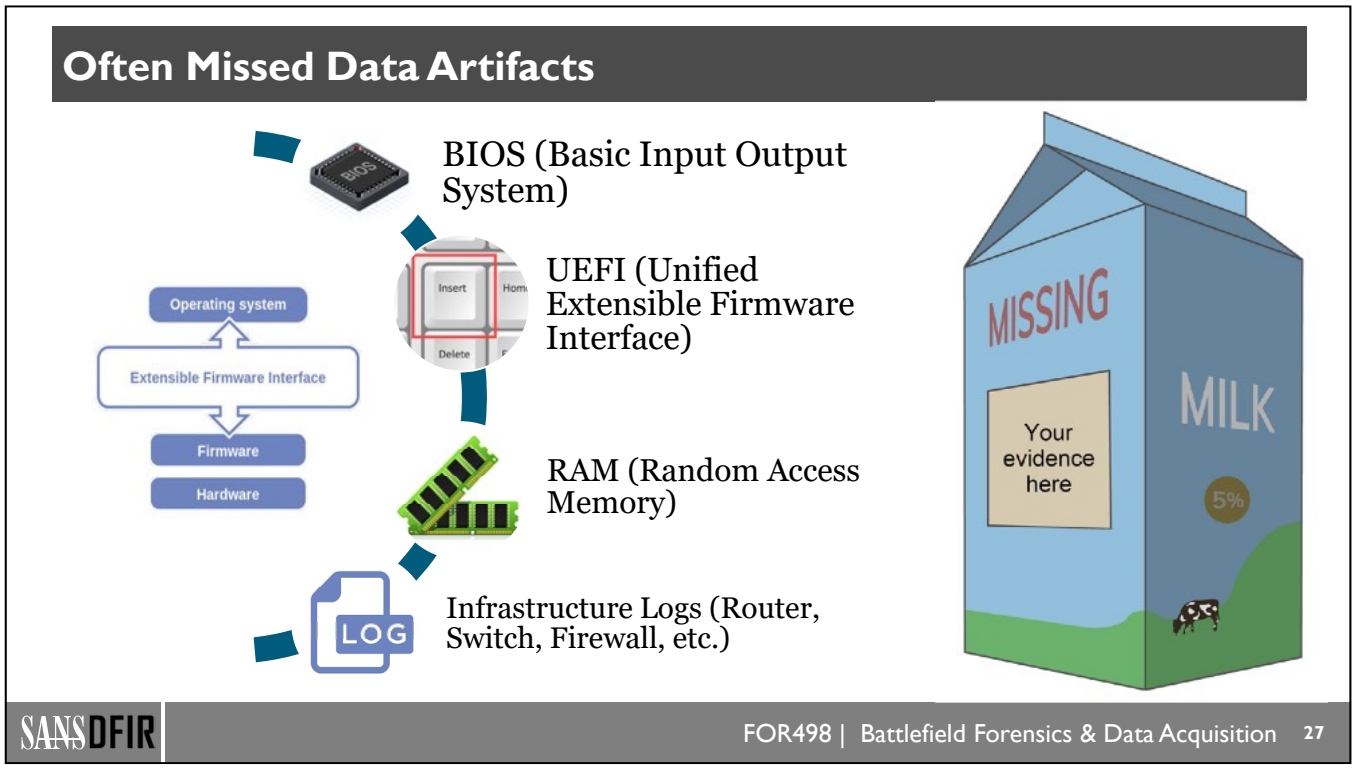
How did the data get there?

What was the environment that existed on the computer at the time the content was introduced, and did that contribute to, or is that environment responsible for the data being present?

How did that data interact with the other files found on the drive?

Provide Legitimate Alternatives To Opposition Findings (If One Exists)

This does not mean the SODDI (Some Other Dude Did It) defense, nor does it mean the virus defense, unless of course these are legitimate possibilities.



There can be no question that 8 GB or more of data is a significant amount of data. A computer system's Random Access Memory (RAM) holds important, even critical, data that is found nowhere else on the computer. Yet in many investigations, this much data and more is routinely "thrown away", because an examiner does not understand the importance, or simply does not know how to collect it.

Network connections, running processes, encryption keys including Bitlocker keys, passwords, malicious software, chat logs, and many more artifacts are examples of the types of evidence that are routinely extracted from memory. More importantly, these same artifacts almost always do NOT exist anywhere on the hard drive. In any given case, it can be very likely that exculpatory, or indeed inculpatory, evidence exists in memory, and nowhere else. More than ever before, if the computer is on at the time of seizure, there is almost never a reasonable excuse, other than personal safety, to not forensically image the RAM. Certainly the lack of preservation will be a sticking point in any legal proceeding.

The Basic Input Output System (BIOS)[1], and more recently, the Unified Extensible Firmware Interface (UEFI)[2] on a computer contains information critical to the interpretation of data. The date and time used by the computer to label file metadata is derived from here, and as such it is critical that this information is collected at the time of device seizure. Without it, an analyst has a much more difficult, if not impossible, task of correlating the actual date/time that an activity occurred.

[1] BIOS | `https://for498.com/9lfcy`

[2] UEFI | `https://for498.com/ywrsm`

You Can't Unbake a Cake

Measure twice, cut once

- Many investigations have been compromised before they start, simply from improper evidence collection.
- There are artifacts that may be destroyed completely, or rendered entirely inaccessible, due to incorrect decisions being made by a first responder.
- Is there active encryption on the drive of a computer that is on at the time of seizure? Hopefully you did not shut it off before then!

In the case of whole disk encryption, how an examiner responds in the initial stages could be the difference between extracting evidence from a computer, and never being able to access the computer data. Although an examiner will, in most "dead box" cases, be able to forensically image a hard drive that contains whole disk encryption, the examiner will not be able to mount the image or access the data without passwords/decryption keys, if at all, simply because the data is encrypted at rest.

A very important question to ask at the outset (if you have the luxury) is, "Has anyone attempted any form of self-investigation?" Many an investigation has been significantly impeded by someone poking around through a live system trying to find answers before you were called.

Current ISO Standards

ISO/IEC	27037:2012 – Guidelines for the identification, collection, acquisition and preservation of digital evidence
	27041 – Guidelines on the assurance aspects of digital forensics (Ensuring that the appropriate methods and tools are used properly)
	27042 – Guidelines on what happens after digital evidence has been collected (analysis and interpretation)
	27043 – Guidelines on broader incident investigation activities, within which forensics usually occurs
	27050 – Concerns electronic discovery (in 4 parts)

Standards exist for the collection and handling of digital evidence. The International Organization for Standardization[1] is the body that oversees ISO Standards. The standard that is related to digital evidence is ISO 27037:2012[2], which governs "Guidelines for the identification, collection, acquisition and preservation of digital evidence."

There are other ISO standards[3] specifically pertinent to digital forensics, such as:

ISO/IEC 27041 offers guidance on the assurance aspects of digital forensics e.g. ensuring that the appropriate methods and tools are used properly.

ISO/IEC 27042 covers what happens after digital evidence has been collected i.e. its analysis and interpretation.

ISO/IEC 27043 covers the broader incident investigation activities, within which forensics usually occurs.

ISO/IEC 27050 (in 4 parts) concerns electronic discovery.

A deeper dive into the family of ISO27k standards can be found at `https://for498.com/yw9zg`

Fun fact[4]: ISO does not stand for International Standards Organization, and is not even an acronym, which is widely believed. It is derived from the Greek word 'ISOS', meaning 'equal'.

[1] ISO | `https://for498.com/xk13l`

[2] ISO standard regarding digital evidence | `https://for498.com/4h-r3`

[3] Other forensics related ISO standards | `https://for498.com/jwtle`

[4] ISO as acronym | `https://for498.com/g5frc`

Getting There from Here

Lethal forensicators are not born or made overnight

- It is one thing to take a course and push a button
- We want to understand the "how" of things as much as or more than the steps to follow when processing evidence

How you get to "lethal" is the same as how you get to Carnegie Hall…

- Practice, practice, practice!

Lethal forensicators do not automatically become so. There is a big difference between the button pushing forensics that is all too common, and the rarified air of the forensic elite.

After working in the field for a great many collective years, the authors have seen far too many environments where examiners are expected to push a magic button and get all the things. At the same time there are far too many environments where capabilities are not expanded on or continuously upgraded. There is no single tool today (nor was there ever) that can perform a complete forensic examination. To become truly lethal, you must not only know what is the best tool for a particular task, but you must also be able to understand and explain how that tool is working.

With that said, knowing what to do is only part of it. Being able to efficiently select and use the best approach for a given task is a skill born of many repetitions and the refinement of technique over time. Like any skill that a person becomes proficient in doing, it takes many successes, and often times failures, to reach the top.

Embrace the Fact That You Do Not Know It All

Recognizing that you know what you know, and recognizing that you do not know what you do not know - this is knowledge.

- Confucius

To be lethal, you must always have an open mind, share knowledge, test theories, and probably most importantly, understand that you do not know what it is that you do not know[1]. Embrace this.

In most every part of the digital use realm, this adage applies. Even more incredibly so in the digital forensics world. Far too often, the author has heard from supposed "experts" in the field that something is a certain way, and it turns out to be false. For example during testimony in a case, an expert testified under oath that a cookie file was being used to hide a picture. The problem was that the picture was 12 KB in size. (No, alternate data streaming was not in use). At the time, and for the particular browser in question, the maximum cookie size was 4 KB. In this case, the expert couldn't understand what the forensic program was telling him, so he guessed at the answer.

As indicated previously, the impact of your role as a forensicator cannot be overstated. Peoples' lives are significantly affected by your findings, and it behooves you to "get it right".

If you approach every investigation (and especially any specific finding to be used in court) with the above in mind, it will hopefully cause you to want to verify your work, and double check it. Just because something was the way it was in Windows 7, does not mean it is that way in Windows 8. In fact, there are a number of settings and artifacts that differ between Windows 7 32 bit, and Windows 7 64 bit!

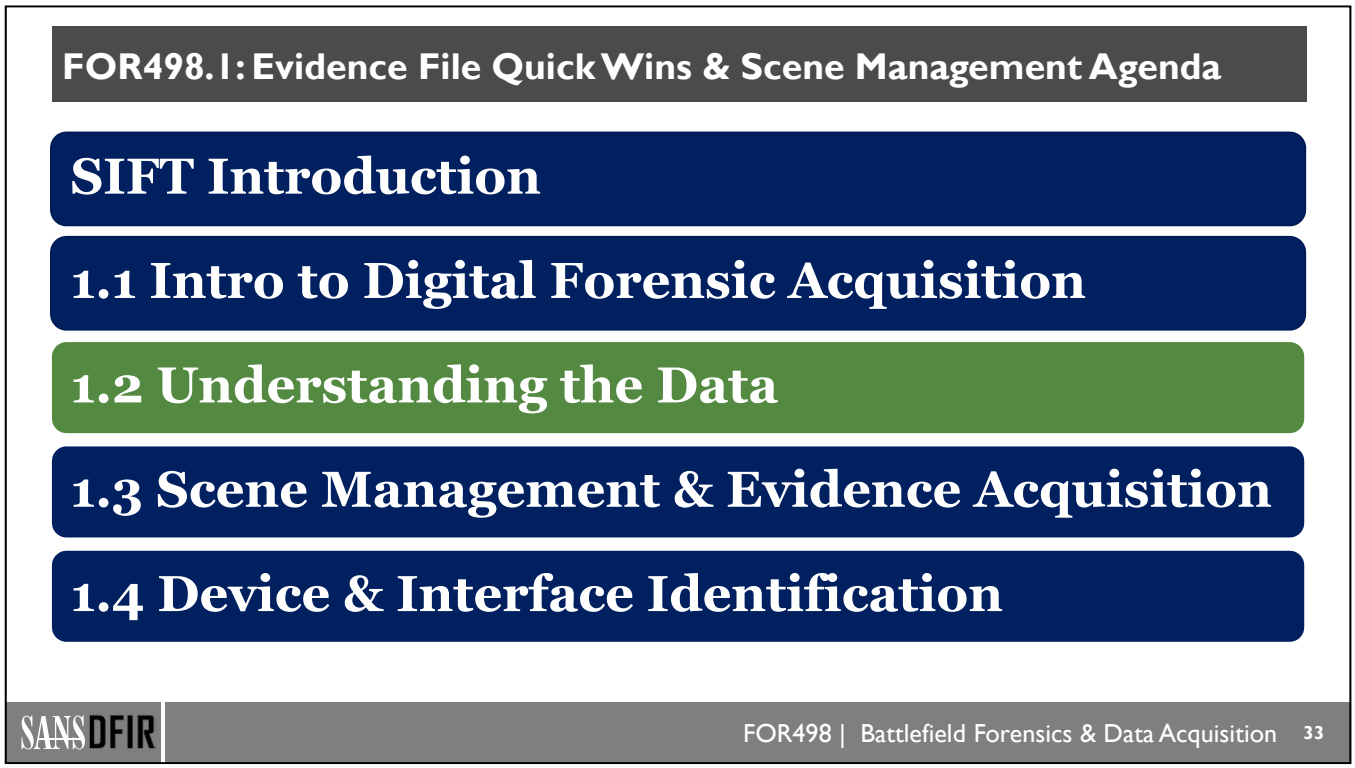
Digital forensics is definitely not the field to be in if you plan on working "9 to 5" only.

[1] Confucius on Knowledge: Knowing What You Know; Knowing What You Don't Know
`https://for498.com/upjkn`

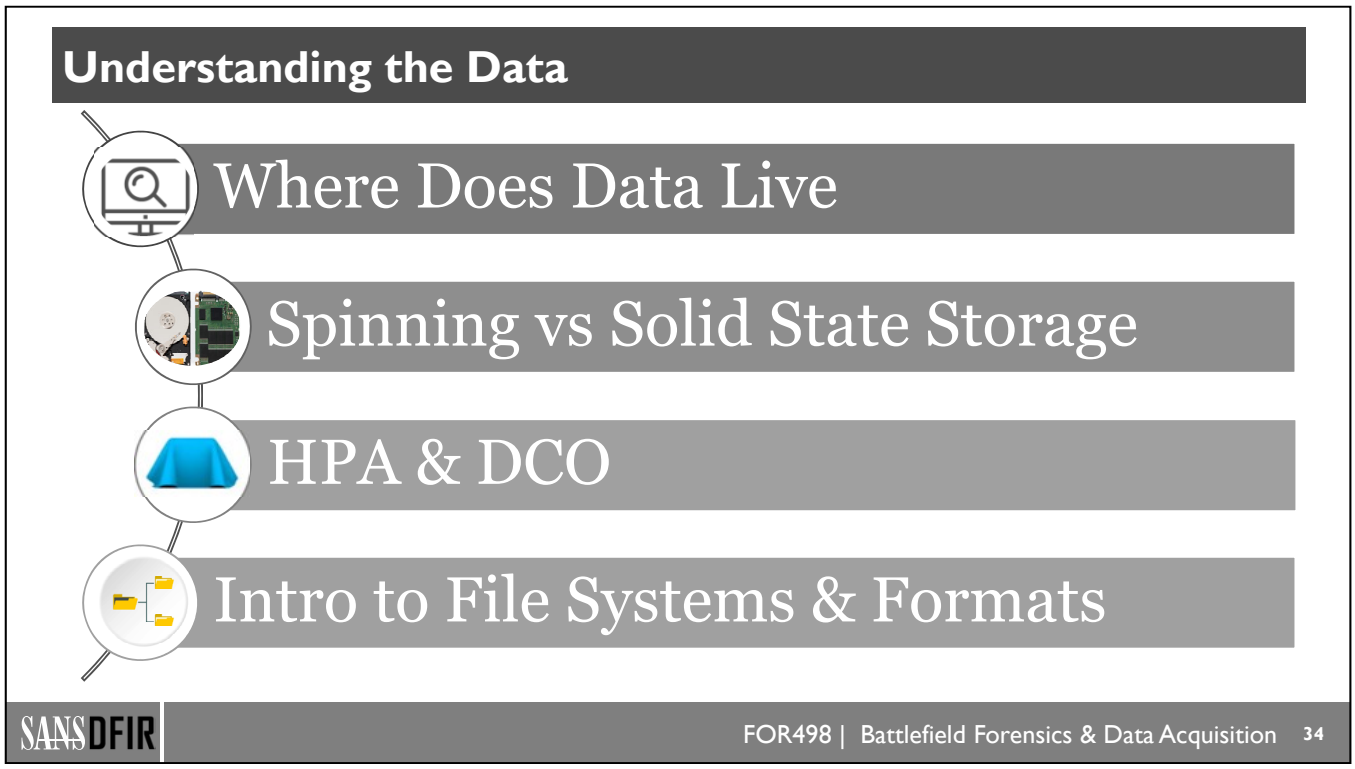
Summary

- Do your own research to dispel the myths
- Storage media is getting larger and larger
- Data is more than just what we get from a hard drive
- If you are touching data, treat it like evidence
- Treat each forensic examination as though your future depends on it
- You can't possibly know it all

This page intentionally left blank.



This page intentionally left blank.



This page intentionally left blank.

Where Does Data Live These Days?

- Phone
- Network
- Traditional
- Internet of Things
- Cloud storage
- Removable
- Unorthodox

As our lives are connected more and more by technology, we find ourselves (and will continue to do so) having to deal with more and more devices that contain a larger and larger amount of storage. To further complicate this, the devices that access and store data may not even possess this capability in and of themselves. Cloud storage and device syncing present new challenges in the realm of digital evidence collection.

Cloud services are popular because they can reduce costs, bring advanced capabilities to a platform that may not be possible otherwise, and allow for seamless expansion, as the cloud provider handles the hardware, bandwidth, and storage necessary for the cloud service to scale properly [1].

We cannot of course, ignore the sources of data that have been around for decades which includes traditional/removable storage such as mechanical hard drives (both internal and external) and thumb drives. The available capacity on these devices has grown rapidly in recent years, and as densities continue to improve, these devices will contain even more data.

Finally, we have things like network storage devices such as file servers, dedicated network attached storage (NAS) devices, and the ubiquitous ”Internet of Things” that continues to grow in popularity. Connecting things from light bulbs to dishwashers is becoming the norm, so it is imperative for investigators to understand how these devices store and access the data they need to operate. It is estimated that by the year 2025, there will be over 75 BILLION devices connected to the Internet, all sending data to a myriad of places, some of which could be vital to an investigation [2].

[1] Why are cloud computing services so popular? | `https://for498.com/a5j1b`
[2] IoT: number of connected devices worldwide 2012-2025 | `https://for498.com/vb539`

Where Does Data Live These Days: Phones

- Built in storage ranging up to 512 GB
- Optional MicroSD storage up to 512 GB
- Cloud syncing
  - 5GB to unlimited
  - Who has this data? How do we get it?

Cell phones continue to push the boundaries of what is possible with a mobile device. Faster processors, bigger built in storage, ever increasing bandwidth with the rollout of 5G services [1], and more will continue to raise the importance of phones in just about every investigation.

Some phone manufacturers, like Google, offer unlimited cloud storage/syncing with their devices for things like photos and videos [2]. To the end user, this unlimited storage is seamless in that the phone manages backing new photos and videos up to the cloud and keeping everything in sync. This of course raises the possibility of recovering data even if the originating device is lost, stolen, or damaged to the point where it cannot be accessed. By being able to target and collect against cloud providers, entire new avenues of information become available to digital forensics practitioners. Other services like iCloud can contain far more than just photos as well, including files, SMS, and so on. Up to 5 GB of data is included, with the option to buy additional space as needed [3].

The capabilities of cameras in mobile devices continues to improve as well, which means better quality photos with more details, EXIF data, etc. being available for use during an investigation, provided you know how to locate and preserve such data.

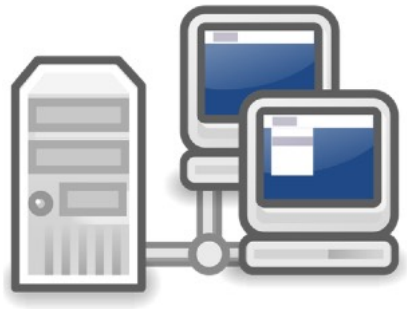
Additional questions to consider:

- What country is the data physically located in?
- Is legal process in your country available to compel the production of data stored in the cloud depending on where it is located?
- Is there more, or less, data available via legal means vs accessing the data with a username and password?

[1] A 5G Device Timeline for 2018 & Beyond | `https://for498.com/j276i`
[2] How the Google Pixel's unlimited photo and video backup works | `https://for498.com/c8zku`
[3] iCloud - Apple | `https://for498.com/tpl5i`

Where Does Data Live These Days: Network

- Traditional file servers
  - Mapped drives
  - Universal Naming Convention (UNC paths)
- Network Attached Storage (NAS) devices
  - Provide the same features as file servers
  - Can also run a multitude of other services
    - Radius
    - Web servers
    - VPN/proxies



Computer networks have existed for decades in one form or another. While the protocols that manage networks have changed, the goal of a network is the same: to share resources among a group of computers and users. In many environments, this involves the Server Message Block (SMB) protocol. SMB provides the following:

"The Server Message Block (SMB) protocol is a network file sharing protocol that allows applications on a computer to read and write to files and to request services from server programs in a computer network. The SMB protocol can be used on top of its TCP/IP protocol or other network protocols. Using the SMB protocol, an application (or the user of an application) can access files or other resources at a remote server. This allows applications to read, create, and update files on the remote server. It can also communicate with any server program that is set up to receive an SMB client request." [1]

Windows contains built-in functionality to expose files and directories using SMB. Another alternative for Windows as well as Linux is Samba [2].

This of course includes the sharing of files, programs, and directories, i.e. data! When a network is involved, we not only have to deal with the data being stored on various network resources such as file servers and Network Attached Storage (NAS) devices, but we may also be interested in the network traffic itself that enables this data sharing between servers and clients.

We have been involved in many cases where the only physical machine in a location contained no direct evidence of the crime. Upon closer look, the bad guy was using virtual machines to run different instances of Windows for nefarious purposes. Inside these VMs, and only inside these VMs, did the subject then connect to the NAS device that was also on the premises. If we were to have just looked at the physical machine, we may not have found the evidence we needed, but by locating both the VMs and the network related storage, the case was made using a combination of triage and follow up questions to the subject. Without the initial triage, questioning the subject would have been much less productive as we would have had to take his word for things vs. going into an interview knowing a good deal about how his infrastructure was set up.

A NAS device is another very common way for files to be shared on a network. These devices are typically small, dedicated computers that an end user can add one or more hard drives to. The NAS device has management software that is used to configure the device, set up permissions and security, install other applications to provide additional services beyond file sharing and so on. Some NAS devices have hundreds of available packages [3] including programming languages, web servers, chat servers, audio and video streaming, VPN, blog software, and backup clients that can push the contents of the NAS out to the cloud. Additionally, NAS devices are often used as the location for backups of other computers on a network.

Devices that fall under this category will often have one or more storage devices that fall under our next category for discussion: traditional.

[1] Overview of file sharing using the SMB 3 protocol in Windows Server | `https://for498.com/syv0h`
[2] Samba - opening windows to a wider world | `https://for498.com/p5t-m`
[3] Packages Synology Inc. | `https://for498.com/5d1g6`

Where Does Data Live These Days: Traditional

- The most common type of device you are likely to encounter
- Depending on the type of storage device, special hardware may be required
  - Obscure tape drives
  - Iomega anything!

In its simplest form, storage devices that fall into this category would be items like a 3.5-inch internal hard drive. These devices range in size from several hundred megabytes to many terabytes! [1]

For internal storage devices, larger capacity drives would use a Serial Advanced Technology Attachment (SATA) [2] interface whereas older, smaller devices would use Parallel ATA [3]. The type of interface present on a storage device would dictate the kinds of equipment needed to access the data on the drive. Most modern computers only contain SATA interfaces, but adapters exist that can easily convert between SATA and PATA interfaces. PATA cables typically use 40 or 80-wire ribbon cables that are two inches wide, whereas SATA cables have 7 wires and are generally a lot smaller in size, coming in at around a quarter of an inch.

Other common situations that fall into this category include RAID arrays, which are storage pools made up of typically more than one storage drive. RAID (Redundant Array of Independent Disks) will be covered in a future section in detail, but for now, understand that several smaller disks can be "combined" into what appears as a single larger volume via a combination of hardware and/or software.

Finally, archival storage devices such as tape drives and older, legacy storage devices such as Zip disks, Bernoulli disks, etc. may be encountered. In these cases, some leg work is often required to track down a device that can access data stored on things like Zip or Bernoulli drives. Organizations such as the FBI have been collecting these devices over many years in case they, one day, seize something that requires such a device. In fact, the FBI has shelves full of old devices that can be leveraged should the need arise during a case.

[1] Seagate IronWolf and IronWolf Pro NAS Hard Drives | `https://for498.com/zvlwq`
[2] Serial ATA - Wikipedia | `https://for498.com/i9p7q`
[3] Parallel ATA - Wikipedia | `https://for498.com/ors3-`

Where Does Data Live These Days: Internet of Things

- The latest wave of devices may have little, if any, local storage
- How can you tell a device's network capabilities by its physical appearance?
- Once you determine where the data is, how do you get it?




SANS DFIR | FOR498 | Battlefield Forensics & Data Acquisition 40

IoT. We have all heard the term, but how often have we had to actually investigate an IoT device? With the recent explosion of devices that fall under the category of the Internet of Things, this scenario will become more and more prevalent in an ever-widening range of cases.

Consider the popularity of technology such as Ring doorbells which automatically upload high quality video to an off-premises location. It is not only limited to doorbells, however. Ring also provides high definition web cams that can push video to the cloud for later viewing for up to 60 days! [1]

The challenge with IoT investigations is twofold. The first problem is knowing the devices exist in the first place. The second is then figuring out exactly how to access and preserve the data that is available.

For the first problem, often the easiest way is to observe network traffic to see the kinds of devices that are communicating. This will often include information like IP addresses (both internal and external), Media Access Control (MAC) addresses (which can lead you to a manufacturer), and so on.

For several years now, people have observed the need for a way to "discover" IoT devices in an environment. Various proposals and specifications have been put forth to allow for the automatic registering of IoT devices into a kind of catalog that can be referenced. [2] Depending on the types of devices, these registrations may be public or private. An example of a framework that handle this kind of thing is Simurgh, which allows for the "Effective Discovery, Programming, and Integration of Services Exposed in IoT". [3]

The second problem is only solvable once the first issue has been addressed. Once a device is discovered, additional research can be done on the device itself based on the manufacturer, labels on the device that may point to a MAC vendor, and so on. Depending on how customized the network hardware is, the MAC address

can lead to where data may exist. Another avenue of investigation is observing and recording the network traffic between IoT devices to see where it is communicating with. The network traffic will contain both IP addresses and ports used for this communication, and this can then be used to determine the owner of the IP addresses, etc.

With the location of the data sorted out, the last step in this phase is to determine how much data is available on the far side of the device. For example, have months' worth of videos been uploaded to a company in California? Does a record exist of every opening and closing of an exterior door that requires a keycard and PIN for access?

Once the location of the data is discovered, legal process (such as a search warrant) can be used to acquire a copy of the data should there not be other means of accessing the data. Consider the case where a device has uploaded video of a location for a date you are interested in. A simple way to gain access to this data would be getting consent by the owner of the device to access some kind of portal that lets you view stored videos. If this is not possible, legal process authorizing the seizure of the data may be necessary.

[1] Ring Security Cameras | `https://for498.com/8rymj`

[2] IoT Needs Open Discovery Scheme | `https://for498.com/2k031`

[3] Simurgh Framework | `https://for498.com/1l23w`

Where Does Data Live These Days: Cloud Storage

- Different from devices that sync data via the cloud
- Options include
  - OneDrive
  - Dropbox
  - Box
  - Egnyte

Recall from our discussions with phones that there are several services, such as Google Photos, that can sync pictures and videos taken on a device and automatically backup/synchronize them to a remote location. In this category however, we are not talking about that kind of cloud storage, but rather the kind that allows end users to store any kind of file and/or directory and have it replicated on a remote system.

These solutions typically involve a client of some kind to be installed on a computer which then designates a "home" folder for the client. Anything placed in this home folder is then uploaded and kept in sync with the remote host [1]. Once data is synchronized, files and folders can then be shared with other users of the same service or shared via unique URL. When sharing with other users on the same service, it becomes possible for all users with access to a shared folder to add and remove files to the share. Any changes made would then be reflected to all other users who have access to that share. To further complicate things, anyone with access to the share can manipulate (upload, download, delete, etc.) files via the vendor's web site in addition to using the client software typically found on a user's computer. Finally, different services allow for setting different levels of permission on shared items (read-only, full access, etc.), offer encryption for the files being stored, and so on, making it a very flexible offering that is capable of meeting the needs of a wide range of people.

Another aspect of cloud storage providers is that most also maintain revisions of files as they change, including the ability to restore files if they are deleted. This is useful in many cases because it allows you to show how a file of interest changed over time, a mass deletion of objects on the account, and so on. While having the user's credentials to the service is required for some of these activities to be possible, it is also possible to serve the provider with legal process in the form of a search warrant, that compels them to not only turn over content, but metadata about the account, including file creation, modification, and deletions.

In fact, it is often a requirement to involve the cloud provider via a search warrant [2], as the data is most likely not recoverable by any other means. In many cases, the data is stored in a proprietary manner in such a way that would make recovering the data impossible, even if you had access to all of the storage devices where the files

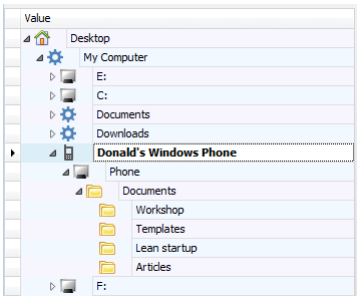
were kept. Additionally, how would you know which files belonged to the user of interest when there may be 100 or 1000 other peoples' files on the same storage devices? As mentioned above, even with a valid search warrant, you may not be able to access the contents of cloud storage if the files were stored in an encrypted fashion. Some providers have the means to decrypt via some kind of shared key whereas other providers allow end users to specify their own encryption key. Without this unique key, recovery of the files in a non-encrypted state is impossible.

[1] How Cloud Storage Works | `https://for498.com/m70hx`

[2] Executing Search Warrants in the Cloud | `https://for498.com/idwpb`

Where Does Data Live These Days: Removable Devices

- USB devices are everywhere
- Not just limited to thumb drives or external hard drives
  - Connecting a phone to a computer exposes internal storage to the host



Removable devices will typically use a USB connection (generally USB2 or USB3 these days), but other connections are possible, including eSATA and Thunderbolt connections. Almost all computers these days have USB ports and as such, USB based external storage is very common, especially as the movement to smaller and thinner laptop computers becomes more popular. Because of how thin laptops are becoming, there is not enough room physically to include something like a DVD reader. Because of this, external DVD readers are often connected via USB as well.

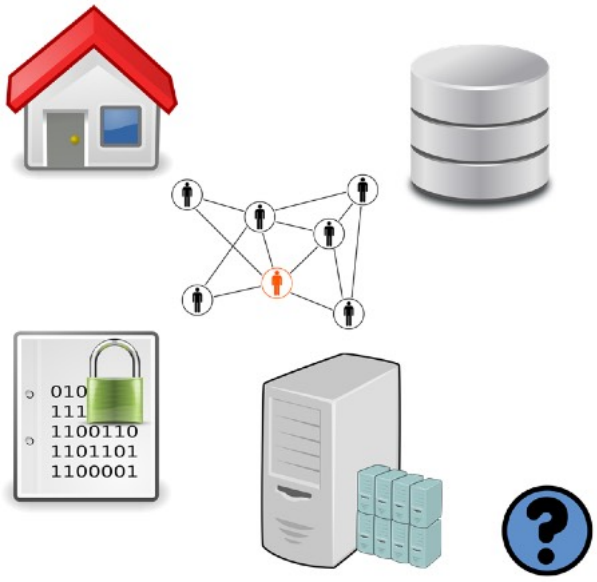
Just like we saw with traditional drives, external drives are also growing in capacity at a rapid pace. In fact, if you have the money, thumb drives of 2TB and larger are available. These modern thumb drives are fast too, reading and writing at well over 200 MB/sec!

One positive side of removable devices from a forensics point of view is the rich amount of evidence that exists as it relates to their use. When devices are connected, removed, reinserted, and/or accessed, the operating system tracks these activities, sometimes in more than one place, in places like the registry, event logs, text log files, etc.

As we will see, there is more to dealing with removable devices than simply knowing they exist. In some cases, the actual device itself may not have been located or seized. Nevertheless, techniques exist that allow you to determine what happened on a computer that involved these removable devices. LNK file analysis, shellbags, and registry forensics provide windows into what was done using a removable device and can significantly improve your understanding of a user's actions on a computer. We will use some of these techniques in other sections of the course.

Where Does Data Live These Days? - Unorthodox

- Self-hosted cloud storage
  - Synology Drive
  - OwnCloud
- P2P networks
  - Retrieve data as needed
- Virtual hard drives
  - VHD(x)
- Virtual machines in general
- Encrypted containers



Over the last decade or so, high speed Internet connections have become the norm [1][2], making it possible to move large files quickly from person to person. The rise of Peer to Peer (P2P) programs [3], from Napster to LimeWire and beyond, created entirely new ways for people to share whatever kinds of files they wanted, from music to videos, or even crocheting patterns. Since many of these P2P networks are made up of computers that only participate in the network while a piece of client software is running, investigating and collecting data from P2P networks can be problematic. Quite often, the solution for operating on these networks is very specific, custom software that only works on a given network.

Another result of fast network connections at home is the proliferation of self-hosted cloud-based storage. These types of software programs allow users to effectively share storage devices on their network with whomever else knows about it. In some cases, this may be a single user, but in other cases, these self-hosted solutions can serve as a kind of closed network that can only be discovered by being invited to it or discovering the network in some kind of post-search warrant scenario. These types of networks work very similarly to the more traditional cloud storage providers, with the major difference being that there is not a single company that can be served legal process as to where a particular user's data may reside. People choose self-hosted cloud solutions for a variety of reasons, including privacy, scalability, and having more direct control over the process. [4]

Another major area of unorthodox storage involves the concept of containers, primarily in the form of virtual hard drives, such as VHDX and VMDK files, as well as encrypted containers like those produced by TrueCrypt, VeraCrypt [5], and PGP. Virtual hard drives can be used in a standalone fashion and mounted as drive letters on a host computer, or used in conjunction with virtual machine software, such as Hyper-V or VMWare/ESXi. Encrypted containers can be created in just about any size and most software also allows for the creation of "hidden" containers inside of another container. In these situations, a user would store the most sensitive things inside the hidden (sometimes also referred to as an inner) container and place other items in the non-hidden container. This way, should someone gain access to the non-hidden container, the

true secrets would still be preserved. While it is possible to detect the use of hidden and non-hidden containers, it is difficult to do so. Even if the existence of a hidden container is shown, little can be done to actually gain access to the hidden container without the proper key/passphrase.

[1] The Majority Of Global Internet Users Using A High-Speed Connection | https://for498.com/qc1a5

[2] Internet and Home Broadband Usage in the United States | https://for498.com/71-8m

[3] A Brief History of P2P Content Distribution, in 10 Major Steps | https://for498.com/6z4wh

[4] Hosted vs On-Premise Private Cloud - VEXXHOST | https://for498.com/as7h5

[5] VeraCrypt | https://for498.com/38keo

Critical Data for Triage and Quick Win Investigations

- In a sea of data, how do we know what to actually collect?
- Focus on data we need to prove who, what, where, and when

As the scope and scale of storage devices continues to expand, the old concept of "image everything" becomes less and less realistic. As we have seen, there are just far too many data sources to collect/image everything available in every case. From the time it takes to do full acquisition to the storage space required to keep it, significant challenges face us now and into the future when the problem will only continue to get worse.

While it will not be possible to just stop fully imaging most devices in a lot of environments (law enforcement for example), what we can do as practitioners is change our approach to investigations by relying more on triage collection and processing to move cases forward. Effective triage processing takes a fraction of the time that a full image takes to produce and does not alter the evidence. This efficiency will continue to go up as storage capacities continue to expand.

When it comes to triage, how do we determine exactly which files/artifacts to collect? When a storage device may contain gigabytes of data, how do we prioritize what to look at? While this is certainly something that becomes easier as an investigator gains experience, we can rely on the past experiences of others who have come before us to aid is in learning this skill.

From a high level, we want to focus our thought process around getting the data that will help us prove who did what, when they did it, and when something happened. As an example, suppose you wanted to show when a certain Word document was created and last changed, as well as who opened it on a given computer. Based on this requirement, you may only need 2-3 artifacts: the Master File Table (MFT), lnk files, and jumplists. The MFT allows you to see file creation and modification times, and LNK files/jumplists tell you which account opened files. By focusing only on these rather small (in comparison with the entire volume) artifacts, these kinds of questions can be answered in minutes vs hours or even days, weeks or months, as is the norm in many environments.

Critical Data Quick Win Examples

- Chat
- Email
- Phone calls
- SMS
- Commonality: SQLite databases

Evidence of communication

- Search terms
- Viewing history
- Problem: Different artifacts for different browsers

Browser history

- EXIF and Geolocation
- Photograph contents
- Network history

Location history

Above we can see several examples of a few categories we may want to focus on when determining what to collect to answer our questions related to who, what, where, and when. This is by no means a comprehensive list of categories and we will see more examples later.

Depending on the kind of data being collected (chats and SMS messages for example), there may be some commonality on how the data is actually stored. In the above example, SQLite database files are used quite frequently for this, especially on mobile devices. This is useful for us because we can simplify our triage collection process to always look for SQLite database files. We know how to access the data in SQLite database files and as such, any application that uses them can be reviewed in some form or fashion with a SQLite database viewer.

As an example of when the opposite is true, consider something like web browsers. Just about every web browser uses a different data format to store its artifacts such as saved passwords, browsing history, bookmarks, cookies, and so on. [1] In some cases, browsers use SQLite to store this kind of information, but in others, entirely different formats are used, such as webcachev, index.dat, html files, and so on. The challenge then becomes, how can we systematically go about finding and extracting information about all the different possible browsers out there to ensure we have a thorough picture of web browsing activity?

The answer is by using a triage tool that allows you, the investigator, to create "recipes" for different applications, artifacts, and so on. Once the recipes are crafted, they can be used against a device to copy key artifacts in a consistent and repeatable manner.

Of course, copying the data off a device is just part of the solution, because many useful artifacts are stored in a format that is not easily interpreted (i.e. in binary vs plain text format). One or more parsers will be required to convert the raw data held in the artifact into a more human readable form.

[1] An Overview of Web Browser Forensics | `https://for498.com/sa6p1`

Physical Storage Device Overview

Spinning

Solid state

The two most common types of physical storage devices you will encounter are hard drives which come in two different varieties: spinning media and flash storage. For most newer devices, both of these devices will have the same kinds of connectors on them for data transfer as well as power. An example of these connections can be seen on the left side of the solid-state device where the bottom connection is for power and the connection above that is for the SATA cable.

These devices come in many physical sizes, from 1 inch all the way up to 8 inches in size, with 2.5 and 3.5-inch drives being the most common.

While the primary goal of both of these types of devices is storing and retrieving data, how the devices go about doing so varies dramatically. For physical devices, the commands to read, write, and move data is generally issued by the host computer to the device, but in the case of a solid-state device, the controller embedded directly on the device has much more control over reading, writing, and moving data. This is due to things such as wear leveling [1]. Wear leveling is a process that helps reduce premature wear in NAND Flash devices, and the SSD software controls when wear leveling occurs. This essentially means that any time power is applied to an SSD device, whether a write blocker is in place or not, the data stored on the SSD may change, because the SSD itself is rearranging the data stored vs. an external process. This can lead to other challenges that typically do not exist with spinning media, such as a hash value discrepancy between a source device and a forensic image of the device. [2]

As the name implies, spinning disks use magnetic, rotating platters to store data, while solid state devices store data in flash memory and have no moving parts at all. In addition to flash memory, SSDs typically contain a controller and in some cases, some kind of cache to temporarily hold data before it is written to flash memory. Spinning disks on the other hand, have a read/write head attached to an actuator arm that moves across the platters to read and write data to the various platters. Several things determine a spinning disk's capacity, including the number of platters in the device as well as the density (how closely the ones and zeros are stored) of the platters themselves.

In some cases, a much deeper understanding of the intricacies of these devices is required to recover data from them. In these cases, a specialized device or data recovery expert may be required to accomplish this.

[1] NAND Wear-Leveling | https://for498.com/h9woq

[2] Solid State Drives: Part 6 | https://for498.com/3y48e

Possible Pitfalls with ATA Devices: HPA

- HPA = Host Protected Area
- Allows for hiding a portion of the hard drive from the operating system
- Legitimate uses include recovery partitions or other diagnostics
- Also allows for hiding of data in an area of the hard drive imaging tools may not take into account

The ATA standard [1] provides means to designate a portion of a storage device as hidden from the operating system. This is known as a Host Protected Area (HPA)[2].

While there are legitimate uses of HPA for both legacy reasons related to BIOS limitations, as well as recovery partitions, etc., we want to be aware of HPAs because they can be used to hide data from the operating system, thereby making them invisible during live response. From a forensic imaging perspective, this is less of a concern because most popular imaging solutions can detect the presence of HPAs on a device.

[1] Parallel ATA - Wikipedia | `https://for498.com/e2duv`

[2] Host Protected Area - Wikipedia | `https://for498.com/opxce`

Possible Pitfalls with ATA Devices: DCO

- DCO = Device Configuration Overlay
- Effectively makes a storage device report less sectors than it truly contains
- Legitimate uses include making a larger hard drive appears as if it's smaller in size, for consistency purposes
- As with HPA, allows for hiding of data in an area of the hard drive imaging tools may not account for

The ATA standard also allows for treating a larger hard drive as if it were smaller in order for different drives to report the same exact number of sectors, etc. This is known as Device Configuration Overlay (DCO)[1]. DCO is invisible to the operating system, BIOS, and end user.

As with HPA detection and mitigation, various imaging hardware and software either detect or detect and remove DCOs prior to imaging. This ensures a complete copy of the data is made at the time of collection.

A great (free!) tool to interact with both HPAs and DCOs is ATATool[2].

Current features of ATATool include:

- List attached ATA devices (both legacy PATA and SATA)
- Display device information including model/serial number, HPA and DCO status
- Modify and reset HPA device status (SETHPA command)
- Modify and reset DCO device status (SETDCO command)
- Corrupt and fix disk ECC data (BADECC command)
- Force Windows device re-detect following HPA status change
- Runs inside a Windows environment (including Windows PE)

ATATool has the following requirements:

- Windows XP SP2 or later (32-bit or 64-bit) including Windows PE
- Administrator rights (elevated on Windows Vista and later)

[1] DCO | `https://for498.com/8hwqd`
[2] ATATool | `https://for498.com/ds7lf`

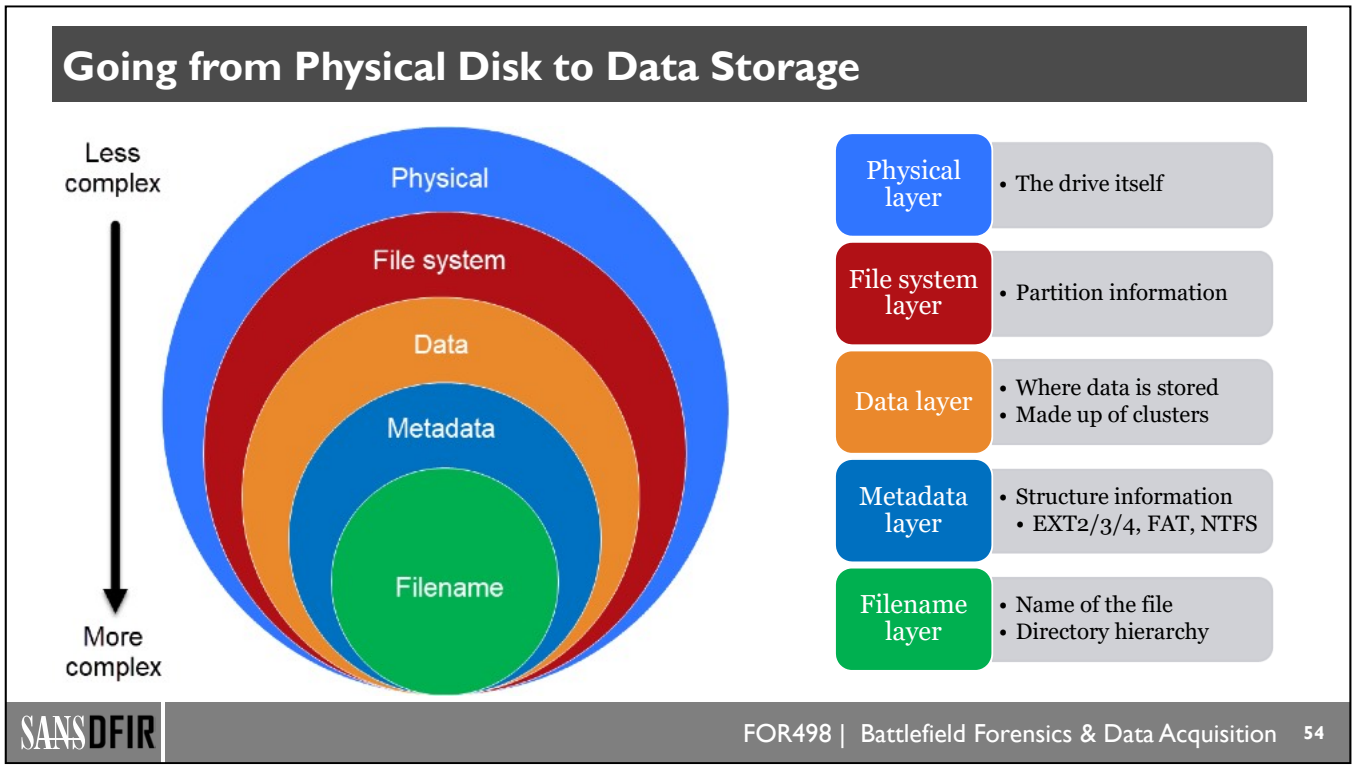
Differences between Spinning and Flash Media

	Spinning	Solid state
Access time	10 ms	0.1 ms
Read/write speed	50-100 MB/sec	200-500 MB/sec
Weight	500 grams	50 grams
Power usage	6 watts	2-3 watts
Cost	5-6x cheaper/GB	-

In general, solid state devices deliver far more than their spinning media counterparts in all areas except for cost. While the gap is closing, from a price per gigabyte perspective, traditional spinning media is much cheaper than an SSD of the same size. As of fall of 2018, a 2 TB SSD cost approximately $349.00 vs a 2 TB mechanical disk cost of $59.00, for a difference of approximately 5.9 times the cost between the two.

In all other areas except for cost, SSDs provide faster access to data, faster read and write speeds, lower weight and power consumption than a spinning media devices. These factors, along with the fact that the physical dimensions of SSD devices are often smaller than traditional drives, have led them to becoming the default storage device in most new desktop and laptop computers sold in recent years. As the costs of SSDs come down, this trend will continue.

One other positive factor of dealing with SSDs is the potential to be able to image the devices faster, thanks to their faster read speeds. There are, however, caveats we must be aware of when it comes to imaging SSDs, that do not exist when dealing with more traditional storage devices. We will cover these caveats in a later section.

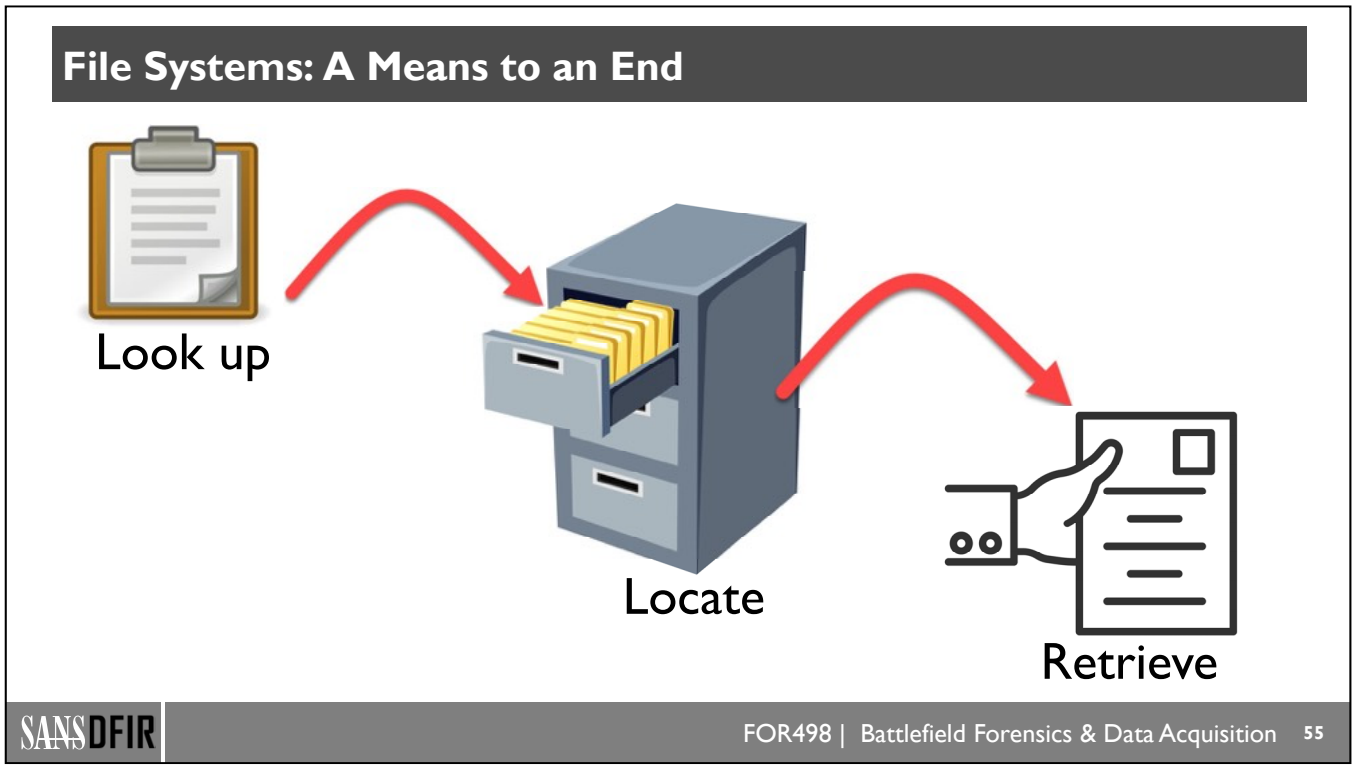


The circles above represent the different layers of organization that exist on a typical storage device. As the circles get smaller, more and more structure, and therefore complexity, is added to the previous layer above it. Looking at things in this manner also makes it easier to understand the relationships between the different layers and the kinds of information that are available as you move between the layers.

At the physical layer, storage devices are a collection of many small storage blocks called sectors. There is not any particular structure or grouping defined as it relates to these sectors at the physical layer, although a very common standard is that a physical sector will be 512 bytes in size.

The file system layer allows for dividing collections of sectors into one or more groups, called partitions. A physical device's sectors can be divided into one or more partitions. Each of these partitions becomes a distinct "bucket" for use by subsequent layers. Any given sector can only be a part of a single partition (or not associated with a partition at all).

The sectors in a partition are then further grouped into clusters. Once sectors are arranged into clusters, the cluster becomes the smallest addressable unit of storage for things in the other layers (metadata and filename).



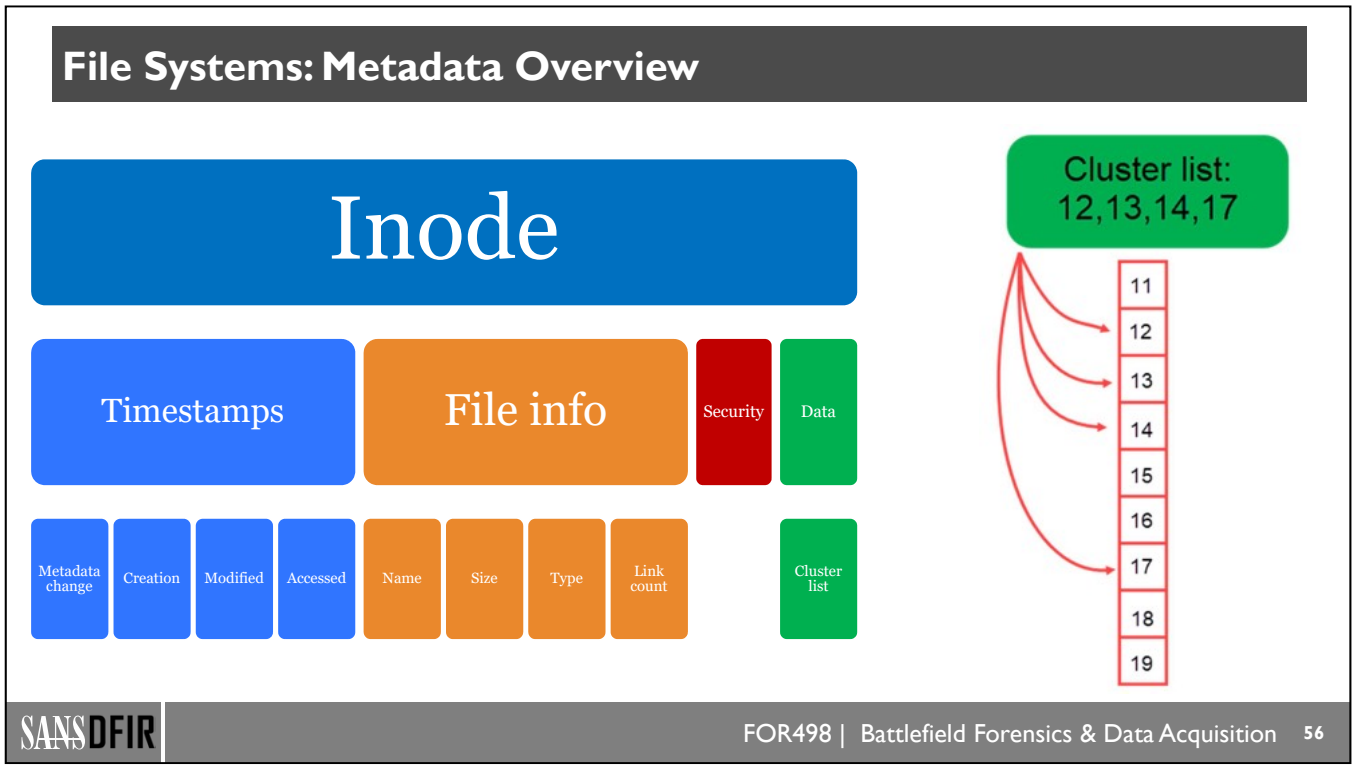
File systems in simple terms, are responsible for organizing and retrieving data. In many ways, they can be viewed as a filing cabinet that is filled with different folders, arranged by customer or date, along with a reference to quickly locate data within the filing cabinet. The key to any filing system is knowing where to find the data you are interested in, and then retrieving the files quickly (vs. manually rummaging through every cabinet and folder within them looking for things). Following along with the filing cabinet analogy, the reference could be implemented by looking up a customer name on a master list and being directed to filing cabinet number 4, drawer 2, folder 5, and so on.

But what if you didn't necessarily need the information contained in folder 5, but rather, you only needed to know about the data in folder 5, like when it was created, or last updated, or how many documents will be found there, and so on? This kind of information is known as metadata [1], because it describes the data, vs. being the data itself. In our filing cabinet lookup system, perhaps the master list would also contain a timestamp for the last time a customer was updated, or how many documents existed in a customer's folder. If this information was available to you and you were only interested in folders that contained data which was updated in the last week, you would not have to go into the file cabinet at all to determine this.

By using the metadata about a file, you could easily find out which customers were updated within the last week and then, based on that shorter list, get the actual files from the filing cabinet without too much trouble.

As we will see next, file systems follow a similar pattern to look up, locate, and retrieve files.

[1] What is Metadata? | `https://for498.com/btg8a`



The basics of any filesystem, regardless of whether they are used in UNIX or Windows, are similar, but the actual implementations are different as far as what the structures look like in their native format. Reducing things to a common knowledge base will help an investigator in that similar techniques can be employed across a wide variety of filesystem types, while only the specific tool that is used changes.

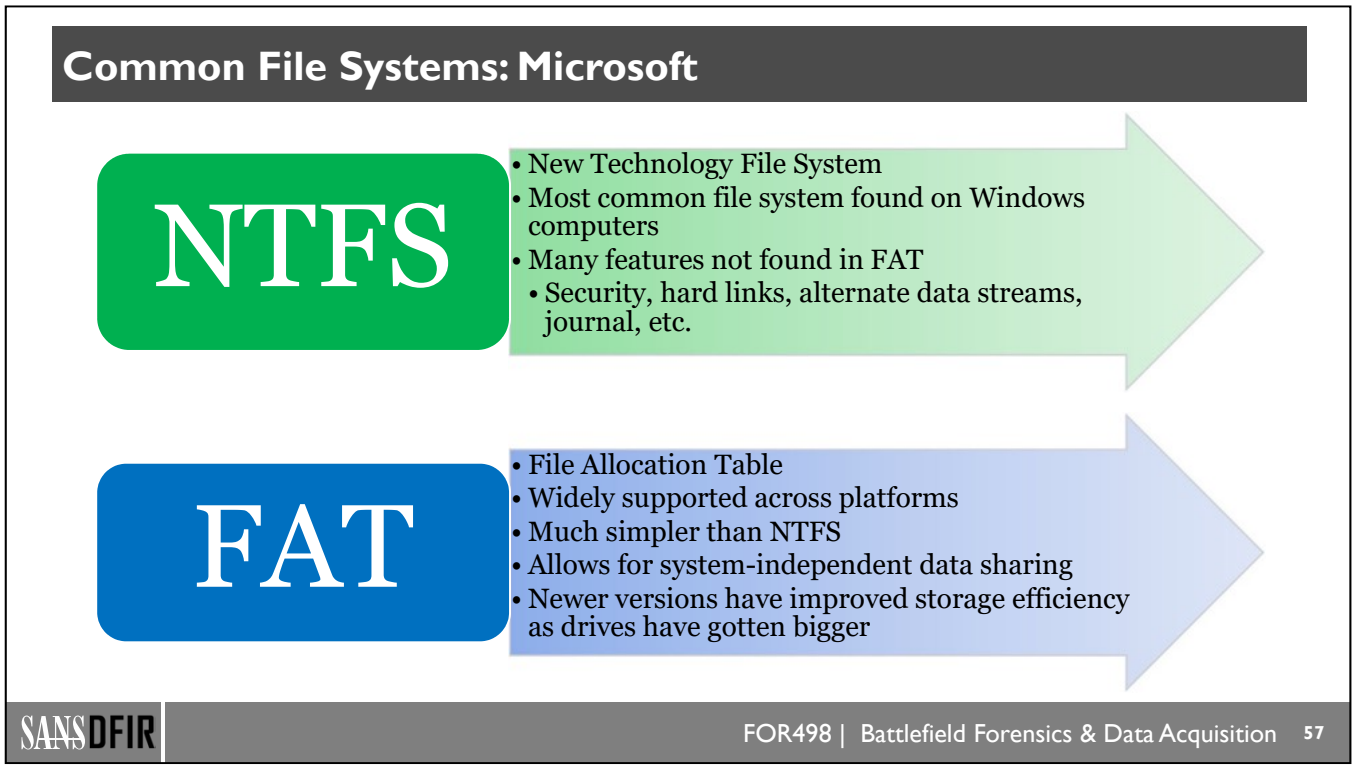
In most filesystems, elements such as the name, a type, a pointer to the file's data, the data size, timestamps, and a security mechanism will exist. Some have more robust features (NTFS and ExFAT), whereas some have less (FAT).

In the case of the above diagram, the Inode (synonymous with MFT Entry for NTFS) at the top is the overall "container" for the items found underneath it.

In the example above, the file's data is contained in 4 clusters as shown in the cluster list on the right. In addition to knowing which clusters are used to hold the data, it is also just as important to know the correct order to read the clusters so the data to be reconstructed properly.

Thinking back to our discussion about metadata, everything under the Inode block above would be considered metadata in that it tells us about the data that is actually stored in the file. The timestamps would help us determine which files we want based on created date, the size would help us locate large files, and so on. Contrast this to how you would have to do this without this data being tracked. In the case of wanting to find large systems, every file would have to be looked at and the size calculated. This would be much more time consuming than being able to quickly look it up in the file system's metadata collection.

Notice too that the metadata about the file actually tells us where to get the contents of the file, via the cluster list. By following the clusters listed in order, the exact contents of the file can be retrieved.



In 1989, Microsoft and IBM began work on OS/2, an operating system that never really caught on. Out of that failure came several new pieces of technology, including the forerunner of what would become NTFS, or New Technology File System. NTFS is the most common file system you will encounter when looking at Windows devices.

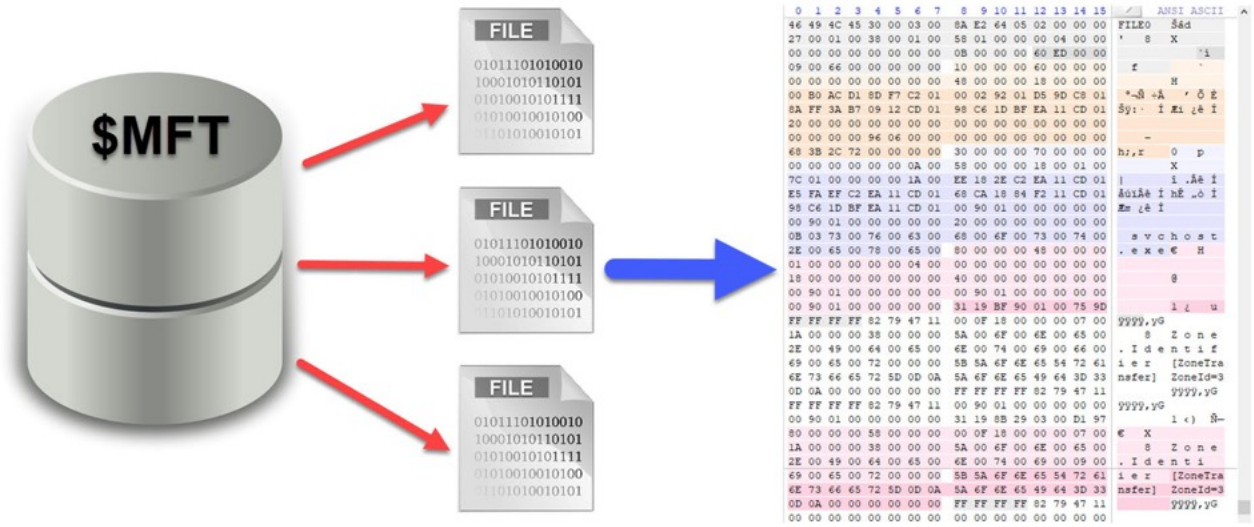
The FAT filesystem has been around since the early 1980s and is one of the simplest filesystems. It contains no security features, few timestamps, and several hacks that have allowed it to still be used today.

Both FAT and NTFS can have varying cluster sizes for a variety of reasons. FAT, because of the way it was written, was forced into using larger and larger cluster sizes as hard drive capacity increased. NTFS on the other hand, was designed to keep cluster sizes at 4KB by default. Each of the file systems has a default cluster size [1], but this can be changed when partitions are formatted.

The common Microsoft file systems have various means to keep track of things we saw in our metadata overview. The next few slides will provide an overview of how each file system goes about tracking metadata and retrieving data.

[1] Default cluster sizes by file system | `https://for498.com/kfed-`

Common File Systems: NTFS

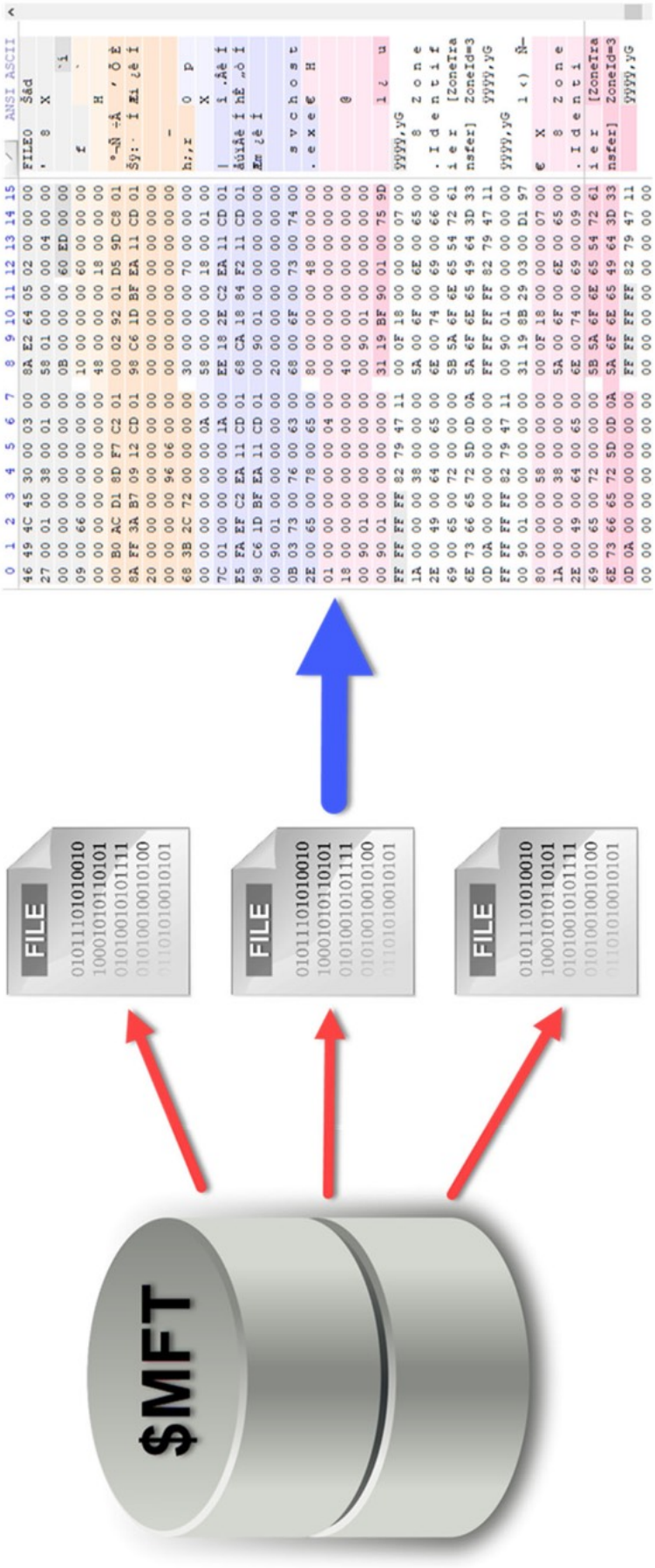


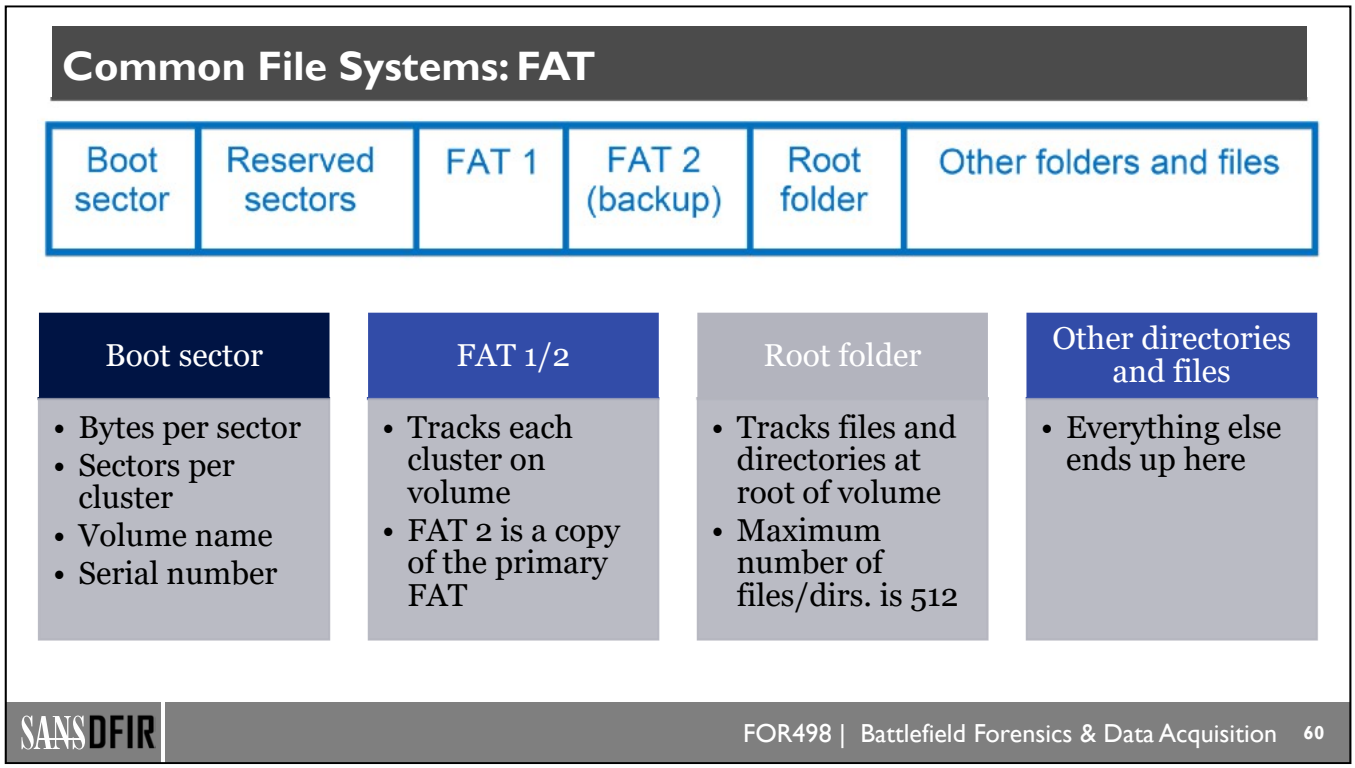
The heart of NTFS is the MFT, or Master File Table. This file is organized into records that are typically 1024 bytes in length (in rare cases they can also be 4096 bytes in length). Every file and directory (a directory is just a special kind of file in NTFS) gets a record in the MFT, including the $MFT file itself.

A FILE record is made up of several parts. First, a header contains information such as the unique identifier for the FILE record, the entry and sequence numbers. There is also a flag that tracks whether the FILE record is currently allocated (i.e. tracking an active file that you could find using File Explorer in Windows).

Following the header is a collection of attributes that are used to record various metadata such as timestamps, file size, location, and so on.

Courses such as FOR508 provide a much deeper dive into NTFS including the full breakdown of the header and common attributes into their corresponding properties. We will, however, take a closer look at some of these properties in a future section.





There are four variations of FAT: FAT12, FAT16, FAT32, and exFAT. The major difference in each is the size of addressable entries in the FAT, which will be described later. The exFAT filesystem is the newest version and can be found in Windows versions after VISTA SP1 and latest versions of Windows CE 6.0.

The FAT filesystem[1] is one the most common PC filesystem around as it is compatible with so many different computers. FAT is reliable because it keeps a table of files and free space.

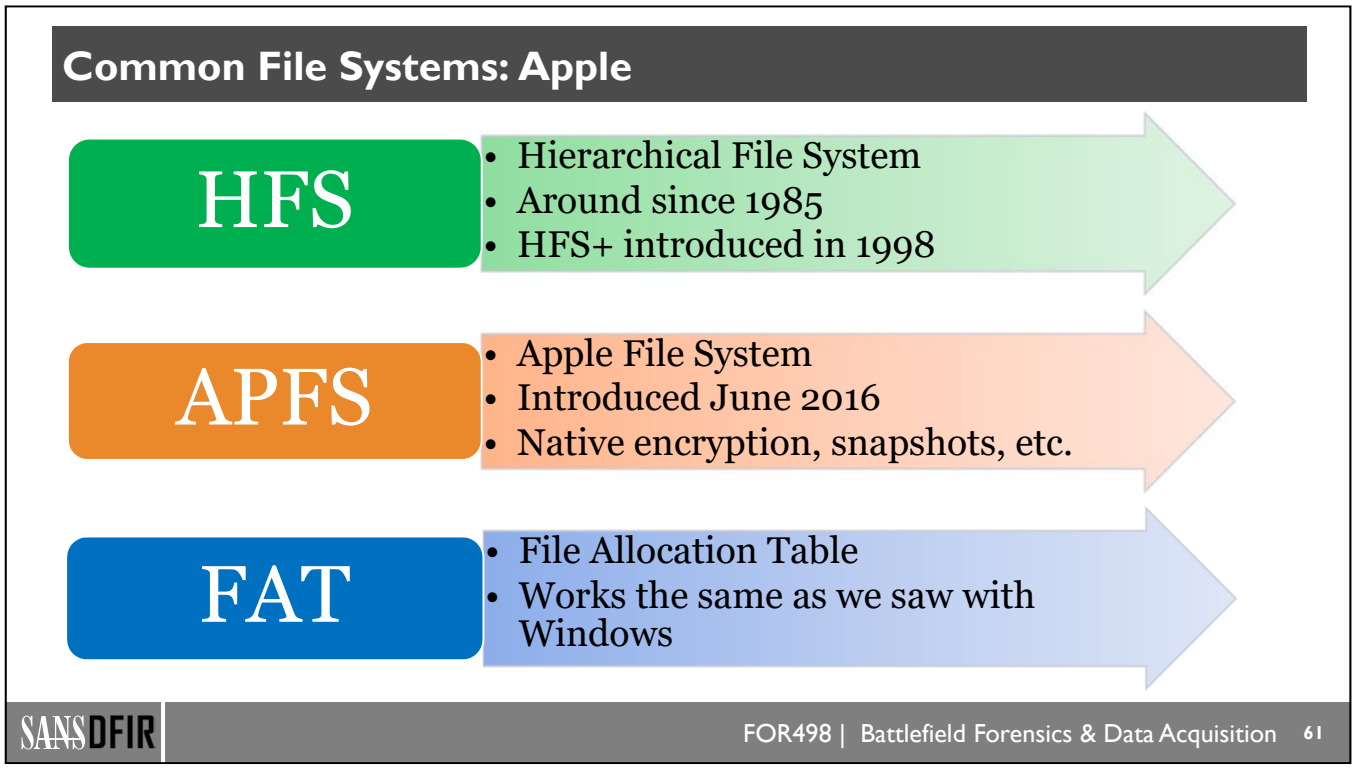
The number at the end of the FAT is a multiple of how many clusters can be addressed on the filesystem. For example, on a FAT16 system, it can address 2 to the 16th power or 65,536 clusters.

FAT32 was introduced with Windows 95 OSR2. With 32-bit FAT entry (FAT32) support in Windows 95 OSR2, the largest size hard disk that can be supported is two terabytes!

The extended FAT filesystem (exFAT) was introduced with Windows CE 6.0 in September 2006. It was provided for desktops and servers in Vista SP1 and Server 2008 in March 2008. Windows XP support was provided in January 2009 in KB Q955704. The file allocation table (FAT) in exFAT behaves differently than earlier FAT filesystems.

The image above shows the general layout for a FAT file system, from the boot sector, the primary (FAT 1) and secondary (FAT 2) file allocation tables, to the root folder and everything else. The purpose of there being two FAT tables is for redundancy. Because the file allocation table is so important, it is backed up in case the primary file allocation table becomes corrupt. Without such protection, catastrophic data loss could occur if the sectors and clusters that hold the primary FAT were damaged somehow.

[1] How FAT Works: Local File Systems | `https://for498.com/pfhyd`



HFS and APFS are Apple developed file systems that are used on everything from iPhones to MacBooks. HFS has two flavors, the original HFS [1] and HFS+[2]. As of Mac OS X 10.6, HFS is not supported and can only be used in read-only volumes.

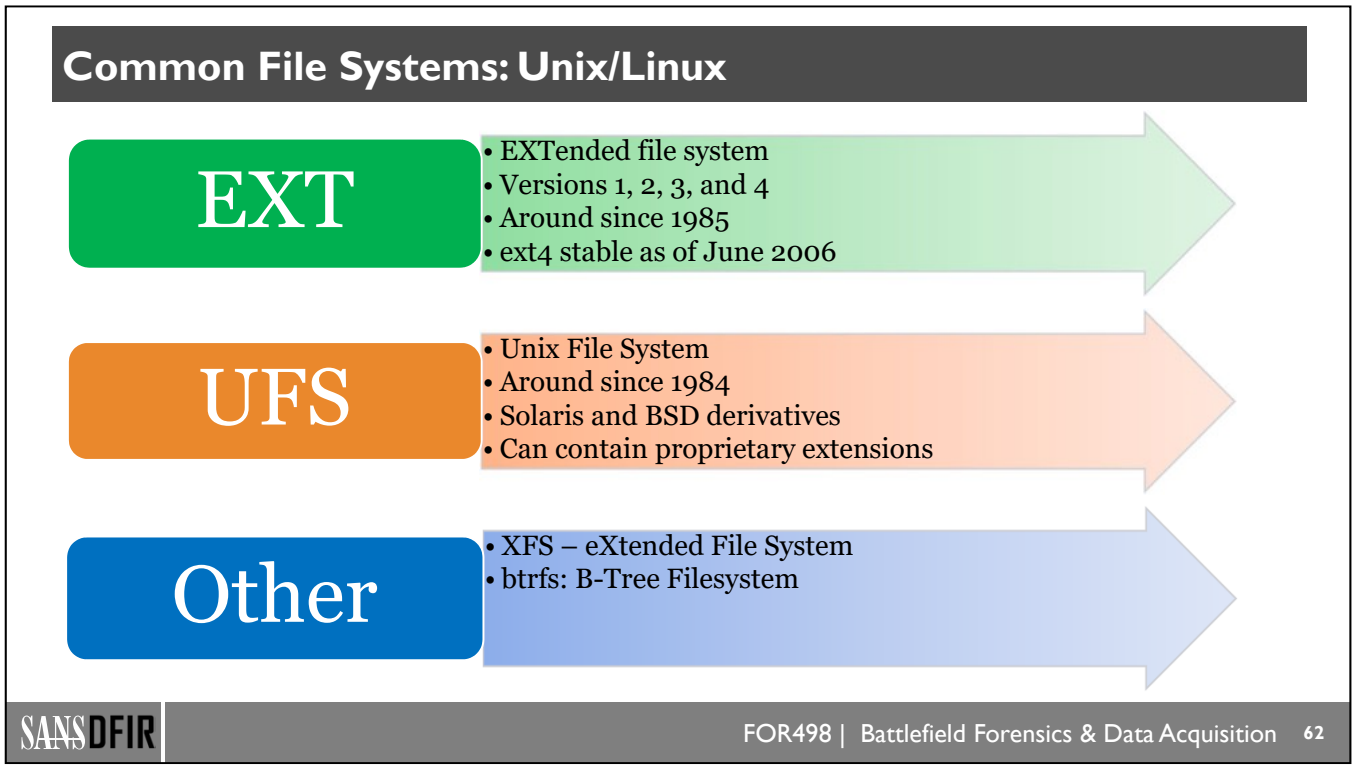
When initially developed, HFS contained many features not found in other file systems such as FAT, including long file names. HFS also uses 32-bit numbers internally and as such, allows it to address much larger volumes. One strange exception to this limited HFS from storing more than 65,535 files on a logical volume.

Like the progression of FAT16 to FAT32, Apple introduced HFS+ in 1998 to address shortcomings of its HFS implementation. HFS+ adds 32-bit block sizes (vs 16-bit in HFS), Unicode support, and hard links. HFS+ also added support for journaling which increases reliability and data integrity, as well as volume encryption support. HFS+ was the default file system used in Apple devices until APFS was introduced in 2017.

Apple File System[3], or APFS, is Apple's current file system which has been optimized for use with SSDs and contains native support for encryption. Additionally, because it uses 64-bit inode numbers (similar to the MFT's entry number), a volume can contain 9 quintillion files or more! Other new features of APFS include snapshots (similar to volume shadow copies on NTFS), metadata checksums, and overprovisioning.

Like most operating systems, Apple also supports reading and writing FAT volumes, but FAT cannot be used as the underlying file system when installing OS X.

[1] Hierarchical_File_System | `https://for498.com/t6b73`
[2] HFS_Plus | `https://for498.com/n-pmz`
[3] Apple_File_System | `https://for498.com/lnum8`



EXT[1] has been around for a very long time (all the way back to 1992) as the first file system created specifically for the Linux kernel. It takes some of its design cues from UFS [2] and has evolved over its iterations to support features such as larger file systems, multiple timestamps for files, journaling, larger directories, metadata checksums, and more granular timestamps, all while maintaining backward compatibility with older versions. [3]

Versions of EXT prior to v3 were prone to significant corruption if power was interrupted or the system was not shut down cleanly. This could lead to anything from a loss of data to a system incapable of booting properly. Version 3 adds support for journaling, which serves to record actions against a file system in an intermediate file that is updated using the concept of transactions that can be reversed (and thereby restoring what a disk looked like before a file operation) which gets the file system back to a consistent state. This may result in data loss, but it significantly improves the resiliency of the file system against corruption.

Version 4 is the current release of EXT which adds things such as 48-bit addressing, extents, block allocation improvements, and more.

The Unix File System (UFS) is commonly used in Berkeley Software Distribution (BSD) style operating systems, such as FreeBSD and OpenBSD and is supported by most Unix/Linux operating systems. UFS was created in the mid-1980s and has gone through many revisions, adding new features and addressing shortcomings along the way. UFS supports Access Control Lists, file system snapshots, logging, etc. like many of the file systems we have already seen.

There are of course other file systems you may run across such as XFS [5] or btrfs, both of which have feature sets that overlap or even go beyond what EXT contains. In addition to computers running Linux, btrfs is an option for file systems in Synology NAS devices. [4].

When you encounter some of the less common file systems such as XFS[5] or btrfs, the forensic tools you are used to using may not work to analyze such a file system. In this case, other techniques or tools will need to be used to conduct a forensic exam.

[1] Ext4 File System | `https://for498.com/qbzmj`

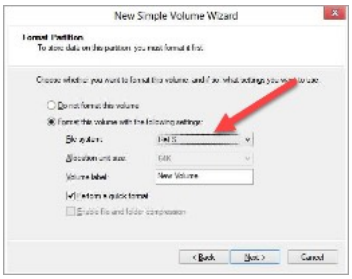
[2] UFS File System | `https://for498.com/-nujx`

[3] Understanding Linux filesystems: ext4 and beyond | `https://for498.com/noxj-`

[4] Synology NAS | `https://for498.com/uizep`

[5] XFS File System | `https://for498.com/4bgf2`

Uncommon File Systems



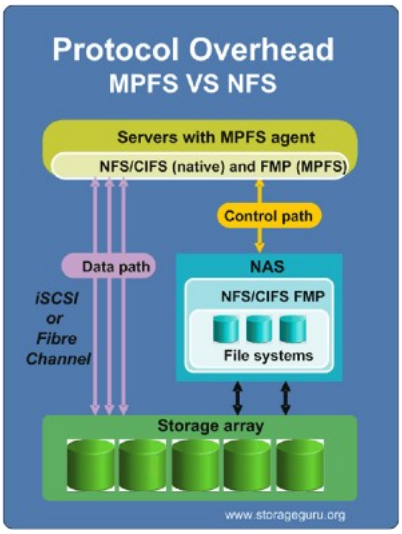

```
  pool: pool
 state: ONLINE
  scan: none requested
config:

        NAME             STATE     READ WRITE CKSUM
        pool             ONLINE       0     0     0
          mirror-0       ONLINE       0     0     0
            /root/disk1  ONLINE       0     0     0
            /root/disk2  ONLINE       0     0     0

errors: No known data errors

  pool: rpool
 state: ONLINE
  scan: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        rpool       ONLINE       0     0     0
          c8t0d0s1  ONLINE       0     0     0
```



The file systems we have covered so far are by no means the only ones you may run into. There are many less frequently used file systems that you may encounter depending on the kinds of cases you work and the environments you must operate in. Some examples of these rarer file systems are below.

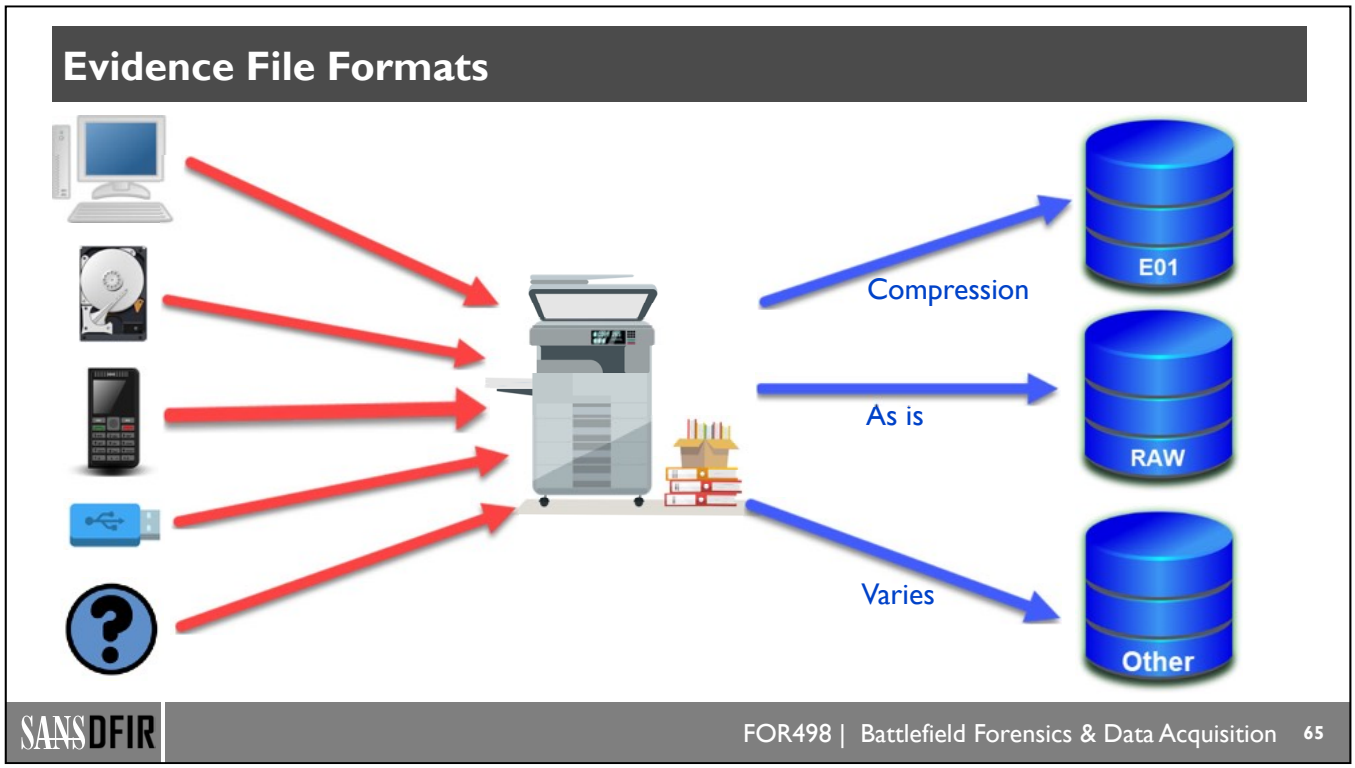
Resilient File System (ReFS)[1] is a new file system developed by Microsoft. The primary goals of ReFS are to overcome some of the shortcomings found in NTFS. As of the end of 2019, ReFS is not bootable and therefore is not going to be found nearly as often as NTFS is. ReFS[2] uses some of the same underlying mechanics that NTFS does, but adds additional things on top of it, such as allocate-on-write and other techniques to avoid corruption and data loss.

ZFS[3] was created by Sun Microsystems, and theoretically can contain a storage volume size of 1 Zettabyte (1 trillion gigabytes). ZFS has many features that allow it to ensure data integrity, seamless storage volume expansion, block level checksums, snapshots, and more. ZFS takes steps to store data redundantly and includes checksums that are verified when data is read. One of the goals of ZFS is to never return corrupted data from a disk.

EMC Corporation developed Multi-Protocol File System (MPFS) as a way to significantly increase performance over what conventional Network File System (NFS) is capable of [4]. MPFS scales to thousands of client computers and is often used in grid computing initiatives.

When you run into a strange or unknown file system, one key thing to do is leverage the people managing the devices using these file systems to help you understand exactly what file systems are in use, how the data can be accessed, and so on. The data on many of these devices can be copied to a device with a more supported file system which, for certain kinds of investigations, may be the easiest approach when dealing with these kinds of storage devices.

[1] What is Resilient File System (ReFS)? | `https://for498.com/k2s7t`
[2] ReFS features and limitations | `https://for498.com/refs`
[3] ZFS - FreeNAS - Open Source Storage Operating System | `https://for498.com/53x9o`
[4] Multi Path File System | `https://for498.com/f9bxv`



Digital forensics investigations often involve creating and examining disk images. A disk image is a copy of a full disk or a single partition from a disk[1]. These images can be in several different formats, from an exact bit-for-bit representation, to one that allows for compressing data, and so on. Before we look at the various formats for storing these copies, let's get a better understanding of why we need to create such images in the first place.

Perhaps the most important reason to create a disk image is to avoid working against original evidence, which is generally never a good idea, especially in digital forensics. If the live device were used for some period of time and encountered a failure of some sort, data could potentially be lost. Making a copy of the original evidence in a way that can be verified removes this risk. What does it mean to verify a copy of the original evidence? In essence, when we verify an image, we want to make sure that the source of the data matches the data in the copy of the data. The most common way to do this in digital forensics is using a cryptographic hash function, like MD5, SHA-256, etc. Once a copy of the data is made, the hash of the source can be calculated which is then compared to the hash of the data in the copy. This verification can happen at any time and is often done more than once in most cases (generally, at least after the initial image creation and once at the conclusion of using the image for forensics purposes).

Other less critical, but important reasons to make a copy of original evidence include:

- Allows more than one examiner to look at the evidence at once
- Once an image is made, its contents can often be read and processed faster than if working on the original (due to the max read speed of the source device for example)
- Removes the need for possibly dozens of physical connections on an examination machine to all of the various data sources an examiner may encounter

[1] Forensic Images | `https://for498.com/9skma`

Common Evidence File Formats

Image formats

- E01/Ex01
- DD/Raw
- AFF4

Container formats

- AD1/L01
- X-Ways
- VHD/VHDX

The goal of image and container files is to preserve information about files from a file system, including the metadata about the files as well as the file's data itself

The various evidence file formats are a means to store the bytes taken from a storage device and represent them in another file. The file where the bytes get stored varies in sophistication depending on the evidence file format being used.

A RAW image is simply an exact copy of each byte as it exists on the source drive. An E01, or Expert Witness, image [1] adds things like a header, followed by chunks of data, usually compressed, and a CRC (Cyclic Redundancy Check) following each chunk of data. Ex01 files are the next generation of the E01 specification which seeks to fix some of the shortcomings of the original E01 specification[2].

Since a RAW image is just a copy of the data as it exists on the source, any additional information about the source must be preserved in a file separately from the source file. This is typically done in a text file that contains details about the source device including how big it is, how many sectors, partition information, who created the image, what software created it, the hash value of the source and destination (which can be used later for validation) and so on.

An E01 on the other hand, since it is more complicated than a RAW file, provides a means to store the information from the text file example above directly in the E01 file itself via properties in the header. In other words, the data about the image (i.e. its metadata) is stored directly in the E01 itself, and as such, no additional text file would need to accompany it. Forensic tools know how to read the header in an E01 and display this information for review, etc.

AFF4 is the newest format and is starting to gain varying levels of acceptance/implementation with tools and examiners. The AFF4 is a next-generation forensic container format supporting features such as [3]:

- Storage virtualization
- Arbitrary metadata storage
- Extensible compression and hashing schemes
- Throughput scalable to high I/O rates

Once an image file is created, it is “ingested” into different forensic tools, either directly or by mounting the E01 in such a way that forensic tools can read the data contained within the image.

Evidence containers can be thought of as a subset of the data that would be found in a disk image. Containers are logical images which generally will contain files added to them by an examiner, along with the file’s metadata. In many cases, only this smaller subset of files is needed, and a full disk image would be unnecessary and inefficient to provide[4]. These logical containers can then be shared with other investigators, prosecutors, and so on. Another advantage of using containers is that it allows for redacting the actual contents of files while still preserving all metadata. In the case where illicit images or videos are present, redacting the contents of these files allows for easier sharing with people who cannot lawfully possess such material. Several vendors have their own container format, including X-Ways, AccessData, and OpenText/Encase [5].

[1] Expert Witness Compression Format | `https://for498.com/3pnxm`
[2] Encase image file format | `https://for498.com/6uhke`
[3] AFF4 Standard Documents | `https://for498.com/owzup`
[4] X-Ways Evidence File Containers | `https://for498.com/z74lf`
[5] Expert Witness Disk Image, EnCase L01 Logical | `https://for498.com/dib8-`

Summary

- Data lives everywhere
- In quick wins, some data is far more important than others
- Spinning hard drives and solid state hard drives both hold data, but that is where the similarities end
- Understand that there may be areas of the hard drive that are not visible to you
- Account for the entire hard drive space
- You must have an understanding of file systems, as well as evidence formats

This page intentionally left blank.

Exercise 1.1

Do You See What I See?

Synopsis: In this exercise, you will take a provided evidence file (E01),and using a Linux VM and VMWare Player, you will cause the evidence file to boot up as though it were a normal computer.

Average Time: 50 Minutes

This page intentionally left blank.

Exercise 1.1 Takeaway

- There is nothing quite like being able to see the user's environment as the user did.
- In an investigation where a forensic image already exists, but time is of the essence, data can be interpreted and extracted much more quickly from a "working" computer.
- Extracting data from a working computer removes the need for a myriad of tools typically used by the examiner to make sense of databases and other repositories when parsing data from a "dead box" image.
- Artifacts in the form of desktop notes can be extremely quick wins. Most examiners never consider the notion of a user having notes on their desktop, nor do most forensicators even know where to find or parse them on a dead box image.

This page intentionally left blank.

SIFT Introduction

1.1 Intro to Digital Forensic Acquisition

1.2 Understanding the Data

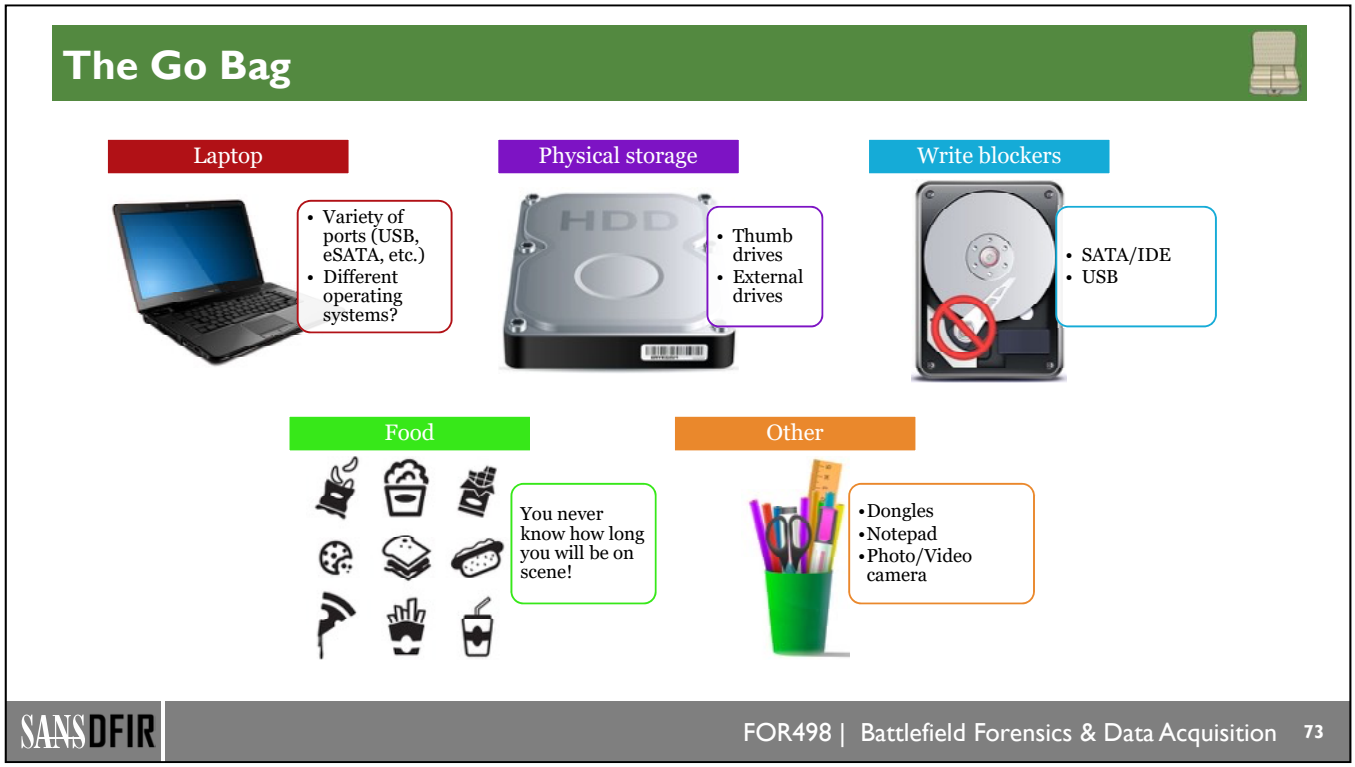
1.3 Scene Management & Evidence Acquisition

1.4 Device & Interface Identification

This page intentionally left blank.

Scene Management & Evidence Acquisition

This page intentionally left blank.



One of the essential things to get correct is the go bag. This can come in many forms, from a simple backpack to an elaborate case with custom inserts for everything from write blockers, to storage devices, and everything in between.

Regardless of the type of go bag you create, there are some best practices to consider when it comes to what you include in your set up. While every case can be different, in general, the following items will serve you well in the field:

- Laptop(s)
- Storage (hard drives and thumb drives)
- Write blockers
- Faraday bags
- Camera/video recorder
- Notebook
- Dongles
- Food

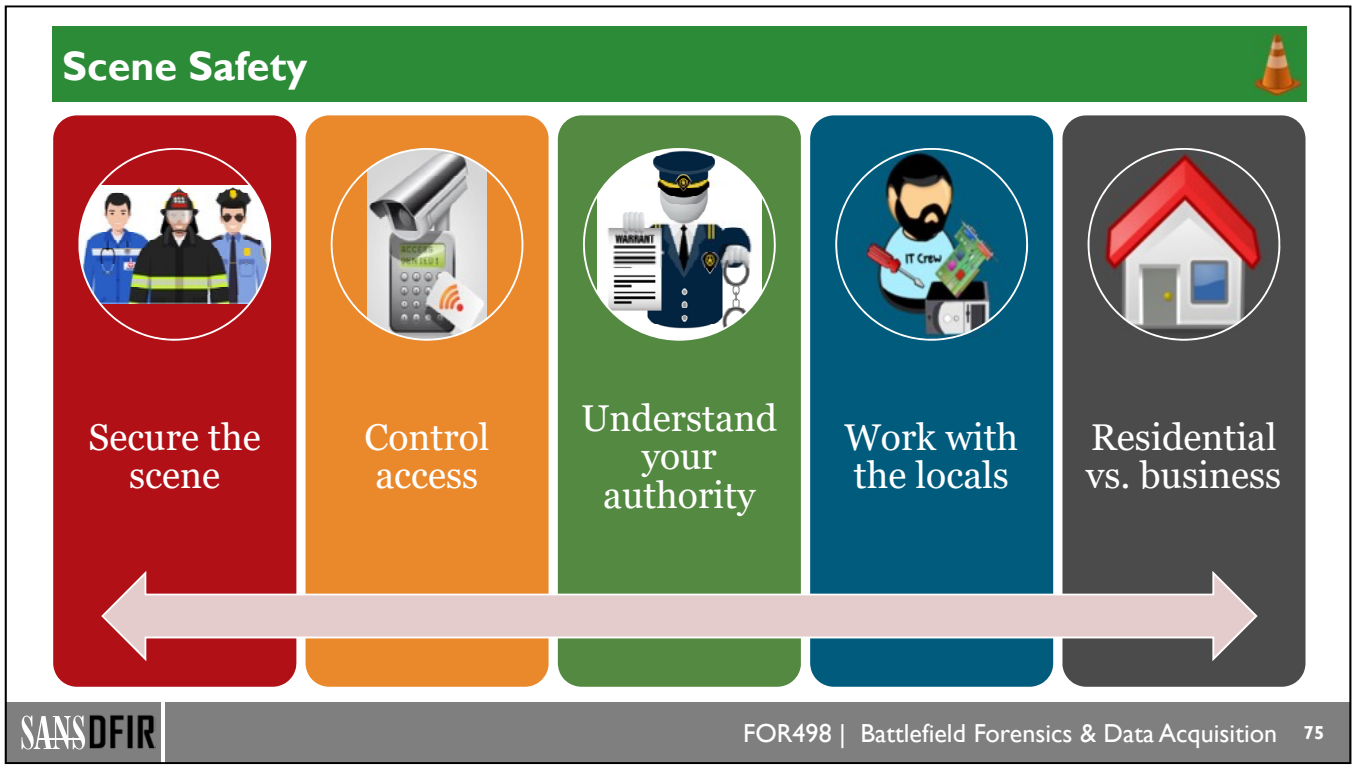
At least one laptop should be available, and possibly more than one, should you need to use different, platform specific tools, or simply to provide some assurances should Murphy's law strike. Laptop hardware should be adequate to run the software you use for everything from triage to imaging, and in some cases, full forensic review in the field. Having a wide variety of ports for peripherals is also a good idea, because dongles and external storage can quickly fill up available USB ports on a machine. Having a faster interface such as eSATA is also helpful, especially when creating forensic images. Many forensic programs rely on dongles for licensing. If you are lucky enough to have enough dongles to keep a copy in your go bag and another set in the lab, this is ideal. Many times however, this is not feasible from a cost perspective, so be sure you grab the dongles you need from the lab before heading out.

When it comes to storage, you would ideally have some idea of what to expect from a requirements standpoint before deploying, but this is not always possible. Because of this, it is a good idea to bring as much storage as you can to a scene in the form of hard drives that can be accessed via some kind of dock, thumb drives, or more traditional external drives with USB or eSATA interfaces. Storage can be used for many things, including imaging devices, triaging computers, and so on.

In some cases it becomes necessary to preview devices such as hard drives or thumb drives in the field. When this happens, it is important to follow proper forensic methodology and ensure access to these devices are done via a write blocker to minimize the chances that something could change. Of course, in certain situations, like dealing with SSD drives, it may not be possible to prevent the device from changing, but we always want to be sure we take every precaution we can (and document them!).

Other things that may be required include faraday bags (to isolate equipment you cannot, or do not want to, turn off), and camera and video equipment for documenting a scene.

Finally, every go bag should include a notebook for documenting your actions while on scene as well as a robust collection of food and snack items. Having something as simple as a few protein bars can make your life a lot easier when you end up being on scene for longer than anticipated, which happens more often than not!



Scene Safety: What Does It Look Like?

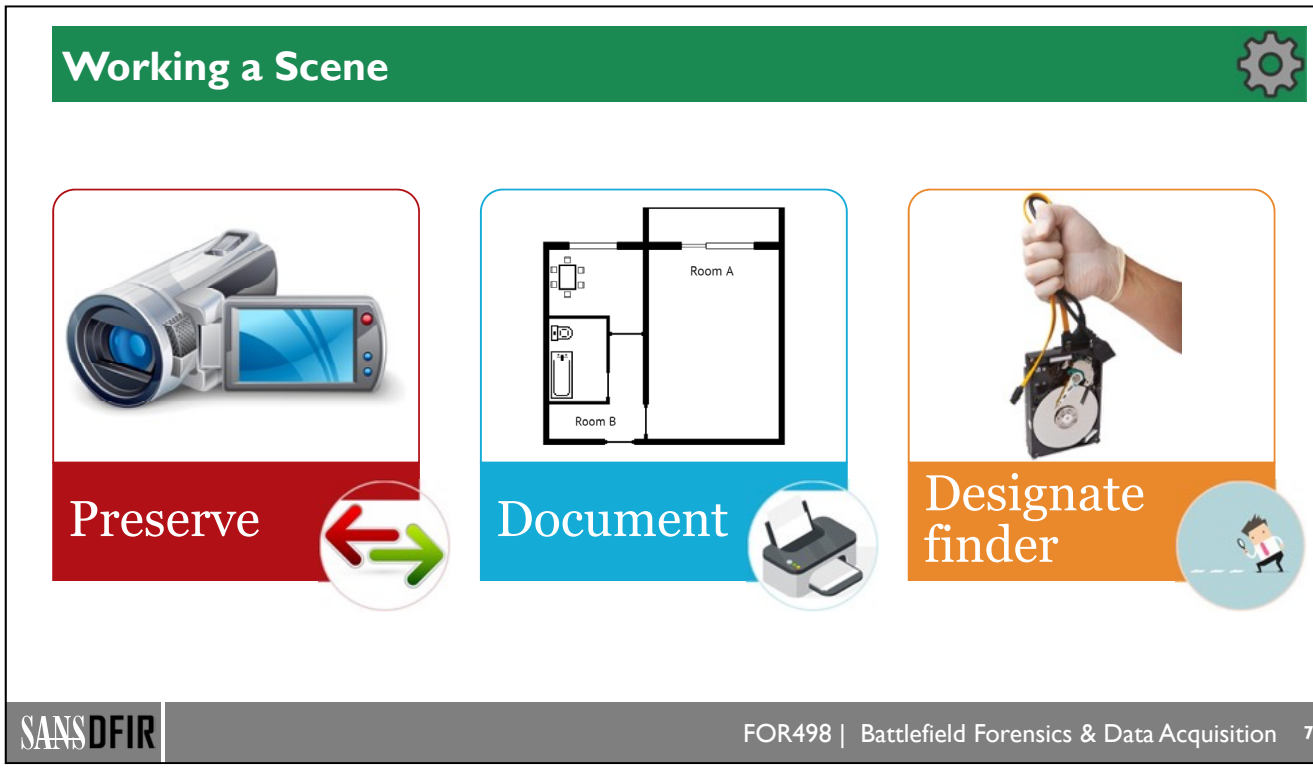
Securing the scene is a critical first step in just about every investigation. This may differ depending on the type of incident you are dealing with, as securing the scene in a law enforcement capacity vs. an internal matter can look quite different in their execution. If entering an unknown environment, it is often necessary for law enforcement to clear the space and account for anyone who may be present before other incident responders enter. In the case where the environment is more well known, such as at a business, this may simply involve identifying who is working in the space you need to operate in.

Once a scene is secure, it is time to transition to working the scene. This generally involves removing most personnel from a location prior to documenting the pre-search state. Not only does this allow for a more efficient process, it eliminates the possibility people will interact with the scene before documentation is started. Generally, one or two people should remain that will document the initial state of the scene.

In all scenarios, controlling access to a scene is paramount. The goal is to know who is in a space and what access people have to devices in that space for the duration you have control. In general, people who do not have a need to be there should not be allowed to be in the space while you are on scene. This looks different when conducting an operation at a residence via a search warrant vs. consent at a business because in the case of a search warrant, you will essentially be able to do what you want as it relates to control of the space, whereas with a business, less direct control will be possible in many cases.

Some kind of authorization to be at a location is assumed, and this would involve either consent from a business/third party or some kind of legal process (search warrant) for law enforcement. At a minimum, you should document who as at the location and determine who has access to what devices at the location (both physical and ideally, remote access). For situations involving law enforcement, this would often include officers initially clearing the area and making sure all people and threats are known and accounted for, limiting access to a location, and so on.

In cases where additional assets are available, such as when operating in a business environment, a lot can be gained by working with the locals because they will have an understanding of the systems, personnel, access rights, and so on in their organization. With that said, it becomes important to vet the person you are working with to ensure sensitive information and investigative techniques are protected.



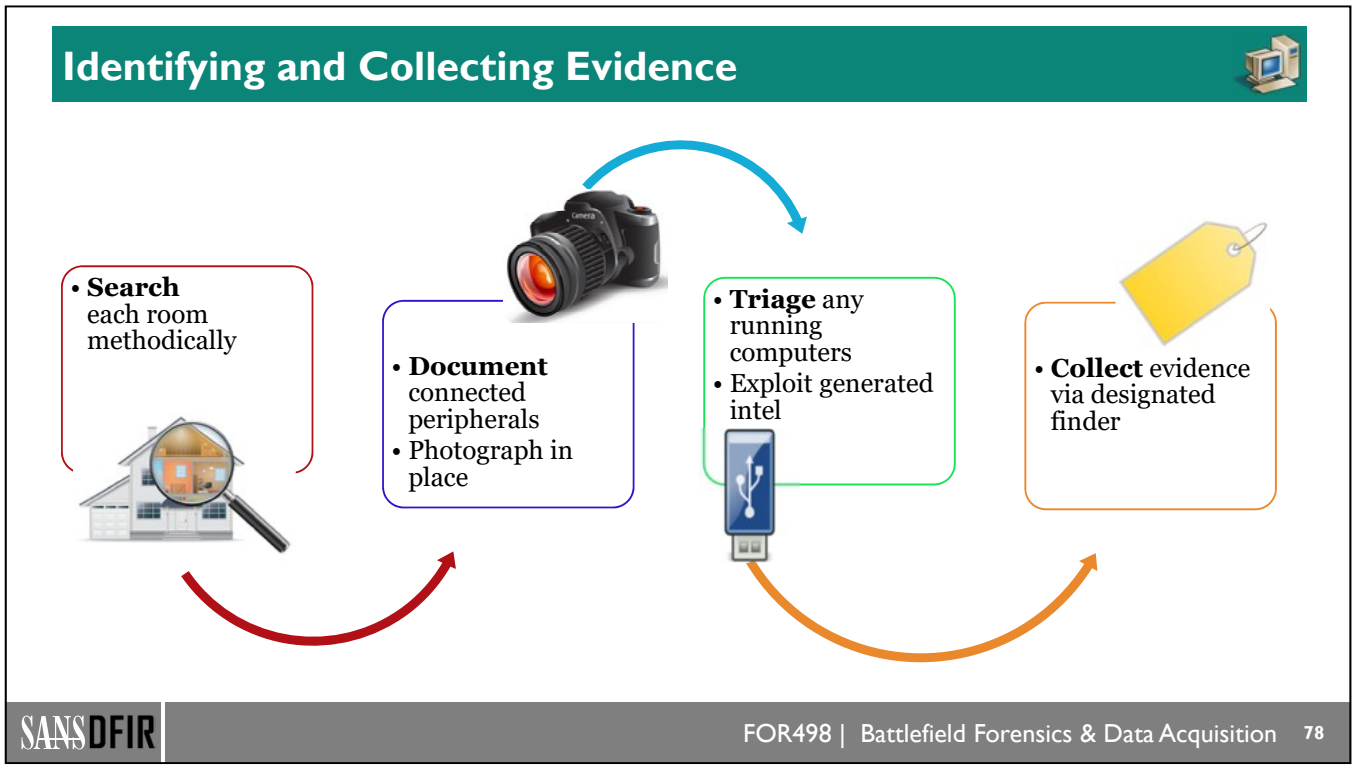
Once a scene is secure, the next stage is to document the pre-search state of the location. This generally involves at least photographs of every area to be searched, but it is suggested to use video to capture the "before" status of the search as it is generally easier to do and allows for automatically recording of any audio that may be of interest by the person preserving the scene. Just as we want to capture the state of things before the search, we also want to capture the state of things after the search as well. As such, a post-search walk through will be done to document, either via photographs or video, how things look at the conclusion of the search.

The reason for capturing the before and after is to avoid any contention down the road from the owner of the location saying that things were broken, taken, disrupted, etc. Having a visual record of the state of the location alleviates this possible issue.

In addition to photos or video, it may also be helpful to sketch out the floor plan of the location. In certain cases, such as apartments or a business location, it may be possible to get floor plans from the apartment management office or the business.

Once the initial walk through is complete, each room should be given a designation via some kind of signage, such as "Room A, Room B" or "Room 1, Room 2" and so on. In the case of a business where there is already existing room designation (either by number or name), these can be adopted to save time. These room designations will be used later when items are located and collected.

Finally, it is important to designate a single person as the "finder" which means they will be solely responsible for taking any evidence found into custody. The reason for this is to greatly simplify the chain of custody. By having a single person responsible for collecting and documenting the items to be taken, it becomes much easier (in court or otherwise) to verify who did what, when it was done, and so on.



With the previous steps completed, it is time to begin the search. Recall that up to this point a very limited number of people were actually in the location who were documenting things. Since that is complete, a larger number of people can be released into the location to search for electronic evidence. The number of people searching usually depends on how big the location is as well as any time constraints that may be present for a given location. The latter concern would come up more in the context of a business vs. a search warrant.

As searchers find items, several things may need to occur. If a device is powered on, a triage/preview of the device should be considered which would include memory collection, checking for encryption, extracting key artifacts for processing on scene (battlefield forensics), and so on. Tools such as osTriage (available to law enforcement) or KAPE provide these capabilities. We will discuss KAPE in detail later in the course. The state of powered on devices should be captured before any interaction with the device via photograph or video to preserve the state of things prior to triage. Once the running device is processed, follow best practices when shutting down the device (clean shutdown, pulling the plug, putting a device in airplane mode, etc.).

Once the device is powered down (or for any devices found in a powered off state), it is ready for collection. At this point, the finder should be called to the location. The device should be photographed in place, along with any other devices connected, such as external hard drives, network cables, printers, etc. Be sure to photograph the front and back of the device if possible. If any serial numbers or other identifying information is present, photograph and document those as well. Once documentation is complete, the finder will collect the device, along with any necessary peripherals such as power cords for laptops, etc.

In general, items such as monitors, keyboards, mice, web cams, and such are not collected unless they have some means to store data or are somehow directly relevant to proving some aspect of your case.

For every device collected, several very important steps must be taken, including creating a chain of custody, adding items to an inventory, and storing the items collected in a proper container.

A chain of custody is a detailed record of everyone who maintained possession/control of a piece of evidence. It includes the full name and exact timestamp when possession was taken and when possession is passed on to another person, checked into evidence, etc. It is critical that there are no gaps in the times. An example may look like this:

Date	Name	Purpose
2018-12-01 11:15:27	Eric Zimmerman	Collection
2018-12-01 12:06:27	Eric Zimmerman	To evidence
2018-12-01 12:06:27	Kevin Ripa	Checked into evidence
2018-12-04 08:16:07	Kevin Ripa	From evidence
2018-12-04 08:16:07	Eric Zimmerman	For forensic review

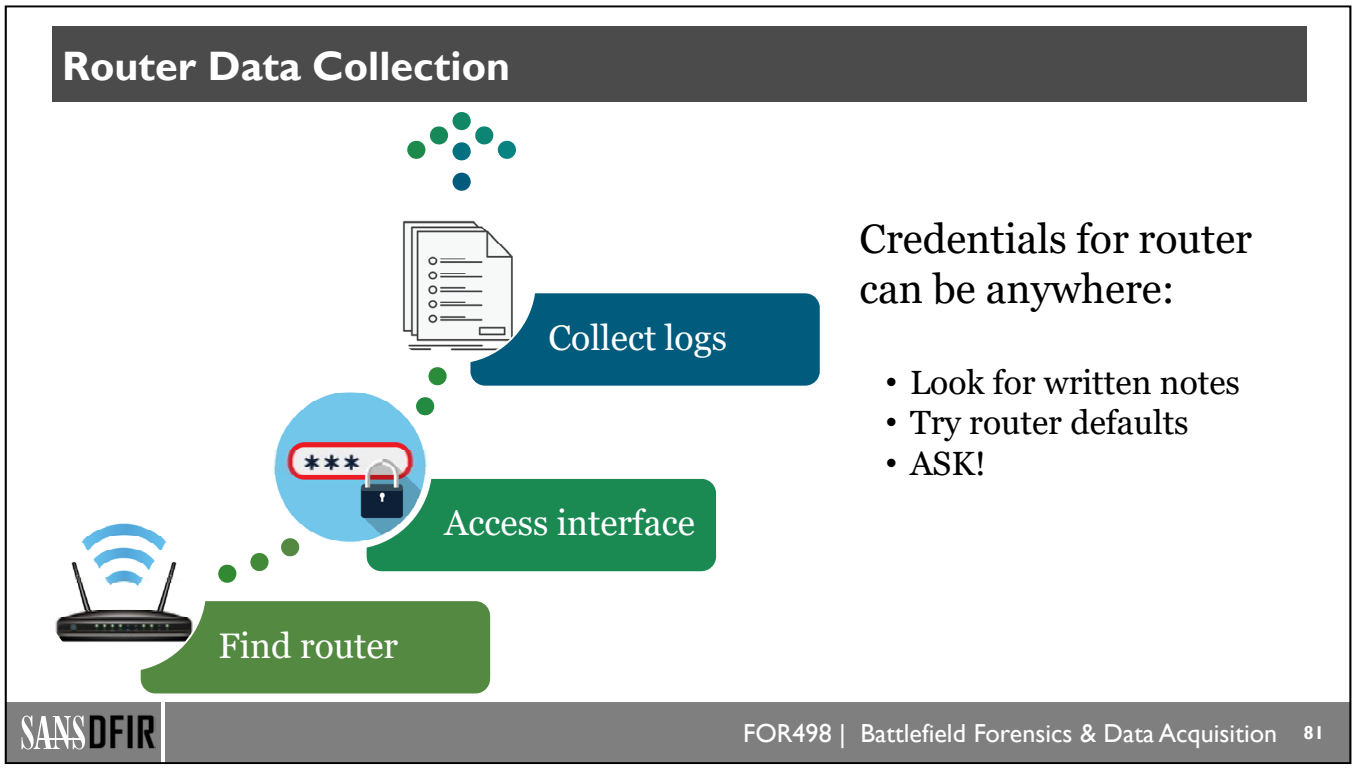
Notice how, with the exception of the initial collection, each interaction has the same exact time, but different names and purposes. This is what is meant by not having gaps in the times in the chain of custody.

Each device should have its own chain of custody created at the time of collection. This will greatly simplify things down the line, should devices need to go to different people for examination purposes. Consider the opposite case, where several devices, such as phones and laptops, are placed into a single box and a chain of custody is created for the entire box. If one person looked at the phones and another the laptops, it would be necessary to split the box into two separate collections, one for each type of device. By creating a chain of custody per device up front, this is avoided.

Once the chain of custody is complete, the device should be added to the inventory of items collected from the location. The inventory, along with photos and video taken during a search, keeps track of exactly what was found and removed from a location. The inventory can be a single list of items or can be broken down by room should there be a lot of items to be collected. For each device on the inventory, be sure to include the room designation where it was found and the make, model, and serial number.

Once a device has been added to the inventory, it should be placed in the appropriate type of storage container. In most cases, a plastic evidence bag should be used for electronic devices. If there is concern about needing fingerprints from a device, use a paper bag instead. For desktop computers, it is also recommended to place evidence tape over the power supply plug as a reminder the device should not be powered up unless absolutely necessary. For laptops, it is recommended to collect the power supply as well.

At the end of the search, the inventory will be reviewed and signed by both the finder and the original owner of the equipment, or a designated person who “owns” the site (homeowner, manager, etc.). Like the pre and post search videos/photographs, this eliminates any contention about what was seized when it comes time to return devices, etc.



Routers can be a great source for information about a network, including DHCP logs, web history, other computers, wireless SSID and password, and more. Of course, the details on the network are also present, including public IP addresses, MAC addresses, and so on.

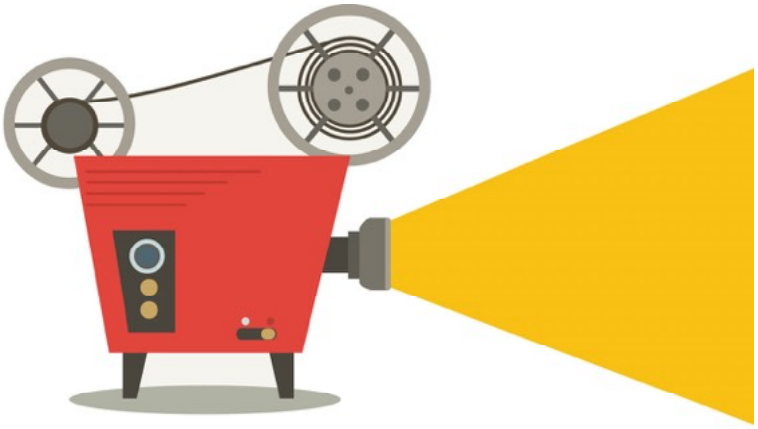
The first step is finding the router itself. The router will usually be located in the same place as a cable or DSL modem. In general, they will have one or more antennae on them, which makes finding the device a bit easier.

Once you have located the router, the next step is accessing the device. In many cases, the router may still have its default username and password in place. There are websites available that list some of the more common ones, but we prefer a quick Google search of the router's make and model along with the phrase "default credentials" or "default password" which should get you the information you need to log in.

If the credentials have been changed, asking for the information can sometimes be successful depending on who you are dealing with.

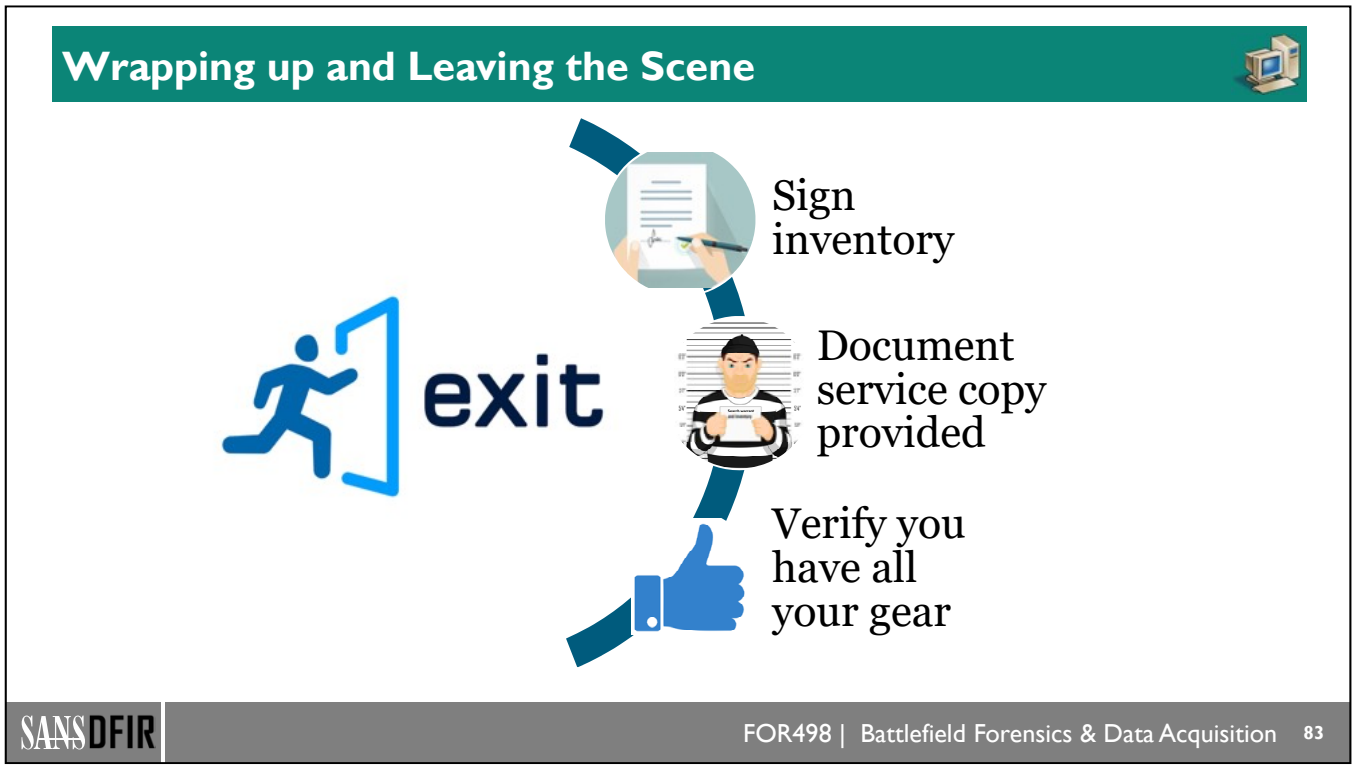
Once access to the device has been gained, a significant amount of information is available. For many of the details, it may be necessary to take screen shots of the various parts of the interface. In some cases, it is possible to download log files from the router, as well for things like DHCP or firewall logs. Be sure to check each area of the router to ensure the data is collected.

Movie Time!



- 1_1: Router ID Description
- 1_2: Router Data Collection

This page intentionally left blank.

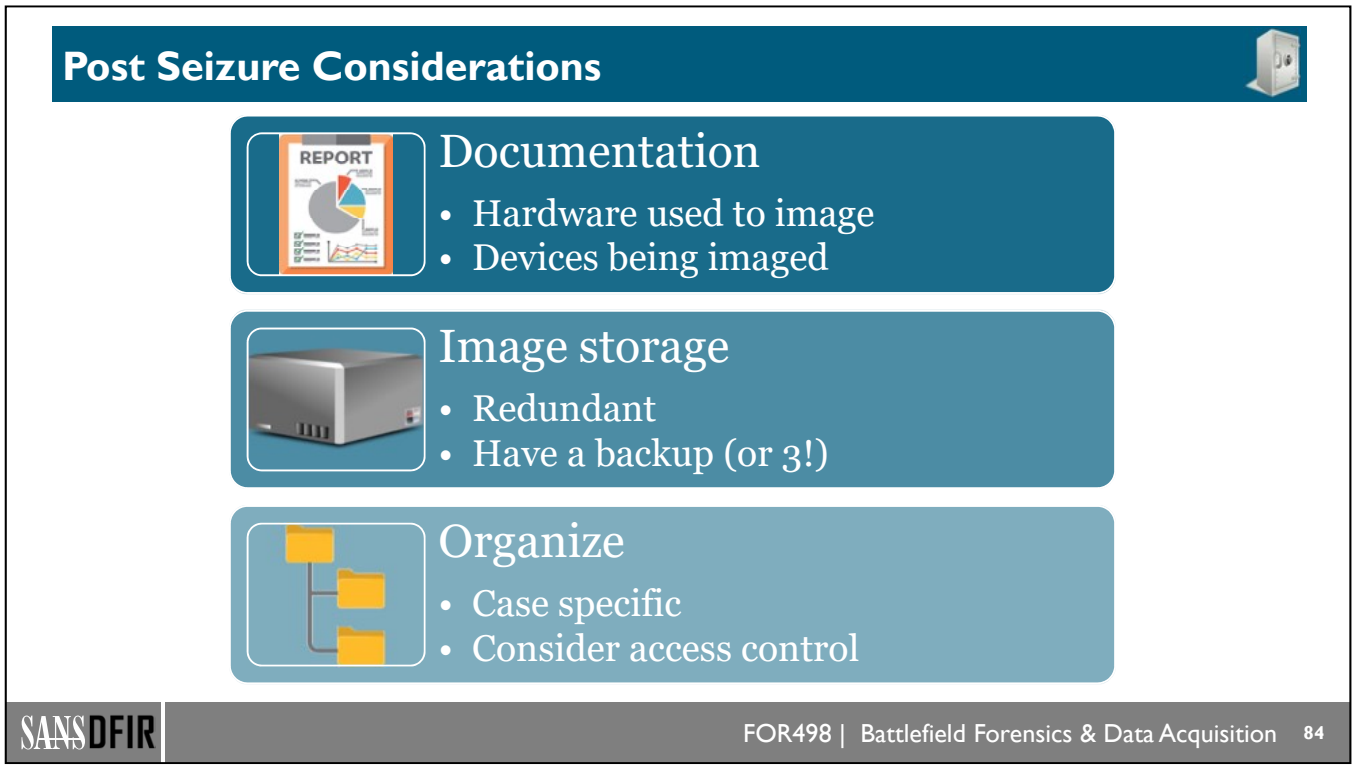


At the conclusion of a search, the inventory should be reviewed and signed by both the finder and the owner of the property. This ensures that what is being collected, is reflected on the inventory, and that both parties agree on the list.

Once the inventory is signed, a copy should be left with the owner or contact person. When a search warrant is involved, a copy of the warrant should also be left with someone. It is often a good idea to take a photograph of the owner or contact person holding the service copy of the warrant as well as the signed inventory. This prevents any argument later that these items were not given to the owner/contact person (and it is rather fun to do!)

When exiting the premises, be sure to check and double check that you retrieved all your gear from the location. This includes taking down room labels, collecting anything from your go bag, verifying all the collected devices are accounted for, and so on. Once this is done, the scene can be released.

Once the scene is released, the collected devices should be transported to a secure location and checked into evidence. If this is not possible for logistics reasons, be sure to take adequate steps to protect collected evidence from tampering or theft. For example, do not leave items in a car overnight outside secured space.



Just because the search is over doesn't mean there isn't anything else to do, and often what you put in place before you even go out on a search will make your life easier once you get back to the lab. The following are some things to consider when setting up procedures for dealing with evidence once it has been collected.

Documentation

- Be aware of the physical storage devices inside a computer. There may be more than one and there may be RAID (Redundant Array of Independent Disks) involved. Recording the details of both the computer and each storage device inside is critical.
- There is a difference between an enclosure and the actual storage device inside of it. Be sure to document both. Understand that there will be a serial number on the enclosure, but this is NOT the same as the serial number for the hard drive inside.
- For each physical device, be sure to document the following:
  - Make, model, serial number, and capacity;
  - The room the device was found;
  - Where the storage came from (i.e. what enclosure was it in).
- Maintain chain of custody of all items taken from other devices.

Imaging

- When imaging devices, ensure proper forensic methodologies are used such as write blockers, etc.
- Document the hardware and software tools being used to image a device.
- Devices should be imaged to an accepted format such as E01, DD, AFF4, etc.
- Document who imaged the device, when it was imaged, and what the results were.
- Be sure to document any failures or discrepancies.

Storage of Forensic Images

- Storage of images should be done on a device that provides for redundancy/file system resilience such as a NAS device or other file server solution.
  - Synology NAS devices are cost effective, scalable, and robust
- Storage devices should employ security to ensure only authorized people can access case related data.
  - NTFS group permissions
  - Individual accounts per user who needs access to data
- When working with case data, a copy of the data should be made to avoid working with original forensic images.

Storage Organization

- Storage should be organized in such a way to make it easy for investigators and server administrators to find data.
- Depending on how many cases are being created, and how long those cases need to be around, possible solutions include:
  - Single drive letter pointing to the global storage location;
  - A unique case identifier, such as a case number, at the top level;
  - Sub folders for Images, derivative data, reports, etc.
- When a case is closed, archive the case data per recommended data retention policies.
  - Move case data to offline storage device;
  - Can be individual hard drives or a secondary NAS with tighter security restrictions;
  - Consult with legal professionals to ensure you follow any regulations or guidelines for your industry.

Backups

- In addition to data redundancy on the storage device itself, regular backups should be done and maintained offsite to ensure data is available in the case of a disaster.
- All case related data should be backed up regularly.
- Consider encrypting the backups depending on the security of the offsite location.

Summary

- Ensure the Go Bag is always ready to Go
- Plan for as many scenarios as you can
- Personnel safety is always the number one priority
- You only get one chance to properly collect evidence
- The router is often overlooked as a source of data
- Document the scene before and after collection
- Photos are a must, but video to augment the photos is best
- Make sure you have all your gear before leaving the scene

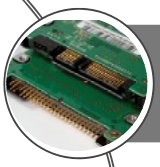
This page intentionally left blank.

FOR498.1: Evidence File Quick Wins & Scene Management Agenda

- SIFT Introduction
- 1.1 Intro to Digital Forensic Acquisition
- 1.2 Understanding the Data
- 1.3 Scene Management & Evidence Acquisition
- 1.4 Device & Interface Identification

This page intentionally left blank.

Device & Interface Identification



Storage Interfaces

BIOS/UEFI

Data Collection Before the Data Collection

This page intentionally left blank.

Storage Device Interface

In the early days of mainstream forensicating, Small Computer System Interface (SCSI) [1] and Integrated Drive Electronics (IDE) [2], also known as Parallel Advanced Technology Attachment (PATA), were the interfaces of the day when talking about hard drives. There were, and continue to be a few different SCSI interfaces, and these will be discussed on a coming slide.

As time progressed, the hard drive interface changed to the Serial Advanced Technology Attachment (SATA) [3]. These weren't the only interfaces in a computer. Floppy Disk connections also existed but have been deprecated from the motherboards of today.

eSATA, USB in various versions, Firewire, Thunderbolt, SAS, CF, SD, NVME, M.2, PCIe, and it becomes easy to see how it would be difficult to keep up on all of these interfaces. But keep up we must.

Any examiner should be well up to speed on the connections for the SATA interface, as that is the most commonly used one today. Examiners that have been around for as long as yours truly certainly are also familiar with PATA. But many have never had to deal with the myriad of SCSI connections, numbering certainly north of 10. In fact, 3-4 are rather commonly used today.

We are currently experiencing a significant change yet again, in consumer grade interfaces, with the increasing market share of solid-state drives looking more like sticks of RAM than hard drives. In these, there are currently 5 commonly seen connection styles.[4] To the uninitiated, they look the same, but certainly are not, as anyone who has had to image one has found out!

The manufacturers of various devices often drive the need for different storage types and interfaces. There can be no question that any hard drive including SSD using a SATA interface could not be used in any of the slimmest laptops of today. The MacBook Air and the Dell XPS, just to name two, simply are not thick enough to house the interface connections necessary to contain a SATA hard drive.

As if understanding and recognizing the various interfaces isn't enough, we as forensicators must have the ability to interact with these devices in a responsible (ideally write blocked) manner. This usually starts with being able to connect the device in some way to an acquisition computer. We then are faced with two concerns. The first being that we generally don't have every possible interface on a computer, and the second being that even if we have the interface, we can't write-block it. This is certainly a situation where SAFE Block would be indispensable. Notwithstanding this, the alternative would be to use an adapter of some kind to convert the drive interface into something we CAN write block. This is usually found in the form of an {*insert interface name here*} to USB adapter; USB being possibly the most universal interface in use today. Maybe it is no coincidence that the 'U' in USB stands for Universal!

In situations such as this, it is best if the forensicator has a wide array of adapters at her disposal, and a potential pipeline for the times when she doesn't. This is yet another great argument for getting to know the forensicators in your community. Without this, your only option is to be able to source the adapters, if they even exist (and they usually do). The problem is that with the most unique ones, an adapter may have to be sourced directly from China. Even if it is within your own country, it is probably not being sold on Amazon and can't be had with an overnight purchase. Of course, we also know from experience that discovering this need never happens on a Tuesday morning. It happens at 8:00 PM on a Friday evening of a long weekend.

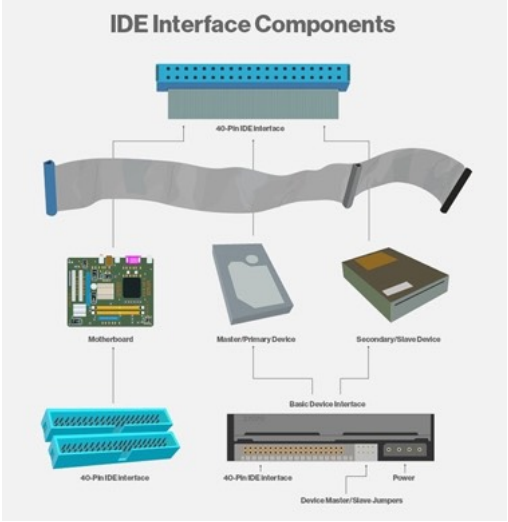
[1] SCSI | `https://for498.com/c08ng`

[2] IDE | `https://for498.com/5rlpn`

[3] SATA | `https://for498.com/j4h2s`

[4] SSD Interfaces | `https://for498.com/4kvjy`

IDE: Integrated Drive Electronics



Integrated Drive Electronics (IDE) [1] hard drives (also known as Parallel Advanced Technology Attachment or PATA) were the first mainstream, consumer grade storage drives on the market. The personal computer (PC) market began its meteoric rise with the advent of Windows 95 and its Graphical User Interface (GUI) based style of interaction, led by the invention of the mouse. Its data transfer rate was a blistering 17 MB/s.[2] In 1998, advances were made to the point that IDE supported a transfer rate of 33 MB/s, and at the point that it started being deprecated for the SATA interface and standard in 2005, it was an amazing 133 MB/s. Of course these transfer rates were under optimal conditions.

This hard drive connected to the computer's motherboard via an interface with 40 pins. The cable had the capability of connecting to two hard drives, and their role was defined by the use of jumpers to set a hard drive as either a master or a slave on the system. The master typically held the operating system. With two drives being on one cable though, a user could expect to see a marked decrease in response speeds when both drives were in use. Another factor affecting the speed of operation was the channel layout. Both read and write functions happened on the same channel. To this day the same occurs with most drive configurations, which is one of the arguments for Redundant Array of Independent Disk (RAID) configurations. When using most configurations except RAID 1 and Just A Bunch Of Disks (JBOD), the read and the write happen on separate channels.

Most motherboards of the day had at least 2 headers, or connection ports for IDE cables, allowing for up to 4 hard drives to be connected at once. Today most motherboards no longer have PATA headers, or connections.

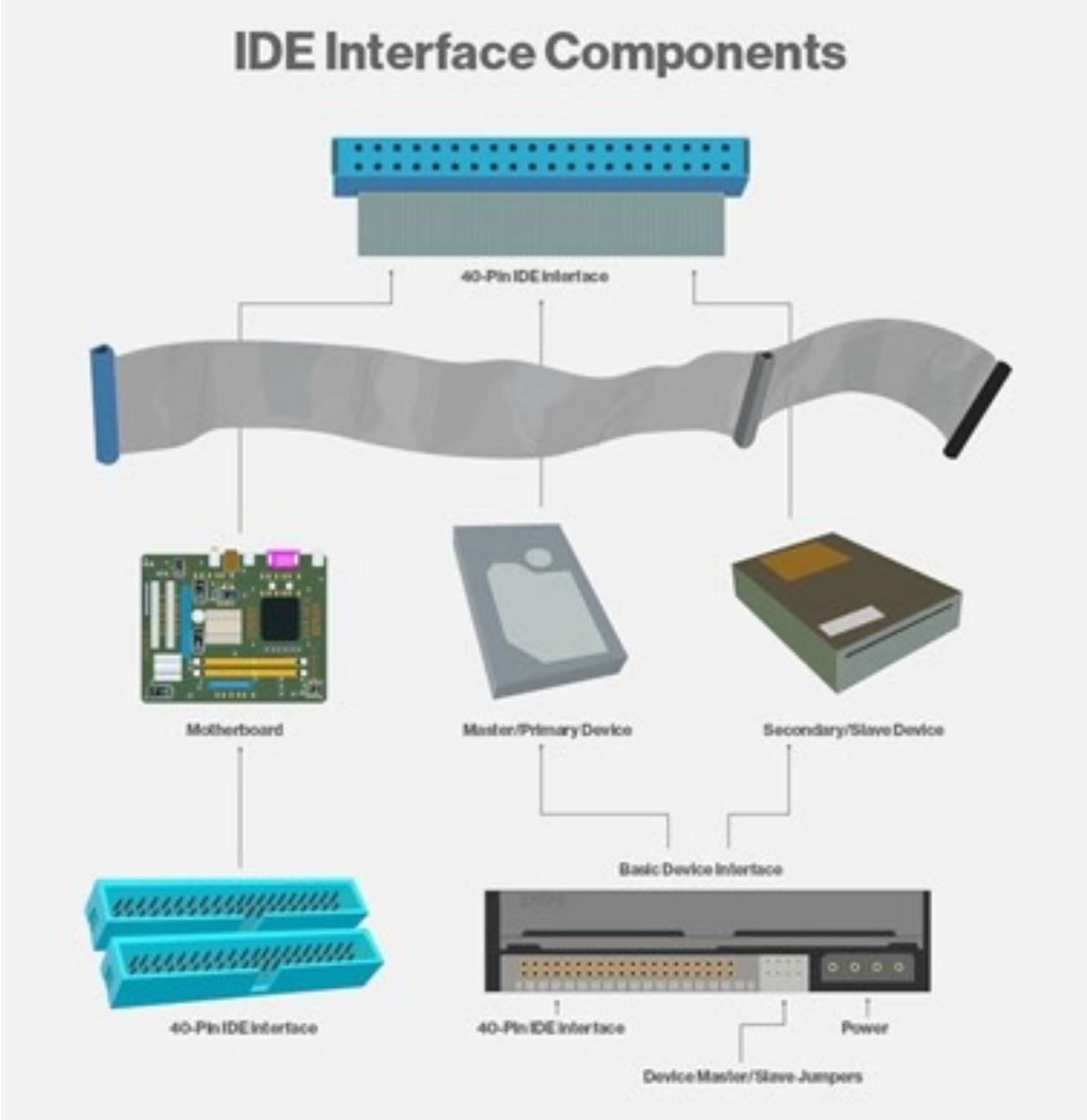
PATA was available on both 3.5", or desktop hard drives, as well as 2.5" or laptop hard drives. One of the features that had the largest impact on data input output (I/O) rate was the rotational speed of the hard drive itself. 3.5" hard drives came in speeds of 5400 RPM and 7200 RPM, and laptop drives came in those same speeds, but also had a 4200 RPM version. Rotational speed had an impact on cost, but also affected power consumption. In the case of laptop drives, cooling was another considerable factor, because faster drives ran substantially hotter. This continues to be a factor today.

It is not terribly uncommon to see these drives still in use today, especially in the case of some enterprise applications. Most hardware acquisition devices have the capability to directly connect to these devices even today, and there is no shortage of easily sourced adapters that convert IDE (PATA) to SATA.

It is a certainty that a forensicator will come across these drives for the near future, so being able to interact with them is a must.

[1] IDE (PATA) hard drives | `https://for498.com/5rlpn`

[2] Data Transfer Rates for Various Media | `https://for498.com/ripek`



SATA: Serial Advanced Technology Attachment

1.8" Micro SATA hard drive with Micro SATA connector

Serial Advanced Technology Attachment (SATA)[1] appeared on the consumer market around 2003 in the form of SATA 1, and by 2008, SATA was being used in 99% of consumer computers. SATA describes both the connection type, as well as a protocol.

Unlike PATA, SATA only allows for the connection of one hard drive per channel. Thus, motherboards typically will have no less than 4 SATA connectors.

A distinct advantage of SATA over PATA is the ability of the drive to be hot-swappable. That is to say that notwithstanding the boot drive, any other drive could be plugged or unplugged without having to shut down the computer first. SATA could not do this natively, but rather leveraged Advanced Host Controller Interface (AHCI). Not all SATA capable motherboards had this controller, but it is quite commonplace today.

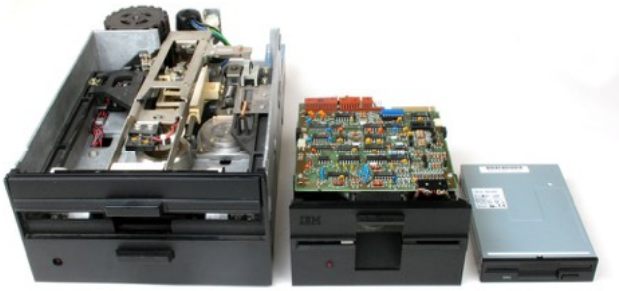
SATA is available on 1.8” hard drives found in some ultra slim laptops, 2.5” hard drives, and 3.5” hard drives. The interface is also seen on both rotational and solid-state media. In the case of 1.8” drives though, the format is a micro SATA (mSATA). Rotational speeds in spinning hard drives are the same as for PATA drives, however Western Digital manufactures a hard drive called the Raptor (aka Velociraptor), which has a rotational speed of 10,000 RPM.

SATA 1 had a data transfer rate of 150 MB/s back in 2003, allowing it to surpass the fastest PATA I/O speeds. A year later, SATA 2 came out with an I/O rate of 300 MB/s. It took another 4 years for the transfer rate to double again to 600 MB/s with SATA 3 in 2008. It is important to note that the I/O speed of the technology trumps potential I/O speed of a hard drive. For example, if a user plugs a SATA SSD with a 520 MB/s I/O speed into a SATA 2 connection, the user will see 300 MB/s at best, and not the rated 520 MB/s. From the perspective of a forensicator, the choices you make in your acquisition equipment are affected by this as well.

[1] Serial Advanced Technology Attachment (SATA) | `https://for498.com/j4h2s`

Photo Credit
Dsimic CC BY-SA 3.0, `https://for498.com/il742`

Other Magnetic Media



There are a great deal of additional non-standard (at least today) storage devices that exist. Jaz and Zip drives still surface from time to time in forensic examinations, and usually they come in a banker's box full of the tapes, but no player to play them! Floppy disks are still plentiful at the 3.5" (124 KB/s I/O) form factor, but 5.25" (62 KB/s I/O) and 8"(31 KB/s I/O) still come up from time to time as well[1]. Floppy style drives connected to the motherboard via a 34-pin ribbon cable that was not visually unlike an IDE cable.

Jaz and Zip drives were storage media created in an attempt to provide the consumer with storage sizes larger than the 1.44 MB that a floppy disk could hold. They did not last long, but there were a number of different form factors for the disks they played, and this causes problems today when trying to find a drive to play them in.

These drives typically connected to the computer via USB, Serial, or SCSI interfaces.

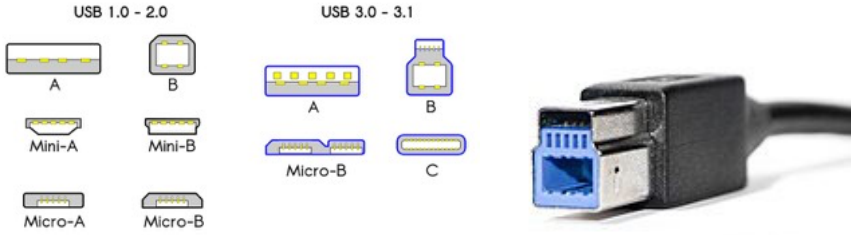
Just to add more complexity, tape drives also add to the confusion. Many enterprises use tape drives for backup operations. These drives connect via SCSI, but the tapes themselves are anything but standard. It is not uncommon to have to source not only the proper drive to play a particular tape, but try to source the specific software and version that created the backup on the tape! There are literally dozens of different form factors with differing compression and density capabilities. It is often touted that tape storage is dying out, but considering that Oracle recently released the T10000C tape drive[2] that can hold 5 TB (240 MB/s I/O) of uncompressed data, it would seem that rumors of the death of tapes are greatly exaggerated.

[1] Interface Bit Rates | `https://for498.com/wfq7e`

[2] Oracle T10000C | `https://for498.com/iea9l`

USB: Universal Serial Bus

Connector Type	USB 2.0 Image	USB 3.0 Image
A		
B		
Micro-B		
Mini-B 5 Pin		-
Mini-B 4 Pin		-
C		



Universal Serial Bus (USB)[1] is more of an industry standard that establishes specifications, as opposed to a storage medium, however since it is ubiquitous in connecting storage media to computers, it must be discussed. As well, USB is the defacto conversion standard when trying to find an adapter between two dissimilar interfaces.

USB was first released in 1996, and its first generation had an I/O speed of 1.5 MB/s. It was quickly replaced by USB 2.0 in 2000, which had a considerably enhanced I/O speed of 60 MB/s. In 2010, USB 3.0 was released, with an I/O speed of 625 MB/s. This vastly exceeded the I/O of the majority of devices that were using it to connect to a computer, and adoption became truly widespread for the use of everyday data transfer, as opposed to an offline, long term storage medium.

Today USB 3.1 is the standard on all Apple products, as well as gaining market share elsewhere. The I/O speed of USB 3.1 is 1250 MB/s. A new USB generation 3.2 was recently released, and touts an I/O speed of 2500 MB/s.

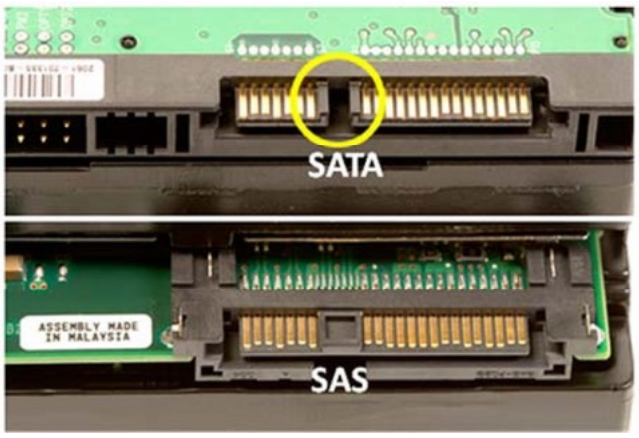
With each new version of USB, there was a cable interface change, and most shops today have bins full of USB cables of one fashion or another. A significant feature of USB is its ability to provide power to the peripheral it is connected to. It can provide a stable 5 volts of power and a constant 500 milliamps of draw. Not only is this enough electricity to power 2.5" external hard drives, but it also provides enough current to charge most portable devices.

In the case of 2.5" external hard drives, and in order to exploit the full capabilities of the I/O in relation to the drive, connection interfaces are no longer through an adapter in the device that converts USB to SATA. It is actually USB direct to the Printed Circuit Board (PCB) of the drive. It is important to understand that significant amounts of I/O transfer speed can be lost at every connection point. "Direct to board" lowers the connection points of a device from 3 to 2. A potential 33% gain in I/O speed.

[1] USB | `https://for498.com/lxkih`

Photo Credits: Viljo Viitanen `https://for498.com/f5vyx`

SAS: Serial Attached SCSI



Serial Attached Small Computer System Interface (SAS)[1] is a line of hard drives primarily created for the enterprise market, and equaling, if not exceeding market share over SCSI drives in today's data centers.

These drives come in 2.5" and 3.5" form factors. Their actual connection interface looks very similar to the SATA interface. In fact, a SAS computer connection can receive any SATA 2 and newer drive. You cannot, however, plug a SAS drive into a SATA computer connection due to the physical bridge between the data and power contacts on the drive itself.

SAS itself is now in version 4, with I/O speeds of 2400 MB/s, but has been around since 2005. It attains such monumental speeds in part due to the rotational speed of the drive itself. SAS drives, although available in 7200 RPM speeds, are typically 10,000 or 15,000 RPM. The drive is also constructed to much higher standards than a SATA drive. All of this obviously results in a higher cost per drive.

Some other differences between SAS and SATA[2], despite their looking very similar is that SATA can use up to a 1-meter cable, and the power cable is separate, whereas a SAS cable can be up to 10 meters, with the power cable and data cable inside one sheath. The drawback to SAS besides price though, is the storage size. SAS drive sizes are usually smaller than commensurately priced SATA drives.

[1] SAS | `https://for498.com/gh5n9`

[2] SAS vs SATA | `https://for498.com/34-h9`

SCSI: Small Computer System Interface

Although we have already covered a Small Computer System Interface (SCSI) drive previously in the form of SAS drives, they just scratch the surface of what is available from this interface. In the enterprise world, SCSI is ubiquitous, and has been going back many years.

Generally speaking, SCSI, drives have always been much more expensive, although much smaller in storage size, than their non-SCSI counterparts such as PATA and SATA. As with SAS this is typically because of higher rotational speeds that subsequently contribute to faster I/O. They have a much lower failure rate, and better reliability. This is attributable (besides because of construction) to something called areal density, or the density of sectors per platter. We discuss this further in the section on data recovery.

One of the difficulties facing forensicators with SCSI drives is that there are no less than 18 different SCSI interfaces that have been introduced over time. 6-8 of them being quite common. In fact, one of the first hardware write blockers sold by Guidance Software was a SCSI device.

When faced with the acquisition of a SCSI drive that may be more than a couple of years old, the examiner's biggest challenge is going to be finding not only the proper PCI card, but the proper hardware drivers for that card. Adapters are quite uncommon for SCSI technology simply because there is not normally a use for such devices. In most cases, SCSI hard drives are not found in singles, but in arrays of multiple drives. In these cases (besides for exceptional situations), the forensic image is being collected logically through the device housing all the drives. Imaging single, standalone SCSI drives is probably the largest challenge for an examiner, notwithstanding massive datasets.

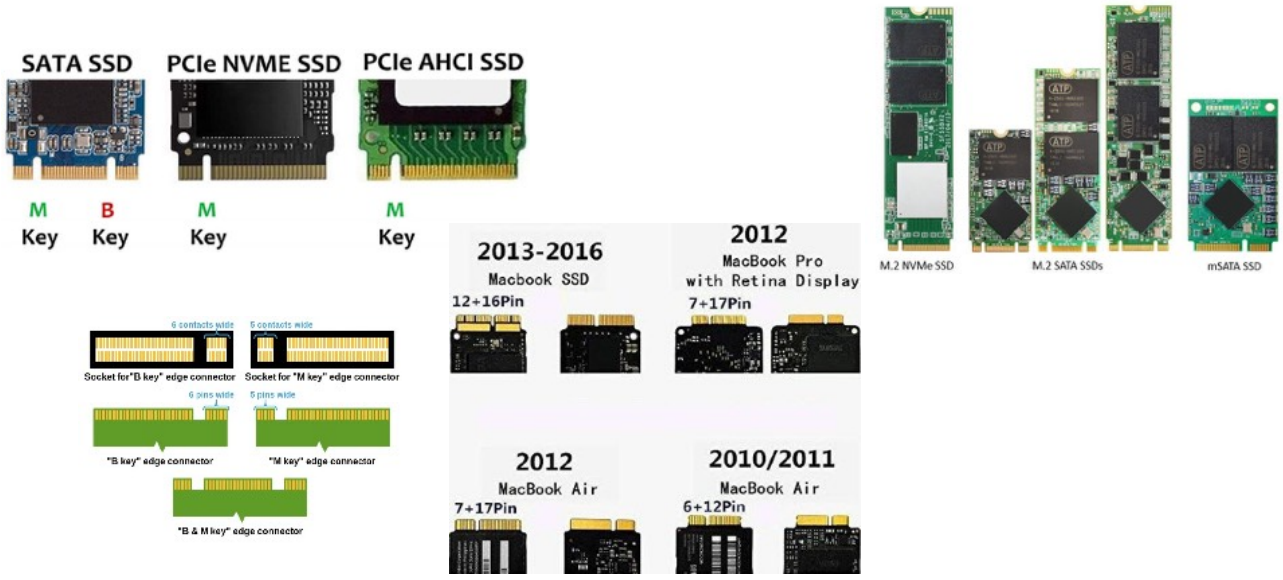
PCIe: Peripheral Component Interconnect express

Peripheral Component Interconnect express (PCIe) is the next generation of hard drives. The PCIe is an actual circuit board that is inserted into a slot on a computer's motherboard. It has historically been used to allow for the interfacing of non-standard drives, to allow for expansion space, and to allow for components that did not come natively on a motherboard of the day, such as a wireless card.

It deserves mentioning in this module about connections and interfaces because it is very commonly carrying a slot for the latest breed of SSD, as well as, in some cases, BEING the SSD. Currently there is no faster hard drive interface than PCIe. A PCIe hard drive utilizing the NVMe protocol can attain data read speeds of 4000 MB/s.

In the case of PCIe cards that contain a removable SSD, the SSD can be taken off and placed in a write block device for imaging. But in the case where the PCIe IS the SSD, it must be imaged either in place, or the PCIe card must be moved to a computer that has the capacity to write block at that level. Currently SAFE Block is the only manner in which to write block at a PCIe interface level.

NGFF (m.2): Next Generation Form Factor



Besides the early SSDs that are still widely in use today that carry the SATA interface, there is a new breed of SSD. It is called the Next Generation Form Factor (NGFF), more commonly known as m.2. A great deal of confusion surrounds these drives because although they are all called m.2, some use the Advanced Host Controller Interface (AHCI) common to the SATA (even though the connection is entirely different), and others use the newer Non-Volatile Memory Express (NVMe) protocol. Both NVMe PCIe and AHCI PCIe use the PCIe channel, but that is where the similarity ends.[1]

In terms of speed, there is no comparison. NVMe is consistently 4 times faster than AHCI. But how do you know which is which? They all (mSATA, m.2 SATA, m.2 NVMe, as well as Mac flavors) look mostly the same. Much like a stick of RAM. But in fact, on closer inspection of the connection, they are not the same. Each of them has a slightly different pin out, as well as something called a 'key', or slot in the connection interface. Even though a drive with an 'M' key and a drive with a 'B' key simply look reversed, there are actually different numbers of pins on the short side. Add to that the Apple SSDs, and all bets are off. These are not compatible with their non-Apple brethren and again, have different pin outs, even among their different models.

You would be excused for mistaking these for sticks of RAM or memory. They are essentially just that. RAM is used in a computer because it is solid state, and therefore, much faster than a spinning hard drive. Imagine if you could use your RAM as your hard drive. Challenge accepted. This is NVMe SSD. Now that is not to say that it is as fast as RAM. While it seems like the I/O speeds rival that of RAM, the SSD only achieves those speeds in sequential reads and writes. RAM achieves those speeds even during random reads and writes. So RAM is not disappearing any time soon.

Adapters for imaging these drives are quite prevalent, and the latest hardware write blocker/imager devices all have optional adapter kits for these drives. Again, you must be careful though, as it is easy to mix these drives up!

[1] NVME vs AHCI | `https://for498.com/mi5s1`

Fibre Channel

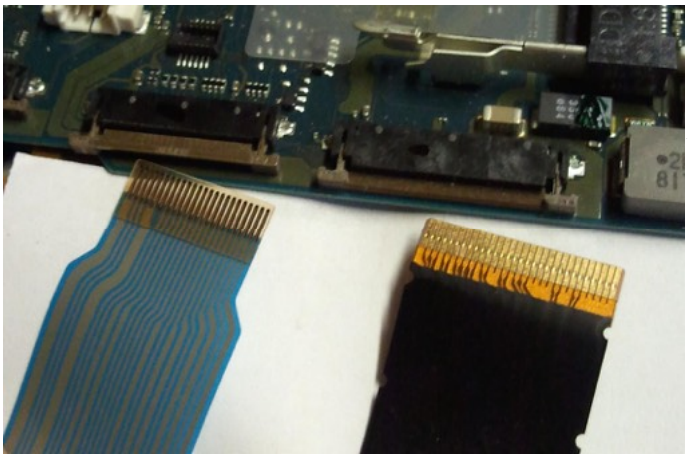
Fibre Channel (FC) is a drive interface used exclusively within enterprise and data centers. The FC hard drive is substantially more expensive than any of its counterparts. As an example, the cheapest normally addressable 300 GB FC hard drive retails for $300.00 USD as of January 2019. $300.00 USD would buy a standard 10 TB SATA HDD! When I say, “normally addressable FC”, what I mean is that in many cases, FC drives are not normally addressable. FC drives are very commonly used in large drive arrays such as Dell EMC solutions, and as such, need all the efficiency that can be squeezed out of them. FC drives are normally built with 520-byte sector sizes [1], rather than the standard 512 bytes. This allows the Error Correction to be occurring within each sector, rather than before and after it. This comes at extreme cost. The same 300 GB FC drive with 520 bytes per sector is $735.00 USD in January of 2019.

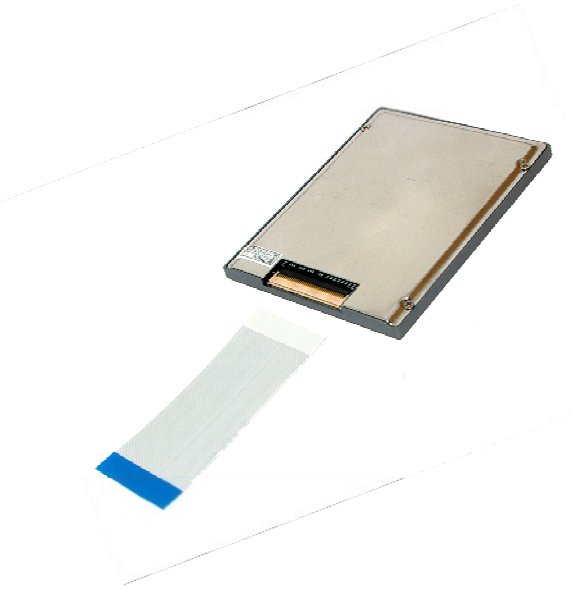
There simply is no adapter to convert the FC interface to anything else such as USB. In fact, it is a challenge just to find the appropriate adapter that will allow for this type of drive to connect to a normal computer. Remember that being FC, this drive uses glass cables (fibre optic) to transfer data. Once you find an adapter (called a T-Card adapter), you need a fibre optic cable as well as a fibre PCIe card to connect it all to a computer. The next issue is getting a normal computer to recognize and communicate with an FC drive of 520 bps. The only combination this author has found to date is via SAFE Block as a write blocker, and then EnCase will see the physical drive.

Having said all of the above, there are only very specific circumstances in which you will ever need to image a single FC drive from out of an array. Under most circumstances, internal data will be collected logically, and the examiner is using the host array controller to access the drive. This happens in a live environment though, with no ability to write block.

[1] Advanced Standard Formats | `https://for498.com/a-qw1`

ZIF/LIF: Zero/Low Insertion Force

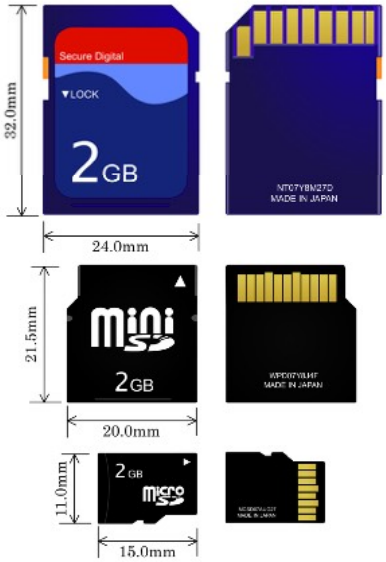




Zero Insertion Force (ZIF) and Low Insertion Force (LIF) drives were a very short lived (but still found) format of drive connection. ZIF/LIF was used primarily for 1.8" PATA drives in ultra-thin notebooks but have since been phased out. ZIF/LIF as a connecter is still used today and continues to attract wide use for connecting different components inside laptops.

When Hitachi and Toshiba had their short run with ZIF connections, they actually created their ZIF drive interfaces for different THICKNESSES of the ribbon cable, so that you would ruin the cable if used in the wrong drive. Tableau has ZIF and LIF adapters, and the ZIF cables are actually labeled for Hitachi or Toshiba.

CF/SD Compact Flash/Secure Digital



Compact Flash (CF)[1] and Secure Digital (SD)[2] are most commonly seen as storage media inside small portable devices such as cameras, video cameras, and cellular devices. If you have any of these cards in your devices, you probably have a reader for them that is USB and can be write blocked in that manner. In addition, many CF cards can be write-blocked via a physical switch on the card itself.

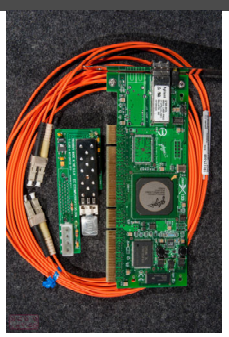
There is no substitute for a good card reader. In the case of microSD cards, it is commonplace to see an adapter where the microSD card slides into a miniSD adapter, which slides into an SD adapter and into the computer. You have now introduced 3 levels of complexity where there should only be one. As well, this method is not designed for forensics, and introduces problems of its own.

[1] Compact Flash | `https://for498.com/va30h`

[2] Secure Digital | `https://for498.com/eb24z`

Adapters

- IDE 2.5 – 3.5
- Micro SATA to SATA
- IDE to SATA
- ZIF/LIF
- SAS to SATA
- SATA to USB
- SCSI to SCSI
- Fibre to PCIe
- SSD styles to PCIe


When it comes to adapters, much like anything else, you get what you pay for. Cheap cables lead to packet loss, which you will not immediately attribute to a lousy cable. One great example is the SATA to USB 3.0 cable made by Apricorn[1]. This cable has a SATA connection at one end and a USB connection at the other. It is perfect for imaging laptop SATA drives. Apricorn specifically makes one, and although it is more expensive than most, it is also the best. Any examiner that has spent any time with adapters like this can attest to the frustration of cheap cables.

A great way to amass connectors and adapters is to not throw them away[2][3][4]. Our lab is as responsive as it is because we maintain equipment from over 20 years ago. We may not use a part for 10 years, but the one time we need it, it sure feels good to have it and not have to spend hours trying to source one. Not to mention how long it takes to get things shipped in, sometimes to remote locations.

One last word to the wise. If an adapter or cable is worth buying, it is worth buying two.

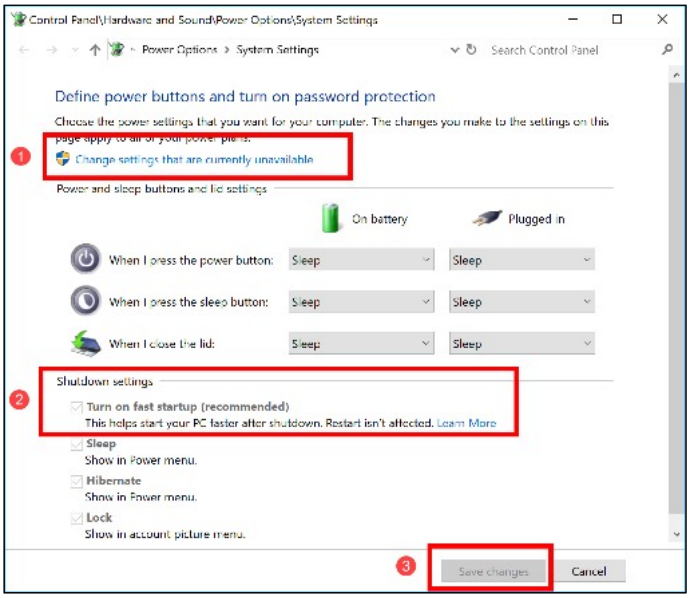
[1] Apricorn SATA to USB | `https://for498.com/fe2m3`

[2] Tableau Adapters | `https://for498.com/q0dy8`

[3] CRU Adapters | `https://for498.com/endr3`

[4] Fibre to PCI Kit | `https://for498.com/vsmjd`

Modern Standby



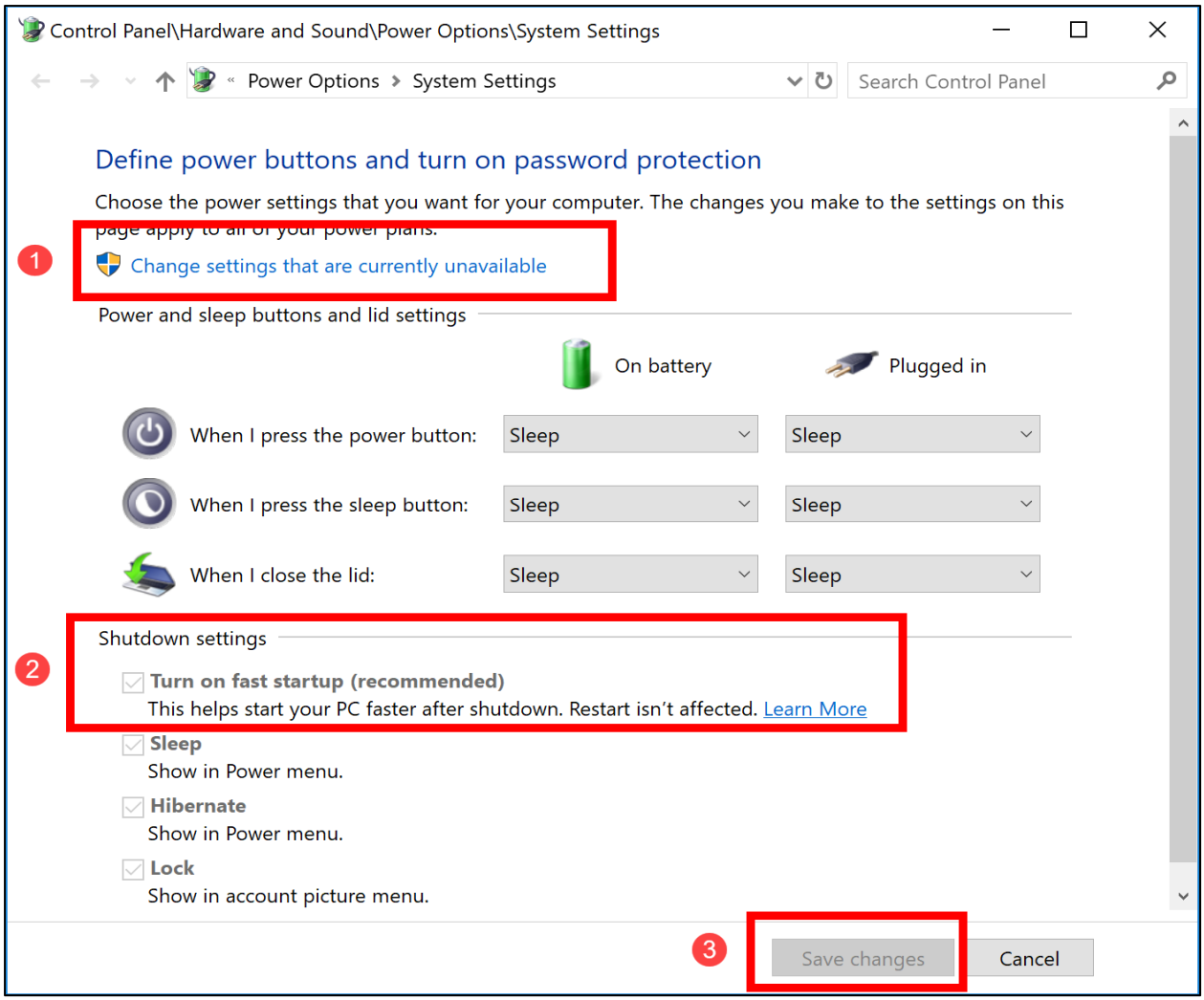
Modern Standby[1] is a feature that was introduced in Windows 8. Modern Standby creates various different layers of awareness within the operating system. It will allow the operating system to wake itself up at various times and to various levels in order to perform various different functions, such as updates and virus definition downloads. Because of this function, a computer by default, does not actually ever completely shut down. Even in the event that you click on the Windows Start button and select Shut Down, the computer is not actually completely shutting down. It is merely hibernating the kernel so that when you press the power button the computer comes on significantly quicker than from a cold boot state. A component that works within Modern Standby is a function variously called Fast Startup[2], or Fast Boot, and it is this component that affects us for the purpose of this slide.

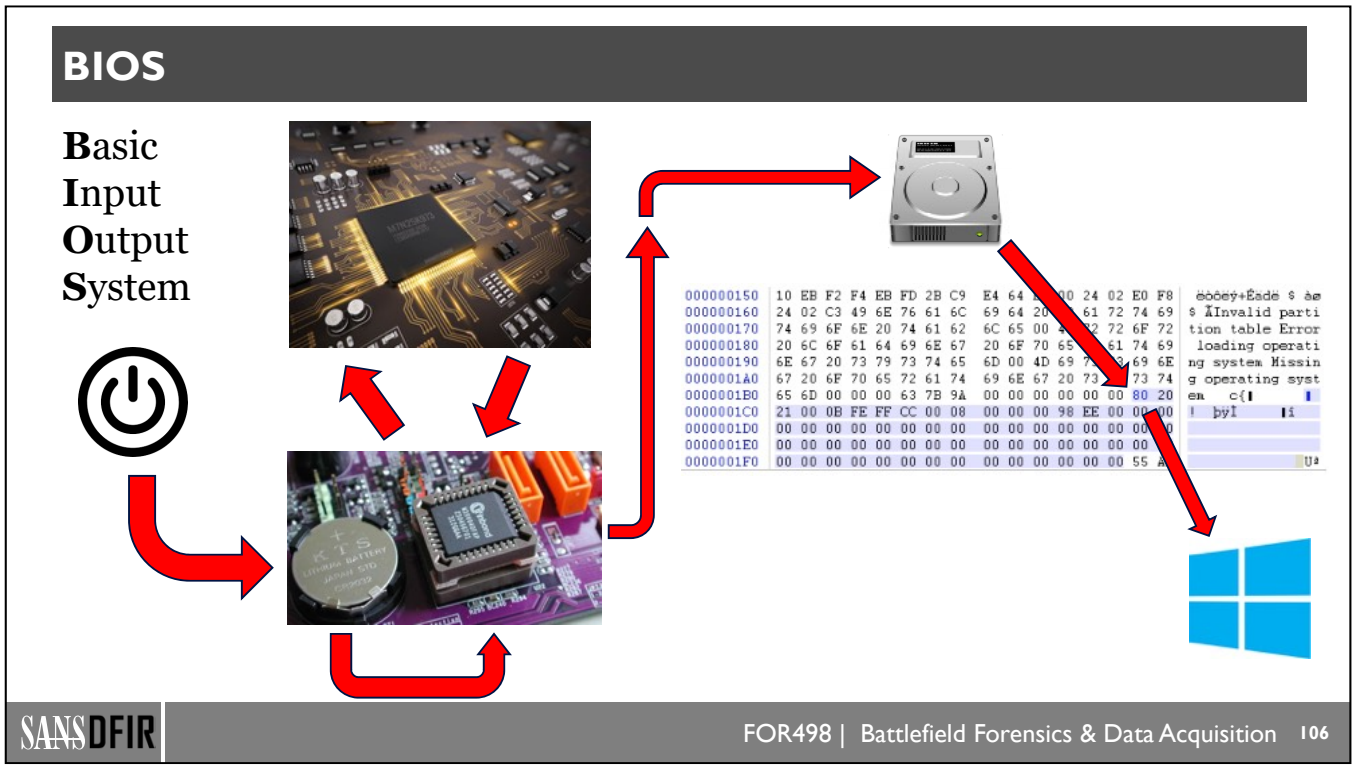
When attempting to enter the BIOS on a computer that has this feature enabled, the user will typically be unsuccessful. You cannot enter the BIOS while a computer is on, and with Fast Startup enabled, the computer technically is always on. If you want to start the computer into the BIOS, in most cases you will have to turn this Fast Startup feature off.

It is worth mentioning that Fast Startup can only function when the hibernation feature on a user's computer is available. If the user has disabled the hibernation function, Fast Startup will not even be a visible option in power settings. In the interest of saving size and space, we have disabled the hibernate feature in your FOR498 VM, and as a result you will not see the settings in the above slide within your VM. It is highly likely however, that you will see these settings on your host computer, suggesting it is a Windows computer.

[1] Modern Standby | `https://for498.com/mw60l`

[2] Fast Startup | `https://for498.com/brkmx`





BIOS stands for Basic Input Output System[1]. It is essentially an operating system on a small chip through which changes to the computer system can be made. To be clear, we mean the computer system and not the operating system. The BIOS will allow for the maintenance of settings such as security, boot order, date, time, connected devices, sound and video settings, and much more. It is a commonly held belief that these settings are kept in the BIOS, but in fact, they are kept in the Complementary Metal Oxide Semiconductor (CMOS) chip[2]. The BIOS merely holds the 'editing software' to make changes to the settings in the CMOS.

Ever wonder what happens when you press the power button on a computer? First, power is allowed to access the CMOS and read its settings. The CMOS is able to maintain those settings even when the computer is turned off, because of a battery beside it on the motherboard. Power is then passed through all buses and components on the motherboard to determine if they are operational, and if anything has changed since the last time it was accessed. If all is well (and it usually is), you will hear a short beep. This beep is provided after the completion of the Power On Self Test (POST). [3] The single short beep denotes that everything is in order. If a problem was detected, there would a different beep or series of beeps. These beeps, or POST codes, can be used to diagnose the problem. Suggesting the POST was ok, the BIOS information (but really CMOS information) is referred to and asked what is available for attached storage. It then starts going through this storage based on the settings in the boot order instructions. It will look first for a Master Boot Record (MBR) on a given device. If it does not find one, it moves to the next device in the boot order. Once it finds an MBR, it will read it to find out information about the Volumes and location of the Volume Boot Records (VBR). It will then check to see which one has a bootable flag set. The bootable flag is a setting of 0x80 at the first byte position in a given partition table. If a bootable flag is discovered, the boot process will be handed over to that partition, and Windows starts to boot.

From the moment the handoff to operating system occurs, and actually looking at a desktop, the operating system has caused to be created, deleted, or otherwise altered, no less than 3-5 THOUSAND files on a Windows 10 computer. While being created, deleted, or otherwise altered, these files could well be overwriting deleted data important to your investigation.

The BIOS is a very old technology, and takes very little space on a (usually) removable chip on the motherboard. This chip can be flashed to be updated.

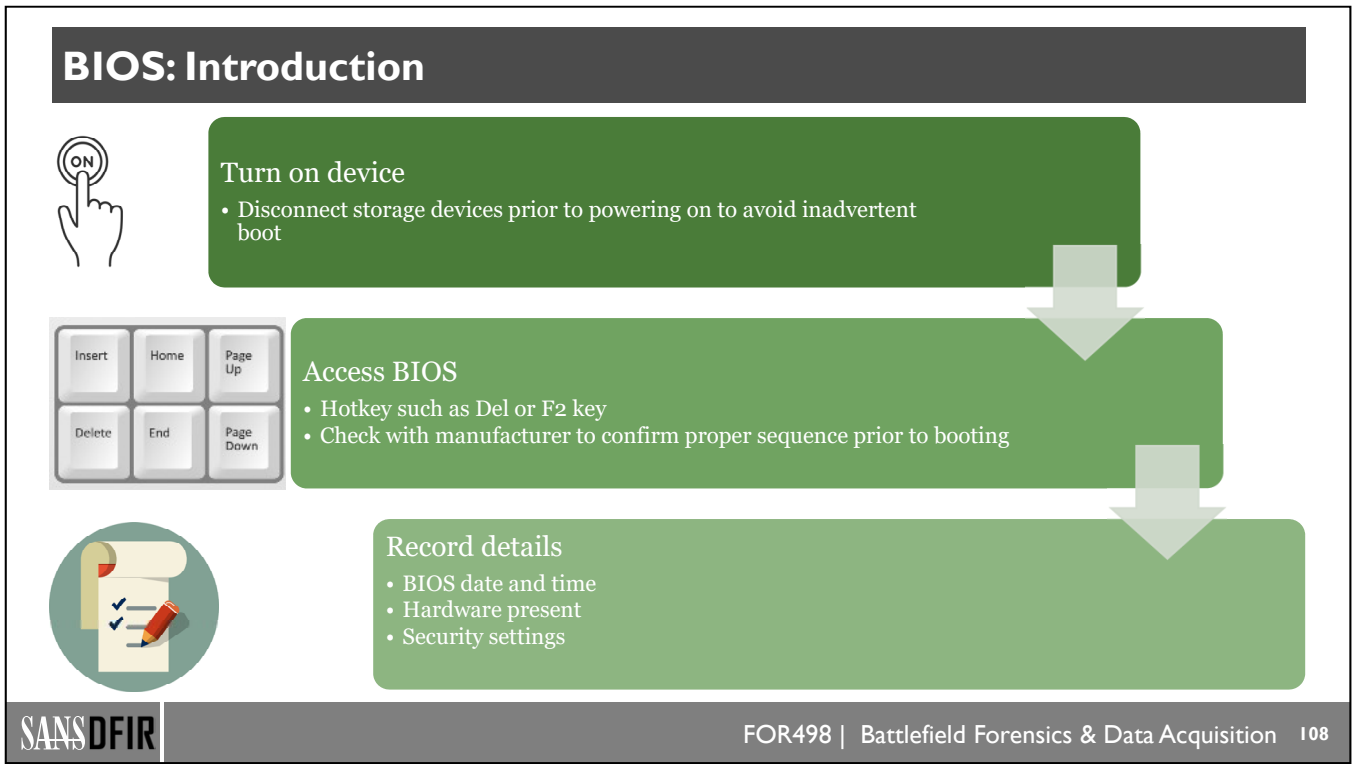
It is very important to realize that access to the BIOS/UEFI can be restricted by administrators through the use of passwords. Without the password, you cannot enter the BIOS/UEFI to collect data or change settings for imaging. Many BIOS in the industry have back doors, and the security can be bypassed[4].

[1] BIOS | `https://for498.com/zsa40`

[2] CMOS | `https://for498.com/u9nc3`

[3] POST and POST codes | `https://for498.com/yrz4j`

[4] CMOS/BIOS/UEFI Security Bypass | `https://for498.com/5z4sq`



Recall from the previous session our discussion on how we can go about accessing a device's internals to locate storage devices. This work is initially required in order to further document the exact state of the device before removing anything from it, to include written documentation, photographs, and so on. Since we already discussed how to get **TO** the actual storage device(s), we are now ready for the next stage in the process.

First, unplug power to the storage devices in the computer case. This is to prevent the computer from booting in case the keyboard shortcut to enter the Basic Input/Output System (BIOS) or Unified Extensible Firmware Interface (UEFI) is missed. Within a few seconds of the power being applied, a message is usually displayed on the screen indicating the proper key press to enter the BIOS/UEFI. When no such message is displayed on bootup, track down the manual for the device you are working on, as this will usually indicate the proper keyboard shortcut. Common messages indicating the proper key to use include: [1]

- **`Press [key] to enter setup`**
- **`Setup: [key]`**
- **`Enter BIOS by pressing [key]`**
- **`Press [key] to enter BIOS setup`**
- **`Press [key] to access BIOS`**
- **`Press [key] to access system configuration`**

In the special circumstance where the power to storage cannot be removed, additional care should be taken to ensure the BIOS/UEFI is entered when booting up. Should the device inadvertently boot up, shut it down and repeat the process. Be sure to document the startup and shut down time accordingly in this case as well. In this situation, additional details may be available in the BIOS/UEFI related to the storage devices including make, model, capacity, and so on. Be sure to record these as well.

Once the shortcut is pressed, the BIOS or UEFI will load. Each of these is a series of menus that allows for both inspecting and changing properties such as boot order, enabling or disabling ports, etc. The interface presented by the BIOS will have a much more basic feel to it in that you have to use the keyboard for navigation, whereas with UEFI, a mouse or other pointing device can be used to interact with screens including menu drop downs, radio buttons, and so on.

While most BIOS/UEFI implementations are somewhat similar, you may have to hunt around in the interface to find the sections containing the information we want to document. We are primarily interested in the following information:

- Date and time as reported by the computer. This will generally be on the first screen that is displayed after the BIOS/UEFI loads.
- Hardware information
- Security information

With the details recorded, power down the device by pressing and holding the power button for 4 seconds, and then remove the storage device from the computer. When the device cannot be removed, the appropriate procedure for the device should be followed. These specific use cases will be detailed in a later section.

[1] How to Enter the BIOS Setup Utility on Most Computers | `http://for498.com/217hs`

Generic BIOS (1)

One of the most fundamental purposes of the BIOS/CMOS (hereinafter referred to as BIOS) is to keep track of the system date/time. Changing the date/time in BIOS will change the date/time in the operating system. As well, changing the time in the operating system will change the time in the BIOS. The BIOS date and time is absolutely critical to a forensicator in determining the accuracy of the date/time stamps on the files in the operating system. Without capturing the BIOS date/time of the subject machine, you may have little to no visibility into the accuracy of metadata within the operating system and files.

The BIOS can be entered and manipulated even if there is no storage media in the computer at the time. This is why it is imperative to do this when the media is removed from the machine. While the hard drive is removed and imaging, it is a good time to perform this. If the drive cannot be removed, access the BIOS after the acquisition is complete. It is very easy to miss the right moment to enter the BIOS and if that moment has passed, the computer will continue to boot into the operating system. If you have already imaged the drive, this altering of data during boot up will not affect your examination.

BIOS examination must be done at the time of acquisition. This is not open for debate. If you image the drive today, but don't verify the BIOS until six months from now, the CMOS battery may have died in the interim and now the BIOS date/time is not accurate, nor will it even match the data in the forensic image.

Entering the BIOS is not as straight forward as it might first seem. Entering BIOS basically fits into two categories based on system type. Is the computer system a custom build, or is it a brand name like Dell, HP, Lenovo, etc.? We will come back to this in a moment.

At the moment the power button of a system is pressed, things start to happen, and you must be ready for them. Usually the first thing seen is a splash screen with the manufacturer's logo on it. In the case of custom builds, you will most likely see a splash screen with the motherboard branding logo on it. This screen usually appears

within 3-5 seconds of pressing the power button, and lasts 1-2 seconds. It is at this moment that you need to press the correct key combination to get into the BIOS. If you miss this moment, the computer will continue to boot.

As regards custom built computers, the key to press to enter the BIOS is almost exclusively the **`delete`** key. Pressing this button repeatedly from the moment of startup initiation will trigger entry into the BIOS. For brand name computers, it becomes more challenging. It seems every brand is different, and even models within a brand can be different. This can range from pressing any of the **F** keys, to having to press a combination of keys. In some cases, we can observe what the proper key stroke should be by watching the startup splash screen. It will sometimes tell us what key needs to be pressed. We can also turn to the Internet and Google for the answer. Barring any of that, odds are that it is one of the **F** keys, and the author has, on more than one occasion, simply swept his finger rapidly back and forth across the **F** keys during start up, and "stumbled" into the BIOS. Again, this is not something you want to do with the subject hard drive in place, unless imaging is complete.

Generic BIOS (2)

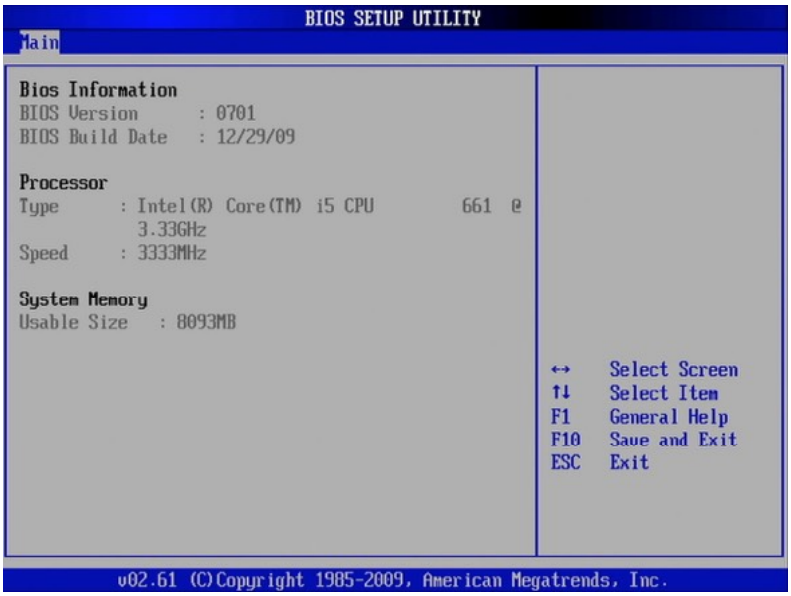
Just about every BIOS is different in some way, which makes it difficult to determine how to move around once you are in. The date and time are the most critical data that an examiner needs from the BIOS, and in most generic BIOS it will be on the first screen you see. Take a photo of this and record it on your evidence intake sheet. Writing it down is not enough, because now you will have to ask others to simply believe that you copied it down correctly.

In most BIOS, the examiner navigates through the possible options using the **`arrow`** keys, the **`enter`** key, and the **`escape`** key.

There are two other pieces of information that we strongly recommend collecting. The processor information and the RAM information. These assist in evidence continuity, as well as identification. The debate rages beyond that about what else to collect. Some examiners will collect the boot order. It is the author's experience that in most computers today, once you remove the storage media for imaging, the boot order changes, making the boot order irrelevant. However there is no such thing as too much information, or too many pictures. Find a system, meet the minimums, and then decide what else you might want to collect depending on a given situation.

In the particular case above, the examiner would press the down arrow key until reaching the System Information and then press Enter.

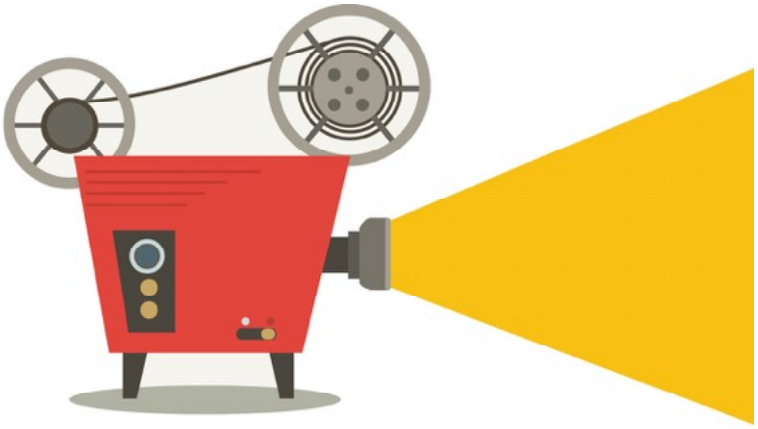
Generic BIOS (3)



We now see the Processor data, as well as the amount of RAM in the system. Take a picture of, and record this information.

When done with the collection of BIOS data, you can simply press and hold the power button for 4 seconds, and the system will shut down.

Movie Time!



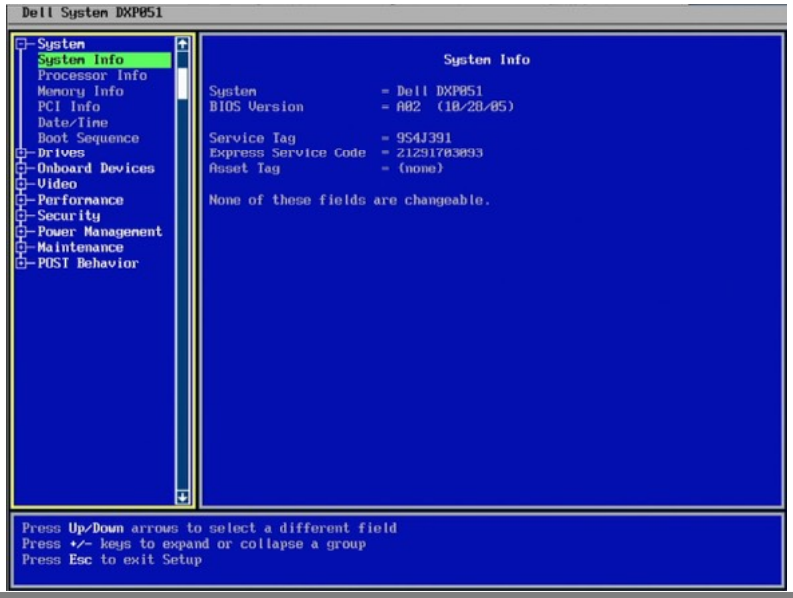
- 1_3: Generic BIOS

This page intentionally left blank.

Brand Name BIOS (1)

As stated before, brand name systems have different keystrokes and combinations for entering the BIOS. In the case of the particular DELL computer used for the demonstration, we see from the splash screen that we must press **`F2`** to enter the BIOS, often referred to as "Setup".

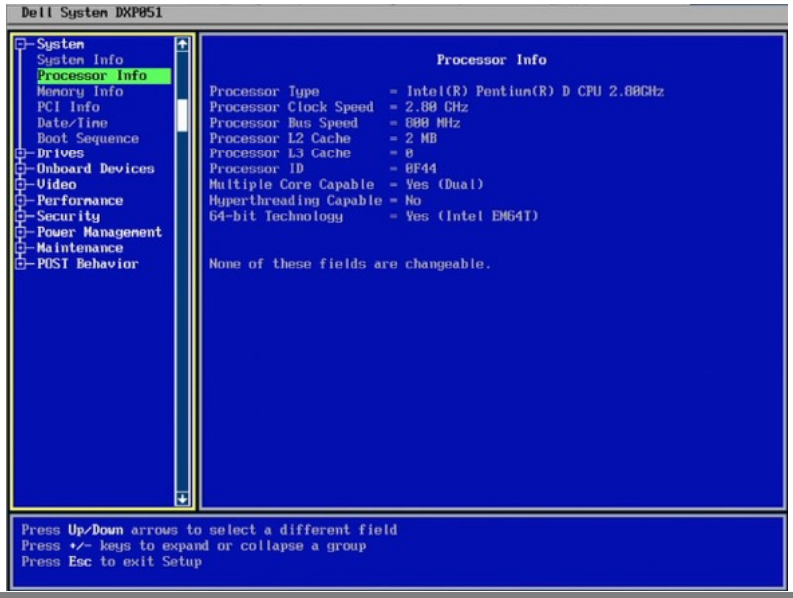
Brand Name BIOS (2)



With many brand name computers, once you are in the BIOS, there may be more useful data to collect. For example serial numbers and Service Tags. With DELL, the Service Tag[1] information is better than a serial number. It is unique to the custom build of that computer, and DELL can provide ownership and configuration information, if necessary.

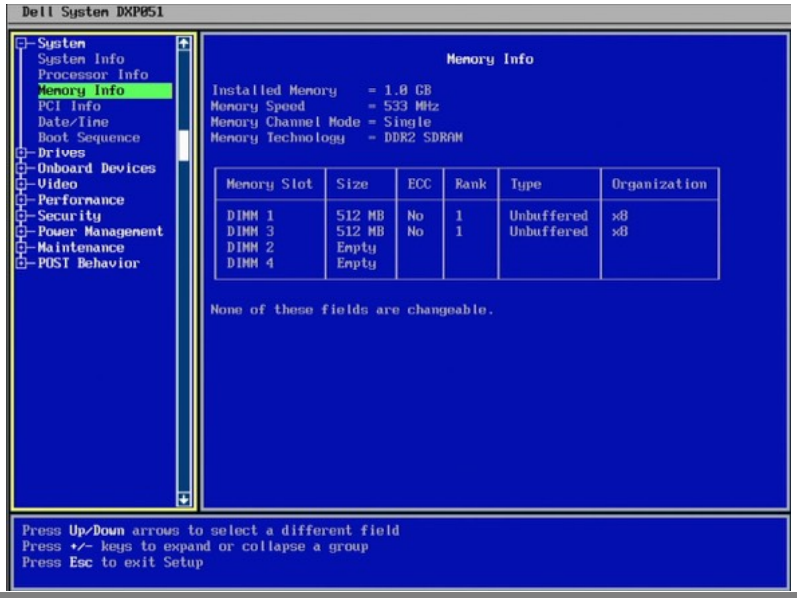
[1] DELL Service Tag Portal | `https://for498.com/g2bi1`

Brand Name BIOS (3)



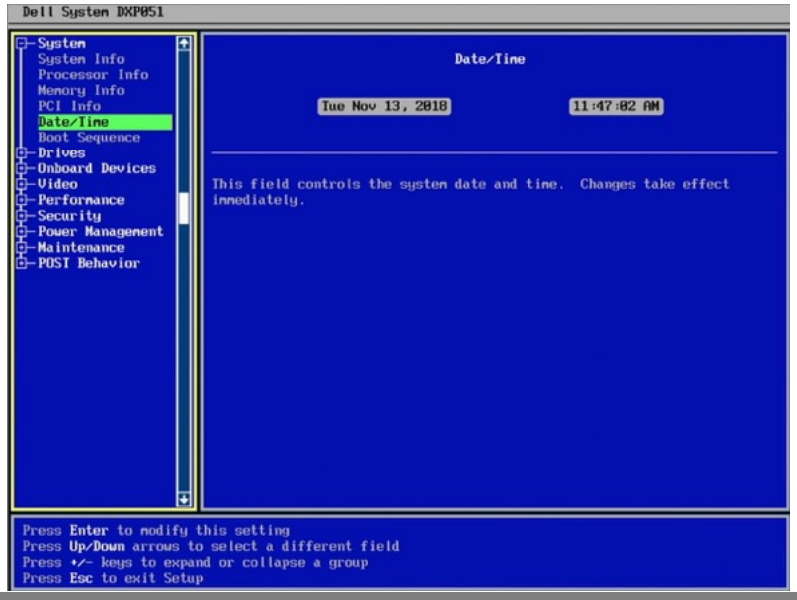
This page intentionally left blank.

Brand Name BIOS (4)



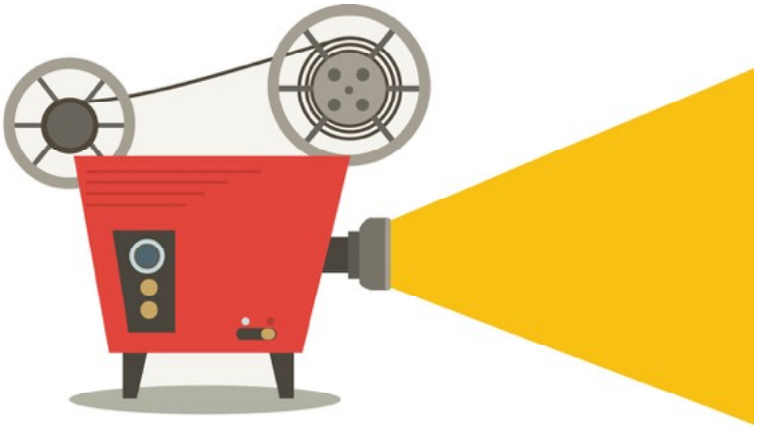
This page intentionally left blank.

Brand Name BIOS (5)



Once done with collection, press and hold the power button for 4 seconds to shut down the computer.

Movie Time!



- 1_4: Branded BIOS

This page intentionally left blank.

UEFI

Unified
Extensible
Firmware
Interface

The Unified Extensible Firmware Interface (UEFI) [1] is the industry replacement for the BIOS. It serves the same purpose, but with a great deal of added features. Besides being more visually appealing, a user can interact with it using a mouse. There are also a great deal more features and granularity to choices, as well as much better explanations on features available within the software. One of the most significant additions within UEFI is the ability to activate and deactivate Secure Boot.

Secure Boot [2] is a feature that checks the bootloader for authorized programs that are running at startup. A Microsoft certificate is stored within the UEFI to check the validity of the operating system items in the bootloader, so that if the computer gets compromised by a rootkit or other malware, it cannot run at bootup. This can extend to other programs that have been authorized, as well as even driver signing.

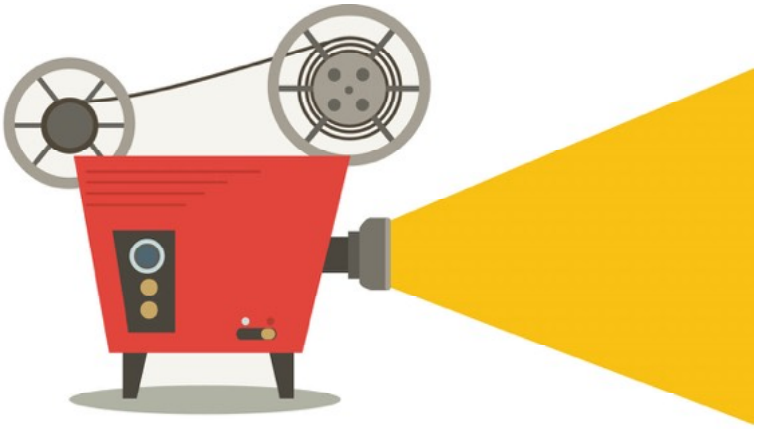
Another feature, and something to keep in mind during evidence collection, is that while booted into the UEFI, network connection is possible.

The UEFI is entered in the same manner as entering the BIOS, as laid out in previous pages, and the data we want to collect is also the same.

[1] UEFI | `https://for498.com/ka-6j`

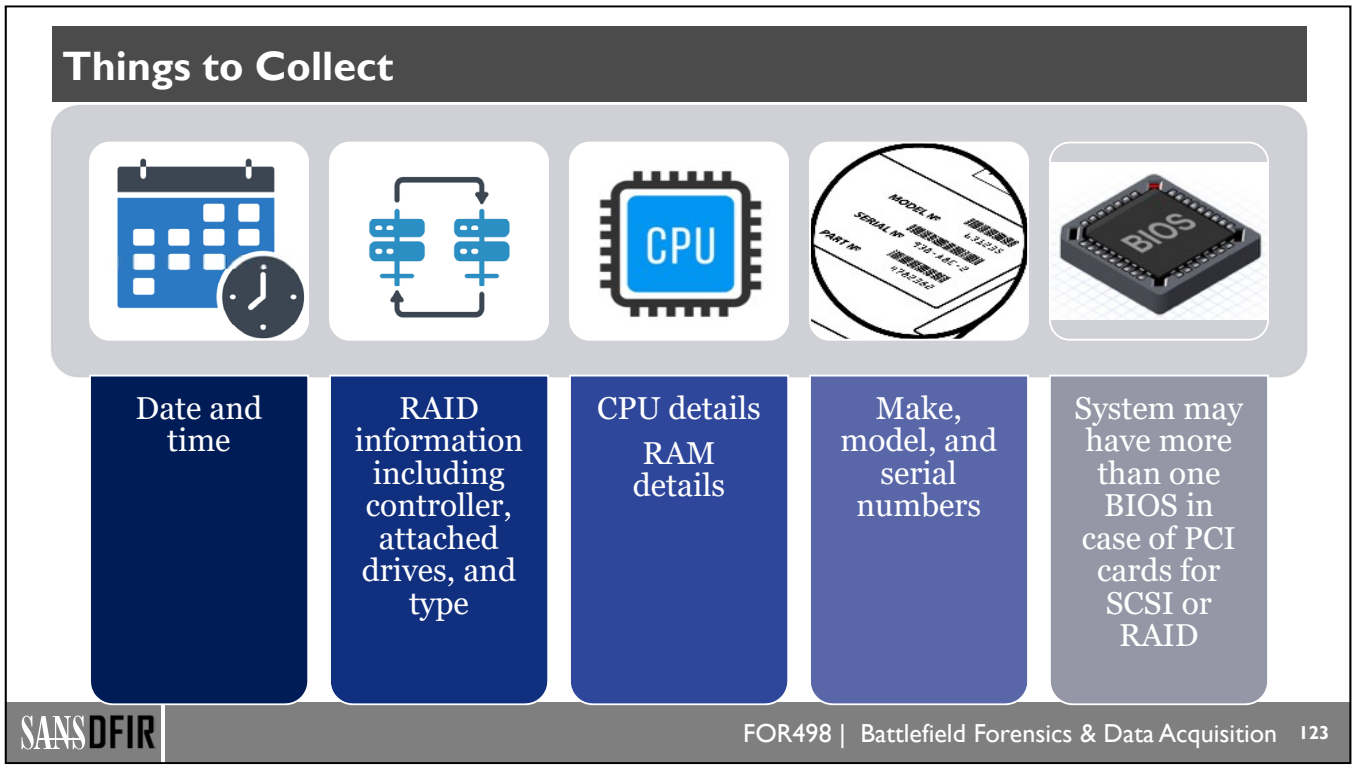
[2] Secure Boot | `https://for498.com/9tmzb`

Movie Time!



- 1_5: Generic UEFI

This page intentionally left blank.



It is worth noting that in the case of RAID, the configurations are made from the BIOS. If the computer being examined has more than one hard drive, and especially if they are the same size, you must assume a RAID configuration. In that case, it will be important to collect the RAID information. In the case of software RAID, the information will be within the system BIOS. In the case of hardware RAID (RAID configured from its own PCI card), the card itself will have a BIOS as well. To enter this second BIOS, you will need to first exit the system BIOS, and then during boot up, watch for another prompt that will allow access to the RAID configuration.

Uniquely in the case of RAID, it is imperative to also collect the RAID card details such as make and model of card.

During the collection phase, you get one chance to get it right. In most cases you are expected to return the equipment to the owner as soon as possible. You may not get a second chance to redo things.

Summary

- There are a great many different interfaces
- A robust adapter kit is a must
- Entering the BIOS/UEFI can be a challenge
- Information from BIOS/UEFI must be documented and collected

This page intentionally left blank.

Exercise 1.2

Interface ID & BIOS/UEFI

Synopsis: In this exercise, you will identify various hard drive interfaces, and then perform the steps necessary to access the BIOS/UEFI of your host computer and collect information.

Average Time: 10 Minutes

This page intentionally left blank.

Exercise 1.2 Takeaway

- BIOS is being deprecated for UEFI
- An examiner must know how to access BIOS/UEFI
- Without BIOS/UEFI information you may not be able to correlate data
- You can't image what you don't recognize
- You must have the correct adapters for a variety of storage media

This page intentionally left blank.