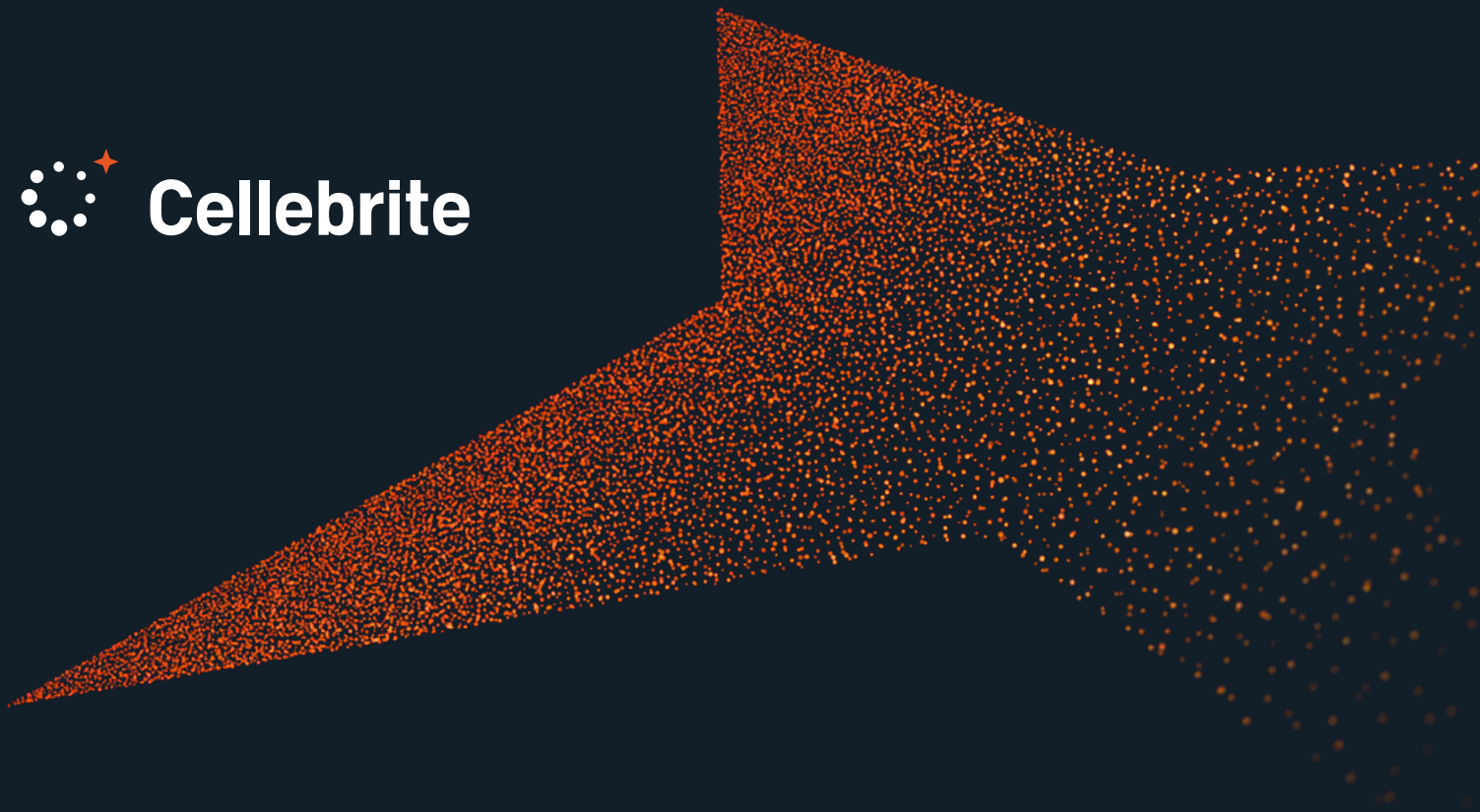


Cellebrite

Preparing for Court Testimony

What happens when you press that button?

2021 REVISION

Introduction

It is the responsibility of the examiner to testify to the integrity of the tools they use. This includes forming an understanding of the actions the software performs during extractions and parsing. Validating the findings presented by the software is an integral part of the job as we need to determine how the data landed on the digital media. This guide is aimed at providing terminology and methods used by Cellebrite so that you have clear and concise answers when asked to explain "what happens when you press that button".

Note: This document reflects the capabilities of the tools at the time of writing and is subject to change.

Cellebrite specializes in mobile device extraction and analysis via several different platforms, including **UFED** for extraction, **Physical Analyzer** for examination, **Reader** for sharing and review purposes, and **Pathfinder** for analysis.

Cellebrite also enables computer acquisition via **Digital Collector** and Computer data review through **Cellebrite Inspector**.

All Cellebrite Tools are designed to protect the integrity of the data at every step. However, there are several things that operators must bear in mind;

- Examiners should adhere to the same best practices for any of the UFED extraction tools.
- Consider isolating your forensic computer from the internet while performing examinations.
- Ensure that digital evidence is stored securely including backups.
- Remember that no mobile device is ever truly in a "write block" state while it has power.

UFED tools are designed to be as unobtrusive as possible. But sometimes it is necessary to take steps that leave a mark on the evidence.

Some extraction methods require a client (aka agent) to be installed on the device to facilitate the extraction. This client is a small piece of software installed to the devices' main memory and may be seen on the extracted data.

Utilizing an agent is comparable to leaving footprints at a crime scene; while certainly less desirable and should be minimized as much as possible, it is sometimes necessary in order to tend to victims or recover evidence. This type of necessity is acceptable in court as long as it is carefully documented and explainable. As a result, UFED will prompt the user before performing any activity such as installing a client. This also includes leveraging features for extraction like bootloaders and more.

In addition, UFED has a setting that will uninstall the client by default once the extraction is completed. While this is useful for intelligence professionals operating covertly, some law enforcement departments may have the policy of leaving the

client in place. Operators should therefore be guided by their own departments' policy in this regard.

Extraction

There are various types of extractions available which differ between different devices. They can be split up into two groups: **partial** and **full**. Understanding the type of extraction is important to understand the type of data you will / will not get.

An important factor that will affect of the level of extraction available will be the state of the device.

- Hot (AFU) / Cold (BFU)
- **AFU** and **BFU** are both terms used to describe to the security state of the device.
- **BFU** (Before First Unlock) is the initial state when the device boots. Most data is securely encrypted.
- **AFU (After First Unlock)** is the state of the device once the passcode has been entered for the first time. Although the data is always stored in a securely encrypted form on the device storage, most of the data is now accessible for decryption using cryptographic keys that have been loaded in memory after the correct passcode was entered.

Partial

- This would include Advanced Logical, Backups, File System, Hot (AFU), Cold (BFU) and Selective Extractions.
- Limited to files within the Allocated Space.
- File System extractions may include some deleted* and hidden files.

*Deleted files are limited to files marked for deletion but not actually deleted. Similar to those found in a recycle bin.

Backup

This extraction method simply utilizes the device's normal backup functionality. It is limited to the data that would be found in a regular backup and therefore is unlikely to include any sensitive files.

Selective Extraction

A relatively new addition to Cellebrite's extraction methodologies, Selective Extraction allows the operator to select just the applications they are interested in, reducing both the time and storage required for the extraction and satisfying limitations on what data should be accessed, keeping privacy in focus.

Cloud

In some cases, data may be available from the cloud account associated to the device being investigated.

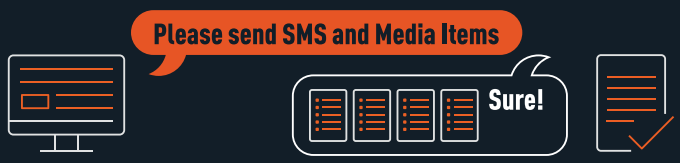
Cloud services such as Apple's iCloud and Google Cloud can be accessed with appropriate authority and credentials and data downloaded.

The data available is usually the result of a device backup and as such can be considered the same.

API Access

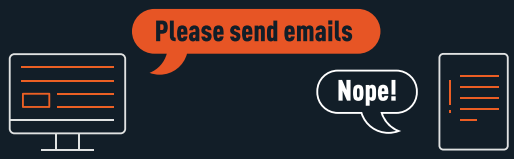
For the most part, Logical Extractions are performed by utilizing the device's API (Application Programming Interface). Just as the API allows commercial third-party applications to communicate with the device OS (Operating System), it also allows for the forensically sound extraction of data.

Upon connection, the UFED device loads the relevant vendor API to the device. It then makes read-only calls to request data. The device replies to valid API requests to extract designated content items such as text messages (SMS), contacts, pictures, etc.



An example of how an API call can be used to obtain data

From a technical standpoint, API-based logical extraction is straightforward to implement, and the results are provided in a readable format. However, it is limited to the scope of the content permissible by the device vendor. In some cases, data such as Email is not available via the API.



An example of how the API can limit access to data

For example, files from a third-party application may not be seen by the API and may not be accessible. To access this data, an examiner would need to access the file system and examine the data associated with the specific application. Deleted data is not usually obtained via API access.

Extractions of iOS Devices

Apple introduced encryption to iOS devices with the iPhone 4S in 2013. Since then, physical extractions has not been possible. As a result, Cellebrite introduced several methods for logical extraction of iOS devices. While some of the methods below are no longer in use, you may have extractions that were obtained with these methods, thus we are including them in this document.

Method 1 : Relies on the iTunes backup infrastructure.

Method 2 : Extracts backup data if the device is encrypted and the operator does not know the device passcode.

Method 3 (Legacy): Only available for jailbroken devices but obtained the most data out of these methods.

Full File System: Utilizes the checkm8 exploit for getting high level access to the device and extracting all files. This method will work on other available exploits for jailbroken devices.

More information on Checkm8 can be found at: https://www.cellebrite.com/en/a-practical-guide-to-checkm8

Full Extraction

- Physical Extractions
  - Can be bootloader based or advanced techniques such as JTAG, ISP and Chip Off.
  - Bit-for-bit copy of the original memory.
  - Includes Unallocated space and File Slack.
  - Data may be encrypted.
- Full File System Extractions
  - Full extraction of the active file system.
  - Allocated data only.

Physical extractions are the most comprehensive type of extraction, providing an exact duplicate of the accessible memory.

It is important to note that on newer devices, this data may be encrypted and near-impossible to decrypt off-device.

Using advanced methods such as checkm8 (iOS) or Qualcomm Live (Android), the UFED device is able to obtain temporarily elevated privileges on the device and extract all live files including hidden files.

For more information on extracting Android devices, please review the Android Data Collection Simplified blog at: https://www.cellebrite.com/en/android-data-collection-simplified

Bootloader

Bootloader methods take control of execution during an early stage of the device boot to elevate extraction privileges. This sometimes requires temporarily flashing custom files to the device partitions.

Most bootloaders are specifically designed for certain chipsets and vendors (Kirin, Exynos, Qualcomm, etc.). With newer devices, encryption comes into play, so you need to use the Decrypting Bootloader Option.

JTAG / ISP / Chip-Off

Although no Cellebrite product assists with extraction via these methods, they are important to understand as Physical Analyzer is capable of reading the resulting data.

All of these advanced techniques are hardware based and require some level of device disassembly and are therefore potentially destructive. With that, an associated element of risk to both the data and/or the device exists.

All these advanced methods result in a physical binary image which, assuming it is not encrypted, can be parsed using Physical Analyzer by selecting the appropriate chain.

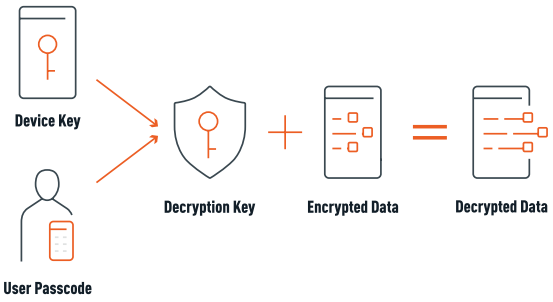
In the rare situations where extraction using a UFED tool fails, the user can simply restart the extraction. The failure does not affect the integrity of the extraction or the data on the device. Failures using destructive methods like Chip-Off, are generally non recoverable.

Encryption & Security

In the past, most mobile devices had a simple passcode that was used to protect the data on the device from unauthorized access. Many techniques existed for bypassing the passcode and this allowed full access to the cleartext (decrypted) data that existed on the device.

In recent years, security has become a much higher priority for mobile device operating system developers and as such almost all modern devices now utilize encryption to protect the user's data from these simple bypass techniques.

Encryption scrambles the user's data using unique encryption keys. Without the correct keys, the data is rendered useless. The encryption keys themselves aren't easily accessible as they are generally protected by both a unique key specific to the device and a passcode chosen by the user. As both parts of the key are required, it means that decryption usually must be done on the device; extracting the encrypted data and attempting to decrypt it 'offline' is not usually feasible.



An example of how data is decrypted using the user passcode and device key

Full Disk Encryption refers to the entire partition being encrypted with the same key. This is similar to how computers typically encrypt data.

File Based Encryption refers to a method where every file on the partition has its own encryption key. This is a much more secure method of encryption and not only allows for different levels of user access but also prevents deleted data from being recovered.

For the convenience of the user, additional methods for accessing the device have been added, such as Fingerprint or Facial ID. These methods offer a convenience for users to unlock their devices in the AFU state, however they do not replace the initial necessary requirement of the passcode to decrypt the data from a BFU state.

Deleted Content

Deleted data may sometimes be recoverable depending on the level of extraction obtained. The encryption type on the device will determine probability of success (Full Disk Encryption / File Based Encryption / No Encryption). Deleted data may be available in the following circumstances:

- It is not actually deleted, just marked for deletion (any extraction type).
- Data still exists in the free pages or Journal/WAL of a database (full extraction). Journals and WALs (write ahead logs) are transaction logs leveraged by databases. They temporarily store data until the database checks and merges the data from the log into the database.
- It still exists in file slack or unallocated space (physical extraction).
- It can be found as a by-product of wear-levelling (NAND level physical extractions).

"Wear-Leveling" is a method for NAND memory to move data around the memory chip to balance the cell usage and prolong the life of the device. It is handled by the controller and not typically seen unless a low-level, pre-controller extraction has been performed. A typical physical extraction would not contain wear-levelling artifacts as it would have been extracted post-controller.

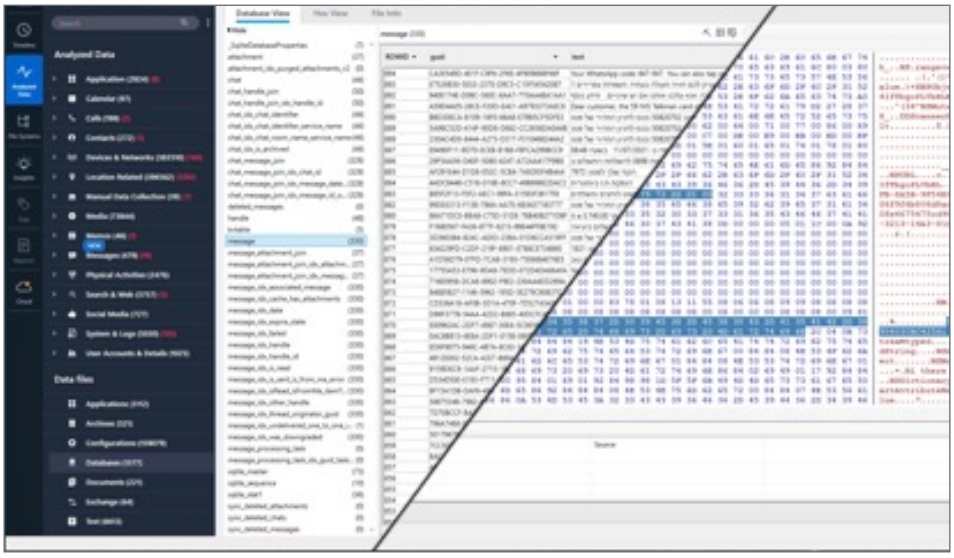
Decoding

The decoding process translates the raw data into a recognizable format.

Data extracted via API's and non-encrypted backups require no decoding because it is intrinsic to these methods, which present files such as pictures and videos as they are seen on the device.

However, data within other files such as databases, must be separately decoded to parse out the content.

Cellebrite Physical Analyzer automatically performs this decoding process, presenting decoded data in both a human readable format and as the raw data it is stored as in the device's memory.



Cellebrite Physical Analyzer displays data in human-readable form and in its raw state

Application Encryption

Some applications may utilize their own encryption or obfuscation to secure the data they contain. This is separate from the encryption of the user memory. These files will be decoded and presented as unencrypted files if supported.

Parsing

Parsing is the process of taking the readable data (after any necessary decoding or decryption) and organising it into a neat, usable format. For example, taking the SMS database and presenting the messages as time-stamped, organized conversations with named parties.

Parsing the data is one of the main functions of Physical Analyzer. This is done by using parsers written by developers who have researched the specific application or artifact. But with over 1 million apps available, it is impossible to support everything.

It is also important to note that even one small change made to an application by the developer will require further research and recoding to support, all of which can take time.

Can parsing miss data?

It is possible for automatic parsing to miss some data. Application parsing typically relies on programming that tells the software to look for and interpret data in certain places on the device, based on the research that was conducted.

For applications and artifacts that aren't supported by parsers, Physical Analyzer offers several options:

File Viewers

Physical Analyser has built in tools for viewing many native file types such as SQLite Databases, Binary PLists, JSON, XML, Protobuf, Text, Image and Video.

When used in conjunction with the directory browser, these features allow operators to navigate, filter and find files of interest and view them manually to find relevant data.

SQL Wizard

SQLite Wizard is a built-in tool within Physical Analyzer that helps you visually decode data from databases. It lets you build SQLite queries and map database fields to Physical Analyzer models. These queries can be saved for future use.

Carvers

Physical Analyzer can search the extraction for various types of data using a process called "Carving".

This means that the data will be searched for defined patterns that match locations, strings or images even if they aren't within known files.

While this method can be powerful to uncover new sources of data, it can also result in many false positives and should be used as a starting point for further research rather than an indication of found evidence.

App Genie

App Genie can search databases for recognizable objects and uses AI to attempt to parse the data without any further user input.

It is important to note that artifacts found using the App Genie are not the result of research and results should therefore be manually verified.

Fuzzy Models

Fuzzy models is an earlier iteration of App Genie. It identifies unparsed databases and attempts to parse them using a set of heuristic processes and rules. Information is split up between Fuzzy Events and Fuzzy Objects. Similar to App Genie, the results should be manually verified.

Enrichments

Cellebrite Physical Analyzer can utilize external data sources, provided by Cellebrite, to enrich the data extracted from the device. For example, a record of a Wi-Fi network BSSID that exists on the device can be supplemented with location data provided by Cellebrite of a previously known location of that particular BSSID, thereby indicating where the device was at that time.

Other examples include the location of Cell Towers or the translation of text between supported languages.

Another type of enrichment is the Media Classification engine, where AI is used to automatically categorize media based on the contents of the image or video.

Verification

Hashing

Hashing is a one-way cryptographic algorithm performed on data which results in a unique set of bytes usually represented by a hexadecimal string, whose length is determined by the hashing algorithm chosen. The data can be hashed any number of times and the resulting hash value should always remain constant. Any difference in the original data will result in a completely different hash value.

Once data extraction is completed, the extracted data is hashed using the SHA256 and/or MD5 hashing algorithm to help ensure data authenticity. This hash is included in the .UFD file and can be used for comparison when the file is opened in Physical Analyzer.

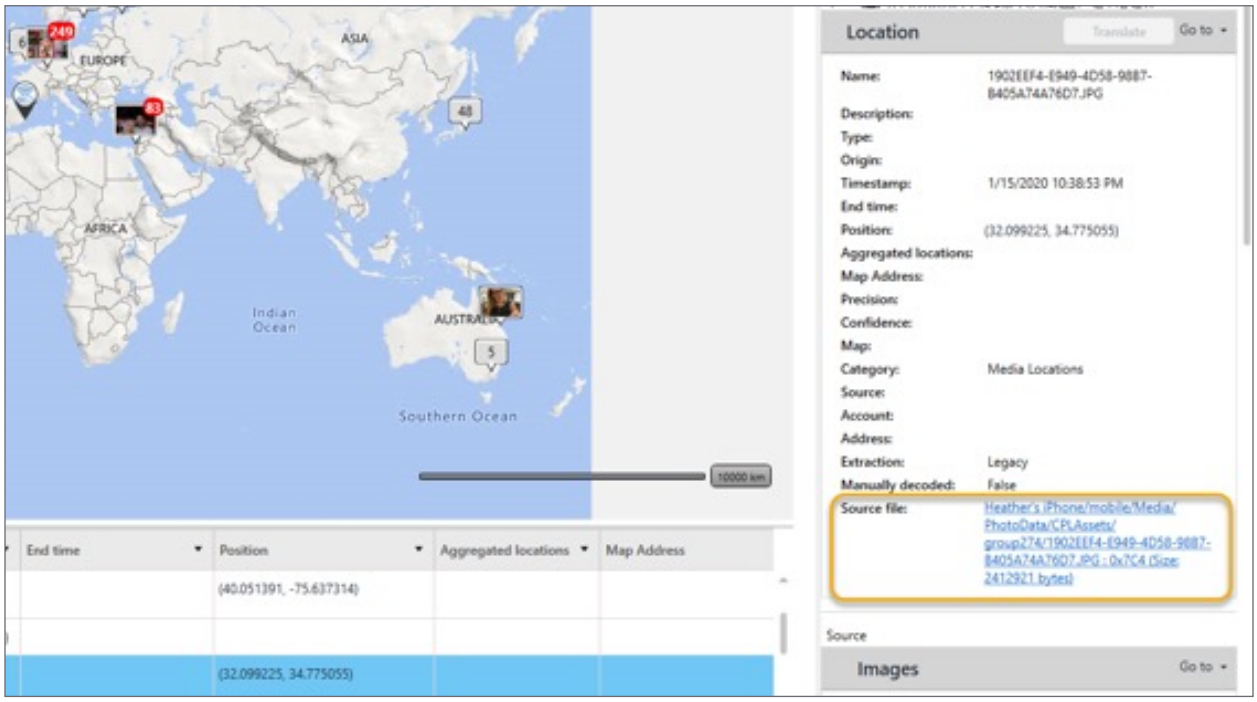
While not impossible for two different files to have the same hash value, the probability of this happening with the MD5 algorithm is one in 9 trillion. The probability of a SHA256 collision is approximately 4.3*1060.

Generating two files with the same hash is considered impossible with modern hash functions such as SHA256. Collision attacks using older functions such as MD5 can be generated by highly-technical advanced actors, however manual inspection of the collided file will reveal a highly irregular form that was generated to create the collision. SHA256 is recommended for forensic use.

Comparing Data

It is good practice to verify some of the parsed data to ensure the values have been decoded correctly and to be able to speak to that validation in court.

This can be done by either following the source of the artifact and manually checking it is correct, viewing the data live on the device (bearing in mind this may result in changes on the device should it be extracted again) or by using a secondary forensic tool. Heather Mahalik produced a series called the "Fundamentals Matter" series by Heather Mahalik includes topics on data validation.



Cellebrite Physical Analyzer provides a link to easily follow the source

Time Stamps & Time Zones

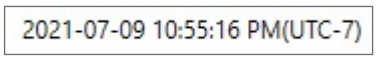
It is important to understand how times are stored on the device being examined. Many different types of timestamp epoch may be used such as;

- **UNIX** (Number of seconds since Jan 1st 1970 UTC)
- **UNIX Millisecond** (Number of milliseconds since Jan 1st 1970 UTC)
- **MAC Absolute** (Number of seconds since Jan 1st 2001 UTC)
- **MAC Nanoseconds** (Number of Nano-seconds since Jan 1st 2001 UTC)
- **Google Chrome** (Number of Microseconds since Jan 1st 1601 UTC).
- Plus many more

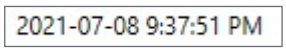
Almost all timestamps will be recorded in UTC and it is up to the tool to apply the appropriate time zone offset. These timestamps are relatively easy to handle as it doesn't matter where the device was at the time the record was made.

However, other timestamps may record as literal strings. These are typically in the local time zone of the device at the time the record was made. For example, the EXIF timestamp within a photograph. These timestamps can be harder to apply time zone offsets to unless you know the time zone that the photograph was taken.

Physical Analyzer can adjust most timestamp offsets to your local time for ease of use. In cases where an offset has been applied the time stamp will include the UTC designation at the end.



An example of an offset timestamp



An example of a Local timestamp

It is also important to validate the accuracy of the time on the device as most timestamps are obtained from the time set on the device, correct or not.

Trusting the data

Cellebrite tools are commercial, which means that the underlying code is proprietary and for that reason, its code is not open for review.

However, it is generally near impossible to falsify UFED extraction data. Furthermore, extractions are subjected to hash calculations at the time of extraction and again when parsed. This means it is simple to repeat and validate any claimed findings should something be called into question.

But as with any software, bugs can exist that may affect the representation of the data in Physical Analyzer. This is one of the reasons why it is important to keep your tools up to date and to validate your work either manually by checking the source of the evidence or by using secondary tools.

It is also common practice to have forensic reports reviewed by colleagues to minimize the risk of mistakes being made in the reporting process.

To read more about Validation of Forensic Extractions see:
https://www.sans.org/white-papers/six-steps-to-successful-mobile-validation

Peer Review

Peer Review of the tool

Numerous organizations conduct reviews of various forensic tools on a regular basis. One of these organizations is the National Institute of Standards and Technology (NIST).

Peer Review testing is designed to confirm a tools limitations and that it is decoding and parsing the data accurately. The results of NIST's peer testing can be found at https://www.nist.gov.

Putting Peer Review Findings in Context

It is important to note that Cellebrite tools are continuously updated to address bugs and to add new features, whereas peer reviews only focus on a single version. A bug found during the peer review will likely have been resolved by the time the review is published.

It is unfeasible that the peer reviewer takes into account all devices, settings and applications that Cellebrite tools are designed to support. Therefore, any peer review is just a small snapshot of the tool in a very specific circumstance.

Peer reviews should therefore only be used as a guide as to the general reliability of the tool in question. The examiner should always seek to validate their findings to ensure their own confidence in their evidence.

Expert Status

In order to give complicated digital evidence in court, examiners are often designated as an "Expert Witness".
The definition of an Expert Witness may vary between jurisdictions but ultimately, it is up to the examiner to prove an understanding of their field of expertise by demonstrating a history of experience, training and knowledge. It is impossible to know and account for every single artifact on a device, but it is important to demonstrate your ability to test, validate and investigate evidential artifacts. It is also important to keep an up-to-date curriculum vitae style document which lists all previous training courses and qualifications, relevant positions or previous expert qualifications.

Training

Cellebrite Training Center includes numerous on-demand or in-person training sessions to help you learn Cellebrite tools and prepare for being qualified as an Expert Witness. There are different training paths depending on the role you are in;

- Mobile Forensics
- Investigative
- Computer Forensics

Cellebrite TRAINING & ADVISORY

Visit www.cellebritelearningcenter.com for more information.

Example Questions

These are a few general questions similar to what you may be asked in court and should be prepared to answer.

- What is Cellebrite UFED?
- Is this tool commonly used by Law Enforcement?
- What training have you received in UFED / in Mobile Phones?
- What certifications do you have in relation to Digital Forensics?
- Is it common for examiners to use multiple tools depending on the make/model of device?
- Does UFED alter the data on the device?
- What type of information was extracted from the device?
- Did you validate the data reported by the tool?
- Have you ever seen data that has been modified by the UFED tool?
- If you examined the phone again now, would you get the same results?
- If you examined the extraction again now, would you get the same result?

Glossary of Common Terms

ADB	Android Debugging Bridge; A command line, client/server tool that allows developers to communicate with an Android device. ADB can be used to install and uninstall apps, run shell commands, backup and restore a device and so on. In a forensic context it can be important (for some makes and models) to enable physical and file system extractions.
Allocated Space	The area on device memory that stores data in an organised manner and contains the operating system and user data. Logical extractions obtain data from allocated space only.
MEID	Mobile Equipment Identifier; A 56 bits number unique to a CDMA device. This is often considered a serial number.
ESN	Electronic Serial Number; Found on CDMA devices and is 11 digits in length and contains both letters and numbers.
ICCID	Integrated Circuit Card Identifier (starts with 89); The unique serial number for a SIM card.
MDN	Mobile Directory Number; A 10 digit phone number.
IMEI	International Mobile Equipment Identifier; A 15 digit unique number to GSM devices.
IMSI	International Mobile Subscriber Identity; A unique number used to associate a user to a mobile network.
Jailbreaking / Rooting	A jailbroken iOS device or Rooted Android device, is one whose owner has taken steps to bypass its factory settings, including built in security and other restrictions. Jailbreaking an iOS device allows the user to install third-party apps from sources other than the Apple App Store, while rooting an Android device provides administrative "root" access to its operating system. UFED solutions do not rely on jailbreaking or permanent rooting to perform forensic extractions, although may utilize them, such as with the checkm8 solution.
MSISDN	Mobile Station International Subscriber Directory Number; The device phone number on a GSM phone.
Unallocated Space	The area on a device's memory outside of the defined file system that is available to write data to. Very often, deleted data can be found and carved from unallocated space. A physical extraction is required.

About Cellebrite

Cellebrite is the global leader in partnering with public and private organizations to transform how they manage Digital Intelligence in investigations to accelerate justice and ensure data privacy. We aid organizations in mastering the complexities of legally sanctioned digital investigations with an award-winning software suite and services to unify investigative workflows and manage Digital Intelligence. Cellebrite works with industry leaders to help them protect the public and safeguard assets with efficiency and transparency. Trusted by thousands of leading agencies and companies in more than 150 countries, Cellebrite is helping customers fulfill the joint mission of protecting and saving lives.

- To learn more visit us at www.cellebrite.com
- Contact Cellebrite globally at www.cellebrite.com/contact

Cellebrite