DFIR Review

Upgrade From NULL— Detecting iOS Wipe Artifacts

Heather Mahalik, Ian Whiffin

Published on: Jun 24, 2021

License: [Creative Commons Attribution 4.0 International License \(CC-BY 4.0\)](https://creativecommons.org/licenses/by/4.0/)

Synopsis

Forensics Question: How do you know when an iOS device was wiped?
OS Version: iOS version 13.7, iOS version 14.2
Tools: Cellebrite Physical Analyzer 7.40	

Upgrade from NULL: detecting iOS wipe artifacts

By Heather Mahalik & Ian Whiffin, Cellebrite

One of the most common questions we have been asked is “How do I know when an iOS device was wiped?” Wiping is a solid way of trampling potential evidence on a device. Over the years, we have seen some savvy moves to try to cover wiping a device and having the act be detected. These include:

1. Wiping the device far enough in advance and conducting as much activity as possible to create noise on the device and make it look “used.” *Like Ruth did on the Cellebrite CTF.*
2. Wiping the device and pushing old backup data to it to make it look “used”.

3. Wiping a device and pushing someone else's data to it to make it look "used."

In this blog, we want to help you quickly identify if an iOS device was wiped. Keep in mind, a wipe doesn't mean something nefarious occurred as there are legitimate reasons to wipe a device. We are simply helping you determine the "when" and we leave you with the fun part of determining the "why" behind the wipe.

Test device used in screenshots below

- Ruth Langmore's iPhone X from the Cellebrite CTF. Her device was wiped on **July 27, 2020 at 7:08 PM UTC**. During this time, the offset from UTC is GMT-4 to hit Eastern Daylight Time. So, in local time, the wipe occurred at 3:08 PM.

As always, we validated our findings across several devices to include an iPhoneSE running iOS 13.7 and an iPhone 6S running iOS 14.2.

The Results

Some examiner's rely on dates that are parsed by tools or that seem to make sense based upon how the artifact is named. We want to first identify artifacts that are not as reliable as the others we discuss in this blog. Up first, the **SetupLastExit** date found in **/root/private/var/mobile/Library/Preferences/com.apple.purplebuddy.plist**. This date is NOT the best indication of when a device was wiped. There are many factors that can impact this timestamp. For example, when you setup an iPhone, you can ignore Wallet and other settings. When you go back and add those items, this timestamp is changed. If you change your credit card for Apple Pay down the road, this timestamp will be updated. So really, it reflects the last time the user of the device changed something in the Settings screen.

For our test device, while the date is correct for the wipe time, you can see that **SetupLastExit** occurred a few hours after the wipe. This time reflects when we updated Ruth's Apple Pay.

If you recently wiped and setup a device, you see that red alert in the device to finish setting up the iPhone.

In summary, the **SetupLastExit** is tracking the phone setup and all changes made there. You may find some devices that are wiped and immediately setup, which is fine,

but you may also come across some that aren't fully setup for days or months. And for that possibility alone, we are ignoring this file.

There are several files that you will find that support the correct date/time when an iOS device was wiped. We are going to share a few of our favorite ones that we have relied upon over the years to aid in detecting the correct wipe date. The first is going to reside within the **com.apple.purplebuddy.plist**, which contains useful dates and is accessible from most iOS acquisition methods including iTunes backups all the way to Full Filesystem extractions.

Within the **com.apple.purplebuddy.plist**, we see a timestamp for **GuessedCountry** on 7/27/2020 at 7:13:13 PM which is around the time Heather was setting up the iOS device after wiping it.

You have to think that when the user presses the button to “Erase All Content and Settings” and proceeds that there may be a few minutes between that action and when

the phone actually boots up as a “freshly wiped” device. One of the first screens the user will see is “Select Your Country or Region” which is represented as **GuessedCountry** inside of this plist.

To rely on this time, please make sure the **SetupState** is **SetupUsingAssistant**. If the **SetupState** is **RestoredFromiCloudBackup**, the date may reflect an old wipe on

from the restored device. For iCloud restores, we recommend obtaining a Full File System extraction and examining the artifacts mentioned later in this blog. If that is not possible, you are going to have to examine the creation dates and usage of the device, which will be a manual process.

In summary, there are many dates accessible in **com.apple.purplebuddy.plist**, which is accessible even by just an iTunes backup. We recommend you examine the

GuessedCountry from this plist to get a good grip on the first time the phone was successfully booted after a wipe and when the Country was presented to the user upon that boot (mine suggested United States). Remember, if the device was restored from an iCloud backup, you will have to look beyond the **GuessedCountry** as it may reflect an older wipe date. If you want to know more on what the user did to set the device up after wiping (Restored from iCloud, Fresh setup, Restored from iTunes) refer to the DFIR Review blog published by Heather Mahalik:

<https://dfir.pubpub.org/pub/2q177smo/release/5>

The next file of interest can be found within Full File System extractions. The **containermanagerd** is a series of log files that are updated when, amongst other things, the device boots. They are located in the **/private/var/root/Library/Logs/MobileContainerManager** folder. The files are appended as .log files but include a secondary, numerical extension which is directly related to the age of the file. So, you may see containermanagerd.log.0, containermanagerd.log.1 and containermanagerd.log.2 for example. It is important to note, that the higher the number, the older the file. This means that containermanagerd.log.0 is always the most up to date file.

As we are interested in the date the device was wiped, which will be stored in the oldest file, we need to look at the file with the highest numeric value. The entry we are most concerned with is during the boot process where the device is checked to see if an update has just occurred:

This entry shows that upon booting, the OS build was found to be 17C54 which is the same as the last time it booted. Therefore, no upgrade has taken place.

This entry shows that the OS was being upgraded from build 17C54 to build 17D50.

But when the device is booted for the first time after an erase, the build information is missing, resulting in the entry:

Each of these records is prepended with a date and time that the record was made. The timestamps are in local time of the device at time the record was made. It is important to note that upon first boot the device defaults to US Pacific Time (UTC-8). This may be updated during the setup process.

This means that the first few records (including the entry we are interested in) are shown in UTC-8 regardless of how the device was setup prior to wipe. If we take

Ruth's device and look at the record described, we can see the timestamp is **12:11:38** on **07/27/2020**.

```

Learn more  Extraction Summary (1)  com.apple.purplebuddy.plist  containermanagerd.log.0
Text View  Hex View  File Info
Mon Jul 27 12:11:38 2020 [67] <notice> (0x16d76f000) containermanagerd_init: containermanagerd performing first boot initialization
Mon Jul 27 12:11:38 2020 [67] <notice> (0x16d76f000) -[MCMMigrationStatus _migrateFromManyMarkerFilesToOne]: Migrating from many marker files down to one
Mon Jul 27 12:11:38 2020 [67] <notice> (0x16d76f000) -[MCMMigrationStatus isBuildUpgrade]: Did not find last build info; we must be upgrading from pre-9.3.1 or this is an erase install.
Mon Jul 27 12:11:38 2020 [67] <notice> (0x16d76f000) -[MCMClientConnection _regenerateAllSystemContainerPaths]: Rolling system container directory UUIDs on disk
Mon Jul 27 12:11:38 2020 [67] <notice> (0x16d76f000) -[MCMMigrationStatus writeCurrentBuildInfoToDisk]: Saved last build version of 17F80
Mon Jul 27 12:11:38 2020 [67] <notice> (0x16d887000) containermanagerd_init_block_invoke: containermanagerd first boot cleanup complete
Mon Jul 27 12:11:40 2020 [67] <notice> (0x16d7fb000) -[MCMContainerMigrator performDataMigratorMigrationForClientConnection:withError:]: Performing Data Migration on 17F80
Mon Jul 27 12:11:40 2020 [67] <notice> (0x16d7fb000) -[MCMFileManager standardizeAllSystemContainerACLsAtURL:error:]: Potential containers requiring ACL migration: (

```

If we treat it as a Pacific Timestamp, we need to add 7 hours to bring it to UTC time (due to Daylight Savings) which gives us **7:11PM** which is a few minutes after the device was wiped. Again, it may take a few minutes for the device to finish wiping and reboot.

This seemed a little odd and prompted us to do further testing.

A Secondary Device - Validation:

This test device was wiped at **4:32PM UTC**. It was setup at **5PM UTC** and rebooted at **5:07PM UTC**. But the container information shows **8:33AM**. As we are now in Pacific Standard Time, we must add 8 hours to get to the UTC time of **4:33PM** which we know to be about correct.

```

Tue Nov 24 08:33:49 2020 [64] <notice> (0x16d4db000) -[MCMMigrationStatus _migrateFromManyMarkerFilesToOne]: Migrating from many marker files down to one
Tue Nov 24 08:33:49 2020 [64] <notice> (0x16d4db000) -[MCMMigrationStatus isBuildUpgrade]: Did not find last build info; we must be upgrading from pre-9.3.1 or this is an erase install.
Tue Nov 24 08:33:49 2020 [64] <notice> (0x16d4db000) -[MCMClientConnection _regenerateAllSystemContainerPaths]: Rolling system container directory UUIDs on disk

```

During this research, Ian identified **logd.0.log** which contains shutdown and timezone change information. This file can be found in **private/var/db/diagnostics/logd.0.log**. If you recall, the setup of this device was started at **5PM UTC**. If we look at the **logd.0.log** file, we can see the first line refers to a time zone change.

```

2020-11-24 10:01:49-0700 logd[25]: Time zone changed, updating file headers
2020-11-24 10:07:03-0700 logd[25]: No userlevel firehose clients left

```

This time zone change was a result of the setup process and shows that the timestamp is **10:01 UTC-7**, which is correct to both the time I ran the setup and my time zone location. The second entry is made when the device is shutting down and matches the time that I rebooted the device at **5:07PM UTC**.

Finally, to bring this back to where we started, we can see that the timestamp in **containermanagerd** is now displayed in local time to the device.

```
Tue Nov 24 10:07:40 2020 [72] <notice> (0x16d907000) containermanagerd_init: containermanagerd performing first boot initialization
Tue Nov 24 10:07:40 2020 [72] <notice> (0x16d907000) -[MCMMigrationStatus isBuildUpgrade]: Current build version (17H35 / (null)) equal to last version recorded (17H35 / (null)); skipping upgrade
```

It's worth noting that this file now appears to have a shelf life and is not guaranteed to be in every extraction you open. But interestingly, Ian previously found a record on his personal device that was approximately 3 months prior to the device being available for sale. It wasn't an epoch date and his only assumption was that it was a record of the device being booted up in factory.

Ruth's Device - Validation:

To ensure nothing was misinterpreted we validated Ruth's data in ArtEx, a tool developed by Ian. Below we can see that the **containermanagerd** is identical to what we examined in Physical Analyzer and shows the **07/27/2020 12:11:38** timestamp. (Heather lives in Eastern Time, so 3 hours must be added to match her local time). This is correct, just as we previously noted.

```
AE Viewer
containermanagerd.log.0
A
Mon Jul 27 12:11:38 2020 [67] <notice> (0x16d76f000) containermanagerd_init: containermanagerd performing first boot initialization
Mon Jul 27 12:11:38 2020 [67] <notice> (0x16d76f000) -[MCMMigrationStatus migrateFromManyMarkerFilesToOne]: Migrating from many marker files down to one
Mon Jul 27 12:11:38 2020 [67] <notice> (0x16d76f000) -[MCMMigrationStatus isBuildUpgrade]: Did not find last build info; we must be upgrading from pre-9.3.1 or this is an erase install.
Mon Jul 27 12:11:38 2020 [67] <notice> (0x16d76f000) -[MCMClientConnection regenerateAllSystemContainerPaths]: Rolling system container directory UUIDs on disk
Mon Jul 27 12:11:38 2020 [67] <notice> (0x16d76f000) -[MCMMigrationStatus writeCurrentBuildInfoToDisk]: Saved last build version of 17F80
Mon Jul 27 12:11:38 2020 [67] <notice> (0x16d887000) containermanagerd_init_block_invoke: containermanagerd first boot cleanup complete
Mon Jul 27 12:11:40 2020 [67] <notice> (0x16d7fb000) -[MCMContainerMigrator performDataMigratorMigrationForClientConnection:withError:]: Performing Data Migration on 17F80
Mon Jul 27 12:11:40 2020 [67] <notice> (0x16d7fb000) -[MCMFileManager standardizeAllSystemContainerACLsAtURL:error:]: Potential containers requiring ACL migration: {
    "33749E1C-7717-4BDE-A13D-42AE0CCE3D71" -- file:///private/var/containers/Data/System",
    "24B12A26-1F19-4A28-BE5F-62DB838AB687" -- file:///private/var/containers/Data/System",
    "5C87FAEF-1AFC-48AF-8257-A437440D4746" -- file:///private/var/containers/Data/System",
    "EFB12656-BA70-465E-A268-407010A85B2A" -- file:///private/var/containers/Data/System",
    "7E020931-F714-4CAB-A648-19735B0A31A1" -- file:///private/var/containers/Data/System"
}
```

When examining the Metadata for **containermanagerd.log.0** we see that the Creation time matches when the device was wiped and is shown in UTC.

Extraction Summary (1) ×	
containermanagerd.log.0 ×	
Text View Hex View File Info	
Find: <input type="text"/>	
General	
User ID	0
Group ID	0
File size	14414 Bytes
Chunks	1
Offsets	
Data offset	0x26F0D10EC
Date & Time	
Creation time	7/27/2020 7:11:38 PM(UTC+0)
Modify time	8/12/2020 8:27:31 PM(UTC+0)
Last access time	7/27/2020 7:11:38 PM(UTC+0)
Deleted time	
Change time	8/12/2020 8:27:31 PM(UTC+0)

So, it is easiest to look at the creation data for the **containermanagerd.log.0**, but it's important that you understand the contents and what they mean so you do not misjudge when the device was wiped. As previously stated, there are other file creation times that can be relied upon to identify when the iOS device was booted after a wipe. Below, we look at additional files from Ruth's iPhone.

private/var/mobile/Library/AddressBook/**AddressBook.sqlitedb**– creation time **7/27/2020 7:11 PM UTC**. The **modified date** has NOTHING to do with the wipe time and has no correlation.

Extraction Summary (1) ×	
AddressBook.sqlitedb ×	
Database View Hex View File Info	
Find: <input type="text"/>	
General	
User ID	501
Group ID	501
File size	303104 Bytes
Chunks	1
Row count	165
Offsets	
Data offset	0x1FB6A9323
Date & Time	
Creation time	7/27/2020 7:11:42 PM(UTC+0)
Modify time	8/12/2020 6:51:23 AM(UTC+0)
Last access time	7/27/2020 7:11:42 PM(UTC+0)
Deleted time	
Change time	8/12/2020 6:51:23 AM(UTC+0)

private/var/mobile/Library/CallHistoryDB/**CallHistory.storedat**a– creation time **7/27/2020 7:12 PM UTC**.

Extraction Summary (1) ×

CallHistory.storedata ×

Database View

Hex View

File Info

Find:

General

User ID

Group ID

File size

Chunks

Row count

Offsets

Data offset

Date & Time

Creation time

Modify time

Last access time

Deleted time

Change time

501

501

73728 Bytes

1

146

0x1F638A720

7/27/2020 7:12:09 PM(UTC+0)

7/27/2020 7:12:09 PM(UTC+0)

7/27/2020 7:12:09 PM(UTC+0)

7/27/2020 7:12:09 PM(UTC+0)

A common file that is used to identify a device wipe is the .obliterated file that can be found at /private/var/root.

This is a zero byte file created by the device upon booting after a wipe.

If we look at the .obliterated file on Ruth's device, we can see that it was created at 3:11:25PM (Eastern Daylight Time).

Name: .obliterated
Type: Uncategorized
Size (bytes): 0
Path: DarArchive/root/private/var/root/.obliterated
Created: 2020-07-27 3:11:25 PM(UTC-4)
Accessed: 2020-07-27 3:11:25 PM(UTC-4)
Modified: 2020-07-27 3:11:25 PM(UTC-4)
Changed: 2020-07-27 3:11:25 PM(UTC-4)
Deleted:
Extraction: File System
MD5:
Source file: [.obliterated](#)

By comparing that timestamp to the other times we have found, we can see that this file was created about 14 seconds before the time shown in containermanagerd.

The .obliterated requires a FFS and may not be found in every extraction. It is however, another great reference for finding when the device was wiped.

Bottom line, you can find solid evidence on when an iOS device was wiped. If you only have an iTunes or advanced logical extraction, simply compare the **GuessedCountry** from the **com.apple.purplebuddy.plist**. If you want more correlation to other artifacts compare this date to the creation times of **Addressbook.sqlitedb** and **Call_History.storedata creation** which should be around the time the device was wiped and then rebooted. Other files may make sense too, but these are the ones we rely upon for consistency.

If you have a Full Filesystem extraction, start with the **containermanagerd** creation time and then take a look at the contents and make sure you convert from Pacific time to the local time of the device. Then go to **logd.0.log** and determine the time that the device time zone was updated (unless of course you are in Pacific Time). From here,

you can go to the **com.apple.purplebuddy.plist** or you can simply sit back and feel like you pinned down that pesky assumption you have been chasing. You now know when that iOS device was wiped.

DFIR Review

One of the reviewers indicates that there is no paper as complete and detailed as this one on this topic. The reviewer was able to reproduce the authors' results. Based on their testing, the `GuessedCountry` from `com.apple.purplebuddy.plist` may indicate the moment when the "Select your Country of Region" screen was displayed rather than the time the phone was booted after being wiped.

Another reviewer was able to verify the authors' results with an iPhone 6S running iOS version 13.3 using Cellebrite Physical Analyzer version 7.41.0.8.

One of the reviewers felt that while the paper presented data to support the final conclusion, there should have been additional data sets to recreate the results. The reviewer believed that a summary section or table at the end of the paper discussing the indicators tested and their validity level would have improved the article.

Future Work

No future work was suggested by the reviewers.

Reviewers

Eric Eppley (Methodology Review)

Nickolas Ligman (Methodology Review, Validated Review using Reviewer Generated Datasets)

Johann Polewczyk (Methodology Review, Validated Review using Reviewer Generated Datasets)