Adam Harrison
Principal Consultant - Verizon VTRAC
Adam Harrison (@harrisonamj)

• Who I am:
  • Principal Consultant (Forensic Investigator)
  • Verizon Threat Research Advisory Center (VTRAC | Investigative Response)
  • GREM, GASF, GNFA, GCFA, GCFE, GWAPT, GCIH, GCIA, GSEC (Other Certification bodies are available)
  • blog.1234n6.com

• Why exFAT and Why me:
  • David Cowen (@hecfblog) made me do it
  • 3 months obsessing over exFAT
  • The small things matter
Edge Games v Future Publishing
Edge Games v Future Publishing
Agenda

• exFAT Primer (Timestamps)
• Differences in handling of timestamps
• Tool interpretation of timestamps
• Differences in volumes
• Questions
But Adam, isn’t exFAT just used for moving files from your Mac to your real computer?
exFAT Primer

• Windows Embedded CE 6 (November 2006)
• Windows XP (SP2), Windows Vista (SP1) (Feb/March 2008)
• Linux (FUSE)
• Mac OS X (10.6.5+)
• Android OS & 3rd Party
• SD (SDXC)
• Other
exFAT Primer

• Comprises:
  • Boot Sector / Backup Boot Sector
  • $Bitmap - Tracks of the allocation status of clusters
  • $UpCase - Table of Unicode character mappings.
  • FAT - Linked list for tracking fragmentation
  • File Metadata stored in Directory Entries

• Boot Sector Points to Root Directory Entry (0x60/96 4bytes)
exFAT Primer

- Metadata is stored with Directory Entries (32 byte)
- First Byte == Type Code
- Directory Entry Record, Stream Extension and Filename Extension

<table>
<thead>
<tr>
<th>Type Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x85</td>
<td>Directory Entry Record</td>
</tr>
<tr>
<td>0x83</td>
<td>Volume Name Record, Master Entry (Named Volume)</td>
</tr>
<tr>
<td>0x03</td>
<td>Volume Name Record, Master Entry (Unnamed Volume)</td>
</tr>
<tr>
<td>0x82</td>
<td>Up-Case Table Logical Location and Size</td>
</tr>
<tr>
<td>0x81</td>
<td>Bitmap Logical Location and Size</td>
</tr>
<tr>
<td>0xC0</td>
<td>Stream Extension</td>
</tr>
<tr>
<td>0xC1</td>
<td>Filename Extension</td>
</tr>
<tr>
<td>0x05</td>
<td>Unallocated File Name Record</td>
</tr>
<tr>
<td>0x40</td>
<td>Unallocated Stream Extension</td>
</tr>
<tr>
<td>0x41</td>
<td>Unallocated Filename Extension</td>
</tr>
</tbody>
</table>
# exFAT Primer

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type Code</td>
<td></td>
</tr>
<tr>
<td>Number of Secondary Entries</td>
<td></td>
</tr>
<tr>
<td>Checksum</td>
<td></td>
</tr>
<tr>
<td>Flags</td>
<td></td>
</tr>
<tr>
<td>Reserved</td>
<td></td>
</tr>
<tr>
<td>Created</td>
<td></td>
</tr>
<tr>
<td>Last Modified</td>
<td></td>
</tr>
<tr>
<td>Last Accessed</td>
<td></td>
</tr>
<tr>
<td>Creation Centisecond Offset</td>
<td></td>
</tr>
<tr>
<td>Last Modified Centisecond Offset</td>
<td></td>
</tr>
<tr>
<td>CR Time Zone Code</td>
<td></td>
</tr>
<tr>
<td>Mod Time Zone Code</td>
<td></td>
</tr>
<tr>
<td>Acc Time Zone Code</td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td></td>
</tr>
<tr>
<td>Filename Hash</td>
<td></td>
</tr>
<tr>
<td>Valid Data Length</td>
<td></td>
</tr>
<tr>
<td>First Cluster Address</td>
<td></td>
</tr>
<tr>
<td>Data Length</td>
<td></td>
</tr>
<tr>
<td>Filename</td>
<td></td>
</tr>
<tr>
<td>Slack</td>
<td></td>
</tr>
</tbody>
</table>

![Hexadecimal Values](image)
exFAT Primer

- 32-bit MSDOS timestamps (2 second granularity).
- Value between 0 and 199 to denote the number of centiseconds which should be added to the recorded MSDOS timestamp. Note, no such field exists for the ‘Last Accessed’ value.
- The Time Zone Code (one byte value)
  - Most significant bit denotes whether used.
  - The required offset is then stored as a 7-bit signed integer.
  - The integer itself represents 15-minute increments.
exFAT Primer

0x62 == 98

1000 0100

-64 32 16 8 4 2 1

0.25 x 4

+1

2018-06-26T13:44:48.980000 UTC
exFAT Primer

• Robert Shullich’s GCFA Gold Paper ‘Reverse Engineering the Microsoft Extended FAT File System (exFAT)’ (2009)

• Jeff Hamm’s paper ‘Extended FAT File System’ (2009)

• Jeff Hamm’s post on the SANS Digital Forensics and Incident Response Blog entitled ‘exFAT File System Time Zone Concerns’ (2010)

• Jeff and Robert presentation/ talk entitled ‘exFAT (Extended FAT) File System - Revealed and Dissected’ (2010)

• 1234n6 ‘exFAT Timestamps: exFAT Primer and My Methodology’ (2018)
Differences in handling of timestamps

Well that sounds simple enough. A filesystem with wide support across multiple Operating Systems!
Differences in handling of timestamps

Windows XP (SP2/3), Vista (SP2), 7 and 8

• Creation Time – Recorded with centisecond granularity.

• Last Modified Time – Recorded with centisecond granularity at creation. Recorded with 2 second granularity when copied or moved (Offset = 0).

• Accessed Time – Recorded with 2 second granularity due to exFAT limitation.

• Offset – All timestamps are recorded as a direct representation of the local time of the system, the offset (as configured on the system) is recorded in the associated offset fields. Excluding Vista SP1.
Differences in handling of timestamps

Windows 10

• Creation Time – Recorded with centisecond granularity.

• Last Modified Time – Recorded with 2 second granularity (Offset Unused = 0)

• Accessed Time – Recorded with 2 second granularity due to exFAT limitation.

• Offset – All timestamps are recorded as a direct representation of the local time of the system, the offset (as configured on the system) is recorded in the associated offset fields.
Differences in handling of timestamps

Ubuntu 18.04

• Creation Time – Recorded with nearest second granularity (0 or 100).

• Last Modified Time – Recorded with nearest second granularity.

• Accessed Time – Recorded with 2 second granularity due to exFAT limitation.

• Offset – All timestamps are recorded in local device time, the Timezone fields are consistently set to 00, indicating that they are not in use.
Differences in handling of timestamps

Ubuntu 16.04

• Creation Time – Recorded with nearest second granularity (0 or 100).

• Last Modified Time – Recorded with nearest second granularity.

• Accessed Time – Recorded with 2 second granularity due to exFAT limitation.

• Offset – All timestamps are recorded in local device time **without DST support**, the Timezone fields are consistently set to 00, indicating that they are not in use.
Differences in handling of timestamps

MacOS 10.13.6

• Creation Time – Recorded with centisecond granularity.

• Last Modified Time – Recorded with centisecond granularity

• Accessed Time – Recorded with 2 second granularity due to exFAT limitation.

• Offset – All timestamps are recorded in UTC - (current UTC offset) number of hours. File created at 1200hrs UTC on system configured with UTC+4 (displaying 1600hrs) will have timestamp of 0800hrs. -(UTC offset) recorded in Offset field.
Differences in handling of timestamps

Android (Samsung)

• Creation Time – Recorded with 2 second granularity (Offset Unused - 0)

• Last Modified Time – Recorded with 2 second granularity

• Accessed Time – Recorded with 2 second granularity due to exFAT limitation.

• Offset – All timestamps are recorded in local device time, the Timezone fields are consistently set to 00, indicating that they are not in use.
Differences in handling of timestamps

Android (Paragon Driver)

• Creation Time – Recorded with nearest second granularity (0 or 100).

• Last Modified Time – Recorded with nearest second granularity.

• Accessed Time – Recorded with 2 second granularity due to exFAT limitation.

• Offset – All timestamps are recorded in UTC, the Timezone offset field is consistently set to 80 which indicates an offset of 0.
## Tool interpretation of filesystem metadata

<table>
<thead>
<tr>
<th>OS</th>
<th>Creation</th>
<th>Last Modified</th>
<th>Accessed</th>
<th>Recorded In</th>
<th>TZ Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows XP (SP2/3), Vista (SP2), 7 and 8</td>
<td>Centisecond</td>
<td>Centisecond*</td>
<td>2 Second</td>
<td>Local</td>
<td>Yes</td>
</tr>
<tr>
<td>Vista SP1</td>
<td>Centisecond</td>
<td>Centisecond*</td>
<td>2 Second</td>
<td>Local</td>
<td>0x00</td>
</tr>
<tr>
<td>Windows 10</td>
<td>Centisecond</td>
<td>0</td>
<td>2 Second</td>
<td>Local</td>
<td>Yes</td>
</tr>
<tr>
<td>Ubuntu 18.04</td>
<td>0 or 100</td>
<td>0 or 100</td>
<td>2 Second</td>
<td>Local</td>
<td>0x00</td>
</tr>
<tr>
<td>Ubuntu 16.04</td>
<td>0 or 100</td>
<td>0 or 100</td>
<td>2 Second</td>
<td>Local (-DST)</td>
<td>Yes</td>
</tr>
<tr>
<td>MacOS 10.13.6</td>
<td>Centisecond</td>
<td>Centisecond</td>
<td>2 Second</td>
<td>WTF</td>
<td>WTF</td>
</tr>
<tr>
<td>Android (Samsung)</td>
<td>0</td>
<td>0</td>
<td>2 Second</td>
<td>Local</td>
<td>0x00</td>
</tr>
<tr>
<td>Android (Paragon)</td>
<td>0 or 100</td>
<td>0 or 100</td>
<td>2 Second</td>
<td>UTC</td>
<td>0x80</td>
</tr>
</tbody>
</table>

* - Recorded with centisecond granularity at creation. Recorded with 2 second granularity when copied or moved (Correction = 00).

OSX - All timestamps are recorded in UTC - (current UTC offset) number of hours. File created at 1200hrs UTC on system configured with UTC+4 (displaying 1600hrs) will have timestamp of 0800hrs. -(UTC offset) recorded in Offset field.
Surely we can just rely upon our forensics tools to accurately display this information?
Tool interpretation of filesystem metadata

Autopsy

• User specifies Timezone of evidence on import
• TSK does not process the timezone field
• Reads timestamp as recorded and marks it with the user defined timezone.

“TSK does not process the timezone field, this means that timestamps from Mac images will be off by double the original time zone value”

https://github.com/sleuthkit/sleuthkit/wiki/ExFAT-Implementation-Notes
Tool interpretation of filesystem metadata

Axiom & EnCase

• Defaults to display time in UTC
• Interprets Timestamps as Local Time
• Applies centisecond correction
• Applies offset where available
Tool interpretation of filesystem metadata

FTK 6.1
• Interprets Timestamps as UTC
• Applies centisecond correction
• Ignores TZ offset

FTK 6.2+ (Apr 13, 2017)
• Interprets Timestamps as Local
• Applies centisecond correction
• Applies TZ offset where available
• Lists timestamps as N/A for 00
Tool interpretation of filesystem metadata

FTK Imager (4.1.1.1)
• Interprets Timestamps as Local Time
• Applies centisecond correction
• Applies TZ offset where available
• Lists timestamps as N/A when TZ offset set to 00 (Ubuntu or Android)
• FTK Imager 3.1.1.8 (lite) still displays inaccurate timestamps when TZ offset set to 00.
Tool interpretation of filesystem metadata

X-Ways

• Defaults to display time in UTC
• Interprets Timestamps as Local Time
• Applies centisecond correction
• Applies TZ offset where available
• Marks timestamps with ‘LT’ when offset set to 00.
Tool interpretation of filesystem metadata

X-Ways / WinHex (Scott Pancoast Template)
Tool interpretation of filesystem metadata

Ubuntu 16.04 (SIFT/ FUSE 1.2.3)
• Defaults to display time in UTC
• Interprets Timestamps as UTC
• Applies centisecond correction (rounded to nearest second)
• Ignores TZ offset where available
## Tool interpretation of filesystem metadata

<table>
<thead>
<tr>
<th>Tool</th>
<th>Interprets timestamp</th>
<th>Applies centisecond offset</th>
<th>Applies Timezone Offset</th>
<th>Differentiates between Offset applied or not</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autopsy (v4.7.0)</td>
<td>Yes (Whatever you tell it)</td>
<td>Yes</td>
<td>No</td>
<td>N/A</td>
</tr>
<tr>
<td>Axiom (v2.4.0)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>EnCase (v8.07)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>FTK (6.1)</td>
<td>Yes (UTC)</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>FTK (6.2+)</td>
<td>Yes (Local)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes (Doesn’t display timestamps for 00)</td>
</tr>
<tr>
<td>FTK Imager (4.1.1.1)</td>
<td>Yes (Local)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes (Doesn’t display timestamps for 00)</td>
</tr>
<tr>
<td>X-Ways (19.6)</td>
<td>Yes (Local)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes (LT when 00)</td>
</tr>
<tr>
<td>Ubuntu 16.04</td>
<td>Yes (UTC)</td>
<td>Nearest Sec</td>
<td>No</td>
<td>N/A</td>
</tr>
</tbody>
</table>
Differences in volume creation

So do the differences stop there?
Differences in volume creation

Remove disks or other media. Disk error. Press any key to restart.

Remove disks or other media. Disk error. Press any key to restart.
Differences in volume creation

macOS

Ubuntu
Conclusion

• Verify how tools are deriving displayed metadata (Dual tool)

• exFAT Timestamp idiosyncrasies can be used to draw conclusion as to which OS was used to place files

• exFAT volume Boot sector can be used to draw conclusion as to which OS was used to create a volume

• Anomalous timestamps may indicate evidence tampering
Who’s got questions?
You’ve got questions!
Maryam Nawaz

• Pakistani politician
• Daughter of three-time Prime Minister of Pakistan
• Named in Panama Papers
(b) I have identified the type font used to produce both certified Declarations as “Calibri”. However, Calibri was not commercially available before 31st January 2007 and as such, neither of the originals of the certified Declarations is correctly dated and happy to have been created at some later point in time.