ExifTool basics for DFIR

Taz Wake | Halkyn Consulting Ltd | Originally published 2024-01-03

Introduction to EXIF Data

Exchangeable Image File Format (EXIF) data is a standard that specifies the formats for images, sound, and ancillary tags used by digital cameras and other systems. This metadata is mainly associated with images and provides critical information like camera settings, timestamps, and geolocation information. Traditionally it is used to help photographers catalogue images, ensure colour balance etc.

Like lots of modern technology, this has evolved beyond its original scope and, in DFIR, we use it to include a range of metadata elements from Windows PE (Portable Executable) and Linux ELF (Executable and Linkable Format) files. This expanded functionality opens new avenues in cybersecurity and digital analysis, offering deeper insights into file structures and origins. In this article, we will give an overview of using Exiftool for DFIR purposes.

Understanding ExifTool

ExifTool is a powerful, versatile tool for manipulating metadata within various file types, including image, audio, and video files. Its capabilities extend to both reading and editing metadata, making it an indispensable tool for digital professionals.

Windows PE Header: An Overview

The Windows Portable Executable (PE) header is a fundamental component of Windows, originally specified in the 1980s. The term is used to describe the structure of Windows executables and Dynamic Link Libraries (DLLs), as well as some other file types (SYS, MUI, FON etc). The PE header includes a range of detailed information, such as the machine type, the number of sections, the size of the executable, and the entry point for the code. It is divided into multiple sections, a Microsoft MS-DOS stub, the PE signature, the COFF file header, and an optional header. Each serves distinct purposes in the executable's lifecycle. The COFF Header provides basic information about the file, while the Optional Header delves into specifics like memory allocation and entry points. This structure allows the operating system to correctly load and manage the executable, playing a key role in the functionality and security of Windows applications. Understanding the nuances of the Windows PE header is vital for software developers, security analysts, and digital forensic experts, as it provides insights into how applications interact with the Windows operating system at a fundamental level.

Insights into Windows PE Metadata

Analysing the metadata in the PE header is often a goldmine of DFIR information. Even though it is possible to modify pretty much every field, there are some which are frequently overlooked. As a result, you can use this data to identify things like timestamp manipulation, an example here is when the compilation timestamp is after the file creation timestamp, find unusual language settings or even simply hunt for unusual compilation packages.

Linux ELF Header

The Executable and Linkable Format (ELF) is Linux's counterpart to Windows PE. The ELF header is essential for understanding the architecture, type, and memory layout of Linux executable files. You can read more about the ELF format in a previous article: https://www.halkynconsulting.co.uk/security-resources/downloads/linux-incident-response-elf-files.pdf

Linux ELF Metadata Unveiled

Metadata in Linux ELF files provides valuable information about the binary, such as its version, dependencies, and compilation details, crucial for compatibility and debugging purposes. However, it should be noted that ELF headers contain less DFIR-useful information than PE file headers.

Utilising exiftool on Windows PE and Linux ELF Images

exiftool can be effectively used to extract and analyze metadata from both Windows PE and Linux ELF images. This functionality is crucial in digital forensics and cybersecurity, aiding in the examination of executables for investigative purposes.

PE Header Metadata

```
ExifTool Version Number         : 12.40
File Name                       : ExplorerSuite.exe
Directory                       : .
File Size                       : 3.4 MiB
File Modification Date/Time     : 2024:01:03 22:54:42+00:00
File Access Date/Time           : 2024:01:03 22:54:42+00:00
File Inode Change Date/Time     : 2024:01:03 22:54:42+00:00
File Permissions                : -rwxr-xr-x
File Type                       : Win32 EXE
File Type Extension             : exe
MIME Type                       : application/octet-stream
Machine Type                    : Intel 386 or later, and compatibles
Time Stamp                      : 2012:07:09 14:41:29+01:00
Image File Characteristics      : No relocs, Executable, No line numbers, No symbols, Bytes reversed lo, 32-bit, Bytes reversed hi
PE Type                         : PE32
Linker Version                  : 2.25
Code Size                       : 86016
Initialized Data Size           : 53760
Uninitialized Data Size         : 0
Entry Point                     : 0x16478
OS Version                      : 5.0
Image Version                   : 6.0
Subsystem Version               : 5.0
Subsystem                       : Windows GUI
File Version Number             : 0.0.0.0
Product Version Number          : 0.0.0.0
File Flags Mask                 : 0x003f
File Flags                      : (none)
File OS                         : Win32
Object File Type                : Executable application
File Subtype                    : 0
Language Code                   : Neutral
Character Set                   : Unicode
Comments                        : This installation was built with Inno Setup.
Company Name                    :
File Description                : Explorer Suite Setup
File Version                    :
Legal Copyright                 :
Product Name                    : Explorer Suite
Product Version                 :
```

Example PE header analyzed with ExifTool running under Ubuntu Linux

```
ExifTool Version Number         : 12.57
File Name                       : ExplorerSuite.exe
Directory                       : .
File Size                       : 3.6 MB
File Modification Date/Time     : 2023:08:14 22:42:44+01:00
File Access Date/Time           : 2024:01:03 23:14:10+00:00
File Creation Date/Time         : 2024:01:03 23:14:05+00:00
File Permissions                : -rw-rw-rw-
File Type                       : Win32 EXE
File Type Extension             : exe
MIME Type                       : application/octet-stream
Machine Type                    : Intel 386 or later, and compatibles
Time Stamp                      : 2012:07:09 14:41:29+01:00
Image File Characteristics      : No relocs, Executable, No line numbers, No symbols, Bytes reversed lo, 32-bit, Bytes reversed hi
PE Type                         : PE32
Linker Version                  : 2.25
Code Size                       : 86016
Initialized Data Size           : 53760
Uninitialized Data Size         : 0
Entry Point                     : 0x16478
OS Version                      : 5.0
Image Version                   : 6.0
Subsystem Version               : 5.0
Subsystem                       : Windows GUI
File Version Number             : 0.0.0.0
Product Version Number          : 0.0.0.0
File Flags Mask                 : 0x003f
File Flags                      : (none)
File OS                         : Win32
Object File Type                : Executable application
File Subtype                    : 0
Language Code                   : Neutral
Character Set                   : Unicode
Comments                        : This installation was built with Inno Setup.
Company Name                    :
File Description                : Explorer Suite Setup
File Version                    :
Legal Copyright                 :
Product Name                    : Explorer Suite
Product Version                 :
```

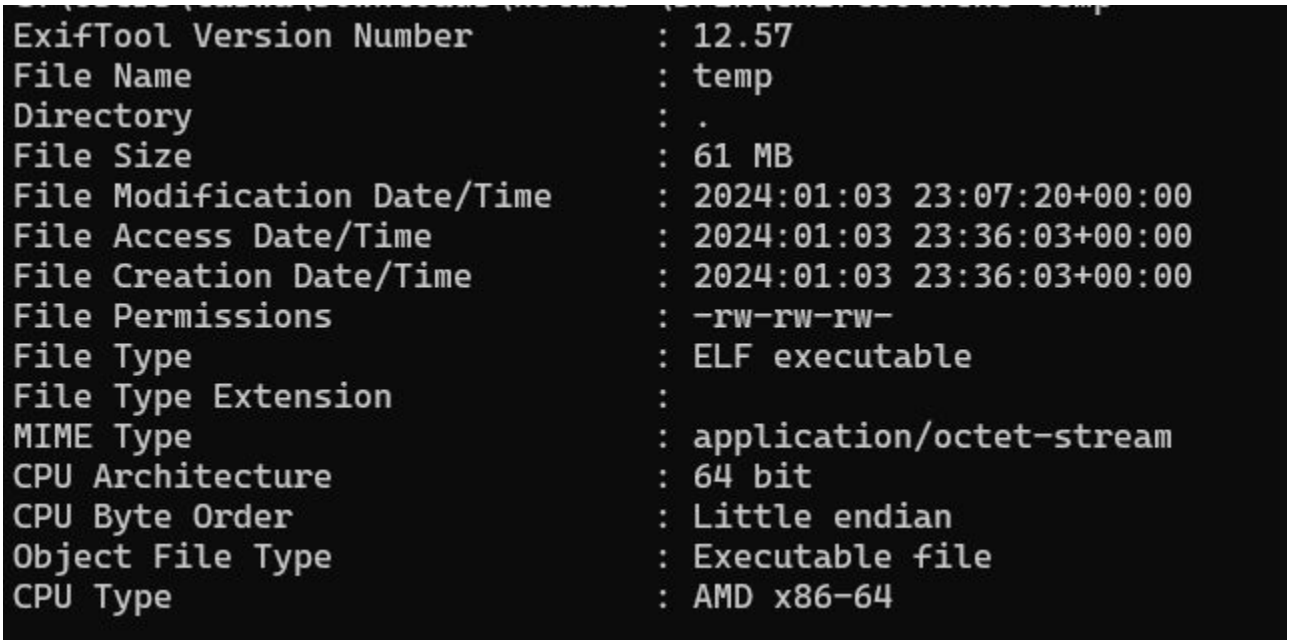
Example PE header analyzed with ExifTool running under Windows

The examples above use ExifTool to analyse the PE header for a file called ExplorerSuite.exe. The exact fields that would be useful will depend entirely on the investigation, but some key points include:

- The File MACB timestamps are extracted from the underlying filesystem. The content here will depend on the filesystem you are using to investigate the file. In the first example above, the file was copied to a Linux partition on 3 January and analysed using the Ubuntu version of the tool. The second example is on the same file, moved to a Windows partition and then analysed with a Windows version of the tool. Note that each tool shows different timestamps - the Linux tool doesn't show the creation time and the Windows tool doesn't show the metadata (inode) change time. It also shows the difference in how NTFS and XFS manage timestamps when a file is copied. **This is a very good example of why it is important to validate your tools**.
- The **timestamp** entry is set when the file is compiled. You might be able to determine malicious timestamp manipulation by comparing this to the filesystem timestamps, compilation should be more or less the same time the file is created, so if you have MACB timestamps which pre-date the compilation timestamp this is unusual. In Windows, this should also match the timestamp in the Shimcache.
- At the end of the output is a collection of fields that can hold useful hints. For example, the **Character Set** can indicate if a non-standard alphabet has been used. The **File Description** and **File Version** are often completed by legitimate applications but less often completed by malicious ones. You can also look for typos or inconsistencies in the **Product Name**, **Company Name**, etc., fields.
- Lastly, look for things that stand out within your organisation. For example, the string "This installation was built with Inno Setup" or references to Delphi, might be rare for legitimate applications in your environment.

```
ExifTool Version Number         : 12.40
File Name                       : temp
Directory                       : .
File Size                       : 58 MiB
File Modification Date/Time     : 2024:01:03 23:07:20+00:00
File Access Date/Time           : 2024:01:03 23:07:25+00:00
File Inode Change Date/Time     : 2024:01:03 23:07:20+00:00
File Permissions                : -rwxr-xr-x
File Type                       : ELF executable
File Type Extension             :
MIME Type                       : application/octet-stream
CPU Architecture                : 64 bit
CPU Byte Order                  : Little endian
Object File Type                : Executable file
CPU Type                        : AMD x86-64
```

Example ELF header analyzed with ExifTool running under Ubuntu Linux



Example ELF header analyzed with ExifTool running under Windows

As the examples above show, the data returned by an ELF file can be significantly less useful during an incident investigation. However, it is important to note this can vary depending on the binary and the nature of your incident. It is still useful, and important, to check files during an investigation.

About Halkyn Consulting

Halkyn Consulting Ltd is a UK-based security and risk management consultancy providing 24/7 incident response, digital forensics, threat hunting, virtual security leadership (vCISO) and security compliance support to organisations worldwide. We deliver clear, practical, vendor-independent advice based on real-world threats — no hype, no upsell.

More free security resources are available at www.halkynconsulting.co.uk/security-resources.

Web: www.halkynconsulting.co.uk | **Email:** info@halkynconsulting.co.uk | **Tel:** +44 1244 940858