# Windows Prefetch for Incident Responders

Taz Wake | Halkyn Consulting Ltd | Originally published 2023-10-30

Prefetch, an essential Windows feature, optimises the speed of frequently used applications by predicting and loading necessary data into memory. When a program runs, Windows records its file layout and loading behaviour in a prefetch file. This file's unique name is generated from a hash based on the program's path, modification timestamp, and size. If the program is moved, a new prefetch file is created upon execution, ensuring accurate tracking.

The main goal of Prefetch is to boost system performance by intelligently caching and loading application data before it's required. By analysing usage patterns and historical data, Windows predicts which applications a user is likely to run, making sure the needed data is in memory for swift access, and enhancing system responsiveness significantly.

Upon an application's first run, Windows generates a prefetch file in the C:\Windows\Prefetch directory. Subsequent runs, of the same executable, in the same folder, update this file. It's important to note that only specific applications, such as those with a .exe extension and Win32-based, qualify for prefetching.

In terms of the structure of a prefetch file, these files are typically named in the format of "AppName.exe-XXXXXXXX.pf".

| Name | Date created | Date modified | Size |
|---|---|---|---|
| LOCKAPP.EXE-05ED7BAD.pf | 21/09/2023 07:31 | 21/09/2023 07:31 | 22 KB |
| LOGONUI.EXE-1BEE4A84.pf | 21/01/2023 11:17 | 21/09/2023 07:31 | 26 KB |
| MICROSOFT.AAD.BROKERPLUGIN.EX-9BB341AA.pf | 13/09/2023 13:49 | 30/10/2023 12:24 | 49 KB |
| MICROSOFTEDGE.EXE-10B8FFA6.pf | 28/06/2021 06:00 | 19/12/2022 00:16 | 35 KB |
| MICROSOFTEDGE_X64_109.0.1518.-30C8AB70.pf | 28/01/2023 03:37 | 28/01/2023 03:37 | 7 KB |
| MICROSOFTEDGECP.EXE-21800523.pf | 17/02/2022 15:43 | 12/12/2022 00:49 | 21 KB |
| MICROSOFTEDGEUPDATE.EXE-0E099B6C.pf | 06/06/2021 19:21 | 30/10/2023 12:55 | 7 KB |
| MMC.EXE-0C6D06A3.pf | 28/06/2021 06:38 | 28/06/2021 07:11 | 33 KB |
| MMC.EXE-9A602585.pf | 29/04/2021 18:30 | 29/04/2021 18:30 | 15 KB |
| MMC.EXE-16E5E545.pf | 25/09/2023 18:56 | 25/09/2023 19:08 | 37 KB |
| MMC.EXE-0872F6EA.pf | 12/12/2022 00:17 | 12/12/2022 00:17 | 33 KB |
| MMC.EXE-44881441.pf | 12/12/2022 00:24 | 19/12/2022 00:14 | 37 KB |

Example of the contents of C:\Windows\Prefetch

The "AppName.exe" represents the name of the executable, and the "XXXXXXXX" represents a hash value derived from the executable's path, modification timestamp, and size. For a few executables (such as svchost.exe) the hash is also linked to command line arguments. As for the internal structure, prefetch files are stored in an uncompressed format. This design choice aims to maximise the speed of access, allowing the Windows operating system to quickly read and utilise the preloaded data without the overhead of decompression.

Since at least Windows 8, Prefetch files have been compressed using the Xpress+Huffman (XPRESS10) compression algorithm. This algorithm employs a combination of LZ77 (a lossless data compression algorithm) and Huffman coding techniques to compress the data efficiently.

SDELETE64.EXE-C8F3326B.pf

| Offset(h) | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | Decoded text |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 00000000 | 4D | 41 | 4D | 04 | 44 | 78 | 00 | 00 | 95 | B8 | A6 | AA | A9 | C8 | AA | BA | MAM.Dx..•¸¦ª©Èª° |
| 00000010 | A9 | B7 | AB | AA | A9 | C7 | CA | 9A | A9 | B7 | AA | 9A | AA | B8 | AA | A9 | ©·«ª©ÇÊš©·ªšª¸ª© |
| 00000020 | 99 | 97 | 99 | 99 | 99 | C8 | AA | 9A | 8A | 97 | 99 | 99 | 89 | A7 | 98 | 89 | ™—™™™Èªš Š—™™‰§˜‰ |
| 00000030 | A8 | 87 | 89 | 99 | 99 | B7 | A9 | 9A | AA | A8 | AA | A9 | A9 | A8 | 9A | AB | ¨‡‰™™·©šª¨ª©©¨š« |
| 00000040 | 99 | B8 | AA | AA | A9 | A8 | AA | AB | AA | B7 | AA | BA | AA | C8 | AA | AA | ™¸ªª©¨ª«ª·ºªÈªª |
| 00000050 | A9 | B8 | AB | AC | BA | A7 | AA | AB | A9 | B7 | AB | AB | B9 | B7 | A0 | BB | ©¸«¬°§ª«©·««¹· » |
| 00000060 | BA | B8 | AA | BA | B9 | B8 | 9B | AC | BA | B8 | AB | AB | BA | A8 | AC | BB | º¸ª º¹¸›¬°¸««°¨¬» |
| 00000070 | AA | B8 | AC | 99 | BA | B8 | BB | 9A | CB | A8 | AC | 9A | CA | B8 | AB | 9B | ª¸¬™º¸»šË¨¬šÊ¸«› |
| 00000080 | AA | A8 | BA | 9A | AA | 08 | B9 | 9A | 0C | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ª¨ºšª.¹š........ |
| 00000090 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | C0 | 79 | 0C | 0B | 00 | 00 | 00 | 00 | 0C | .......Ày....... |
| 000000A0 | 88 | A7 | C2 | 00 | 00 | 00 | 00 | C0 | 88 | AA | C6 | 00 | 0C | 00 | 00 | BC | ˆ§Â....Àˆª Æ....¼ |
| 000000B0 | 86 | AA | B7 | 00 | CA | 00 | BB | A0 | 98 | AA | A8 | C0 | BA | 0C | AC | A0 | †ª·.Ê.» ˜ª¨Àº.¬ |
| 000000C0 | 88 | 9A | 98 | C0 | AA | 00 | 9B | 80 | 87 | 99 | 87 | 00 | AA | C0 | 9B | 80 | ˆš˜Àª.›€‡™‡.ªÀ›€ |
| 000000D0 | 77 | B9 | 87 | 0C | B0 | 0C | 90 | 80 | 87 | A8 | 97 | 0C | C0 | 0B | AC | 80 | w¹‡.°..€‡¨—.À.¬€ |
| 000000E0 | 87 | C9 | 98 | CC | C0 | 0C | C0 | 90 | 90 | A9 | 99 | 0C | BC | 0A | CC | 90 | ‡É˜ÌÀ.À..©™.¼.Ì. |
| 000000F0 | A0 | CA | BA | 00 | B0 | 0C | 00 | 00 | C0 | CC | B0 | 00 | 00 | 00 | 00 | C0 | Êº.°...ÀÌ°....À |
| 00000100 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | D0 | D8 | 45 | 84 | AE | DD | DA | D0 | ........ÐØE„®ÝÚÐ |

Example of a modern Prefetch file viewed in a hex editor

When Prefetch files are written to disk, they are stored in this compressed format, conserving disk space and ensuring optimal usage of system resources. The impact this has on an investigation is that we can no longer simply open the file in a text editor. Instead, specialist tools need to be used, such as PECmd.exe from the Zimmerman Tools.

```
C:\Windows\Prefetch>pecmd -f SDELETE64.EXE-C8F3326B.pf
PECmd version 1.4.0.0

Author: Eric Zimmerman (saericzimmerman@gmail.com)
https://github.com/EricZimmerman/PECmd

Command line: -f SDELETE64.EXE-C8F3326B.pf

Keywords: temp, tmp

Processing 'SDELETE64.EXE-C8F3326B.pf'

Created on: 2021-06-28 06:02:49
Modified on: 2021-06-28 06:02:49
Last accessed on: 2023-10-30 13:27:34

Executable name: SDELETE64.EXE
Hash: C8F3326B
File size (bytes): 30,788
Version: Windows 10

Run count: 1
Last run: 2021-06-28 06:02:39

Volume information:

#0: Name: \VOLUME{01d27dcbf8124cb1-0cf98fe5} Serial: CF98FE5 Created: 2017-02-03 03:16:54 Directories: 10 File references: 64

Directories referenced: 10
```

Example of PECmd.exe output

In the realm of cybersecurity and digital forensics, Prefetch files are goldmines of information. Investigators examine these files to identify executed applications, their timing, and associated actions. The creation timestamp marks the initial run, and the modification time denotes the last instance. Prefetch files provide up to eight timestamps, offering insight into recent executions and referenced files or folders. This detailed information aids in understanding an executable's purpose, uncovering loaded DLLs, and illuminating application functions.

However, there are challenges to consider. Prefetch files experience a delay after execution before being written to disk, introducing slight timing uncertainties.

The delay between a file being executed and the prefetch being run can be attributed to the Windows operating system's optimisation processes. After an application is launched, Windows takes a brief moment to analyse the execution pattern and decide whether it's a candidate for prefetching. This analysis involves evaluating the application's frequency of use and system resource availability. The necessary data is preloaded into memory once Windows determines that prefetching is beneficial. However, this process introduces a slight delay, typically around 10 seconds, before the prefetch file is actually written to disk. During this time, the system optimises the loading process, ensuring that only relevant and frequently used applications are prefetched, minimising unnecessary resource consumption.

Moreover, these files are absent in Windows Server operating systems. Additionally, they are user-editable, posing a risk if attackers delete the C:\Windows\Prefetch directory. In such cases, forensic experts can resort to memory carving techniques, attempting to salvage prefetch files from system memory. This intricate interplay between prediction, execution, and analysis underscores Prefetch's pivotal role in Windows systems analysis and forensics.

You can learn more about Digital Forensics and Incident Response on Windows endpoints in two awesome SANS courses:

https://sans.org/for500 (Windows Forensic Analysis)

https://sans.org/for508 (Advanced Incident Response, Threat Hunting, and Digital Forensics)

---

## 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