CASE 001

THE STOLEN SZECHUAN SAUCE

By Andrew Mendis

# Table of Contents

[**Table of Contents**](#_mikfdc9pm16o)

[**Executive Summary**](#_wcue24189zb)

[**Methodology**](#_pdo82axcs6d2)

[Evidence & Data](#_iujvoi8t5fv)

[Disk Images](#_qn7to4y6xybq)

[Memory Captures](#_gocdkqe3lk7o)

[Files](#_siguzh3zkyab)

[Tools & Programs](#_ytwfrl38kgy7)

[Access Data FTK Imager](#_d0s9qnaq6fww)

[Autopsy](#_bcupieuv8gz5)

[IE10Analyzer](#_evvwbow5b116)

[PowerShell](#_rz8bm4t0s8bp)

[Registry Explorer](#_ty5l474lgrcn)

[VirusTotal](#_1dbwj4ezxube)

[Volatility](#_e188c22e00gc)

[Verifying Integrity of Evidence](#_sm7fqkd59urj)

[Server Disk Image](#_c56jxpjz8ke5)

[Server Memory Capture](#_vaum5t8xx7q9)

[Workstation Disk Image](#_oewmdfgfsshh)

[Workstation Memory Capture](#_6l1yssrvaidg)

[**Investigation**](#_cae2i7r8apsp)

[Q1: What’s the Operating System of the Server?](#_twem79stvpq6)

[A1: Windows Server 2012 R2 Standard Evaluation](#_euzb9bacj00u)

[Q2: What’s the Operating System of the Desktop?](#_mu6hu4csfryc)

[A2: Windows 10 Enterprise Evaluation](#_9gfw1yg8sban)

[Q3: What was the local time of the Server?](#_l931bf2hjrsw)

[A3: Pacific Standard Time](#_ig1xi8vluff)

[Q4: Was there a breach?](#_7njjmp5w9qdc)

[A4: Yes](#_yx30q2c90yry)

[Q5: What was the initial entry vector (how did they get in)?](#_f8ad8uwf3rz5)

[A5: Bruteforce Attack via RDP](#_ypivufekdnbi)

[Q6.1: Was malware used? If so, what was it? Yes coreupdater.exe](#_64ie0io9ubn2)

[A6.1: What process was malicious?](#_xvqyhgbh1ggy)

[Q6.2: Identify the IP Address that delivered the payload.](#_ehos5tdb5air)

[A6.2: 194.61.24.102](#_ykmkszx1tu7h)

[CITADEL-DC01](#_8f65g0r2dv9p)

[Q6.3: What IP Address is the malware calling to?](#_7t8quqob775u)

[A6.3: 203.78.103.109](#_9pdun9e3vfp1)

[Q6.4: Where is this malware on disk?](#_tpwdi9akwpkj)

[A6.4: System32 folder](#_s50t54ay0mwh)

[Q6.5: When did it first appear?](#_vlnw7s1if98x)

[A6.5: It first appeared at 20:24:12 PST on the system](#_mw620te30e8k)

[Q6.6: Did someone move it?](#_yrfmsmruxs1b)

[A6.6: Yes, to C:\Windows\System32\](#_26g6iswqlvdk)

[Q6.7: What were the capabilities of this malware?](#_ytg2uyxqufkm)

[A6.7: Many](#_yamko7bh7x0t)

[Q6.8: Is this malware easily obtained?](#_lkvz4rmt4qqh)

[A6.8: Yes](#_1es71bqg8ec8)

[Q6.9: Was this malware installed with persistence on any machine?](#_17ecllmmuh7v)

[A6.9: Yes](#_n9vy8uxwawcx)

[A6.9.1 & 6.9.2: When? & Where?](#_oswkwn87byax)

[Q7: What malicious IP Addresses were involved?](#_ej1k1mowivst)

[A: 194.61.24.102 & 203.78.103.109](#_o36y9j6dimq9)

[Q7.2: Were any IP Addresses from known adversary infrastructure?](#_kfsvga8tuohr)

[A7.2: Yes](#_5iplno80p3ej)

[Q7.3: Are these pieces of adversary infrastructure involved in other attacks around the time of the attack?](#_tacnufhlhlm1)

[A7.3: Yes](#_xq1x7cc52d60)

[Q8: Did the attacker access any other systems?](#_cs809s9ghlii)

[A8: Yes](#_66u4k9bgzhop)

[Q8.2: How?](#_m7hop9tn26yk)

[A8.2: Via RDP](#_2v8jg4gld7fe)

[Q8.3: When?](#_vp9ws8d5drar)

[A8.3: On 18/09/2020 at 20:36:24 PST](#_ifd5zk71tf73)

[Q8.4: Did the attacker steal or access any data? If so, when?](#_pitd2zsgrmq1)

[A8.4: Yes](#_qw0oqbuxlxre)

[Q9: What was the network layout of the victim network?](#_mmy38zrwxu7i)

[A9: The network layout is quite simple; the network 10.42.85.0/24 has two devices in it. CITADEL-DC01 has the IP address 10.42.85.10 and DESKTOP-SDN1RPT has the IP address 10.42.85.115.](#_yohncamah6rl)

[**Timeline**](#_2bqxthm95czc)

[**Recommendations**](#_255abksgxiug)

[**CITATIONS**](#_qjjwena1hb4t)

# 

# 

# 

# 

# 

# Executive Summary

The following report outlines the results and findings from the Stolen Szechuan Sauce Case. A number of tools were used for the purposes of the investigation. The tools consist of the following; FTKImager, volatility, and Autopsy. The investigation found evidence of a breach which was done through an RDP brute force attack. After the successful breach a payload was created using metasploit and subsequently deployed and executed onto the system. An administrator account was compromised, a number of files were exfiltrated, and persistence was established on the computer. The malware is capable of many functions which will be detailed in the report below. The report will outline the many vulnerabilities in the organization which lead to this breach, a timeline of the events, and recommend changes to prevent further breaches from happening.

# Methodology

## **Evidence & Data**

### Disk Images

**DC01-E01.zip** - Disk image of “CITADEL-DC01” the domain controller.

**DESKTOP-E01.zip** - Disk image of “DESKTOP-SDN1RPT” the workstation computer.

### Memory Captures

**DC01-memory.zip** - Memory Capture for the domain controller “CITADEL-DC01”

**DESKTOP-SDN1RPT-memory.zip** - Memory Capture for the domain controller “DESKTOP-SDN1RPT”

### Files

**SOFTWARE\_Clean** - Software hive for the two systems. Has a separate file with the same name for each in different directories.

**SYSTEM\_clean** - Software hive for the two systems. Has a separate file with the same name for each in different directories.

**Security.evtx** - Security events log file for the two systems. Has a separate file with the same name for each in different directories.

**System.evtx** - System events log file for the two systems. Has a separate file with the same name for each in different directories.

**WebCache.dat** - File containing the cached items from Internet Explorer and Microsoft Edge. Has information regarding web history and file downloads. Has a separate file with the same name for each in different directories.

**$UsnJrnl:$J** - Log file for file activity. Shows a history of when files were accessed, modified, created, and deleted. Important for creating a timeline of file activity when advanced logging hasn’t been enabled on a system. Has a separate file with the same name for each in different directories.

## **Tools & Programs**

### *Access Data FTK Imager*

FTK Imager provides the mounting and access of the Disk Images. Through the imager I’ll be able to gain access to valuable artifacts such as registry hives, and documents with ease.

### *Autopsy*

Autopsy is a powerful and useful tool which scans and indexes the files in the Disk Images from our evidence. Once the scan is finished it will provide hashes of the individual files and detail things such as creation time, last access time, modified times, and many more properties. This will help give a high level overview of events and aid in creating a timeline of the events that took place in the systems and network. Having this kind of insight will help understand how the breach happened and gain insight to subsequently fix the vulnerabilities.

### *IE10Analyzer*

This tool allows us to analyze the WebCache file for Internet Explorer and Microsoft Edge. giving us insight into the history of webpages visited and files downloaded.

### *MFTEcmd*

Allows the conversion of the UsnJrnl file into a CSV table making it more readable.

### *PowerShell*

Powershell allows us to download Eric Zimmerman’s tools, and hash evidence to verify their integrity against the given hash list.

### *Registry Explorer*

Registry Hives will be opened here for investigation. The program helps us view contents in the registry to gather valuable insight into what happened to the systems. It provides a user-friendly interface to explore the values in the hives, helping to extract information and correlate it towards the accuracy of the timeline of events that happened.

### *VirusTotal*

VirusTotal will provide a comprehensive cross-analysis between antivirus/antimalware providers to check the reputation of a file or its hash.

### *Volatility*

Volatility provides analysis of memory captures. It's another powerful tool that gives us insight into what processes were running and what open connections they had. The tool will also allow us to extract running programs and analyze them with other tools.

## **Verifying Integrity of Evidence**

Before starting work on the case, the initial step is to verify the integrity of the evidence. The client had provided MD5 hashes for the evidence which was used as reference to validate the integrity of the files once copies were received. Below are the MD5 values that were hashed post intake.

### Server Disk Image

| *Client Provided Hash* ↴    *Locally Produced Hash* ↴ |
| --- |

### Server Memory Capture

| *Client Provided Hash* ↴  Y  *Locally Produced Hash* ↴ |
| --- |

### Workstation Disk Image

| *Client Provided Hash* ↴    *Locally Produced Hash* ↴ |
| --- |

### Workstation Memory Capture

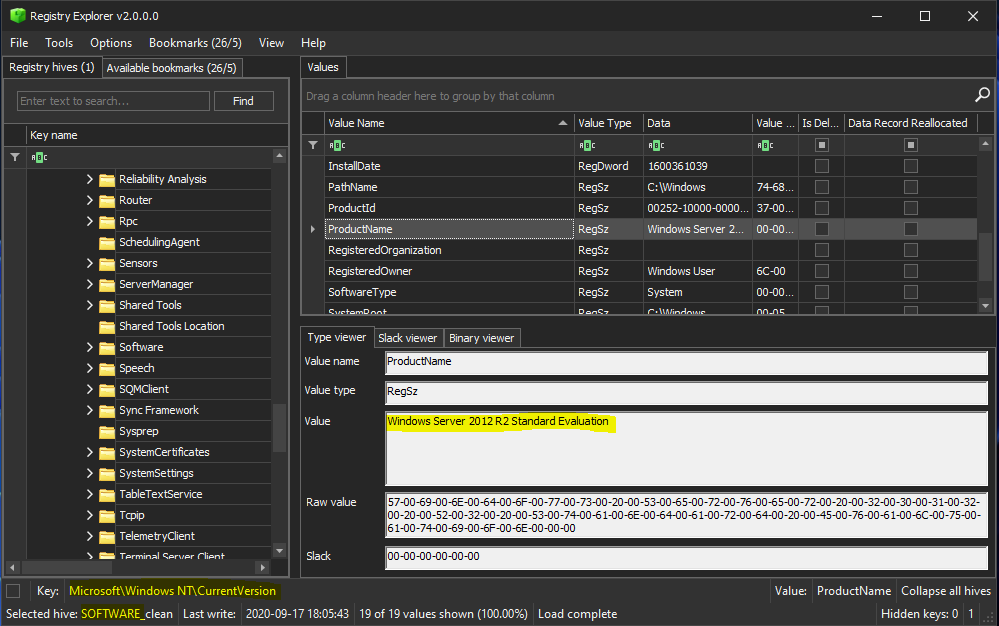
| *Client Provided Hash* ↴    *Locally Produced Hash* ↴ |
| --- |

# Investigation

## Q1: What’s the Operating System of the Server?

### **A1: Windows Server 2012 R2 Standard Evaluation**

Using FTK Imager we can first mount the image of the server to view the SOFTWARE registry hive. Once mounted, we navigate to HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion where the data for the ProductName value tells us what OS the server is running.



## Q2: What’s the Operating System of the Desktop?

### **A2: Windows 10 Enterprise Evaluation**

Using FTK Imager we can first mount the image of the desktop to retrieve the SOFTWARE registry hive. Once that’s done, we navigate to HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion where the data for the ProductName value tells us what OS the desktop is running.

| **1** |  |
| --- | --- |

## Q3: What was the local time of the Server?

### **A3: Pacific Standard Time**

Like with the previous questions we use FTK Imager to first mount the image of the server. Afterwards we retrieve the SOFTWARE registry hive. Once that’s done, we navigate to HKLM\SOFTWARE\ControlSet001\Control\TimeZoneInformation where the data for the TimeZoneKeyName value tells us what the local time zone is on the server.

| **1** |  |
| --- | --- |

## Q4: Was there a breach?

### **A4: Yes**

### Q5: What was the initial entry vector (how did they get in)?

#### **A5: Bruteforce Attack via RDP**

Looking through the security logs we can see that there were multiple failed login attempts originating from a computer called ‘kali’. All trying to attempt to login via a network type connection (Logon Type 3) using the account ‘Administrator’. After many failed attempts they login successfully using either Terminal Services or an RDP Connection (Logon Type 10) originating from ‘194.61.24.102’. The first picture below shows the repeated attempts to login via the **Administrator** account. The first entry is a false positive. It was normal user behaviour, as they logged in with correct credentials after the one bad attempt. The brute force attack started at 20:21:25 PST on 18/09/2020, and after 96 attempts they were able to successfully login at 20:21:48 PM PST.

| **1** |  |
| --- | --- |
| **2** |  |

### Q6.1: Was malware used? If so, what was it? Yes coreupdater.exe

#### A6.1: What process was malicious?

Using the PsList module in **Volatility** “coreupdater.exe\* was found *(Table #1)*. It didn’t seem like a normal windows process or service. Using the “cmdline” module *(Table #2)* in volatility, a check was made to see if it was recently executed and if so what program or process executed it. Not much information was provided other than that it could have been executed from a directory that’s part of the PATH system variable. Afterwards a process dump was run against the PID of the process to get the executable *(Table #3)* to run further analysis. The hash was submitted to Virus total and returned with an 87% positive result *(Table #4)*.

| **1** |  |
| --- | --- |
| **2** |  |
| **3** |  |
| **4** |  |

### Q6.2: Identify the IP Address that delivered the payload.

#### A6.2: 194.61.24.102

We can make a good assumption that the source of the file was from the same IP (194.61.24.102) as the source of the RDP connection from [Q4.1](#_ypivufekdnbi). We know from the VirusTotal [results](#_rxm3l5bujw08) that the payload was created using metasploit, and the source machine’s hostname is [kali.](#_stqvou26fnls) Kali is known for penetration testing and usually comes with metasploit pre-installed.

| **1** |  |
| --- | --- |

### CITADEL-DC01

### Q6.3: What IP Address is the malware calling to?

#### A6.3: 203.78.103.109

Further analysis revealed that the malware has an open socket. Using the NetStat module in volatility, it showed the following result *(Table #1).* The malware seems to be active and communicating with the foreign address “203.78.103.109”. The virus total results *(Table #3)* show that the IP address has a history of communications involving the “coreupdater.exe” executable. It also has other IP addresses associated that also have a history of malware *(Table #3)*.

| **1** |  |
| --- | --- |
| **1** |  |
| **2** |  |

### Q6.4: Where is this malware on disk?

#### A6.4: System32 folder

Running the PsTree module shows that it was executed from the System32 directory.

| **1** | **↘ ↙**  🡓 |
| --- | --- |

### Q6.5: When did it first appear?

#### A6.5: It first appeared at 20:24:12 PST on the system

By analyzing the WebCache.dat file using IE10Analyzer we can conclude that the file was first downloaded to the Administrator’s Downloads folder at 20:24:12 PST.

| **1** | **↘ ↙**  🡓 |
| --- | --- |

### Q6.6: Did someone move it?

#### A6.6: Yes, to C:\Windows\System32\

Using Autopsy we can confirm its only current location on the system by doing a search and asking to view the file in its source.

| **1** |  |
| --- | --- |

### Q6.7: What were the capabilities of this malware?

#### A6.7: Many

The malware is a metasploit payload. Depending on how the module/payload was created it can have a number of capabilities. Below in *(Table #1)* you’ll find some of its detected techniques and capabilities from Falcon Sandbox analysis on Hybrid Analysis’ website. From the results shown we can assume it has the following capabilities; Persistence, Privilege Escalation, Credential Access, Discovery, and Command and Control. These are just the known and detected features of the payload. There could be more that the sandbox/software did not have a chance to detect due to test duration.

| **1** |  |
| --- | --- |

### Q6.8: Is this malware easily obtained?

#### A6.8: Yes

Metasploit is a free to use open-source application used by penetration testers and the like. It comes preinstalled on some linux distributions and has comprehensive documentation made for it. It has a large community base and users willing to do a deep dive can be creating and deploying payloads on systems in a short amount of time.

### Q6.9: Was this malware installed with persistence on any machine?

#### A6.9: Yes

### A6.9.1 & 6.9.2: When? & Where?

The malware was installed on both machines and set up persistence through registry entries and setting itself up as a service. The service is set to start-up automatically. As shown in *(Table #1)* for the Server& *(Table #2)* for the Workstationthe services were installed at 2020-09-18 20:27:49 (20:27:49 PST) for the Server & 2020-09-18 (08:42:42 PST) for the Workstation. The registry keys were also installed at the same time as the services for each respectively as shown in *(Table #3)* for the Server & *(Table #4)* for the Workstation.

| **1** |  |
| --- | --- |
| **2** |  |
| **3** |  |
| **4** |  |

### Q7: What malicious IP Addresses were involved?

#### A: 194.61.24.102 & 203.78.103.109

### Q7.2: Were any IP Addresses from known adversary infrastructure?

#### A7.2: Yes

| **1** |  |
| --- | --- |

### Q7.3: Are these pieces of adversary infrastructure involved in other attacks around the time of the attack?

#### A7.3: Yes

Hybrid analysis shows that 203.78.103.109 has been involved in 7 attacks in total in the past. It has been used for a variety of types of attacks.

| **1** |  |
| --- | --- |

### Q8: Did the attacker access any other systems?

#### A8: Yes

### Q8.2: How?

#### A8.2: Via RDP

Since the attacker has the credentials for the Administrator account on the domain, they can login using RDP over the local network to other devices on the domain with ease. Below *(Table #1)* you will find the attacker logged in to the workstation using the Administrator account on 18/09/2020 at 20:36:24 PST via RDP (Logon Type 10). They logged in via the server as evidenced in *(Table #2)*. The originating IP address is the same as the DC server's local IP.

| **1** |  |
| --- | --- |
| **2** |  |

### Q8.3: When?

#### A8.3: On 18/09/2020 at 20:36:24 PST

The attacker logged in to the workstation using the Administrator account on 18/09/2020 at 20:36:24 PST via RDP (Logon Type 10) as was shown in *(Table #1)* above.

### Q8.4: Did the attacker steal or access any data? If so, when?

#### A8.4: Yes

Autopsy shows us the most recently interacted with files on the systems. We can see some suspicious activity from the Administrator user after the compromise had happened. The attacker accessed a number of files in the Desktop, Documents, and Pictures folders of the user ‘mortysmith’ on the Desktop Device (DESKTOP-SDN1RPT). The attacker also accessed a FileShare called ‘Secret’ on the Server (CITADEL-DC01). It looks like the files on the Desktop were compressed into a zip file called ‘loot.zip’ and the contents of the fileshare on the server compressed into a file called ‘Secret.zip’. A list of the most recently interacted files between the Desktop and Server are shown below in *(Table #1).* The recent and suspicious file called loot.zip was nowhere to be found. At which point I started to inspect the $UsnJrnl file as Autopsy showed an entry in it while searching for ‘loot.zip’ as shown in *(Table #2)*. Using MFTECmd.exe the journals on both devices were parsed and exported into CSV tables. The entries indicate that upon the creation of loot.zip and Secret.zip files they were immediately deleted. We can suspect that before deletion the files were exfiltrated as archiving multiple files into one makes it easier for attackers to download files as one. We can make an educated judgment that ‘loot.zip’ was exfiltrated approximately at 20:46:XX PST before deletion *(Table #4)* and ‘Secret.zip’ was exfiltrated approximately at 20:32XX PST before deletion as shown in *(Table #5)* in their own respective $UsnJrnl entries.

| **1** |  |
| --- | --- |
| **2** |  |
| **3** |  |
| **4** |  |
| **5** | **.**  **.**  **.**  **.** |

### Q9: What was the network layout of the victim network?

#### A9: The network layout is quite simple; the network 10.42.85.0/24 has two devices in it. CITADEL-DC01 has the IP address 10.42.85.10 and DESKTOP-SDN1RPT has the IP address 10.42.85.115.

Looking at the interfaces key we can see the details for the network configuration on each machine. *(Table #1)* shows the network configuration for CITADEL-DC01 & *(Table #2)* shows the network configuration for DESKTOP-SDN1RPT. *(Table #2)* shows a diagram visualizing the information from these two entries into a network diagram.l

| **1** |  |
| --- | --- |
| **2** |  |
| **3** |  |

# Timeline

| **Date + Time** | **Event** |
| --- | --- |
| 18/09/2020 20:21:25 PM PST | Attacker starts bruteforce attack on CITADEL-DC01 |
| 18/09/2020 20:21:48 PM PST | Attacker succeeds in bruteforce attack |
| 18/09/2020 20:24:12 PST | Payload downloaded into the Administrators Downloads folder |
| 18/09/2020 20:24:12 PST | Malware was moved to the \*\System32 folder |
| ↕ | Malware was executed |
| 18/09/2020 20:27:49 PST | Persistence was established via registry key entry and autostart service |
| 18/09/2020 20:32XX PST | ‘Secret.zip’ was exfiltrated |
| 18/09/2020 20:36:24 PST | Moved laterally to DESKTOP-SDN1RPT via RDP connection |
| 18/09/2020 20:40:01 PST | Payload downloaded into the Administrators Downloads folder |
| ↕ | Malware was executed |
| 18/09/2020 20:42:42 PST | Persistence was established via registry key entry and autostart service |
| 18/09/2020 20:46:XX PST | loot.zip’ was exfiltrated |

# Recommendations

**Disable RDP Port 3389**

The events that occurred happened due to RDP being enabled over the internet. Should RDP be needed it should only be used locally. Blocking port 3389 on the firewall removes the threat of anyone attempting this from outside the network all together.

**Stronger Password Policy**

The brute force attack took less than 1 minute to be successful. This indicates that the user “Administrator’ had a weak password. An implementation of a password policy would help prolong a subsequent attack or make the attacker give up altogether due to time constraints.

**2FA for RDP/Domain Login**

Enabling 2FA would essentially nullify brute force attacks in the future given that the attacker doesn’t have the second authentication method. This would in general strengthen the overall security posture of the organization.

**Install/Enable IPS System**

Having an IPS system would have automatically blocked this type of attack.

# CITATIONS

Labs, Fsp. (n.d.). *Process tracking with Event Log Explorer*. Event Log Explorer blog. https://eventlogxp.com/blog/process-tracking-with-event-log-explorer/

Moaistory. (n.d.). *Moaistory/IE10Analyzer: Http://moaistory.blogspot.com/2016/08/ie10analyzer.html*. GitHub. https://github.com/moaistory/IE10Analyzer

*Payload module*. Payload Module - an overview | ScienceDirect Topics. (n.d.). https://www.sciencedirect.com/topics/computer-science/payload-module

Summerson, C., & Lewis, N. (2023, November 29). *How to edit your system path for easy command line access in windows*. How. https://www.howtogeek.com/118594/how-to-edit-your-system-path-for-easy-command-line-access/

USNJRNL forensics extraction for Efficient Investigation | Otorio. (n.d.-a). https://www.otorio.com/resources/usnjrnl-extraction-for-efficient-investigation/

Vinaypamnani-Msft. (n.d.). *4624(s) an account was successfully logged on. - windows 10*. 4624(S) An account was successfully logged on. - Windows 10 | Microsoft Learn. https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-10/security/threat-protection/auditing/event-4624

Volatility filescan – eyehatemalwares. (n.d.-b). https://www.eyehatemalwares.com/digital-forensics/memory-analysis/volatility-filescan/

Free Automated Malware Analysis Service - powered by Falcon Sandbox - viewing online file analysis results for “coreupdater.exe.” (n.d.). https://www.hybrid-analysis.com/sample/10f3b92002bb98467334161cf85d0b1730851f9256f83c27db125e9a0c1cfda6/5f7695f4a553eb21aa0cdfe1

*How payloads work*. Metasploit Documentation Penetration Testing Software, Pen Testing Security. (n.d.). https://docs.metasploit.com/docs/using-metasploit/basics/how-payloads-work.html

*Event ID 4625 without source IP*. Server Fault. https://serverfault.com/questions/683837/event-id-4625-without-source-ip

*How to know when a file was deleted in a NTFS filesystem?*. Information Security Stack Exchange. https://security.stackexchange.com/questions/61166/how-to-know-when-a-file-was-deleted-in-a-ntfs-filesystem

*What is the WebCache folder in AppData/Local/Microsoft/Windows/ Windows 10*. Super User. https://superuser.com/questions/1600395/what-is-the-webcache-folder-in-appdata-local-microsoft-windows-windows-10