The Stolen Szechuan Sauce Forensic Case Study

Daniel (Zhiyi) Deng

Table of Contents

        - Where?
    - What malicious IP Addresses were involved?
        - Were any IP Addresses from known adversary infrastructure?
        - Are these pieces of adversary infrastructure involved in other attacks around the time of the attack?
    - Did the attacker access any other systems?
        - How?
        - When?
        - Did the attacker steal or access any data?
        - When?
    - What was the network layout of the victim network?
- Additional investigation
- Advanced and Bonus Questions
- Summary
- Reference

Executive Summary:

Asian Fusion Wok, a growing restaurant business, experienced a cyberattack in September 2020 and engaged our services as a Managed Security Service Provider (MSSP) to investigate. With the assistance of AFW's executive team, we successfully carried out the forensic process, reconstructed the entire attack process.

The purpose of this forensic investigation was to determine the attack vector, evaluate the extent of the breach, and confirm whether any sensitive data was compromised within AFW's IT environment. This report details the connection between the attack and our investigative findings and provides recommendations for enhancing security measures to prevent similar incidents in the future.

Key Findings

- **Unauthorized Access Detected**:
  Analysis of security logs and network traffic confirmed that both the server and desktop systems were compromised following a series of unsuccessful login attempts from a suspicious IP address.
- **Malware Infiltration**:
  A malicious program was discovered on the system, granting the attacker long-term access to the network and facilitating the transfer of data to an external server.
- **Data Exfiltration Confirmed**:
  Encrypted network traffic revealed data was being sent to a known Command and Control (C2) server, indicating that sensitive information, including the Szechuan sauce recipe, was accessed and extracted.

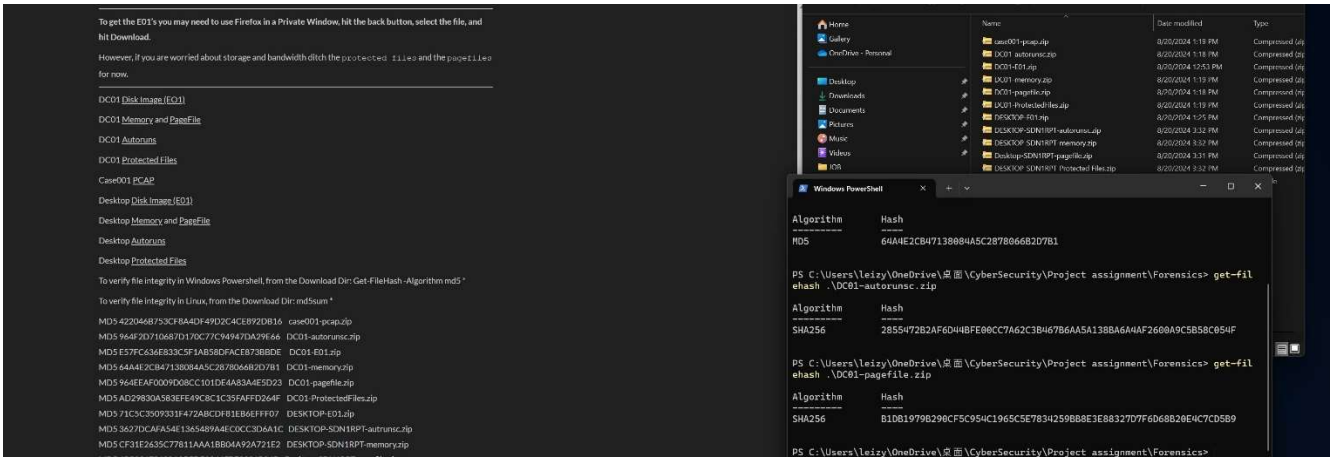
Our investigation determined that the attacker gained access to and exfiltrated sensitive files, including the **Szechuan sauce recipe**. The breach occurred via a brute-force attack on the Remote Desktop Protocol (RDP), allowing unauthorized entry into the system.

To mitigate the risk of similar incidents in the future, we recommend

- **Secure RDP Access**: Use a VPN with MFA for RDP, and regularly audit access permissions.
- **Network Isolation**: Isolate the DC in a separate network segment and prevent Internet exposure.
- **Disable External RDP**: Turn off RDP from the Internet; use secure methods like VPNs or jump servers for remote access.
- **Network Monitoring**: Continuously monitor network traffic and use IDS to detect and respond to suspicious activity.

Forensic Methodology

For the **Collection** phase, the original artifact (such as a disk image or memory dump) is provided by the certified entity, DFIR Madness, which owns the compromised system. This ensures that the evidence is handled and transferred properly, maintaining the integrity and chain of custody throughout the forensic investigation. Before conducting the analysis, we checked the hash to make sure the integrity of the artifact.



The tools for carrying out the forensics:

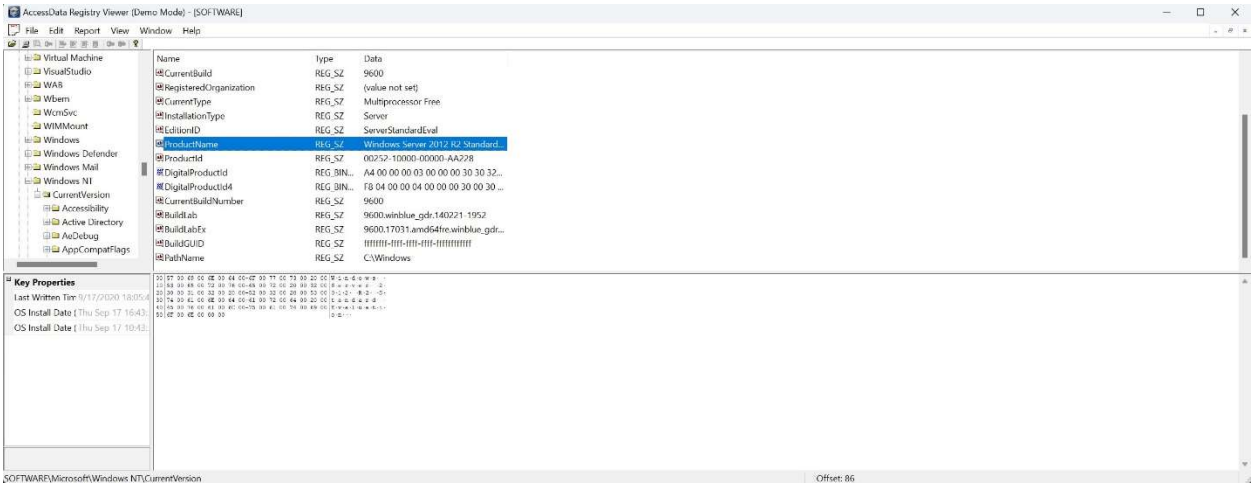
- PowerShell
- FTK imager
- Eric Zimmerman Registry Explorer (FTK registry viewer can be alternative)
- Wireshark
- Event Viewer (Event Log Explorer can be alternative)
- VirusTotal
- AlianVault

For the **Analysis** process, we mount forensic images using FTK imager to examine system images, establish dates, times, and operating systems, and document findings. Review system images for obvious malware and inspect user downloads for any intentional or accidental malware downloads. We used Wireshark to analyze the incident PCAP file and identify potential Indicators of Compromise (IOCs). OSINT tools such as VirusTotal and AlienVault were utilized to gather additional details. Registry Explorer was employed to delve into the evidence and confirm our suspicions. Finally, we drafted this report to document our detailed findings and provide recommendations.

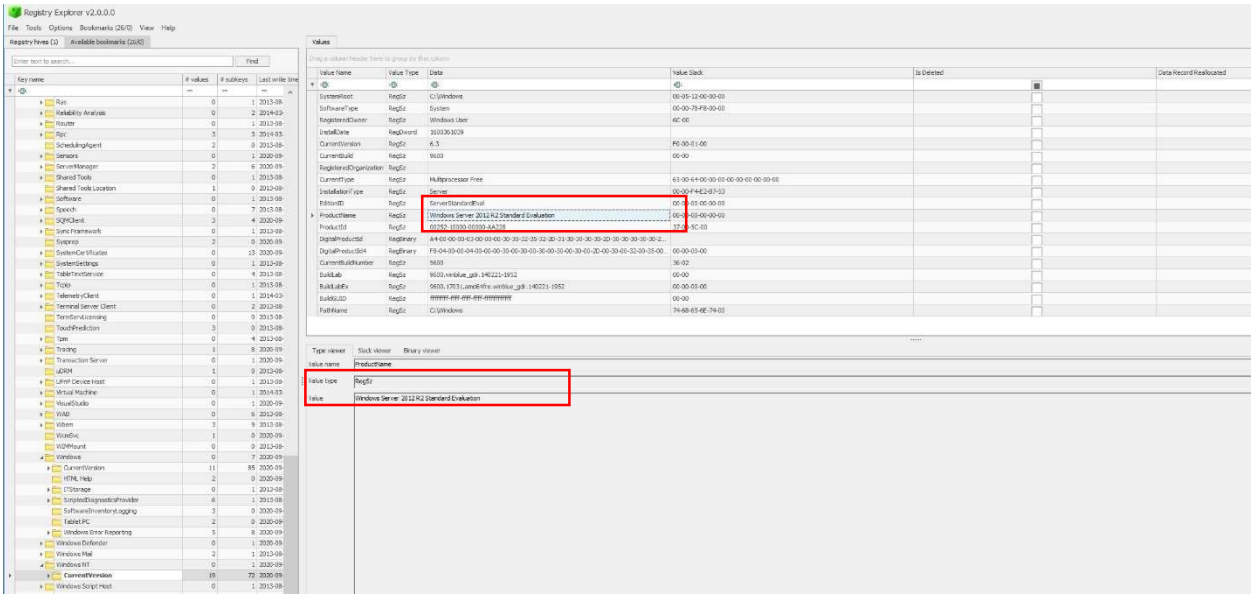
Investigation Result:

What's the Operating System of the Server? Windows Server 2012

We can obtain this information in different ways. One option is to read C:\Windows\System32\license.rtf from a disk image using an FTK imager. However, this path is outdated for modern Windows versions. While older versions like XP used this file for the EULA, newer versions (Windows 7, 8, 10, 11) have shifted to digital agreements and online activation services, so the file may no longer exist or be required.

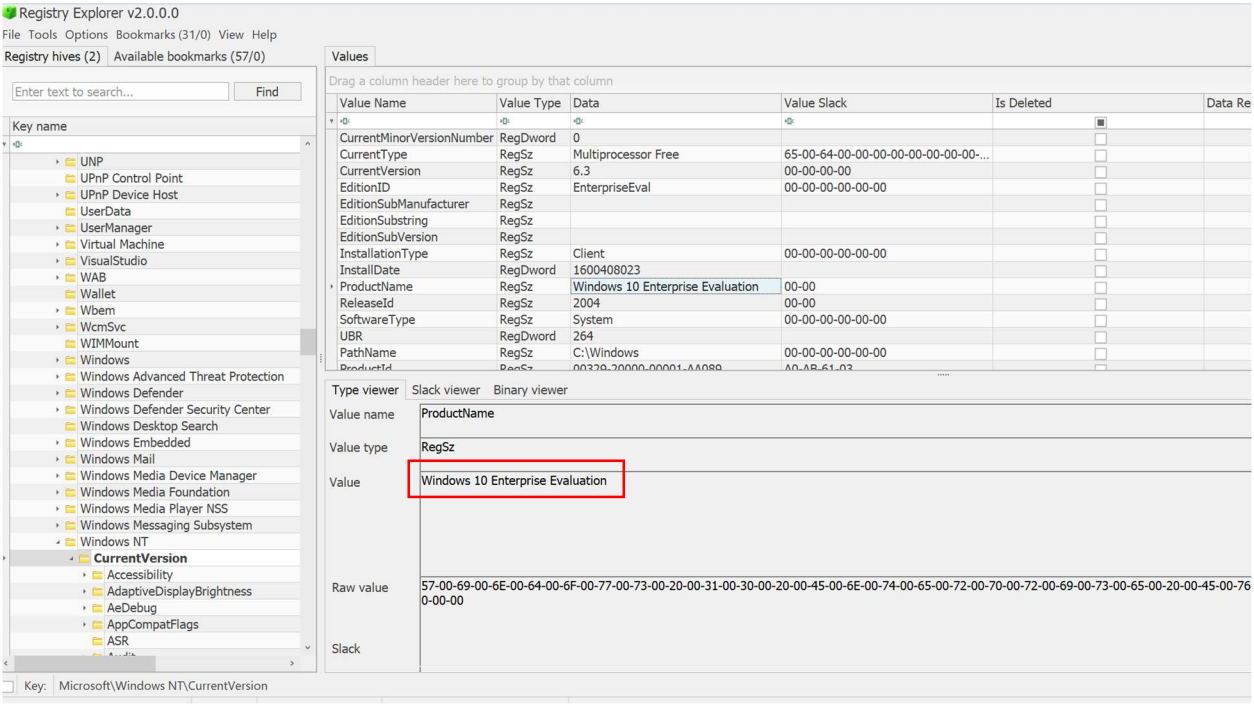


Another way to obtain this information is by mounting the disk image and navigating to C:\Windows\System32\config, where registry hive files like SOFTWARE are stored. These hives contain data such as the HKLM\Software\Microsoft\Windows NT\CurrentVersion key, which provides accurate system version and build details. You can export the registry hives for System, Software, and Security, then use Eric Zimmerman's Registry Viewer to open them.

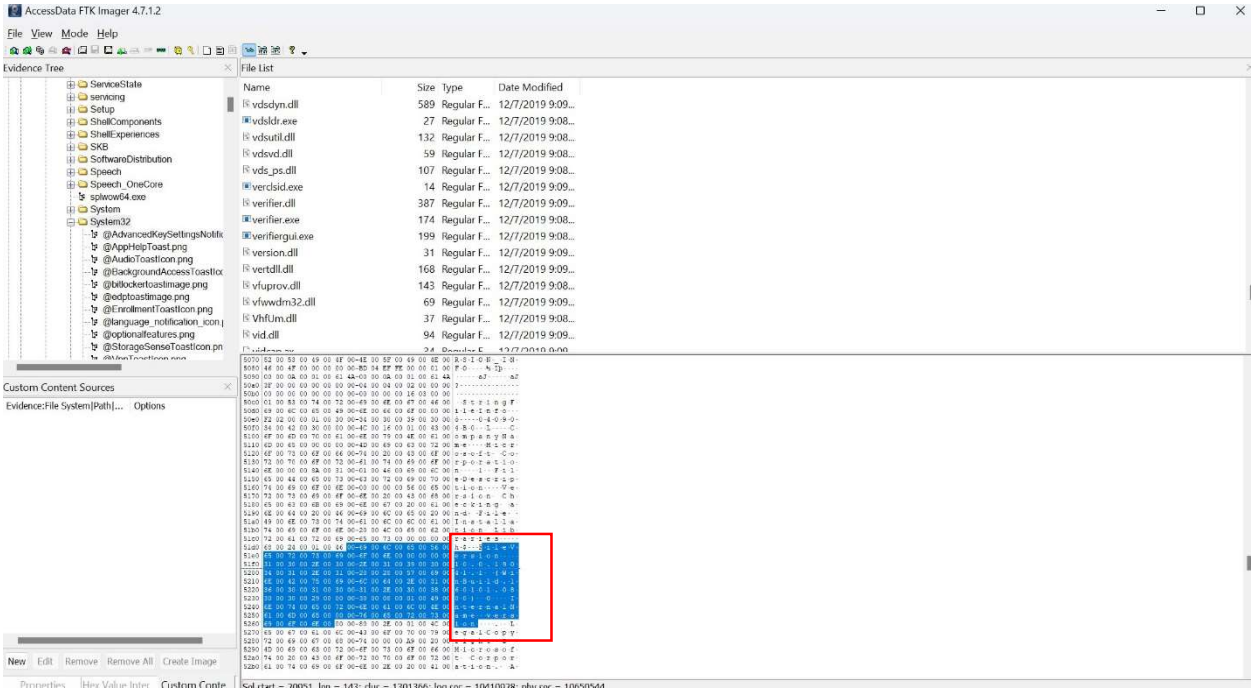


What's the Operating System of the Desktop? Windows 10

Using the same method mentioned above, we can obtain this information, first we use EZ Register Explorer to view the SOFTWARE registry file:



Or we can use FTK imager to read the version.dll file.



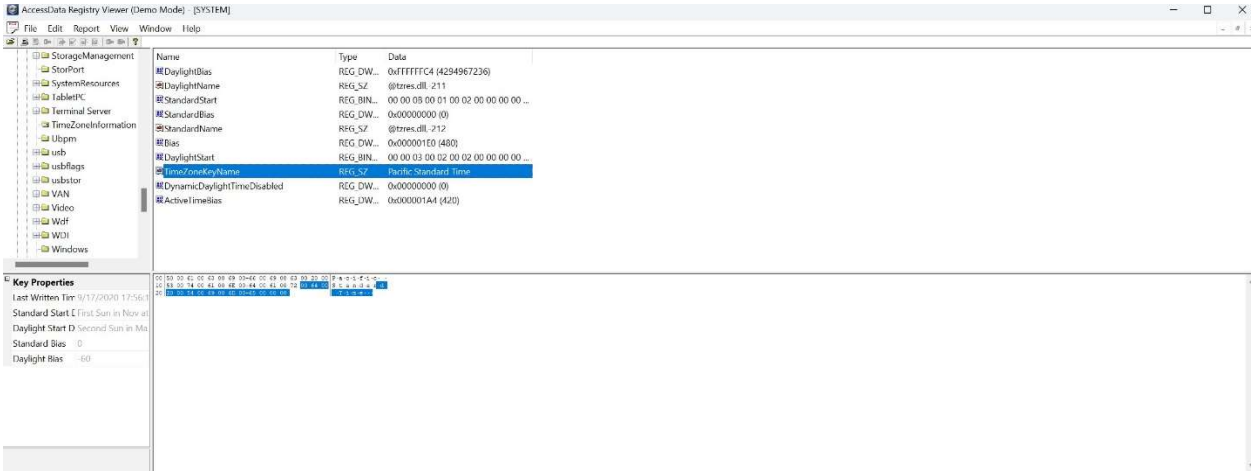
The version number **1.0.0.19041.1** in the version.dll file is indicative of the operating system version. This specific version number corresponds to: **Windows 10 Version 19041**, also known as **Windows 10 May 2020 Update (20H1)**.

What was the local time of the Server? PST

The local time of the server can be found in the **SYSTEM** registry hive. Specifically, it is located under the key:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\TimeZoneInformation

We can obtain this information by exporting **SYSTEM** registry from FTK imager, then read it in FTK Registry Viewer.

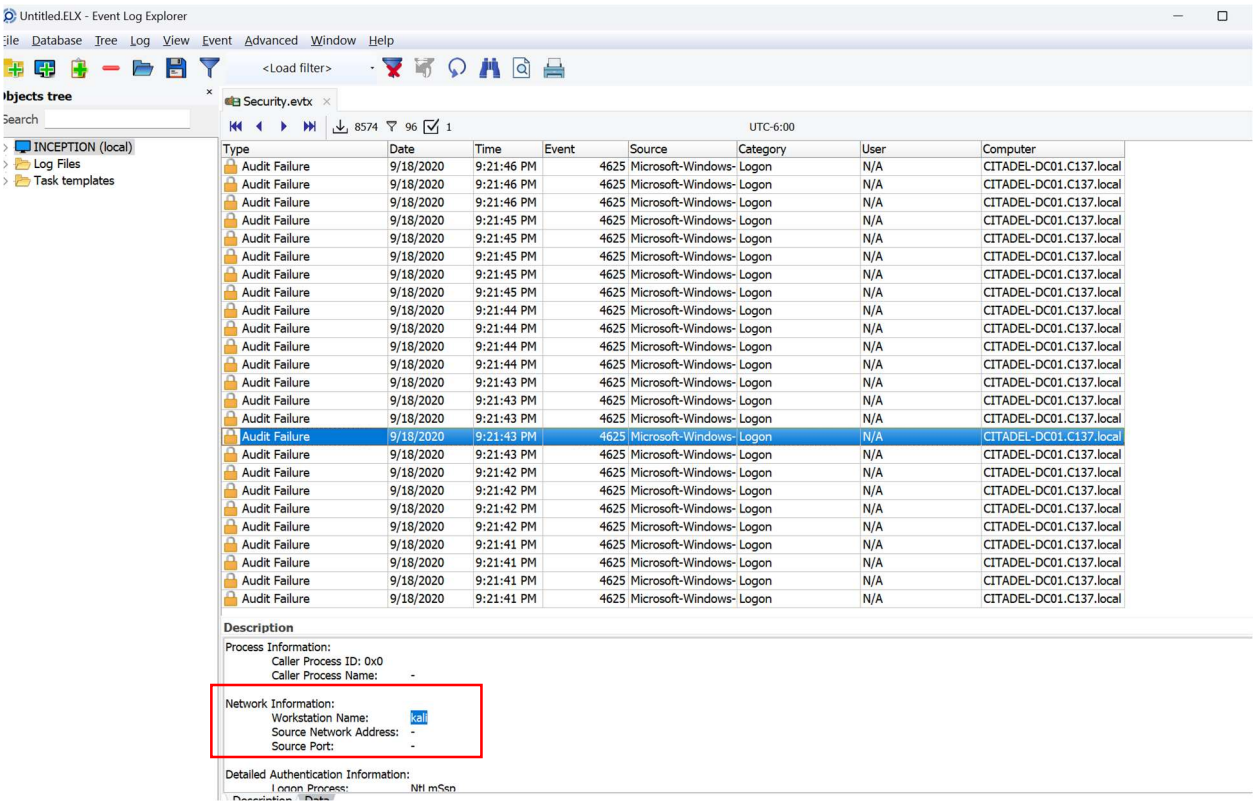


Note: This incident took place in Colorado in September, which falls under UTC -6. However, the PST timestamps in the logs suggest they are using Pacific Time, which differs from the local time in Colorado. To ensure consistency and accurately analyze the incident's timing, it's important to choose a single time zone for all machines and align the timestamps accordingly.

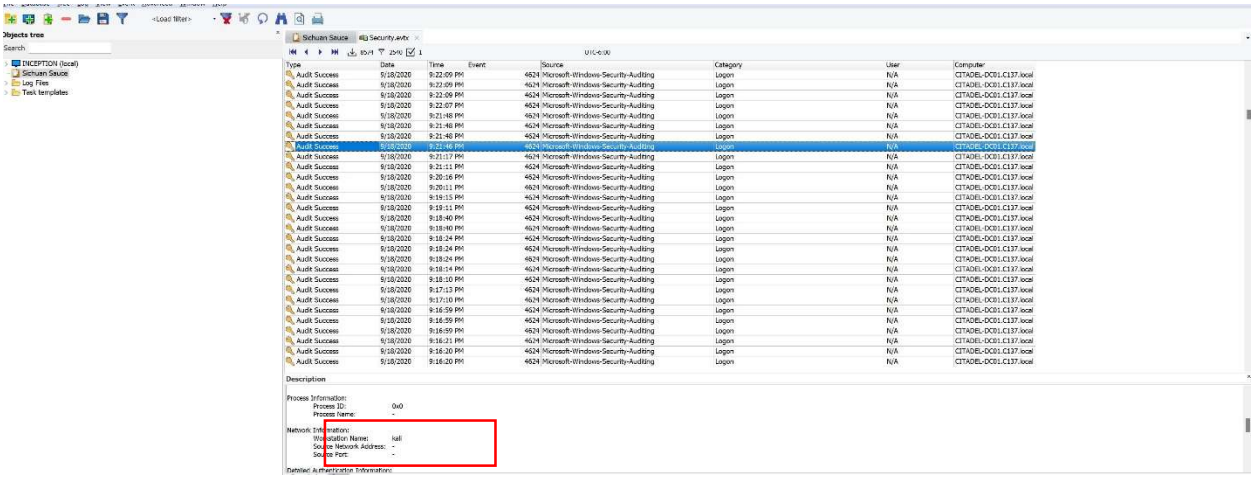
Was there a breach? I will answer it later

What was the initial entry vector (how did they get in)? Brute Force

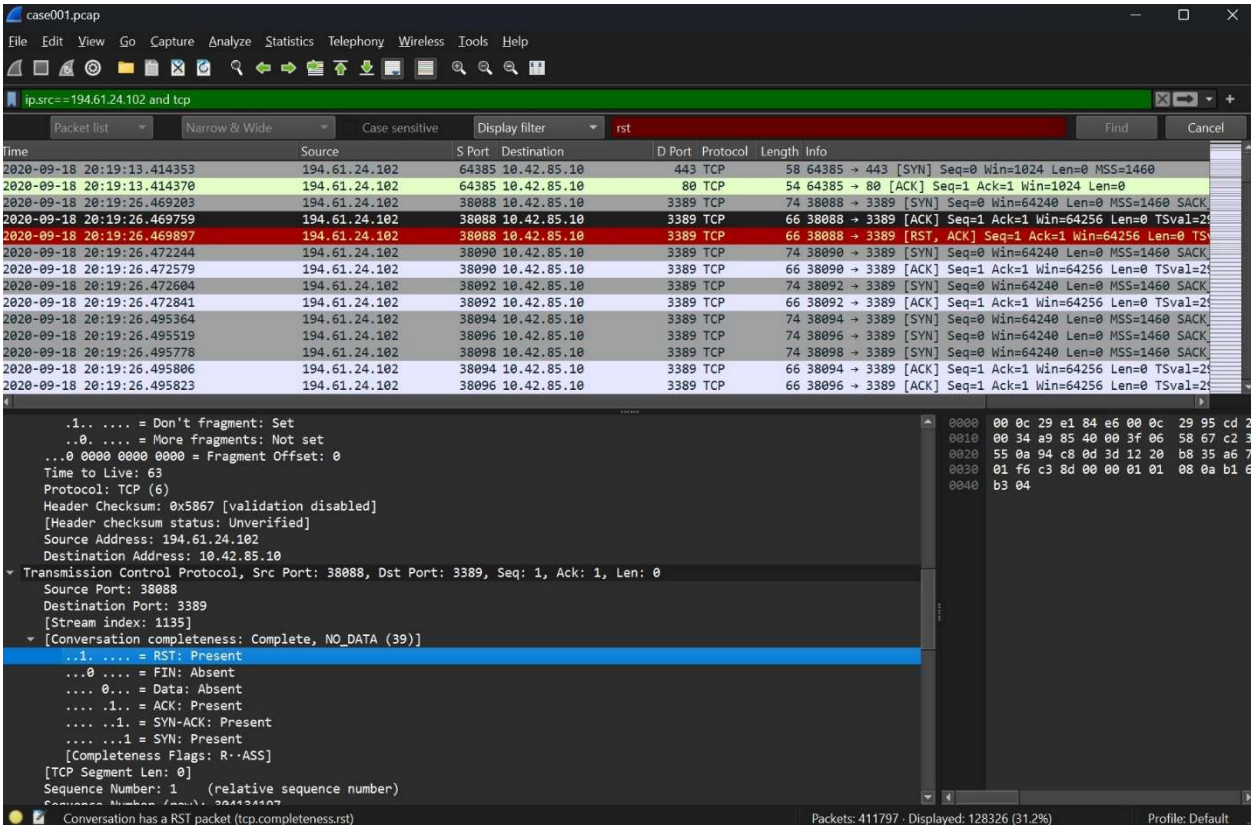
There are two ways to get the information, through PCAP file or Event logs. We will use both approaches. Export the **Security.evtx** log file from the following path: C:\Windows\System32\winevt\Logs\Security.evtx. Once you open this file in **Windows Event Viewer**, you'll notice several event IDs in the 4625, indicating failed login attempts (Microsoft, 4625(F): An account failed to log on., n.d.). These failed attempts occurred in quick succession and originated from a Kali host on 9/18/2020 at 9:21, suggesting a possible brute-force attack on the Domain Controller.



By following the timestamps and filtering event ID 4624, which indicates a successful logon, we confirmed that after the brute force attack, Kali successfully logged into our domain controller (Citadel-DC01) at 9:21:46 PST.

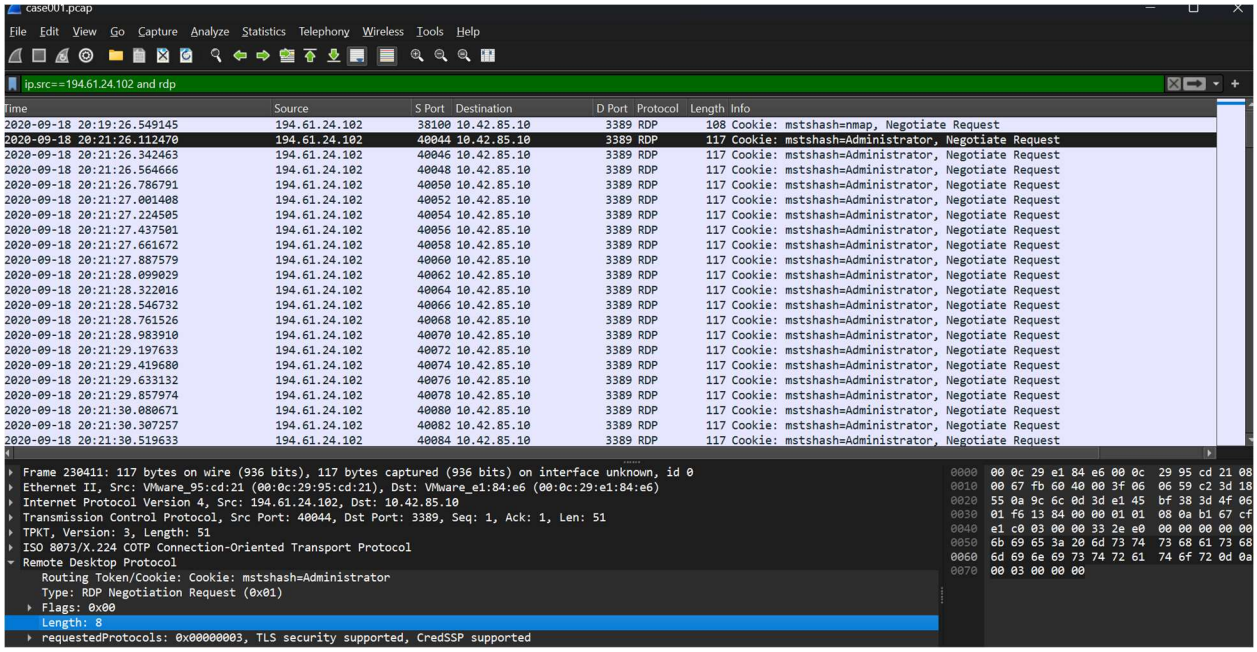


Further investigation using Wireshark to analyze the accompanying PCAP file confirmed this suspicion. The reconnaissance from IP 194.61.24.102 (Kali) occurred on September 18, 2020, around 20:19 Mountain Time (19:19 PST). The initial ping, accompanied by an RST flag, indicated a terminated connection. Following this, a large number of pings were detected targeting the destination IP 10.42.85.10 (our DC) on port 3389 (RDP) within 1 second, suggesting a brute-force attack.



The brute force attack from a Kali machine targeting a Windows server via **port 3389** (RDP) seeks to exploit Remote Desktop Protocol, which is commonly used for Windows remote management and access, by repeatedly attempting to guess valid login credentials.

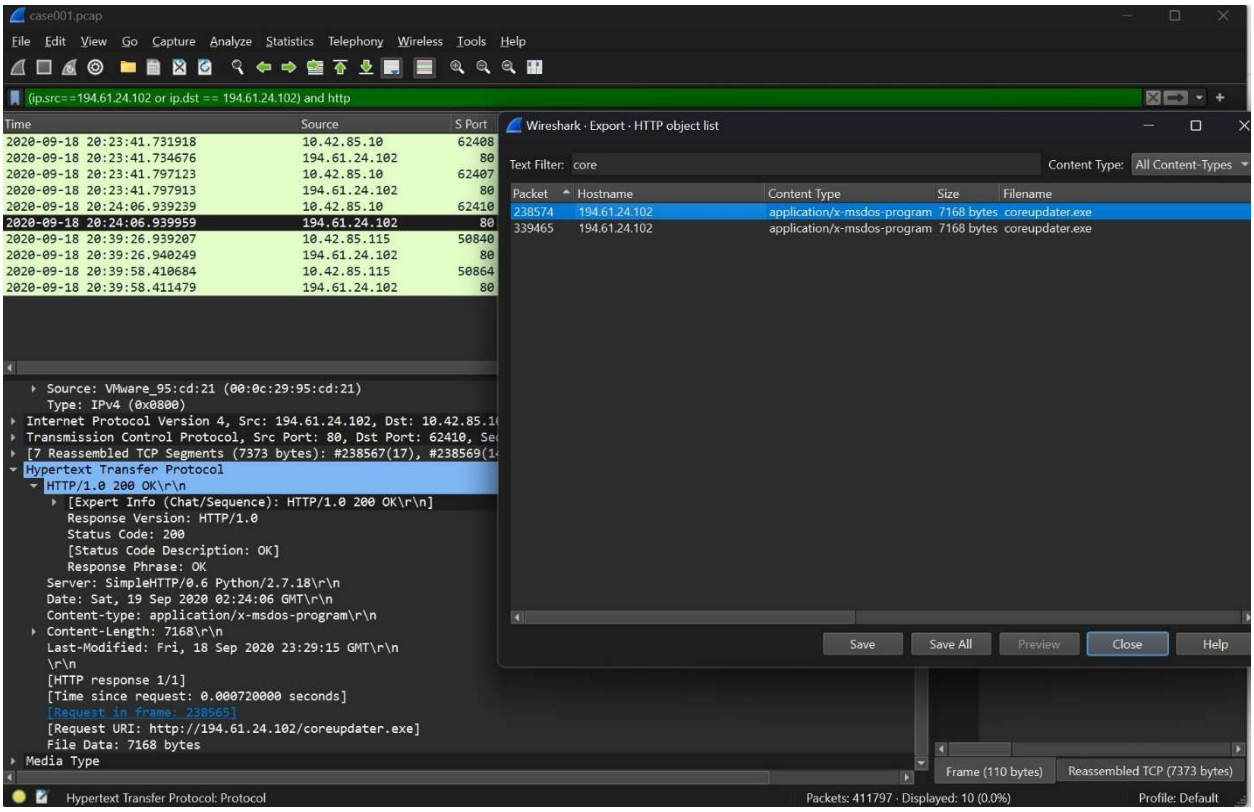
By further filtering out RDP traffic in Wireshark, the presence **of Cookie: mstshash=Administrator** indicates an attempt to connect to the target using the **Administrator** account. This suggests the client is specifically targeting our Domain Controller (DC) and trying to log in with elevated privileges. Given the suspicious nature of the activity, including the large number of pings and brute-force patterns, it's likely that the attacker is attempting to gain unauthorized access to the DC using the Administrator account.



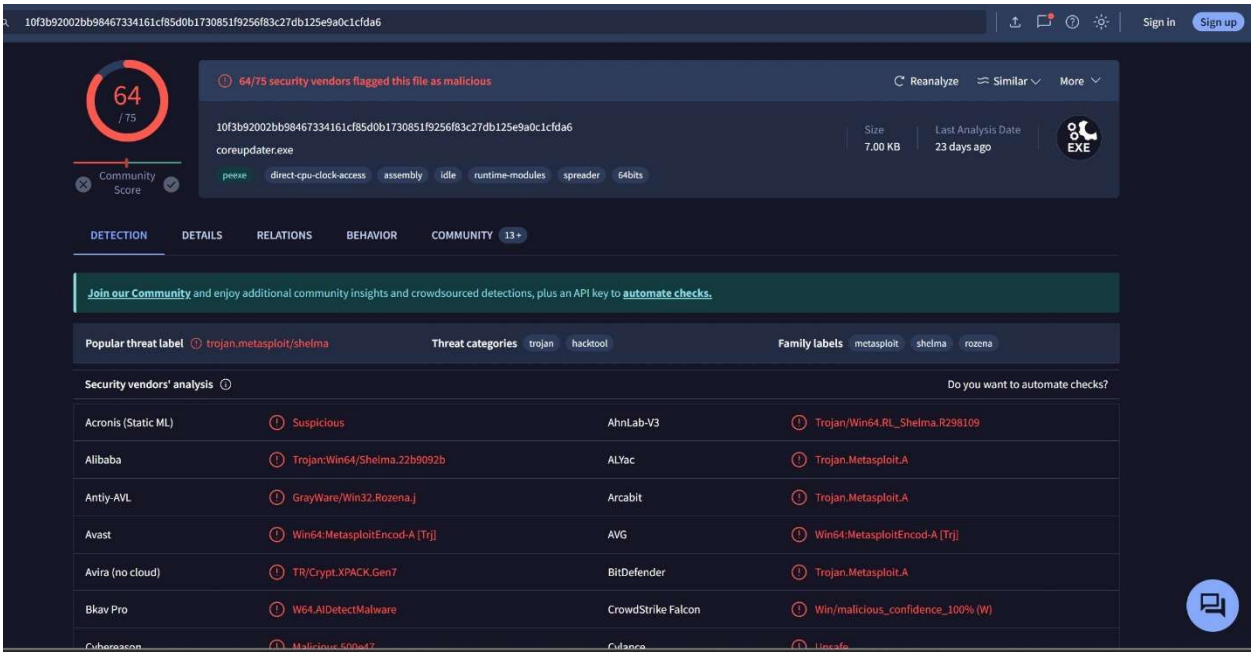
Was malware used? If so, what was it? Yes, coreupdater.exe

While reviewing the PCAP file, I examined all traffic associated with the suspect IP to understand the activity beyond the initial access. I discovered several GET requests from the IP address 194.64.24.102 to a suspicious URI, "coreupdater.exe." Both the Domain Controller (IP 10.45.85.10) and the Workstation desktop (IP 10.45.85.115) sent GET requests to download this file.

To obtain the downloaded file, use Wireshark to extract it by navigating to File -> Export Objects -> HTTP and save. Be careful not to execute this file, just run the hash check.



Run PowerShell in the same folder where the malware is saved and run the command GET-FileHash .\coreupdate.exe to get the sha-256 hash. Check the hash on VirusTotal proves that this is indeed a malware.



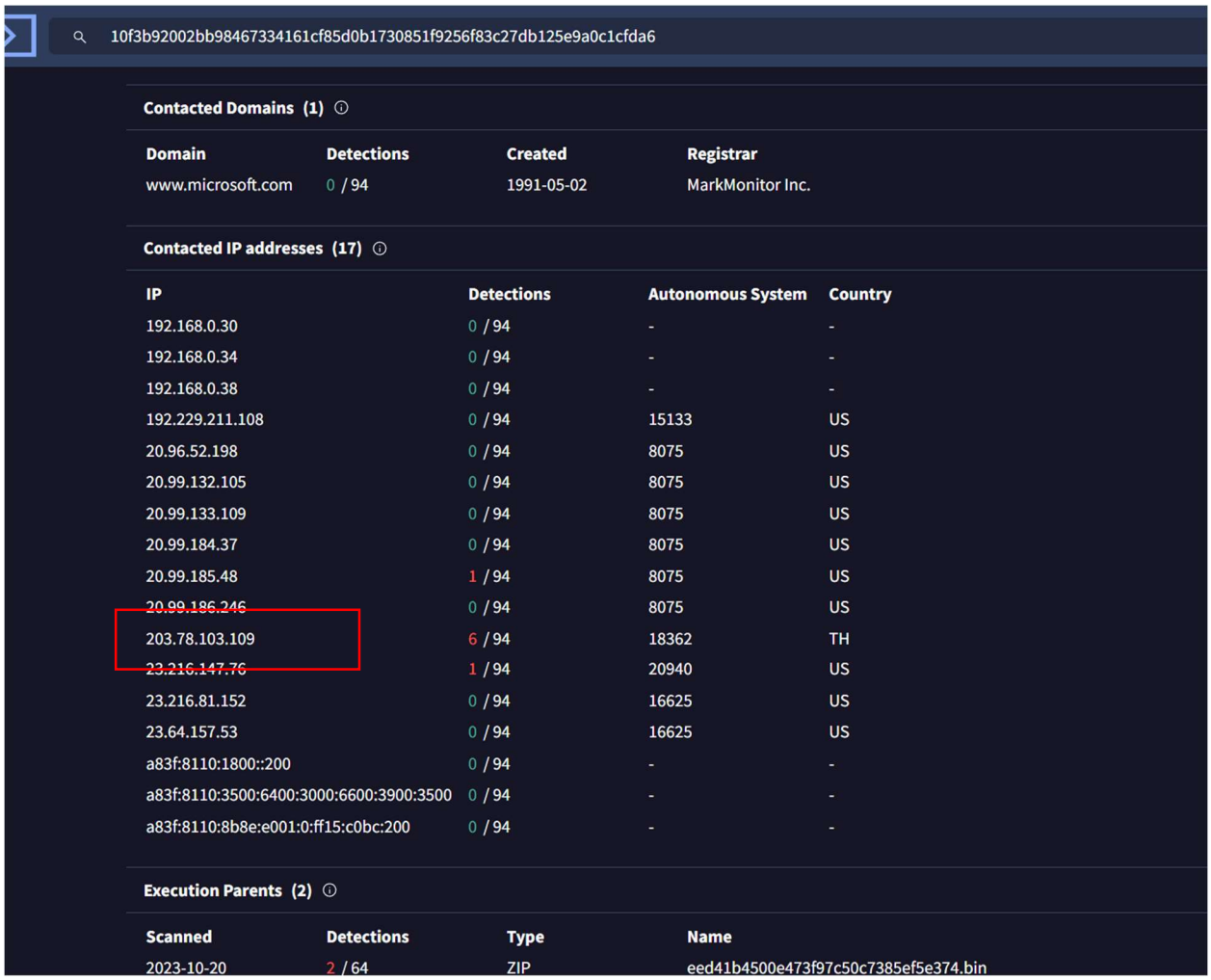
What process was malicious? coreupdater.exe

- Identify the IP Address that delivered the payload. 194.61.24.102

From the PCAP analysis, we can easily obtain this information.

- What IP Address is the malware calling to? 203.78.103.109

For this hash checked on VT, we can obtain IP addresses that are related to this malware in Contacted IP addresses.



10f3b92002bb98467334161cf85d0b1730851f9256f83c27db125e9a0c1cfda6

Contacted Domains (1)

Domain	Detections	Created	Registrar
www.microsoft.com	0 / 94	1991-05-02	MarkMonitor Inc.

Contacted IP addresses (17)

IP	Detections	Autonomous System	Country
192.168.0.30	0 / 94	-	-
192.168.0.34	0 / 94	-	-
192.168.0.38	0 / 94	-	-
192.229.211.108	0 / 94	15133	US
20.96.52.198	0 / 94	8075	US
20.99.132.105	0 / 94	8075	US
20.99.133.109	0 / 94	8075	US
20.99.184.37	0 / 94	8075	US
20.99.185.48	1 / 94	8075	US
20.99.186.246	0 / 94	8075	US
203.78.103.109	6 / 94	18362	TH
23.216.147.76	1 / 94	20940	US
23.216.81.152	0 / 94	16625	US
23.64.157.53	0 / 94	16625	US
a83f:8110:1800::200	0 / 94	-	-
a83f:8110:3500:6400:3000:6600:3900:3500	0 / 94	-	-
a83f:8110:8b8e:e001:0:ff15:c0bc:200	0 / 94	-	-

Execution Parents (2)

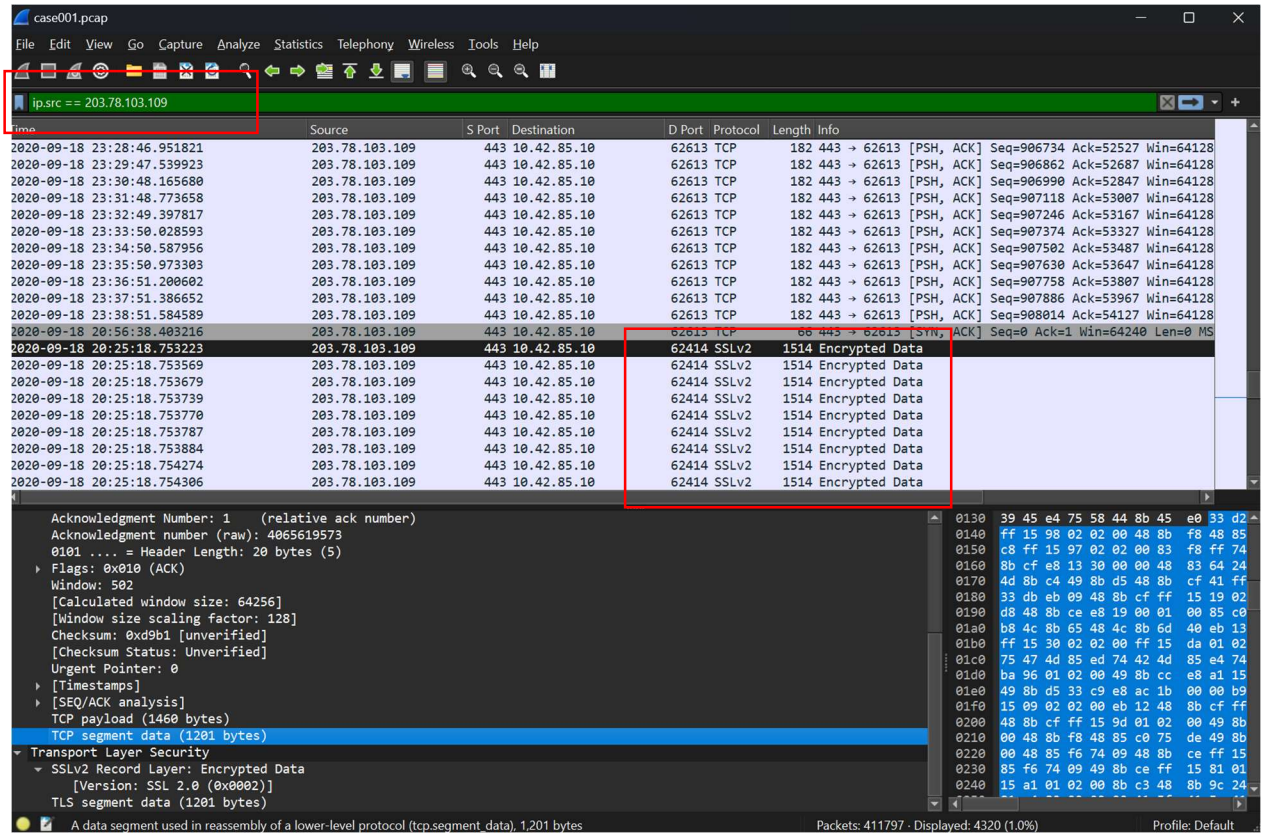
Scanned	Detections	Type	Name
2023-10-20	2 / 64	ZIP	eed41b4500e473f97c50c7385ef5e374.bin

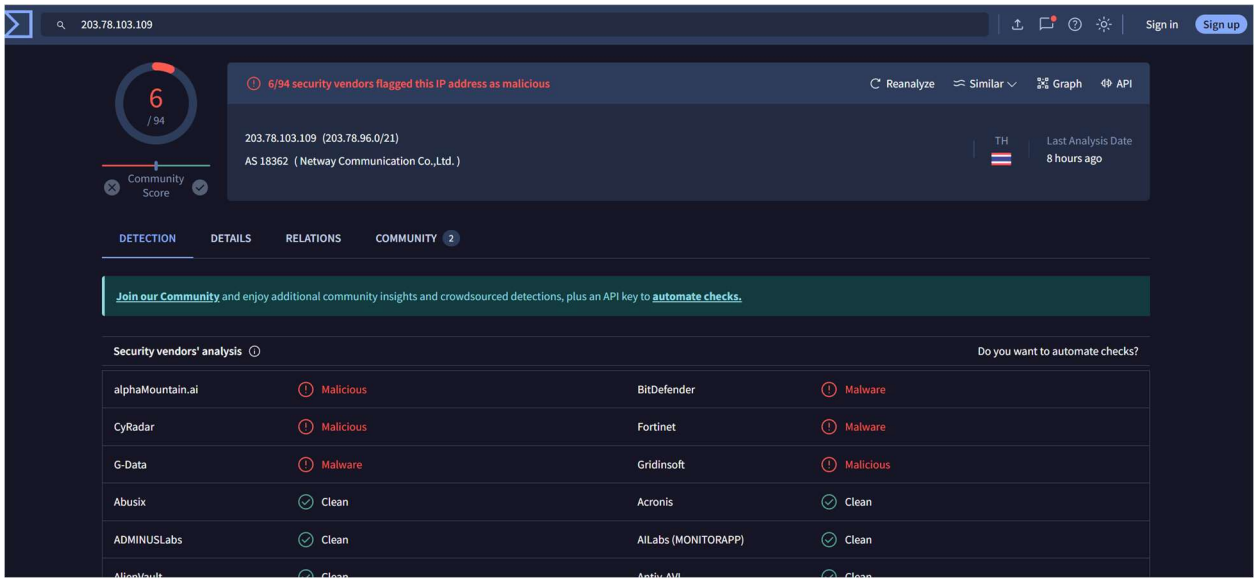
I analyzed the IP addresses in the provided PCAP and identified a match with VT: **203.78.103.109**. A further search on VT confirmed that this IP is listed in their database. Look into the traffic packet:

- **1514 bytes on wire**: This is the total size of the frame, including all headers and data.
- **1460 bytes of TCP payload**: This is the actual data transmitted at the application layer.

- **1201 bytes of TLS segment data**: This represents encrypted data within the Transport Layer Security (TLS) protocol, part of the total TCP payload.

So, the **total amount of data transmitted** in this frame is **1514 bytes**. The **TCP payload** itself (application data) is **1460 bytes**, with **1201 bytes** being specifically the TLS-encrypted portion. And as we can see from the picture below, there are multiple traffic captured from this IP to both DC and Workstation.





This is a typical download malware and post-download communication process: If the malware was indeed downloaded, the reason Wireshark captures other traffic related to IP **203.78.103.109** after the download could be that the malware is actively communicating with this IP address, which might be the attacker's command-and-control (C2) server (Novak, 2021).

This type of behavior is typical for malware:

- **Post-download communication**: After the malware is downloaded and executed, it often establishes a connection with a remote server (like **203.78.103.109**) to receive instructions, **exfiltrate data, or download additional payloads.**
- **Malware calling home**: The captured traffic may represent the malware "calling home" to the C2 server, sending or receiving data (in this case, encrypted via TLS).

Now I can answer this question: Was there a breach? Yes.

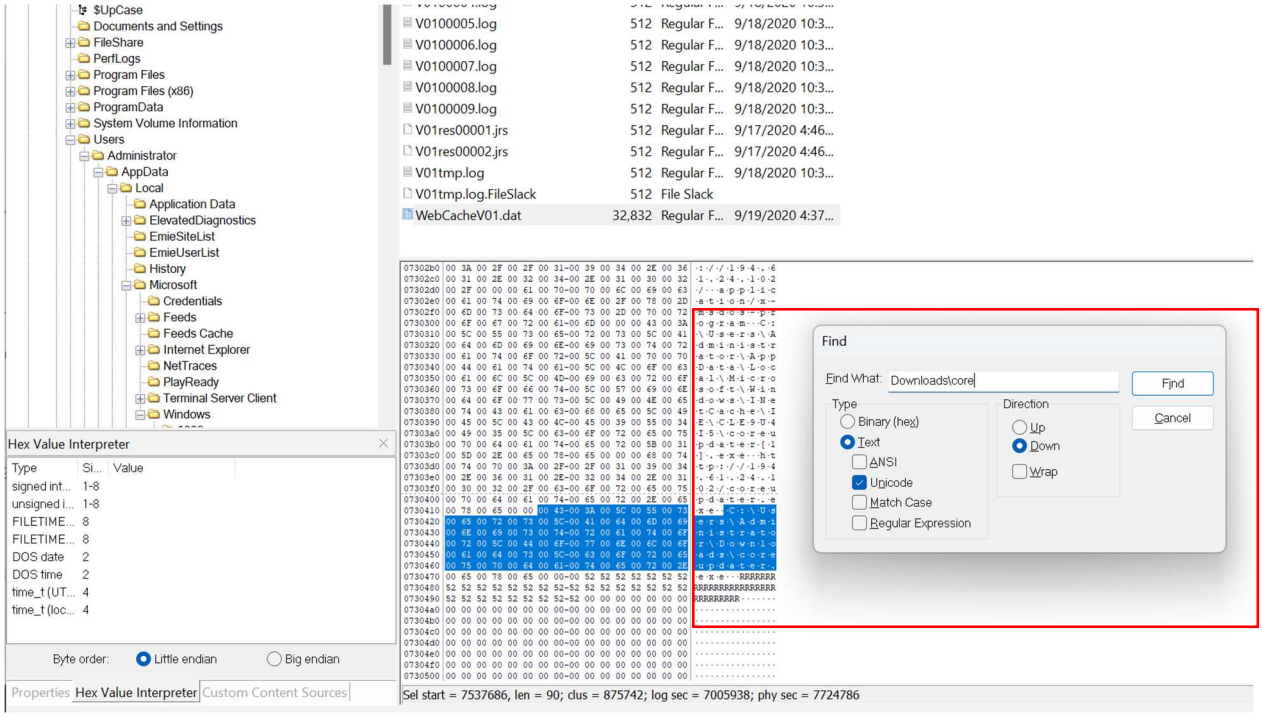
It is reasonable to conclude that the Szechuan sauce recipe has likely been exfiltrated, given the multiple instances of traffic captured between this IP and both the Domain Controller and Workstation. The use of the SSLv2 protocol suggests that the data was encrypted during transmission, likely as part of the exfiltration process.

- Where is this malware on disk? Initially in C:\Users\Administrator\Downloads, Later was moved to C:\Windows\System32\coreupdate.exe

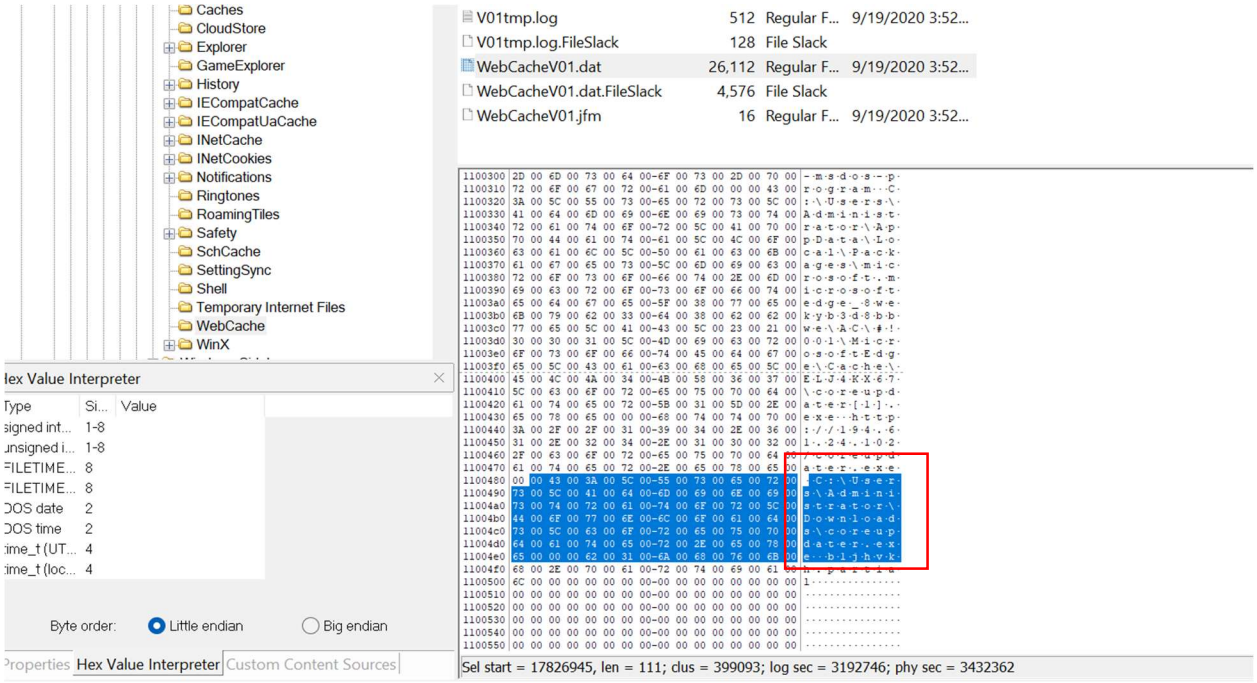
In **FTK Imager**, you can find **WebCache** files, which are related to the browser's history and cached data including downloads (PCmag, n.d.), stored in the user's profile folder on Windows systems. These files are important for web activity. The path is C:\Users\<User>\AppData\Local\Microsoft\Windows\WebCache.

Navigate to this path and parse the WebCacheV01 file. We can find that the coreupdater.exe was initially downloaded to the C:\Users\Administrator\Downloads folder which is the default folder for download.

For DC:

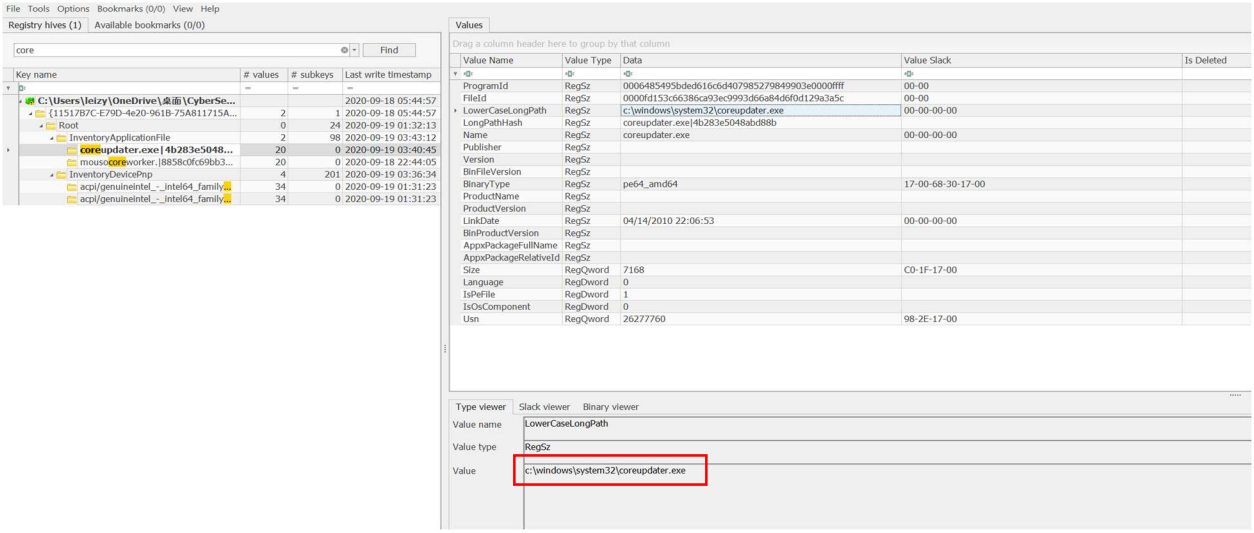


For Desktop:



Has it been moved to another folder? We need to check the **Amcache registry,** it is a registry hive in Windows that stores information about executable files and their metadata,

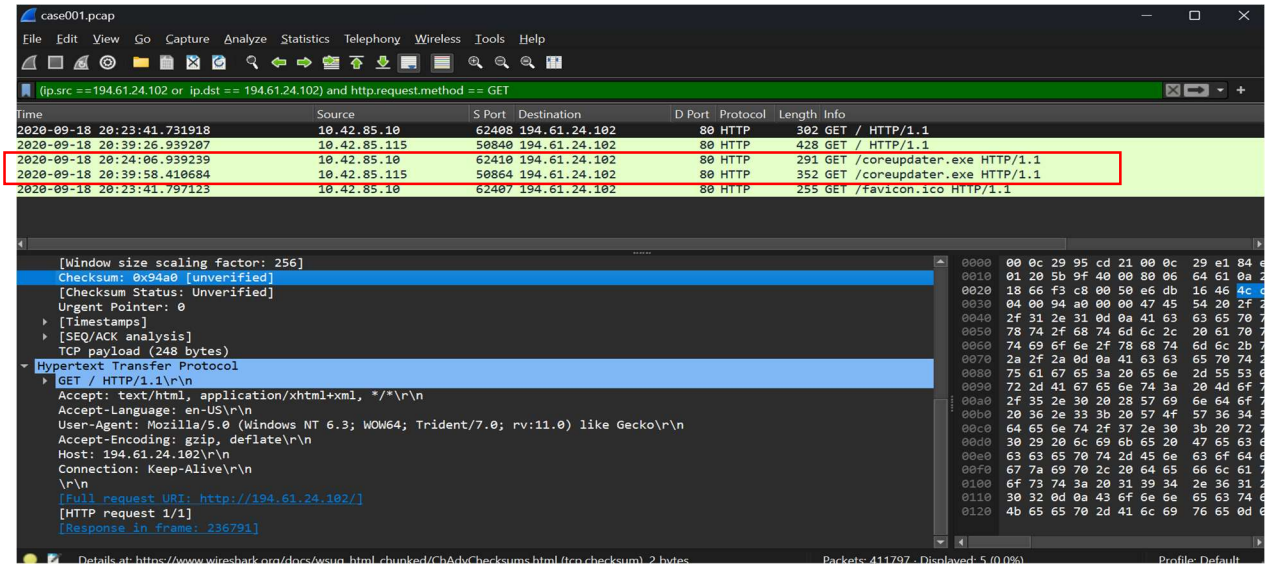
including program installation details. It's useful for forensic investigations as it helps track which programs were executed, when they were executed, and other associated details (ForenSafe, 2022). Access this registry following this path: C:\Windows\AppCompat\Programs\Amcache.hve, and parse it with EZ Registry Explorer, we can conclude that the file has been moved to C:\Windows\System32.



Fun fact: The malware was downloaded for both DC and Workstation, but only the workstation's registry file shows that the malware was moved to C:\Windows\System32 and executed.

- When did it first appear? September 18, 2020, at 20:24:06on DC

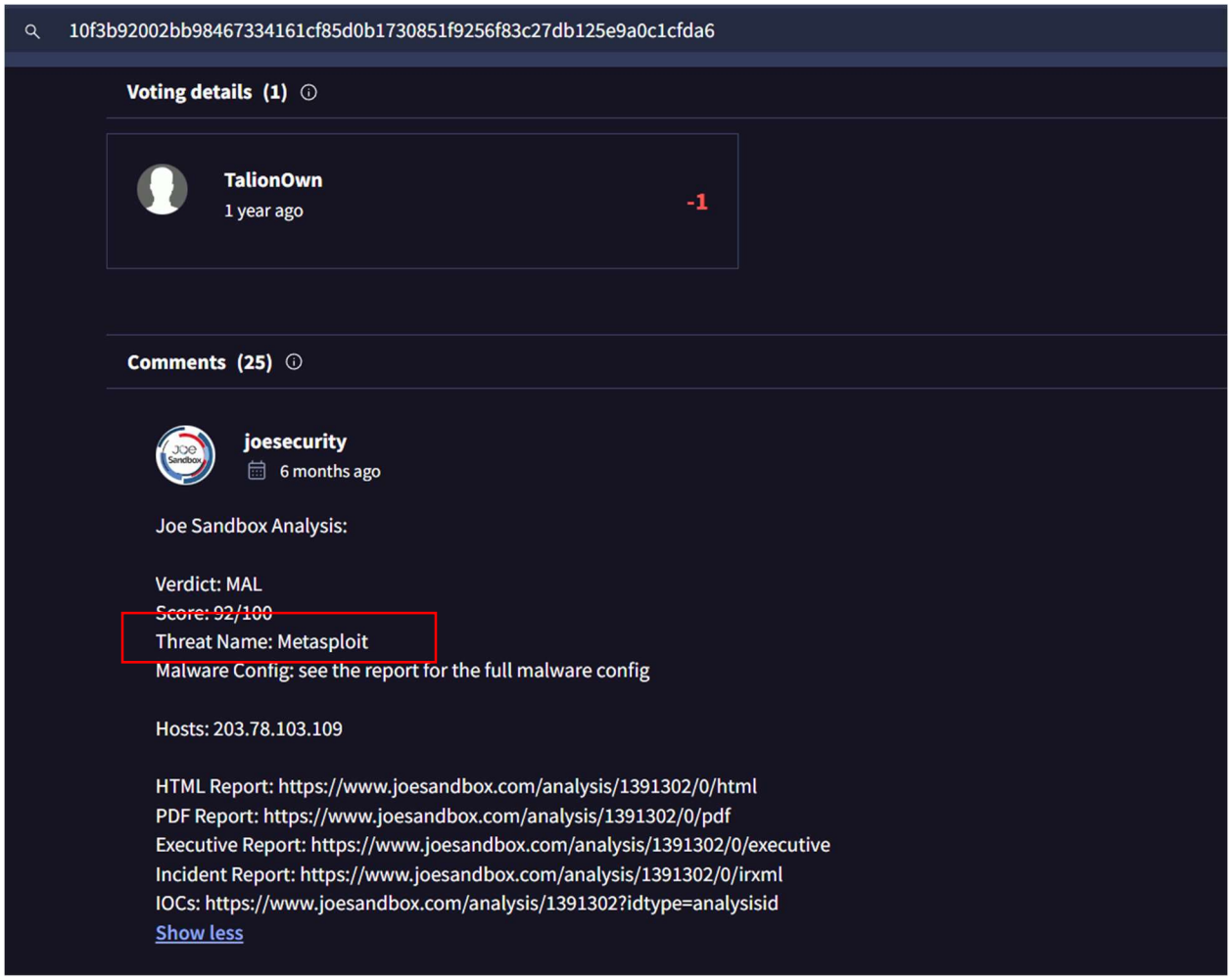
We have identified the malicious IP address and filtered out the GET requests. This packet suggests that a **GET request** was sent to **194.61.24.102** on port **80** (HTTP), which is typically used for downloading or retrieving data from a web server. The user agent making this request is posing as **Mozilla/5.0**, likely mimicking a legitimate browser. The first instance appears to target our Domain Controller (10.42.85.10) on September 18, 2020, at 20:24:06 Mountain Time.



- Did someone move it? Yes. From C:\Users\Administrator\Downloads\ to C:\Windows\System32\. I explained the process in the previous question: Where is this malware on disk?

- What were the capabilities of this malware?

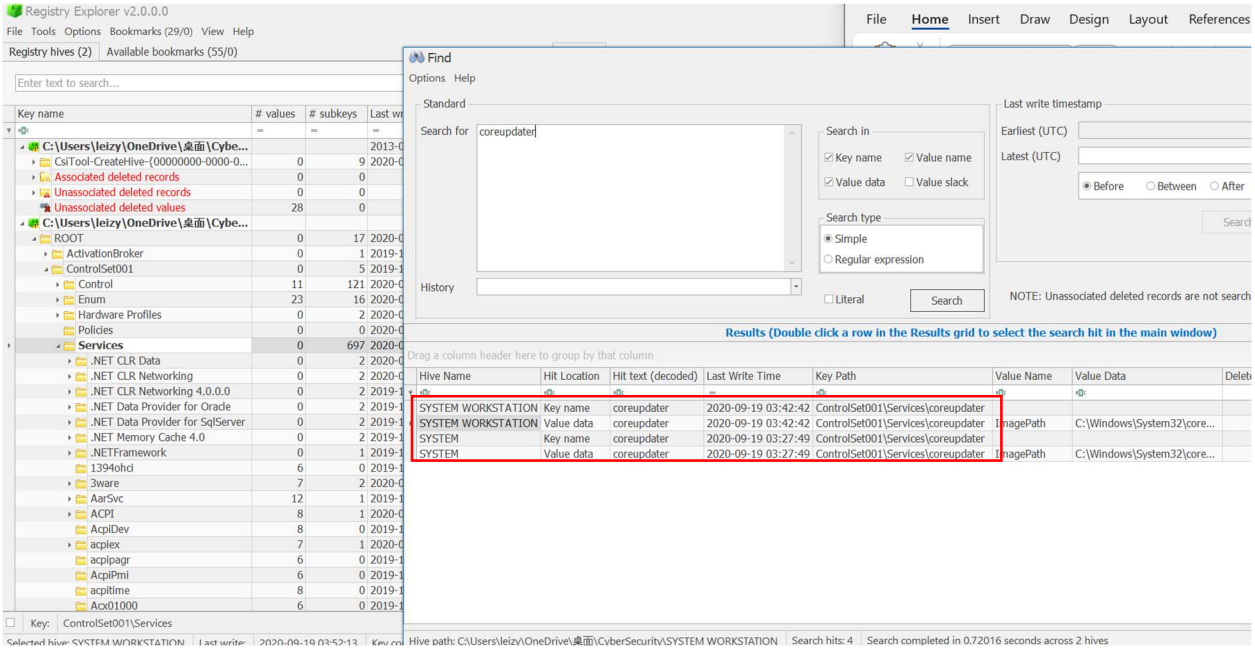
From VT we can see this malware is related to the Metasploit framework. Further research shows that This tool offers a wide range of capabilities by providing the attacker with a command-and-control (C2) shell on the victim's machine. From this point, they can further explore the compromised device, pivot to other devices within the network, or execute additional payloads and commands.



- Is this malware easily obtained? Yes. It comes with Metasploit Framework which is free to download and use (Metasploit, n.d.).

- Was this malware installed with persistence on any machine? When and where? Yes, for both inside the Registry as service, DC: 2020-09-19 03:27:49 PST. Workstation: 2020-09-19 03:42:42 PST

For this I checked the **SYSTEM** registry using EZ Registry Explorer to see if there are any "coreupdater" services installed. We can also see a registry key was created "coreupdater" on both DC and Workstation. the presence of coreupdater under the **Services** registry key likely indicates that the **coreupdater** program is installed as a service and may be set to run persistently on the system.



- What malicious IP Addresses were involved? 194.61.24.102 and 203.78.103.109

  - Were any IP Addresses from known adversary infrastructure? Both belong to adversary infrastructure

For IP address 194.61.24.102, VirusTotal shows a single appearance. In contrast, AlienVault lists 31 pulses for this IP, with tags indicating malicious activity and brute force attempts. For IP address 203.78.103.109, it has been flagged on multiple OSINT sites, including the 2nd picture shown below from VirusTotal (Sandbox, 2024).

LevelBlue/Labs Browse Scan Endpoints Create Pulse Submit Sample API Integration All Search O

IPV4

194.61.24.102 Add to Pulse +

Pulses	Passive DNS	URLs	Files
31	0	0	0

Analysis Overview

Location	Russian Federation
ASN	ASNone
Related Pulses	OTX User-Created Pulses (31)
Related Tags	9 Related Tags Nextray, cyber security, ioc, phishing, malicious, brute force, ssh, zmap, scan Less

External Resources Whois, VirusTotal

Analysis | Related Pulses | Comments (0)

User Created (31)

IOC Records Provided by @NextRayAI

IPv4 Indicator Inactive

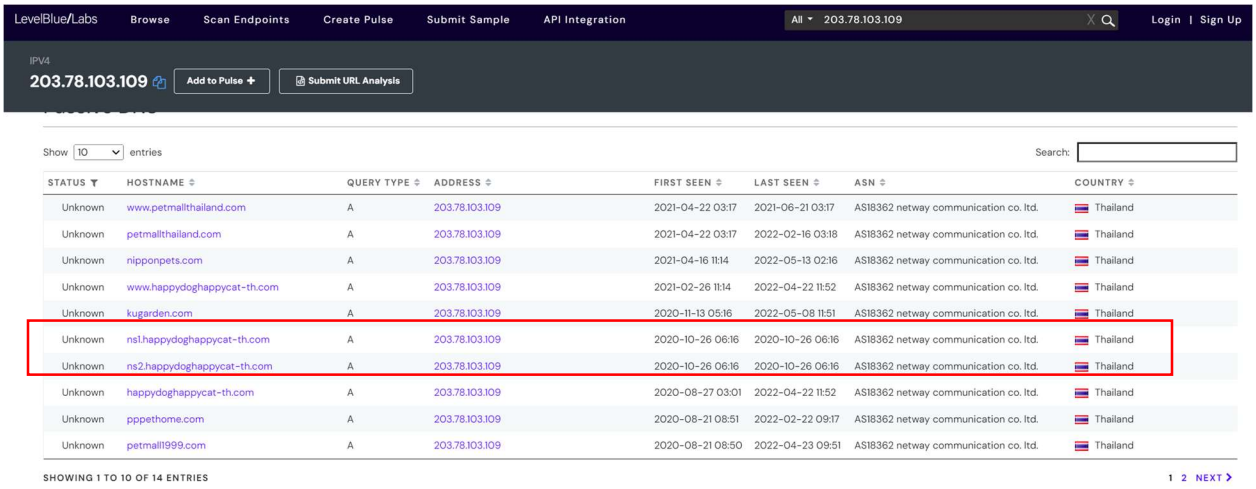
CREATED 2 YEARS AGO | MODIFIED 1 MONTH AGO by NextRay-AI | Public | TLP: White

Public IPs

IP	Domain	Country	Flag	ASN	ASN Name	Malicious
203.78.103.109	unknown	Thailand		18362	NETWAY-AS-APNetwayCommunicationCoLtdTH	true

- Are these pieces of adversary infrastructure involved in other attacks around the time of the attack? Yes

According to AlienVault's data, on October 26, 2020, there were two additional attacks associated with the IP address 203.78.103.109, linked to the DNS hostname happydoghappycat-th.com.

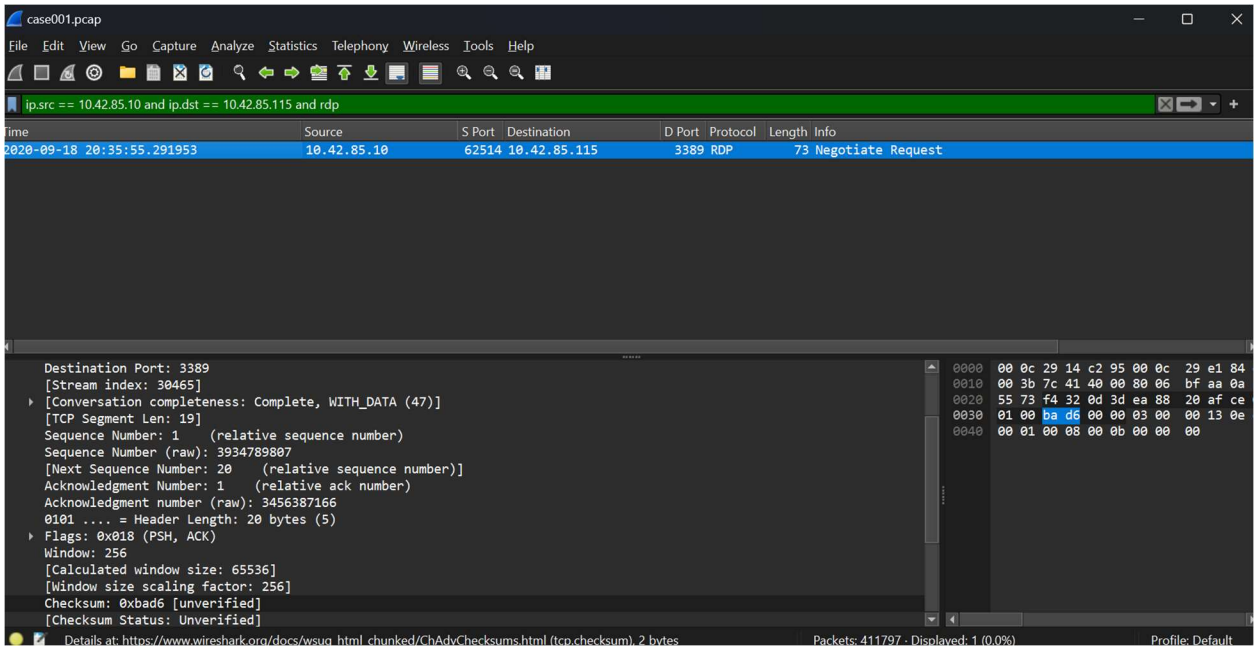


- Did the attacker access any other systems? YES
  - How?

    The attacker accessed the Workstation desktop machine, Desktop-SDN1RPT, via RDP. They used a brute-force attack to crack the Administrator account password on the Domain Controller (DC). After gaining access to the DC, they established a RDP session from the DC to the desktop machine, see picture below.

  - When? 2020-9-18 at 20:35:55 Mountain Time

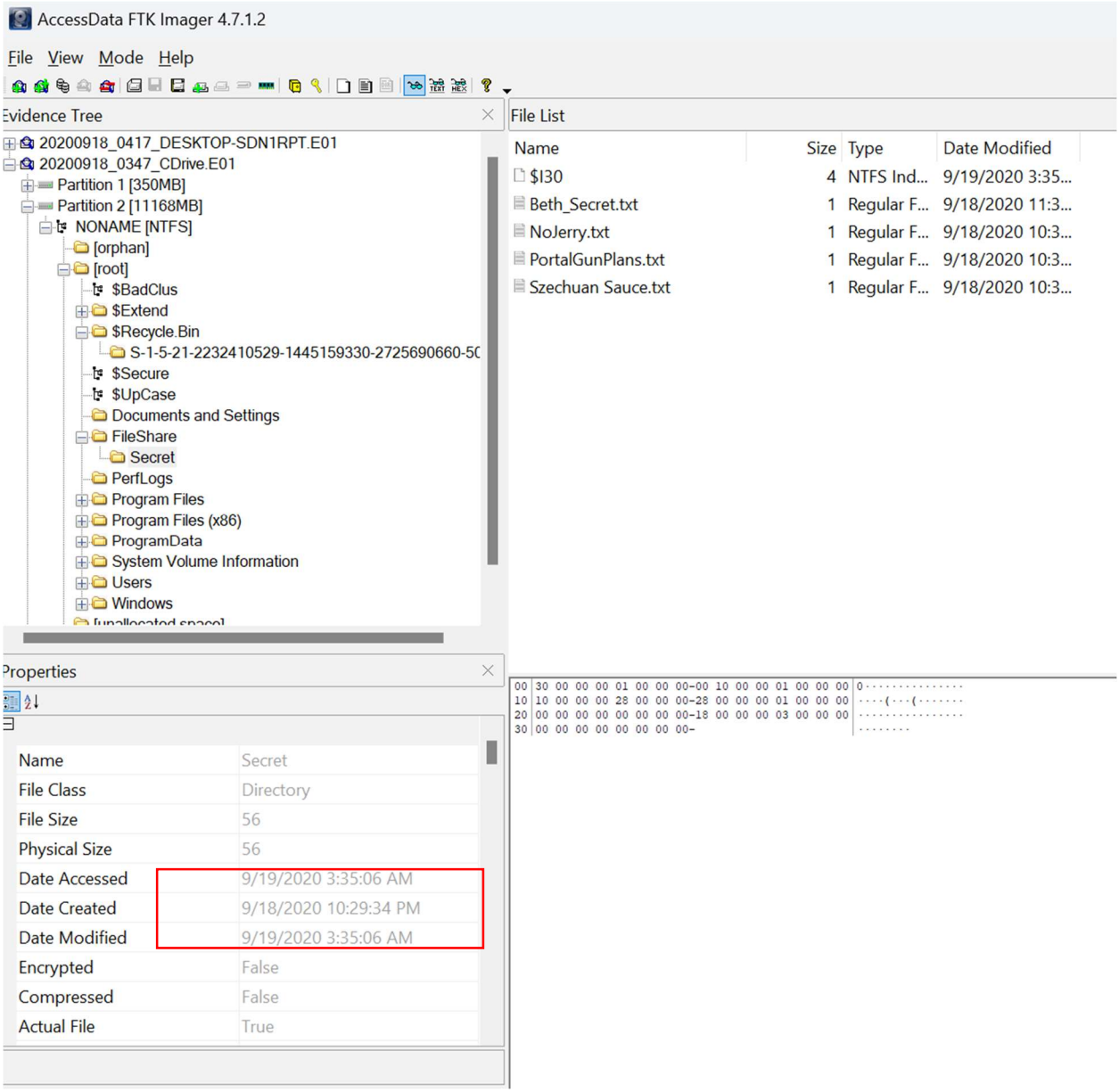
    The PCAP reveals that the RDP session from the Domain Controller to the workstation began on September 18, 2020, at 20:35:55 Mountain Time.



- Did the attacker steal or access any data? When?

  Yes, at 3:35:06 AM PST from the DC. At 3:47:11 AM PST from Workstation. The following data has been accessed and stolen:

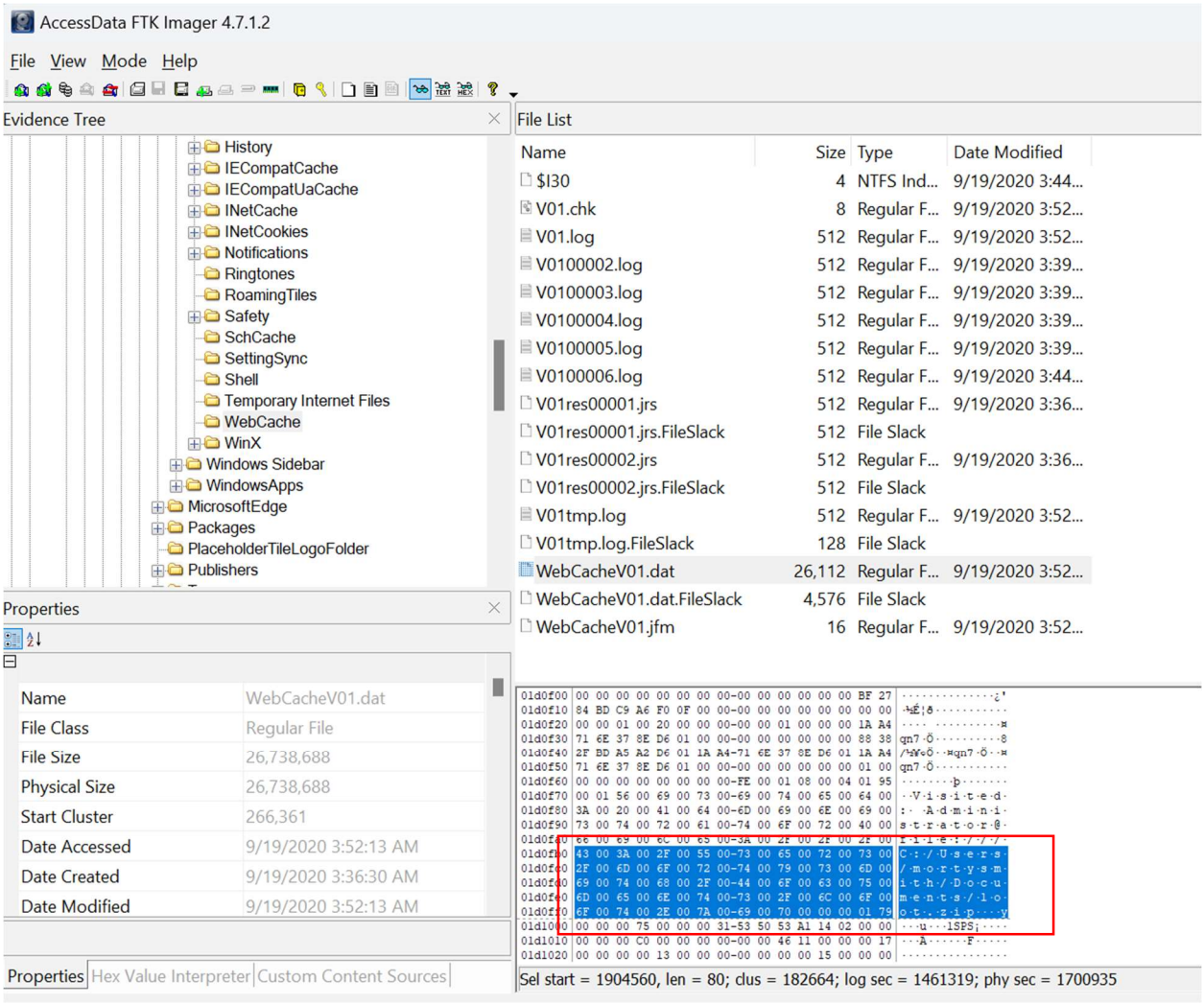
  - Beth_Secret.txt
  - NoJerry.txt
  - Portal_gun.png
  - PortalGunPlans.txt
  - Szechuan Sauce.txt
  - Plans.txt
  - My Social Security Number.txt
  - Thoughts.txt



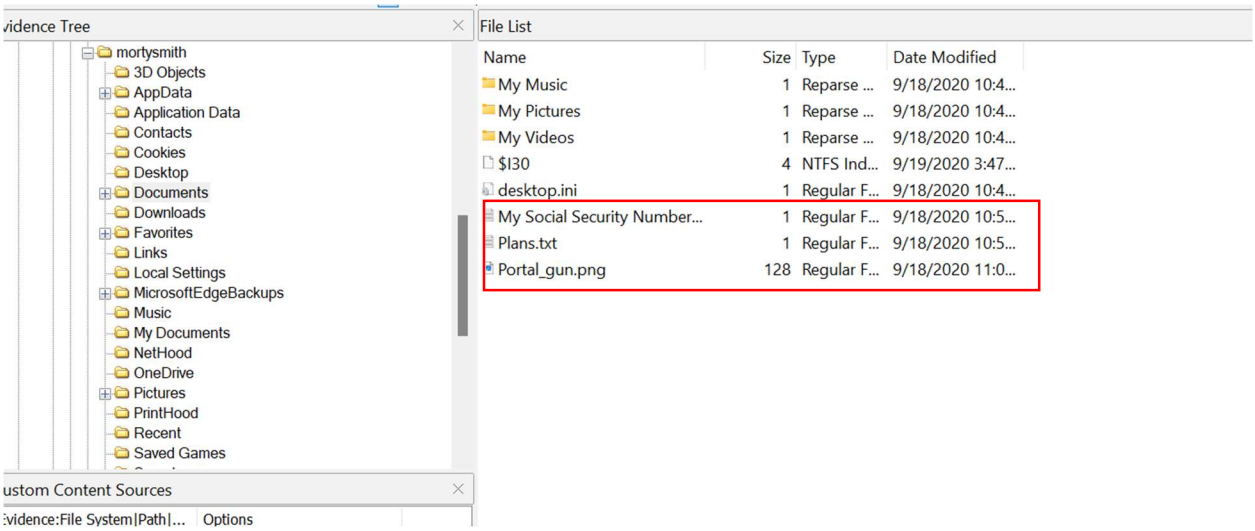
The entire secret folder on the DC was accessed at 3:35:06 AM PST. At this point, we can only confirm that the data was accessed. To determine if it was stolen, further investigation is required.

During data exfiltration, it's typical for a malicious actor to create a staging zip file to consolidate stolen data, enabling a quicker exfiltration process. After searching the **MFT files** (IDERA, n.d.) from the disk images of both the DC and the workstation, I found two files: loot.zip on the workstation and secret.zip on the DC. Given that the DC has a "secret" folder in its FileShare folder, I suspect that secret.zip contains all the exfiltrated data from DC, loot.zip contains all the exfiltrated data from the workstation.

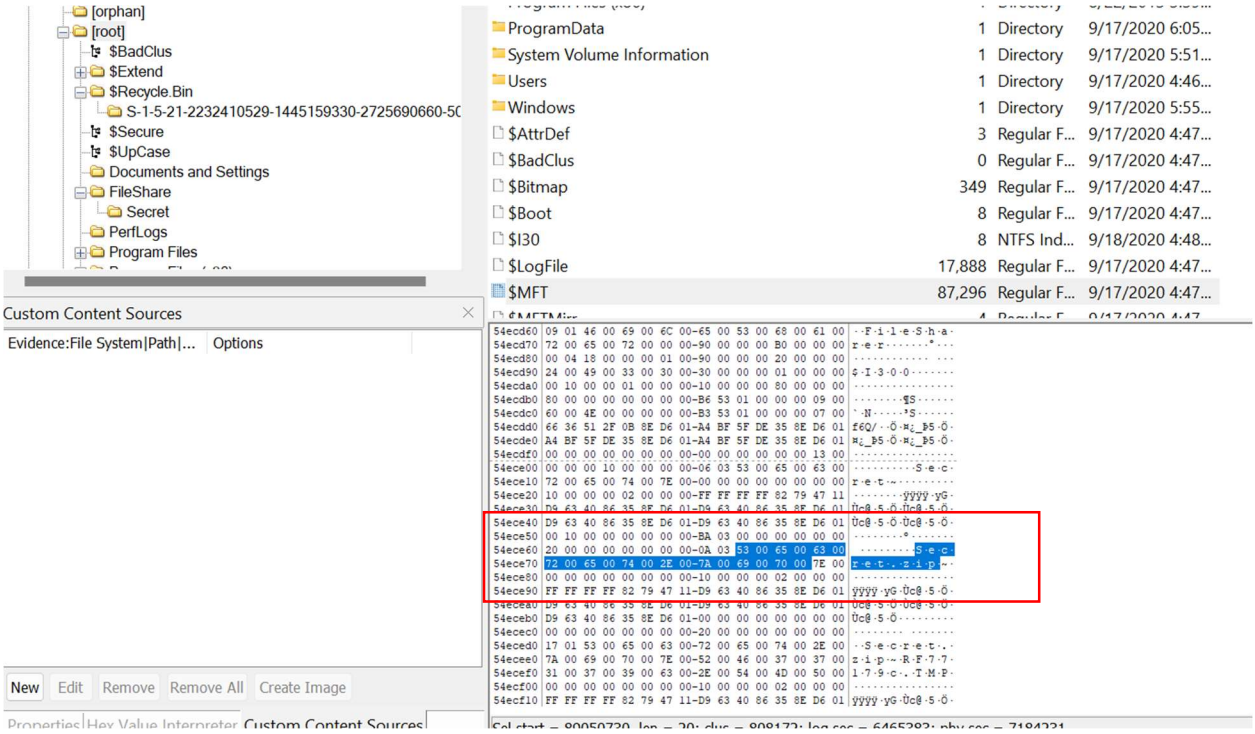
Further investigation is needed on **WebCacheV01.dat file (**I mentioned this file earlier in this report**) and MFT file,** The Master File Table (MFT) is a crucial component of the NTFS (New Technology File System) used by Windows. It contains metadata about every file and directory on an NTFS volume, acting like an index or database that keeps track of important information about files (IDERA, n.d.).



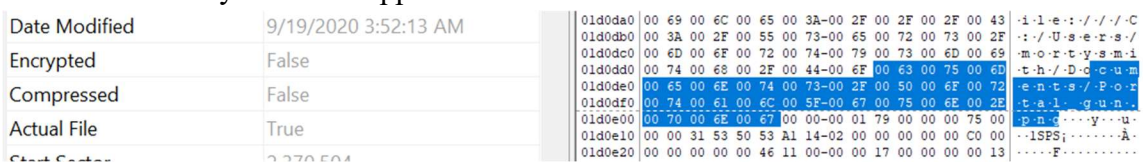
In the image above, we see the WebCacheV01 from the workstation's disk image, which not only reveals the presence of loot.zip but also pinpoints its location at **C:/Users/mortysmith/Documents.** This path requires further investigation, and unsurprisingly, I discovered additional files that were exfiltrated from Morty Smith's workstation to the DC. These documents are: Portal_gun.png, My Social Security Number.txt, Plans.txt and Thought.txt Please refer to the image below. Upon further investigation, I found a file named Thought.txt on Morty's desktop folder. Although it was not located in the same folder as the other files, it was still listed in WebCacheV01.dat, indicating that this file was also accessed.

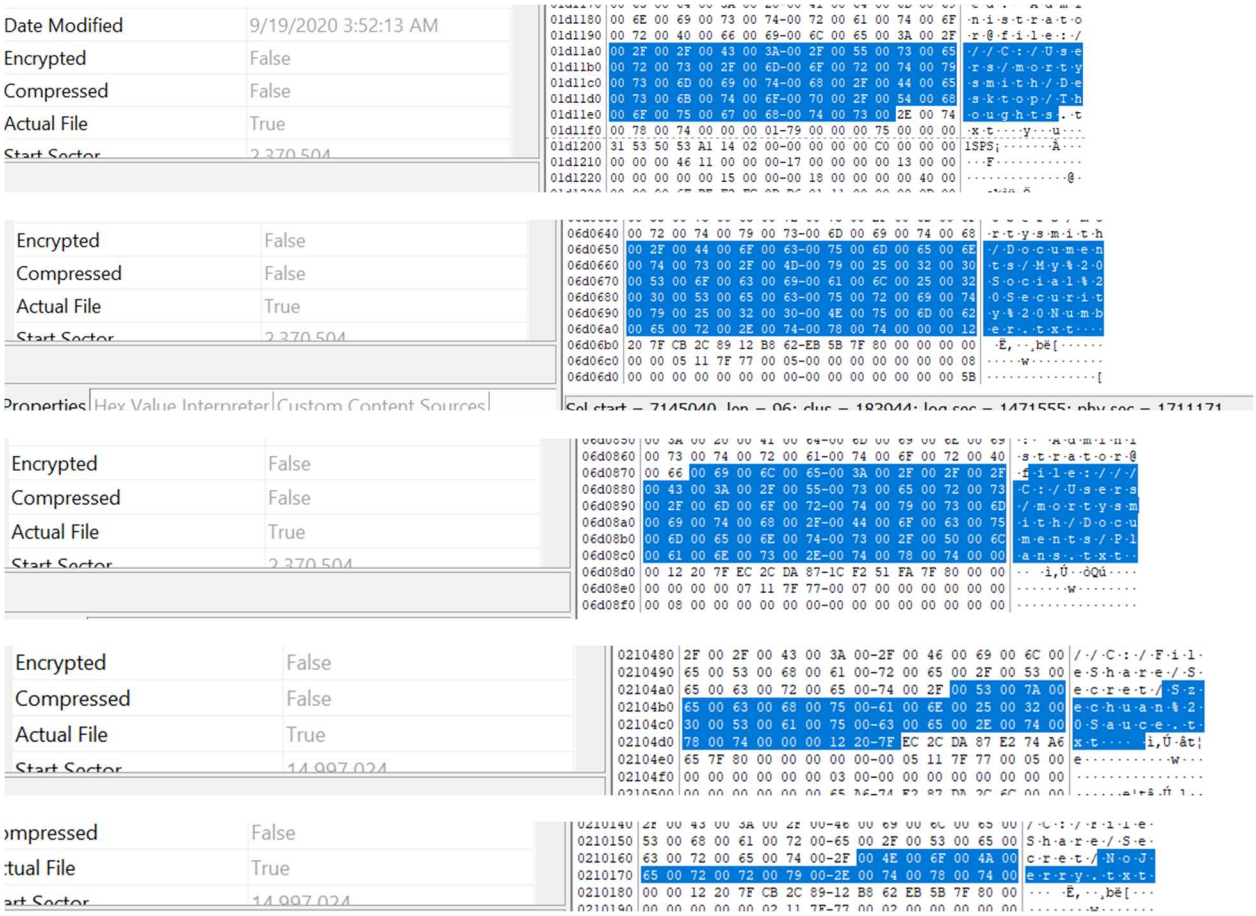


The MFT file on the DC reveals the existence of secret.zip. The attacker successfully compressed the data from both the DC (secret.zip) and the workstation (loot.zip), then exfiltrated both files.



The following images are extracted from the WebCacheV01.dat file on both the DC and workstation. They show the appearance of the accessed and exfiltrated files.

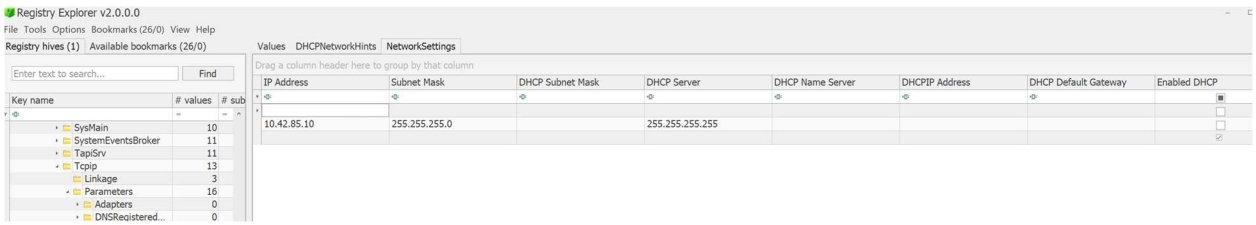




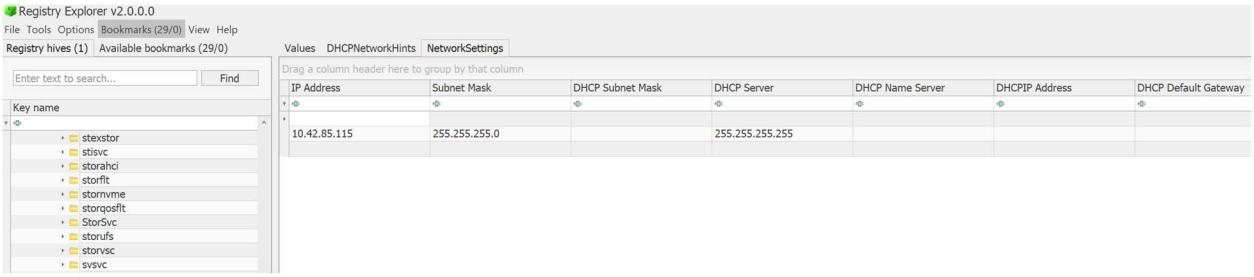
- What was the network layout of the victim network? Two machines in the subnet 10.42.85.0/24. Windows Server DC 10.42.85.10 and workstation 10.42.85.115

To find network configuration details such as IP addresses and DNS servers from a disk image, you can examine the Windows registry key located at (Microsoft, n.d.): HKLM\SYSTEM32\CurrentControlSet01\Services\Tcpip\Parameters

We use EZ Registry Explorer, follow the path inside interfaces we know our DC is 10.42.85.10 inside Subnet with a Mask 255.255.255.0, we can conclude that the Subnet is 10.42.85.0/24

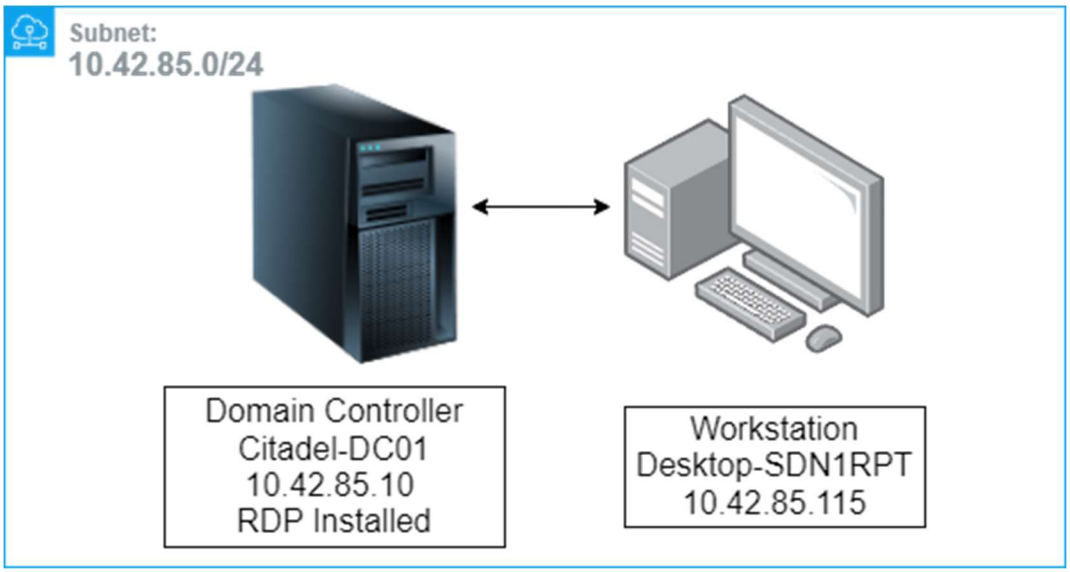


We used the same approach to check our desktop's SYSTEM registry, which revealed the IP address 10.42.85.115 with the same subnet mask. This confirms that within this subnet, there are two hosts: one is our DC, and the other is the workstation desktop.



This network design is problematic. Placing the DC on the same subnet as workstations compromises security, as it allows potential attackers who gain access to a workstation to move laterally and target other system within the subnet.

Topology:



In one of my blog posts, I outlined the best practices for network and system administration, including the importance of a well-structured network layout:

https://github.com/Danieldzy/Windows-Server-Active-Directory-management

Additional investigation

- What architectural changes should be made immediately?

  Given that initial access was gained through an RDP brute force attack on the DC:

  - Secure RDP Access: Place RDP behind a carefully configured VPN.
  - Network Isolation: Ensure the DC is accessible only within an isolated network segment and not exposed to the public Internet.
  - Disable External RDP: Turn off RDP access from the Internet to prevent unauthorized external connections.

- Did the attacker steal the Szechuan sauce? If so, what time?

  Yes, at 3:35:06 AM PST from the DC, the secret folder was access, Szechuan sauce Recipe and other documents was stolen.

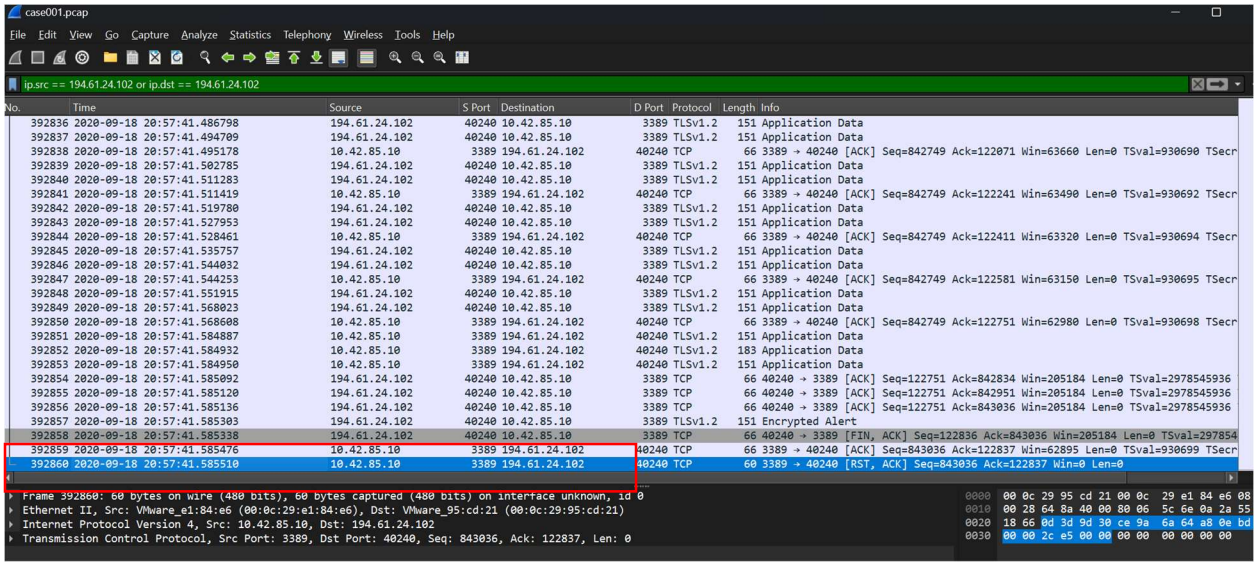
- Did the attacker steal or access any other sensitive files? If so, what times?

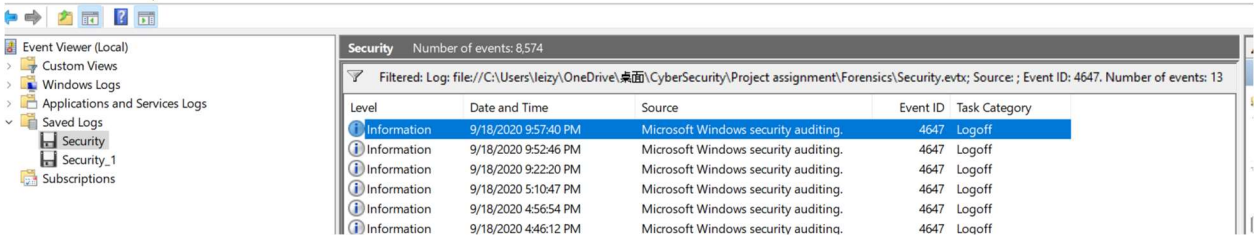
  Yes, as I explained previously: at 3:35:06 AM PST from the DC. At 3:47:11 AM PST from Workstation. The following data has been accessed and stolen:

  - Beth_Secret.txt
  - NoJerry.txt
  - Portal_gun.png (Morty Workstation)
  - PortalGunPlans.txt
  - Plans.txt (Morty Workstation)
  - My Social Security Number.txt (Morty Workstation)
  - Thoughts.txt (Morty Workstation)

- Finally, when was the last known contact with the adversary?

  The PCAP file indicates that the last known connection with the malicious IP was established on 9/18/2020 at 20:57:41 Mountain Time. This is further confirmed by the exported Windows Security event log from the Domain Controller, showing the last successful logoff under Event ID 4647 (Microsoft, 4647(S): User initiated logoff., 2021).





Advanced and Bonus Questions:

- What CIS Top 20 or SANS Top 20 Controls would have directly prevented this breach?

  - CIS Control 8: Malware Defenses (Tenable, CIS Control 8, n.d.)
  - CIS Control 9: Limitation and Control of Network Ports, Protocols, and Services (Tenable, CIS Control 9, n.d.)

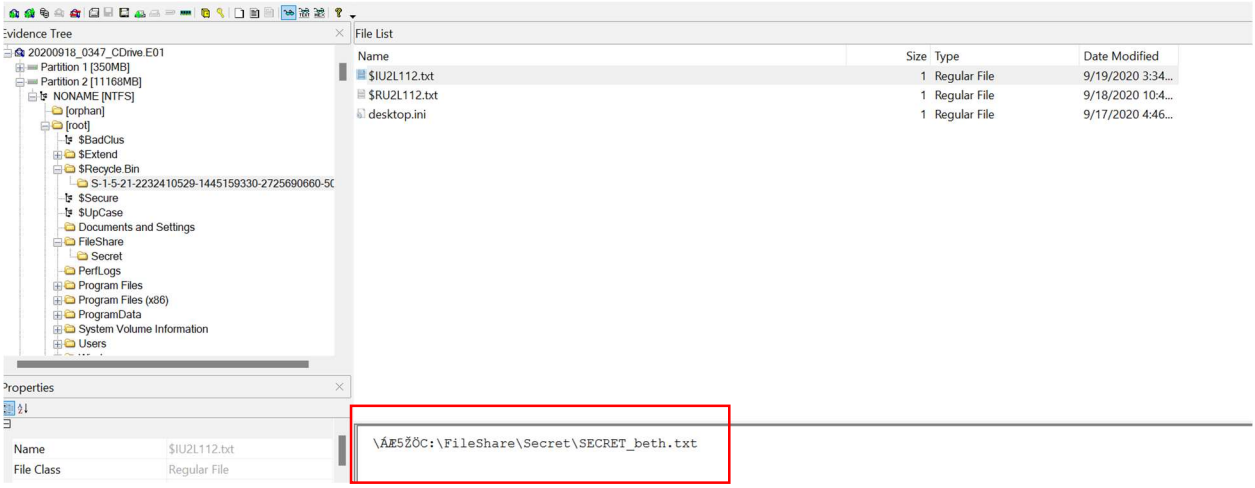
- What major architecture improvement could be made that would have prevented this breach?

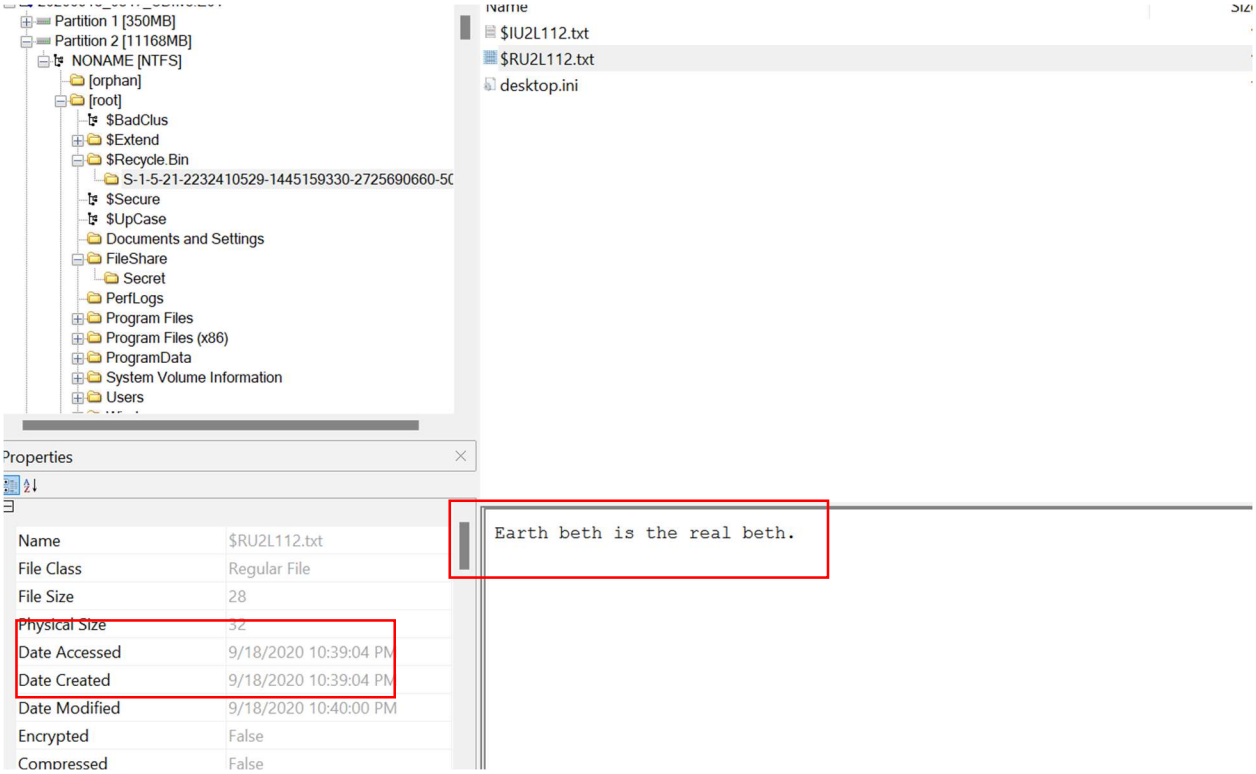
  - Implement VPN Access: Ensure all remote access, including RDP to the Domain Controller, is securely routed through a VPN to prevent unauthorized external access.
  - Deploy IPS: Install an Intrusion Prevention System (IPS) to monitor and block suspicious activities, such as brute force attacks, on the network.
  - Enforce Network Segmentation: Segregate critical systems like Domain Controllers into isolated subnets to limit exposure and reduce the risk of lateral movement in case of a breach.

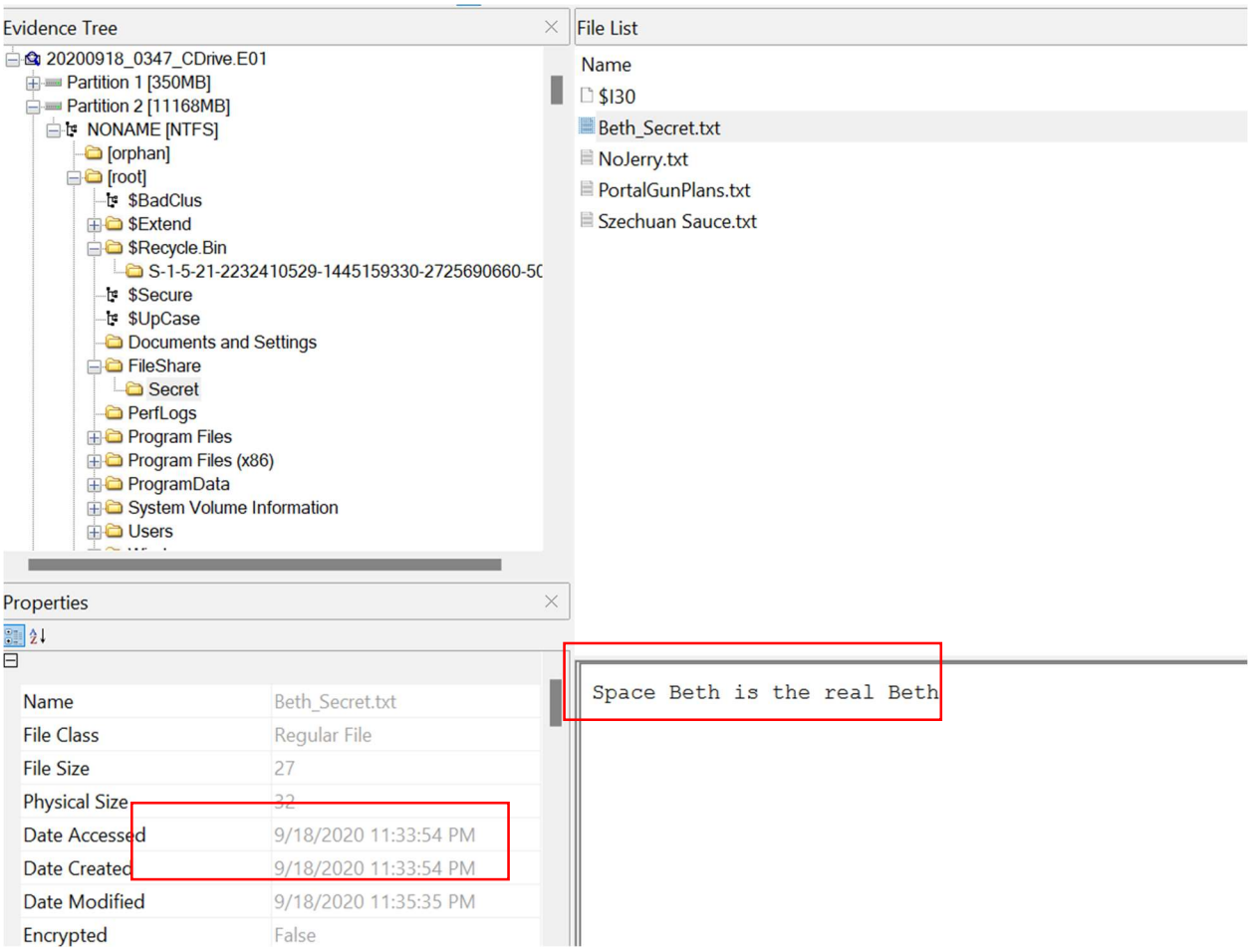
- Can you recover the original file about Beth's Secrets?

  - What was the original name? Secret_Beth.txt
  - Original Contents? Earth Beth is the real Beth.

Based on the image below, we can confirm that in the recycle bin of the DC disk image, the original file, **Secret_Beth.txt**, was modified. The initial content stated: *"Earth Beth is the real Beth."* After the incident, the file was removed and a new file named **Beth_secret.txt** was created, and the content was altered to: *"Space Beth is the real Beth."*







- What file was time stomped? Secret_beth.txt

We can see from the image above, original file was removed to recycle bin, the modified the file was created at 11:33:54 PST.

Summary:

Our digital forensics investigation into the Stolen Szechuan Sauce revealed valuable details about the attack methods and the consequences for the Asian Fusion Wok's IT infrastructure and sensitive files. The investigation confirmed that the attacker successfully breached the network via a brute force attack targeting the Remote Desktop Protocol (RDP) service. This unauthorized access enabled the attacker to move laterally to the desktop machine after compromising the server.

We recommend the following enhancements to strengthen AFW's IT infrastructure, mitigate future threats, and support ongoing business growth.

- **Secure RDP Access**: Ensure RDP connections are routed through a well-configured VPN and enforce multi-factor authentication (MFA) for added security. Regularly review and update VPN settings and user permissions to safeguard access. Disable RDP access from the public Internet entirely
- **Network Isolation**: Place the Domain Controller (DC) in a dedicated, isolated network segment, accessible only to essential internal systems. Use firewalls and access control lists (ACLs) to manage traffic and prevent public Internet exposure.
- **Network Monitoring**: Implement comprehensive network monitoring solutions to detect and analyze unusual activity. Utilize Intrusion Prevention Systems (IPS) and network traffic analysis tools to promptly identify and address potential threats.

References:

ForenSafe. (2022). Retrieved from AmCache: https://forensafe.com/blogs/AmCache.html

IDERA. (n.d.). Retrieved from Master File Table: https://www.idera.com/glossary/master-file-table/

Metasploit. (n.d.). Retrieved from The world's most used penetration testing framework: https://www.metasploit.com/

Microsoft. (n.d.). Retrieved from 4625(F): An account failed to log on.: https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-10/security/threat-protection/auditing/event-4625

Microsoft. (n.d.). Retrieved from TCP/IP and NBT configuration parameters for Windows XP: https://learn.microsoft.com/en-us/troubleshoot/windows-client/networking/tcpip-and-nbt-configuration-parameters

Microsoft. (2021). Retrieved from 4647(S): User initiated logoff.: https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-10/security/threat-protection/auditing/event-4647

Novak, L. G. (2021). Retrieved from Network Forensic Investigation: Identifying Malware in Network Traffic: https://lynnseygrahamnovak.medium.com/network-forensic-investigation-identifying-malware-in-network-traffic-9a6bc32116dc

PCmag. (n.d.). Retrieved from Web cache: C:\Users\<User>\AppData\Local\Microsoft\Windows\WebCache

Sandbox, J. (2024). Retrieved from Joe Sandbox: https://www.joesandbox.com/analysis/1391302/0/pdf

Tenable. (n.d.). Retrieved from CIS Control 9: https://docs.tenable.com/security-center/CIS-CAS/Content/Controls/Foundational/Control-9/Control-9.htm

Tenable. (n.d.). Retrieved from CIS Control 8: https://docs.tenable.com/security-center/CIS-CAS/Content/Controls/Foundational/Control-8/Control-8.htm