Threat Intel Roundup: BreachForum, Vidar, Git, QNAP

Week in Overview[14 May-21 May] - 2024

THREATRADAR
By HADESS

WWW.THREATRADAR.NET

Technical Summary

BreachForum, a notorious online platform for trading stolen data, has been seized by the United States Federal Bureau of Investigation (FBI). The forum's landing page now displays a notice confirming the FBI's control and shows images of the administrators' Telegram profile pictures behind bars. This seizure disrupts a key hub for cybercriminals and represents a significant step in ongoing efforts to combat cybercrime. The FBI is likely analyzing data and activities from the site to identify and prosecute individuals involved.
CVE-2024-4985: GitHub Enterprise Server Authentication Bypass

CVE-2024-4985 is an authentication bypass vulnerability in GitHub Enterprise Server (GHES), affecting versions prior to 3.13.0. The vulnerability exploits flaws in the handling of encrypted SAML claims, allowing attackers to create fake SAML assertions that GHES fails to validate correctly. This vulnerability can enable unauthorized access to GHES instances. Exploiting this involves crafting a specific SAML assertion and sending it to the GHES server. The issue has been addressed in version 3.13.0.
Hijack Loader

Hijack Loader, also known as IDAT Loader, is malware that emerged in September 2023 and has quickly become one of the most widely used loaders. The latest version includes enhanced anti-evasion techniques, such as avoiding inline API hooking, adding Windows Defender exclusions, bypassing User Account Control (UAC), and using process hollowing. This version decrypts and parses a PNG image to load its second-stage payload, which is modular and primarily focuses on injecting the main instrumentation module. Hijack Loader is currently ranked 6th on the ANY.RUN Trends Tracker and delivers payloads like Amadey, Lumma Stealer, Meta Stealer, Raccoon Stealer V2, Remcos RAT, and Rhadamanthys.

Vidar Stealer is a malware variant known for its ability to steal sensitive information from infected systems. Recently, changes have been observed in the response headers of servers associated with Vidar Stealer's infrastructure. These changes are being tracked using specific queries and tools like Censys. Vidar Stealer is often deployed to exfiltrate data such as passwords, cryptocurrency wallet contents, and other personal information.
QNAP QTS - QNAPping at the Wheel (CVE-2024-27130 and Friends)

QNAP QTS is a NAS operating system that has been found vulnerable to multiple security issues, including CVE-2024-27130, an unauthenticated stack overflow bug that allows remote code execution. Researchers discovered this vulnerability and others while analyzing the shared codebase of QTS, QuTSCloud, and QTS hero. The vulnerabilities stem from legacy code and poor security practices, such as hardcoded credentials and memory corruption issues. Patches have been released for some vulnerabilities, but others remain unaddressed.

PDF.js is an open-source JavaScript library used for rendering PDF documents in web browsers. It allows users to view and interact with PDFs directly within the browser without needing external plugins. The library is widely used for its ease of integration and functionality, enabling features like text selection, searching, and document navigation. However, it must be regularly updated to avoid potential security vulnerabilities that could be exploited by attackers.

CVE-2024-32002 is a security vulnerability discovered in PDF.js, affecting versions prior to 3.0.0. This vulnerability allows attackers to execute arbitrary code on the user's system by exploiting a flaw in the way PDF.js processes certain PDF files. An attacker can craft a malicious PDF file that, when opened, triggers the vulnerability and executes code with the same permissions as the user running the browser. Updating to PDF.js version 3.0.0 or later mitigates this risk.

Key Findings

it is crucial for organizations and individuals to prioritize remediation and patching efforts to safeguard their systems and data. The following key findings highlight the importance of proactive measures to mitigate risks associated with various vulnerabilities and threats:

- BreachForum
- CVE-2024-4985
- Hijack Loader
- Vidar
- QNAPping
- PDF.js
- CVE-2024-32002

🚨 Vulnerability of the Week

Git CVE-2024-32002

This repository contains a Proof of Concept (PoC) for CVE-2024-32002, a Remote Code Execution (RCE) vulnerability in Git submodules. The exploit demonstrates how a malicious payload can be triggered via a recursive clone of a Git repository.

Before running the PoC, create the following repositories on your remote Git server or feel free to change the names as needed:

- hulk.git
- submod.git
- smash.git

Update the repository paths in the PoC script accordingly if you change the names.

The exploit leverages Git submodules to execute a payload on the target system when the repository is cloned recursively. This PoC is tested on macOS.

HULK_REPO="git@github.com:safebuffer/hulk.git"
pullme_REPO="git@github.com:safebuffer/submod.git"

# Final Exploit Repo
SMASH_REPO="git@github.com:safebuffer/smash.git"

Triggering the Exploit
To trigger the exploit, run the poc.sh script and then execute the following command:

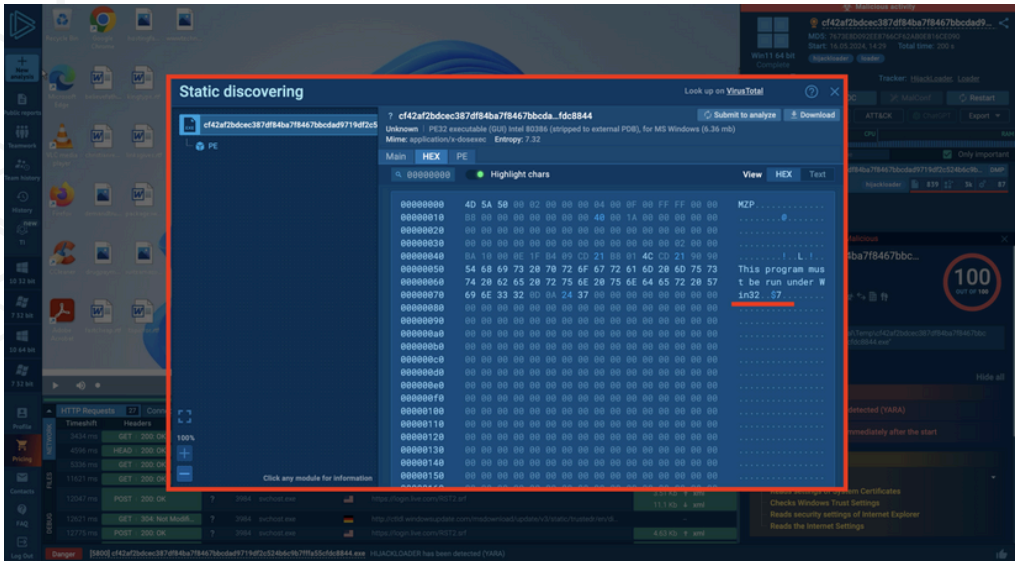
git clone --recursive $SMASH_REPO

Payload
The default payload in this PoC opens the Calculator application on macOS. You can change this to any payload of your choice.

/System/Applications/Calculator.app/Contents/MacOS/Calculator

https://x.com/safe_buffer/status/1791850146201755921
https://github.com/safebuffer/CVE-2024-32002

Art of Detection



https://x.com/anyrun_app/status/1792526734484439511

The new version of Hijack Loader employs a modular architecture, which primarily focuses on the injection of the main instrumentation module. This architecture allows the malware to load additional modules as needed, making it more flexible and adaptable to various environments.

To enhance its stealth and evade detection, the malware has adopted several advanced techniques:

- Avoiding Inline API Hooking: By avoiding this common detection method used by security software, Hijack Loader can operate more covertly.
- Exclusion for Windows Defender: The malware adds an exclusion to bypass Windows Defender antivirus, reducing the chances of being detected and removed.
- Bypassing User Account Control (UAC): This technique allows the malware to gain elevated privileges without triggering security prompts.
- Process Hollowing: This method involves injecting malicious code into legitimate processes, making it harder for security solutions to detect the malicious activity.

Security researchers identified seven new modules associated with Hijack Loader in March and April 2024, indicating ongoing development and enhancement by the threat actors.
Background on Hijack Loader
Hijack Loader, also known as IDAT Loader, emerged in September 2023 and has quickly gained popularity. It is now one of the most widely used loaders, ranking sixth in the ANY.RUN Trends Tracker. This ranking is based on the analysis of public sandbox submissions.
Common Payloads Delivered by Hijack Loader
The following are some of the common payloads delivered by Hijack Loader:

- Amadey
- Lumma Stealer
- Meta Stealer
- Raccoon Stealer V2
- Remcos RAT
- Rhadamanthys

In a specific analysis session, the second-stage payload did not download because the Command and Control (C2) server was inactive at the time. Despite this, the detection mechanisms of ANY.RUN remain robust.
Latest Hijack Loader Indicators of Compromise (IOCs)
The following are the latest IOCs for Hijack Loader, collected from the Malware Trends Tracker. These artifacts are dynamically updated with new public analysis sessions and uploads to ANY.RUN.
IPs

- 185.215.113.67
- 193.233.132.139
- 185.172.128.76

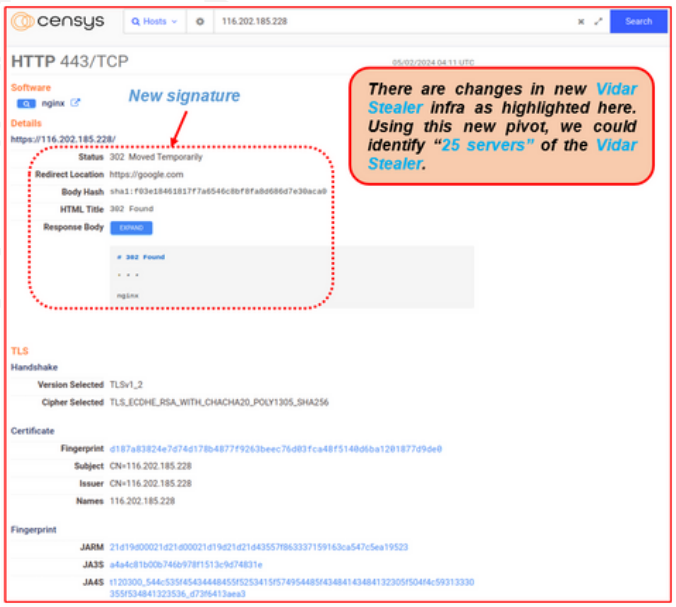
Hashes

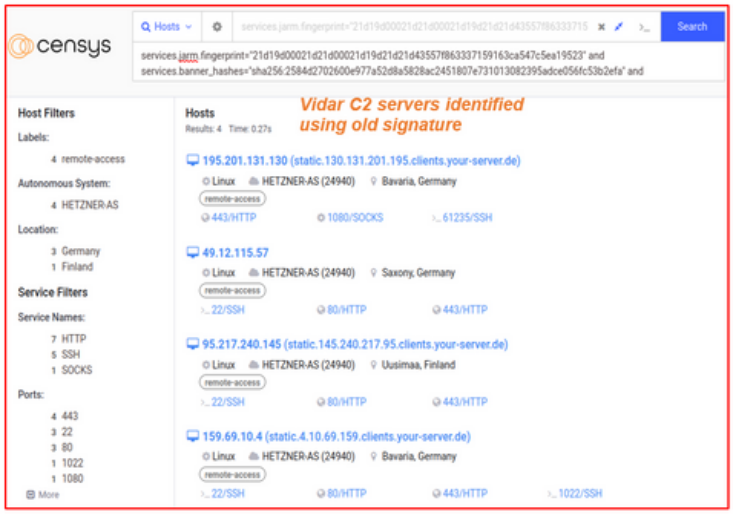
- 86BCCBACD8E9FDE23FF236155EE47F866DD7DD51C6129ED340034810A10705B3
- 0AE58BE8D7058E40926FDB51B76043D109B96B91AA9FA2950DBB8A3626185E0F
- A38DA72082FC2DC1F60B3B245E1F2382D5F8C1D08EBC397DD0D81CC9F74EBBE6

URLs

- mail.zoomfilms-cz[.]com
- discussiowardder[.]website
- wxt82[.]xyz

Malware or Ransomware





https://x.com/Cyberteam008/status/1792756439003676864

Recent investigations into the Vidar Stealer infrastructure have revealed notable changes in the response headers associated with its command and control (C2) servers. These modifications could indicate evolving tactics and techniques employed by the threat actors behind Vidar Stealer, potentially impacting detection and mitigation strategies.
Overview of Vidar Stealer

Vidar Stealer is a notorious piece of malware primarily used to exfiltrate sensitive information from infected systems. This includes passwords, credit card information, and other personal data. The malware operates by communicating with C2 servers to send collected data and receive instructions.

Our analysis, supported by data from Censys and other sources, highlights specific changes in the response headers returned by the C2 servers of Vidar Stealer. These observations are critical for understanding how the malware infrastructure is adapting to evade detection.

Changes in Response Headers
The following changes have been identified in the response headers of Vidar Stealer's C2 servers:

- Content-Type Alterations: The Content-Type header has shown variations, possibly to mimic legitimate traffic and evade content-based filtering systems. Previous headers indicated generic content types, while recent changes include more specific and varied content types.
- Server Identification: There are changes in the Server header, with some instances showing no server information and others using common web server identifiers. This inconsistency can be a tactic to confuse automated detection systems.
- Cache-Control Modifications: Adjustments in the Cache-Control header settings have been observed. These modifications might be aimed at influencing how responses are cached and thereby managing how quickly changes in the infrastructure are propagated.

Analytical Insights
The changes in response headers suggest a deliberate attempt by the operators of Vidar Stealer to adjust their infrastructure's fingerprint. This makes it harder for security tools to reliably identify and block traffic associated with the malware. Such adaptability is a common trait among sophisticated threat actors aiming to maintain persistence and evade detection.

0Day

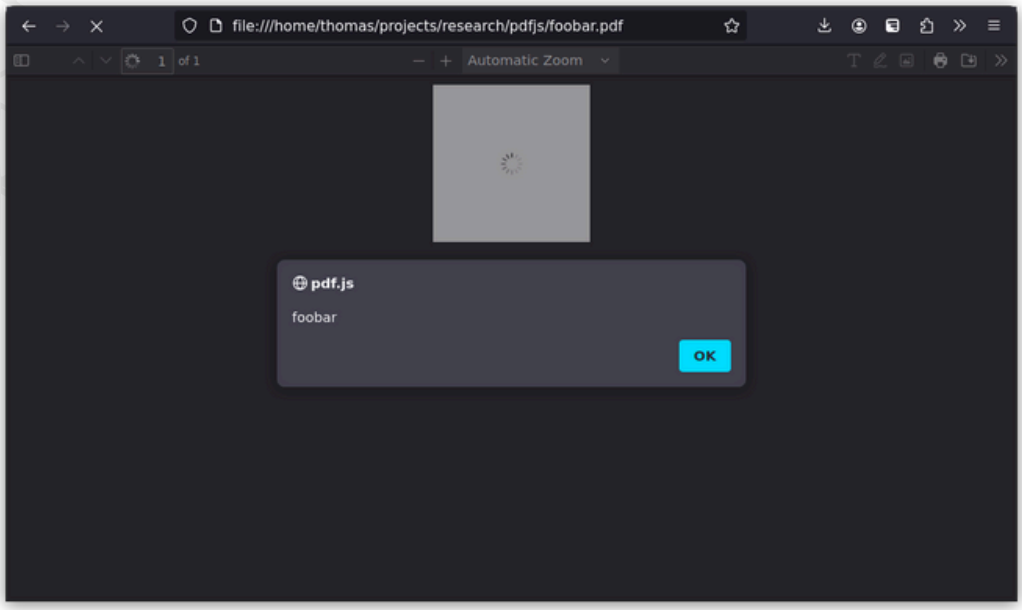
A critical vulnerability, tracked as CVE-2024-4985, has been discovered in GitHub Enterprise Server (GHES), allowing unauthorized access to instances without requiring pre-authentication. This flaw affects all GHES versions prior to 3.13.0.

The vulnerability exploits the way GHES handles encrypted Security Assertion Markup Language (SAML) claims. An attacker can craft a fake SAML claim containing valid user information. When GHES processes this fake SAML claim, it fails to correctly validate its signature, thus granting unauthorized access to the GHES instance.
Proof of Concept (PoC):
To demonstrate the exploitation of this vulnerability, follow these steps:

1. Open your preferred penetration testing tool.
2. Create a Web Connection Request.
3. Select the "GET" request type.
4. Enter your GHES URL.
5. Add a fake SAML Assertion parameter to your request. You can reference an example of a fake SAML Assertion parameter from GitHub documentation.
6. Check the response from GHES.
    - If the response returns an HTTP status code of 200, the authentication bypass using the fake SAML Assertion parameter was successful.
    - If the response returns a different HTTP status code, the bypass attempt failed.

https://github.com/absholi7ly/Bypass-authentication-GitHub-Enterprise-Server

1Day



https://twitter.com/Dinosn/status/1790226682562978210

This post details CVE-2024-4367, a vulnerability in PDF.js found by Codean Labs. PDF.js is a JavaScript-based PDF viewer maintained by Mozilla. This bug allows an attacker to execute arbitrary JavaScript code as soon as a malicious PDF file is opened. This affects all Firefox users (versions below 126) because PDF.js is used by Firefox to display PDF files. It also impacts many web- and Electron-based applications that indirectly use PDF.js for preview functionality.
If you are a developer of a JavaScript/TypeScript-based application that handles PDF files in any way, we recommend checking that you are not indirectly using a vulnerable version of PDF.js. See the end of this post for mitigation details.

PDF.js is commonly used in two scenarios:

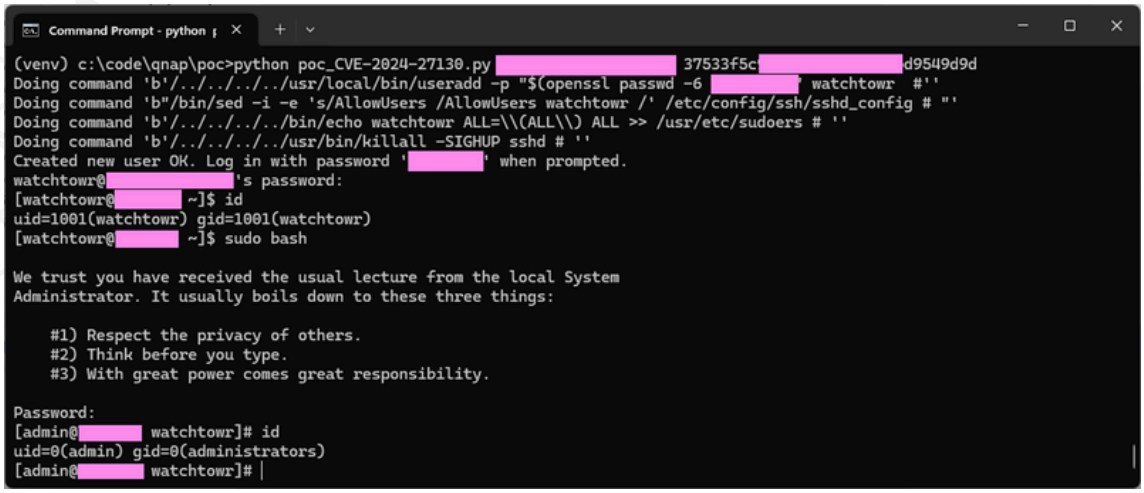
1. Firefox's Built-in PDF Viewer: PDF.js is Firefox's built-in PDF viewer. If you use Firefox and have ever downloaded or browsed a PDF file, you've seen it in action.
2. Web and Electron Applications: PDF.js is bundled into a Node module called pdfjs-dist, which has around 2.7 million weekly downloads on NPM. Websites use it to provide embedded PDF preview functionality, which is utilized by everything from Git-hosting platforms to note-taking applications.

The PDF format is notoriously complex, with support for various media types, intricate font rendering, and even rudimentary scripting. This complexity makes PDF readers a common target for vulnerability researchers. While PDF.js avoids memory corruption issues by being written in JavaScript rather than C or C++, it introduces its own set of risks.

Surprisingly, this bug is not related to the PDF format's scripting functionality. Instead, it stems from an oversight in a specific part of the font rendering code.
Fonts in PDFs can come in various formats. For modern formats like TrueType, PDF.js mostly relies on the browser's font renderer. For other formats, PDF.js manually turns glyph (character) descriptions into curves on the page. To optimize performance, a path generator function is pre-compiled for each glyph by creating a JavaScript Function object with a body (jsBuf) containing the path instructions.

https://x.com/thomasrinsma/status/1792503111317119185

NDay



https://x.com/watchtowrcyber/status/1791419479580430643

NAS devices, typically used in multi-user environments such as offices, are attractive targets for attackers due to the potential for acquiring large amounts of sensitive data. Given the long legacy and history of security weaknesses in QNAP's QTS codebase, we decided to conduct a thorough analysis, focusing on three main variants: QTS, QuTSCloud, and QTS hero.
Initial Analysis

We began our exploration with QuTSCloud, a VM-optimized version of QTS. After obtaining and setting up a virtual machine, we delved into the system, discovering a Linux-based environment with various middleware components exposed via HTTPS for management purposes. Notably, the middleware was written in C, which is notorious for security issues such as memory corruption.
Vulnerability Discovery

Our initial examination revealed several bugs, mostly memory corruption issues like double frees and buffer overflows. One significant find was CVE-2024-27130, an unauthenticated stack overflow bug that could lead to remote code execution. Here's a high-level overview of our discovery process:

1. Exploring the Web Server: We found that the web server, thttpd, executed numerous CGI scripts written in C. These scripts were prime candidates for vulnerabilities.
2. Fuzzing for Bugs: By sending long inputs to these CGI functions, we triggered several crashes, including the stack overflow in CVE-2024-27130.

CVE-2024-27130 is a particularly concerning bug due to its unauthenticated nature. By exploiting this stack overflow, an attacker can execute arbitrary code. Here's a simplified exploitation sequence:

1. Identify Vulnerable Endpoint: We targeted a specific CGI script that handles file size requests.
2. Craft Malicious Payload: Using a long input, we caused a segmentation fault, indicative of a stack overflow.
3. Execute Arbitrary Code: By carefully crafting the payload, we managed to control the execution flow, leading to remote code execution.

QNAP has released patches for some of the vulnerabilities we found. The following products have been updated:

- QTS: Version 5.1.6.2722 build 20240402 and later
- QuTS hero: Version h5.1.6.2734 build 20240414 and later

For those still affected, we recommend taking systems offline or heavily restricting access until patches are available. Additionally, users should monitor for unusual activity that could indicate exploitation of these bugs.

The Topic of the Week

https://x.com/vxunderground/status/1790745829239562694

BreachForum, a notorious online platform known for facilitating the exchange of stolen data, has been seized once again. The current display page on the forum reveals that the United States Federal Bureau of Investigation (FBI) has taken control and is actively reviewing the site.

Upon visiting BreachForum, users are now greeted with a stark notice indicating the site's new status. The message clearly states that the forum is under the control of the FBI. In addition to this, the notice prominently features an image of the current administrators' Telegram profile pictures, depicted behind bars, symbolizing their capture and the forum's shutdown.

The seizure of BreachForum marks another significant victory in the ongoing battle against cybercrime. This forum has long been a hotspot for cybercriminals to trade in stolen data, including personal information, financial records, and other sensitive materials. The FBI's intervention is a critical step in disrupting these illicit activities and bringing perpetrators to justice.

BreachForum has faced law enforcement action before, reflecting its persistent role in cybercrime. The forum's repeated shutdowns highlight both the resilience of cybercriminal communities and the ongoing efforts by global law enforcement agencies to curb their operations.

With the forum under FBI control, the focus will likely shift to analyzing the data and activities that occurred on the site. This could lead to further arrests and the dismantling of other cybercriminal networks connected to BreachForum. Users who frequented the site for illegal purposes are now at risk of identification and prosecution.

The seizure of BreachForum serves as a reminder of the relentless efforts by law enforcement to combat cybercrime. While the forum's closure will disrupt cybercriminal activities temporarily, it also underscores the need for continued vigilance and cooperation in the cybersecurity community to address the evolving threat landscape.

HADESS

cat /etc/HADESS

"Hadess" is a cybersecurity company focused on safeguarding digital assets and creating a secure digital ecosystem. Our mission involves punishing hackers and fortifying clients' defenses through innovation and expert cybersecurity services.

Website:
WWW.HADESS.IO

Threat Radar
WWW.THREATRADAR.NET

Threat Radar is a powerful threat intelligence platform that combines advanced analytics, machine learning, and human expertise to deliver actionable intelligence to organizations. It continuously monitors various data sources, including the deep web, dark web, social media platforms, and open-source intelligence, to identify potential threats, vulnerabilities, and emerging attack patterns.