Mini Stealer: Possible Predecessor of Parrot Stealer

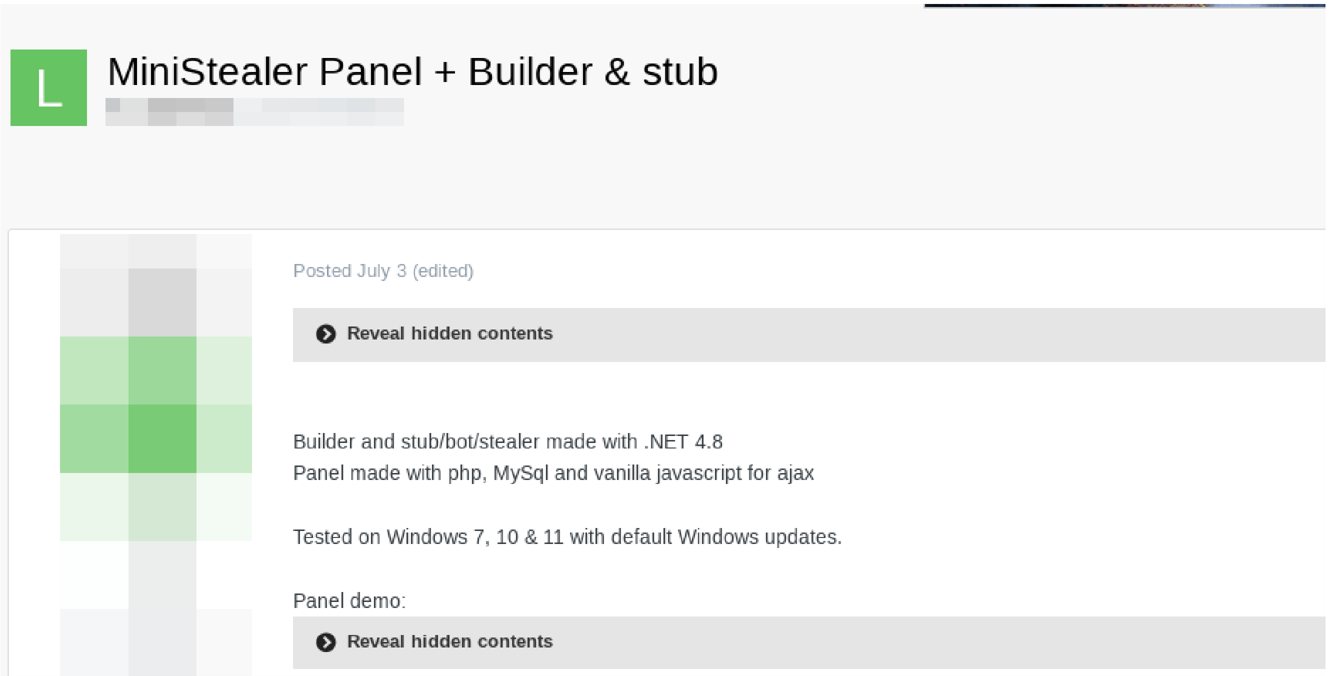
blog.cyble.com/2022/08/29/mini-stealer-possible-predecessor-of-parrot-stealer/

August 29, 2022

Mini Stealer's Builder & Panel released for free

During a routine threat hunting exercise, Cyble Research and Intelligence Labs (CRIL) discovered a post on a cybercrime forum where a Threat Actor (TA) released MiniStealer's builder and panel for free.

The TA claims that the stealer can target operating systems such as Windows 7, 10, and 11. Using such builders, TAs can easily generate malicious payloads. MiniStealer mainly targets FTP applications and Chromium-based browsers.



L

MiniStealer Panel + Builder & stub

[redacted]

[redacted]

Posted July 3 (edited)

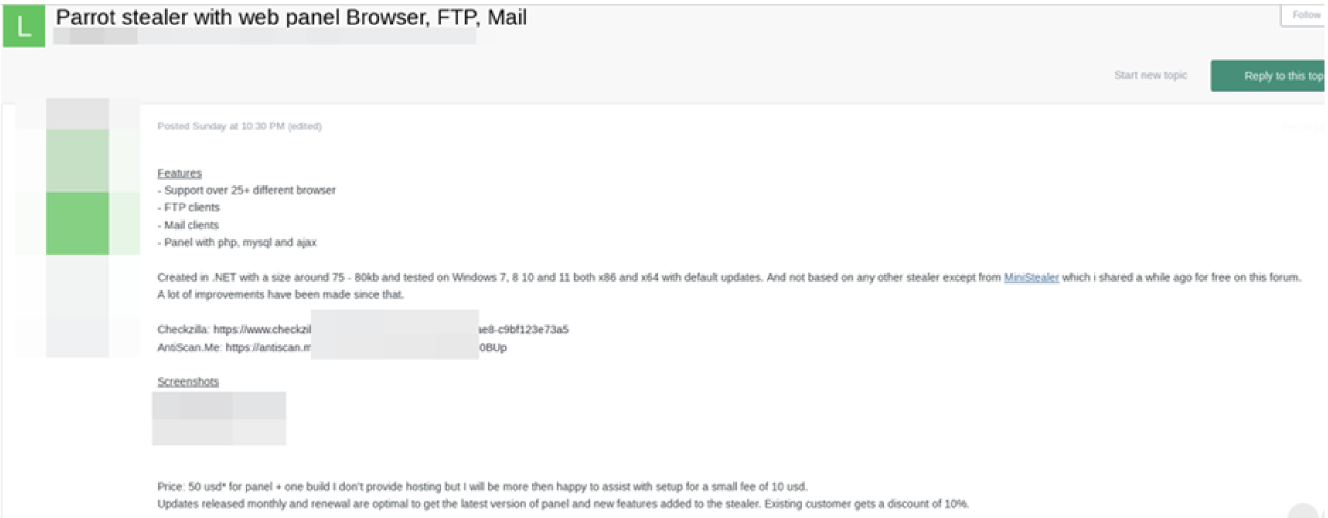
Builder and stub/bot/stealer made with .NET 4.8
Panel made with php, MySql and vanilla javascript for ajax

Tested on Windows 7, 10 & 11 with default Windows updates.

Panel demo:

Figure 1 – Post on a cybercrime forum

Nearly a month after the release of MiniStealer, the same TA made a post selling Parrot Stealer's builder and panel for USD 50. The TA stated that this stealer is based on MiniStealer. We suspect that the TA might have added the functionalities in Parrot stealer which were missing in MiniStealer.



L

Parrot stealer with web panel Browser, FTP, Mail

[redacted]

[redacted]

Posted Sunday at 10:30 PM (edited)

Features
- Support over 25+ different browser
- FTP clients
- Mail clients
- Panel with php, mysql and ajax

Created in .NET with a size around 75 - 80kb and tested on Windows 7, 8 10 and 11 both x86 and x64 with default updates. And not based on any other stealer except from MiniStealer which i shared a while ago for free on this forum.
A lot of improvements have been made since that.

Checkzilla: https://www.checkzil[redacted]ıe8-c9bf123e73a5
AntiScan.Me: https://antiscan.m[redacted]0BUp

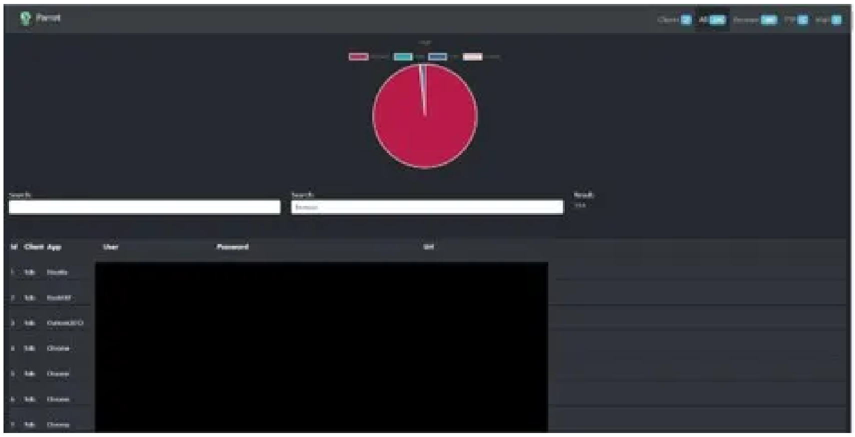
Screenshots

[redacted]

Price: 50 usd* for panel + one build I don't provide hosting but I will be more then happy to assist with setup for a small fee of 10 usd.
Updates released monthly and renewal are optimal to get the latest version of panel and new features added to the stealer. Existing customer gets a discount of 10%.

Figure 2 – TA Selling Parrot stealer

The figure below shows the Parrot stealer panel.



[redacted]

Figure 3 – Parrot Stealer Web Panel

Builder and Web Panel

The zip file leaked by TA contains two folders, as shown below. These folders contain the following files:

Builder:

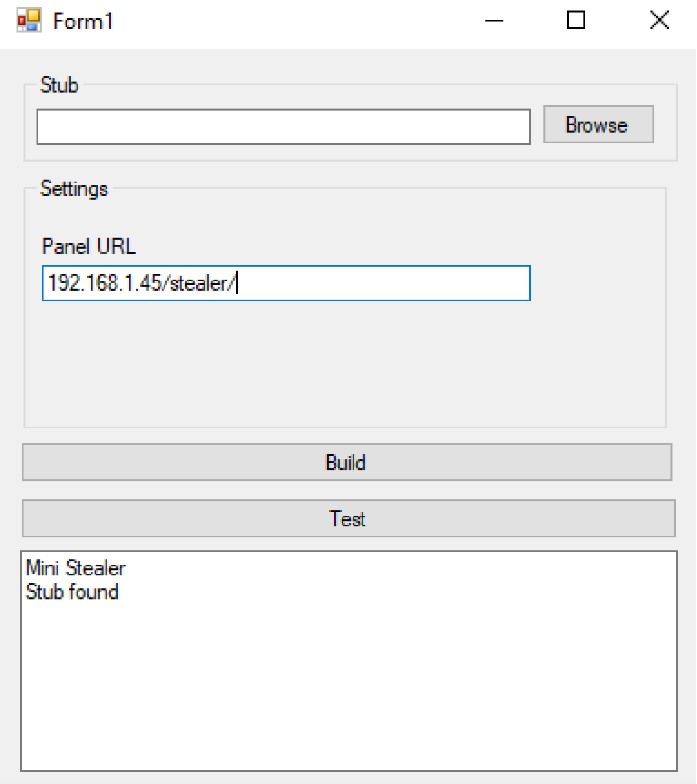
MiniStealerBuilder.exe, Stub

Panel:
Web Panel Source code

Figure 4 – Leaked Files

The builder released by the TA is a .NET-based binary. It has the functionality to add the Command and Control (C&C) details to the payload. The builder loads a file named “stub,” which is the actual payload, and then writes the C&C details to it for generating the final payload.

The test button shown in the figure below sends the Test Logs to the C&C server to check if the connection can be established. These logs contain three connection strings TestUser, TestPass, and TestHost.

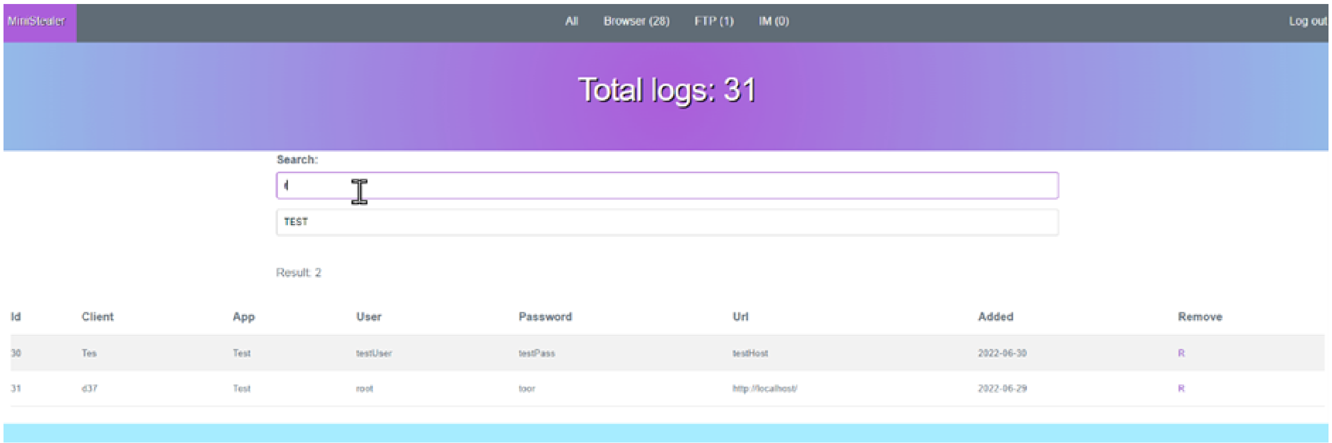


Panel URL: 192.168.1.45/stealer/

```
Mini Stealer
Stub found
```

Figure 5 – MiniStealer Builder

The TA has also released the source code of the web panel, which can be used to receive stolen data from a target network. The figure below shows the web panel.



Total logs: 31

Search: TEST

Result: 2

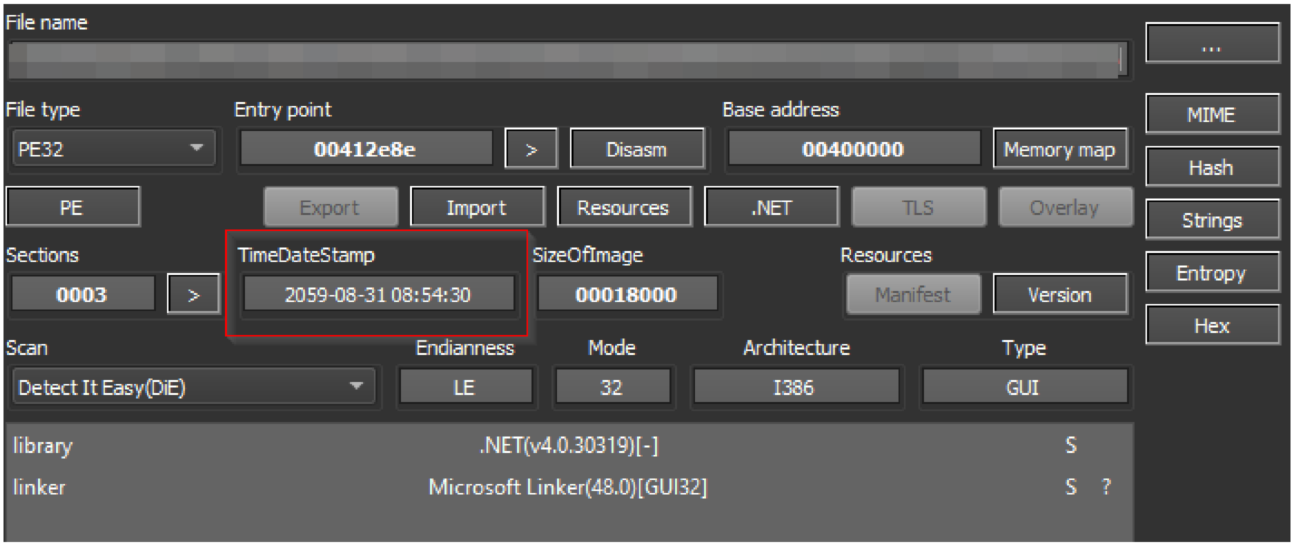
Id	Client	App	User	Password	Url	Added	Remove
30	Tes	Test	testUser	testPass	testHost	2022-06-30	R
31	d37	Test	root	toor	http://localhost/	2022-06-29	R

Figure 6 – MiniStealer panel

Technical Analysis

(Sample SHA256:*e837a0e6b01ca695010ee8bc4df57a6a9c6ef6e2c22e279501e06f61f0354f67)*

Mini Stealer is a 64-bit .NET-binary that uses Timestomping. Timestomping is a technique that modifies the timestamps of a file. Adversaries use this technique on their payloads to deflect any unnecessary attention during forensic investigations.



File name: [redacted]

File type: PE32

Entry point: 00412e8e

Base address: 00400000

Sections: 0003

TimeDateStamp: 2059-08-31 08:54:30

SizeOfImage: 00018000

Scan: Detect It Easy(DiE)

Endianness: LE

Mode: 32

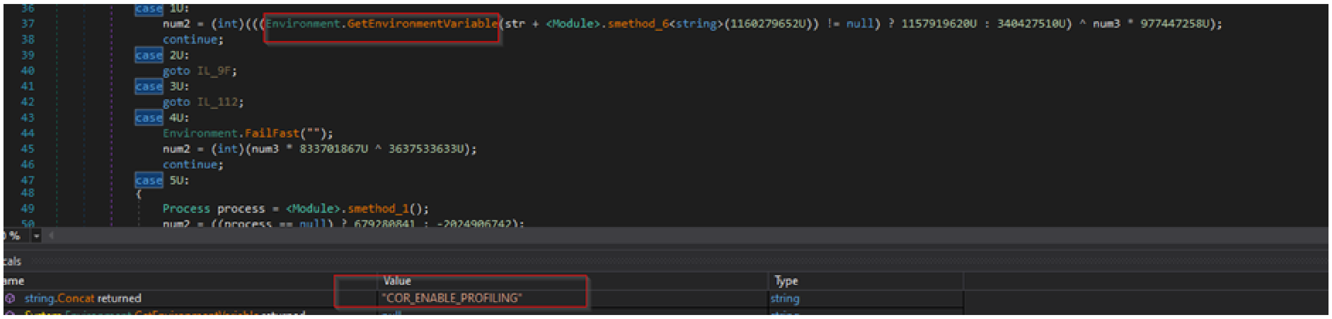
Architecture: I386

Type: GUI

```
library    .NET(v4.0.30319)[-]                 S
linker     Microsoft Linker(48.0)[GUI32]       S ?
```

Figure 7 – File Details

The stealer uses multiple AntiAnalysis checks to prevent debugging of the sample. To detect profiling, the code verifies if the *COR_ENABLE_PROFILING* environment variable is present and set to 1. Profilers are designed to monitor, troubleshoot, and debug managed code executed by the .NET Common Language Runtime. The figure below showcases the stealer detecting profiling.

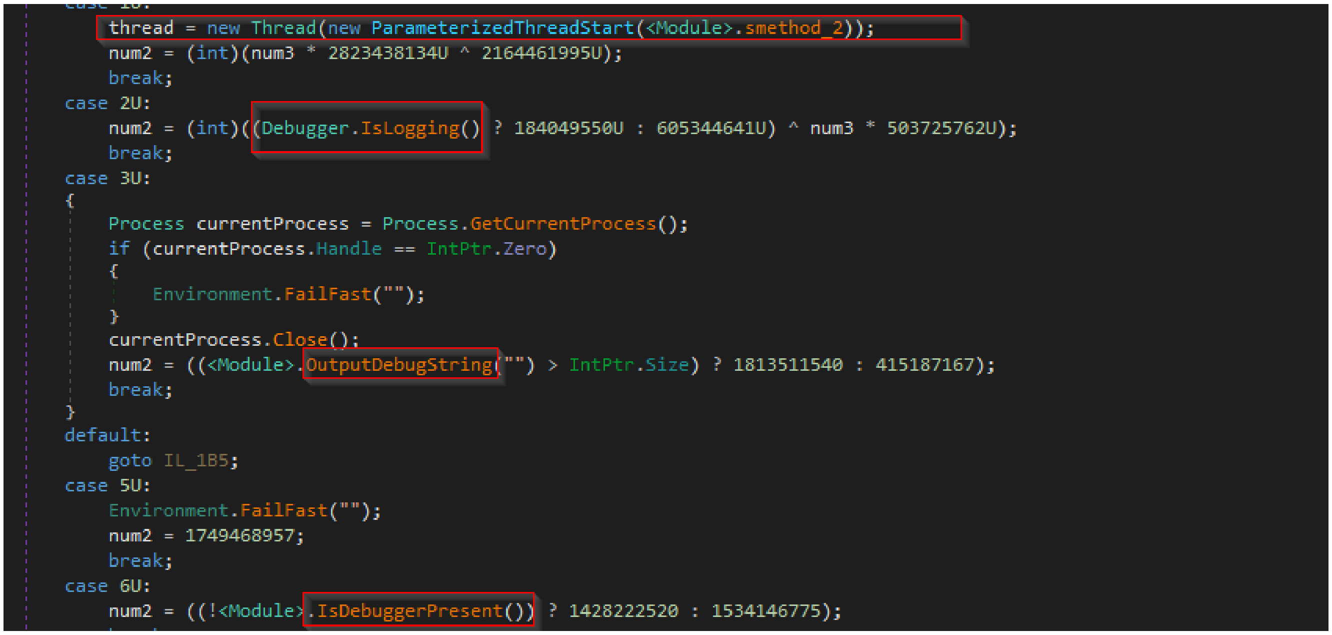


```
        case 1U:
            num2 = (int)(((Environment.GetEnvironmentVariable(str + <Module>.smethod_6<string>(1160279652U)) != null) ? 1157919620U : 340427510U) ^ num3 * 977447258U);
            continue;
        case 2U:
            goto IL_9F;
        case 3U:
            goto IL_112;
        case 4U:
            Environment.FailFast("");
            num2 = (int)(num3 * 833701867U ^ 3637533633U);
            continue;
        case 5U:
        {
            Process process = <Module>.smethod_1();
            num2 = ((process == null) ? 679280841 : -2024906742);
```

```
Name                        Value                     Type
string.Concat returned      "COR_ENABLE_PROFILING"    string
```

Figure 8 – Detecting profiling

This stealer spawns a thread for continuously checking if the stealer payload is being debugged. To check for the presence of debuggers, this thread executes methods such as *IsDebuggerPresent, OutputDebugString*, and *Debugger.IsLogging.*

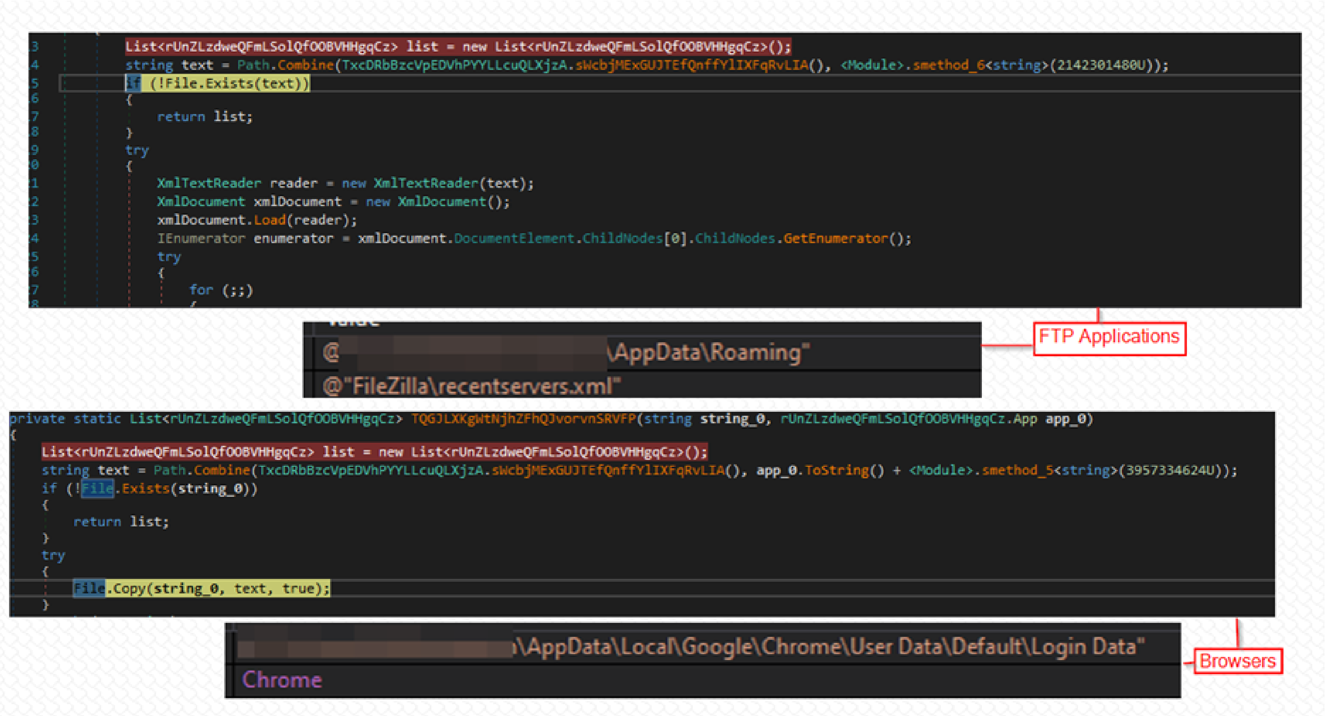


```
        thread = new Thread(new ParameterizedThreadStart(<Module>.smethod_2));
        num2 = (int)(num3 * 2823438134U ^ 2164461995U);
        break;
    case 2U:
        num2 = (int)((Debugger.IsLogging() ? 184049550U : 605344641U) ^ num3 * 503725762U);
        break;
    case 3U:
    {
        Process currentProcess = Process.GetCurrentProcess();
        if (currentProcess.Handle == IntPtr.Zero)
        {
            Environment.FailFast("");
        }
        currentProcess.Close();
        num2 = ((<Module>.OutputDebugString("") > IntPtr.Size) ? 1813511540 : 415187167);
        break;
    }
    default:
        goto IL_1B5;
    case 5U:
        Environment.FailFast("");
        num2 = 1749468957;
        break;
    case 6U:
        num2 = ((!<Module>.IsDebuggerPresent()) ? 1428222520 : 1534146775);
```

Figure 9 – Anti-Analysis

This stealer payload steals data from the following Chromium-based browsers and FTP applications. The stealer appears to be in the development stage as several FTP applications are hardcoded in the stealer, but it does not appear to target all of them.

The TA might have added these functionalities in Parrot Stealer, which is suspected to be an upgraded paid version of MiniStealer. For the FTP application, the stealer steals data from configuration files. For browsers, the stealer copies certain files for exfiltration present in the *AppData\Browser* directory, which stores user session and login credentials, as shown in the figure below.



```
        List<rUnZLzdweQFmLSolQfOOBVHHgqCz> list = new List<rUnZLzdweQFmLSolQfOOBVHHgqCz>();
        string text = Path.Combine(TxcDRbBzcVpEDVhPYYLLcuQLXjzA.sWcbjMExGUJTEfQnffYlIXFqRvLIA(), <Module>.smethod_6<string>(2142301480U));
        if (!File.Exists(text))
        {
            return list;
        }
        try
        {
            XmlTextReader reader = new XmlTextReader(text);
            XmlDocument xmlDocument = new XmlDocument();
            xmlDocument.Load(reader);
            IEnumerator enumerator = xmlDocument.DocumentElement.ChildNodes[0].ChildNodes.GetEnumerator();
            try
            {
                for (;;)
```

```
@[redacted]\AppData\Roaming"
@"FileZilla\recentservers.xml"
```

```
private static List<rUnZLzdweQFmLSolQfOOBVHHgqCz> TQGJLXKgWtNjhZFhQJvorvnSRVFP(string string_0, rUnZLzdweQFmLSolQfOOBVHHgqCz.App app_0)
{
    List<rUnZLzdweQFmLSolQfOOBVHHgqCz> list = new List<rUnZLzdweQFmLSolQfOOBVHHgqCz>();
    string text = Path.Combine(TxcDRbBzcVpEDVhPYYLLcuQLXjzA.sWcbjMExGUJTEfQnffYlIXFqRvLIA(), app_0.ToString() + <Module>.smethod_5<string>(3957334624U));
    if (!File.Exists(string_0))
    {
        return list;
    }
    try
    {
        File.Copy(string_0, text, true);
    }
```

```
[redacted]\AppData\Local\Google\Chrome\User Data\Default\Login Data"
Chrome
```

Figure 10 – Stealing Data

Over 25 Chromium-based browsers:

Chrome, AvastBrowser, AVGBrowser, Browser360, CCleanerBrowser, CentBrowser, Chedot, Citrio, CocCoc, ComodoDragon, CoolNovo, Coowon, ElementsBrowser, EpicPrivacyBrowser, IridiumBrowser, Kometa, LiebaoBrowser, Maxthon, OperaGX, OperaNeon, Orbitum, QIPSurf, Sleipnir, SlimJet, Sputnik, SRWareIron, uCozMedia, Vivaldi, Yandex,

Over 20 FTP Applications:

Filezilla, FlashFXP, AutoFTPManager, AutoFTPPro, BitKinex, BulletproofFTP, ClassicFTP, CoreFTP, CuteFTP, Cyberduck, Dreamweaver, FreeFTP, DirectFTP, ftpcommander, FTPEXPLORER, FTPRush, FTPvoyager, LeapFTP, MultiCommander, SmartFTP, SuperPutty, TotalCommander, TurboFTP, WSFTP, WinSCP

Conclusion

The availability of free malware builders and panels can assist TAs in carrying out attacks, as TAs do not need to invest time and money to get malware payloads for cybercrime purposes. There is always the possibility that TA might have released the builder and panel of MiniStealer for free for marketing purposes and to build a reputation amongst themselves on cybercrime forums.

The TA’s behavior further reinforces this theory as after 1 month; they began selling a paid stealer, which is suspected to be based on MiniStealer. CRIL continuously monitors emerging threats and has observed a surge in the use of stealer malware by TAs.

Our Recommendations

We have listed some essential cybersecurity best practices that create the first line of control against attackers. We recommend that our readers follow the best practices given below:

How to prevent malware infection?

- Avoid downloading pirated software from warez/torrent websites. The “Hack Tool” present on sites such as YouTube, torrent sites, etc., contains such malware.
- Use strong passwords and enforce multi-factor authentication wherever possible.
- Turn on the automatic software update feature on your computer, mobile, and other connected devices.

- Use a reputed anti-virus and internet security software package on your connected devices, including PC, laptop, and mobile.
- Refrain from opening untrusted links and email attachments without first verifying their authenticity.
- Educate employees in terms of protecting themselves from threats like phishing’s/untrusted URLs.
- Block URLs that could be used to spread the malware, e.g., Torrent/Warez.
- Monitor the beacon on the network level to block data exfiltration by malware or TAs.

MITRE ATT&CK® Techniques

Tactic	Technique ID	Technique Name

Execution	T1204	User Execution
Defense Evasion	T1497.001 T1070.006	Virtualization/Sandbox Evasion: System Checks Indicator Removal on Host: Timestomp
Credential Access	T1555 T1539 T1552 T1528	Credentials from Password Stores Steal Web Session Cookie Unsecured Credentials Steal Application Access Token
Discovery	T1087 T1518 T1057 T1007	Account Discovery Software Discovery Process Discovery System Service Discovery
Command and Control	T1071	Application Layer Protocol
Exfiltration	T1041	Exfiltration Over C2 Channel

Indicators of Compromise (IOCs)

Indicators	Indicator type	Description
d65def0ad7f1b428bc1045cf2214b82f	**MD5**	Malicious binary
e2beda0ef5d1c38bb96fb7eb6ee25990073e6a17	**SHA1**	
e837a0e6b01ca695010ee8bc4df57a6a9c6ef6e2c22e279501e06f61f0354f67	**SHA256**	