

Deep Analysis of Redline Stealer: Leaked Credential with WCF

By S2W

Published: 2022-05-24 · Archived: 2026-04-05 14:17:58 UTC



Author: Jiho Kim | S2W TALON

Last Modified : 2022.03.03.

Press enter or click to view image in full size



Photo by [S Migaj](#) on [Unsplash](#)

Executive Summary

Redline Stealer, which is currently being distributed, has changed the C2 communication method and the way of delivering the collected information from the previous Redline Stealer, but the overall execution flow is the same.

Redline Stealer has hard-coded encoded data such as C2 Server IP and Unique ID, and the XOR Key required to decode this data. When Redline is executed, the value is extracted first. After that, the information is collected and leaked by referring to the configuration data received from the C2 server, and the collected information is composed of **Environment Details** and **Credential Details**. The collected information includes system

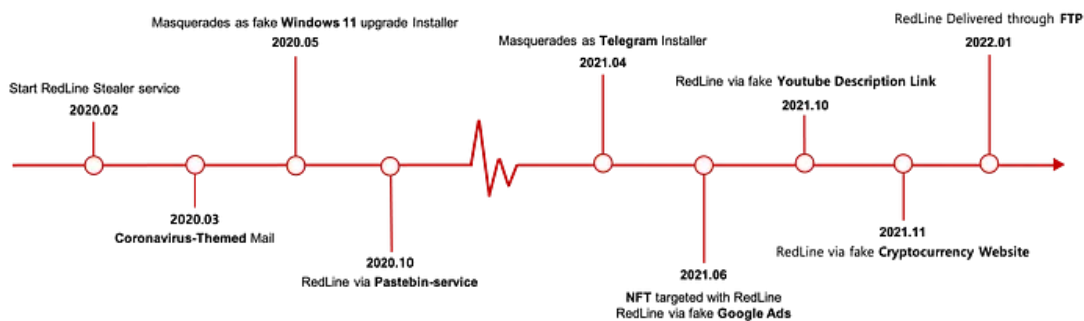
information, browser credentials, crypto wallet information, FTP information, Telegram and Discord information, etc.

After collecting and leaking information, Redline Stealer also has the ability to download executable files and perform additional malicious actions.

Introduction of Redline Stealer

Since its release in February 2020, Redline Stealer has been delivered through various channels. Redline Stealer is mostly distributed through Phishing Emails or malicious software disguised as installation files such as Telegram, Discord, and cracked software. However, recently, Phishing Link that downloads Chrome Extension containing Redline Stealer by abusing **YouTube Video Description and Google Ads** is utilized, or Python Script that runs Redline Stealer through FTP is being distributed.

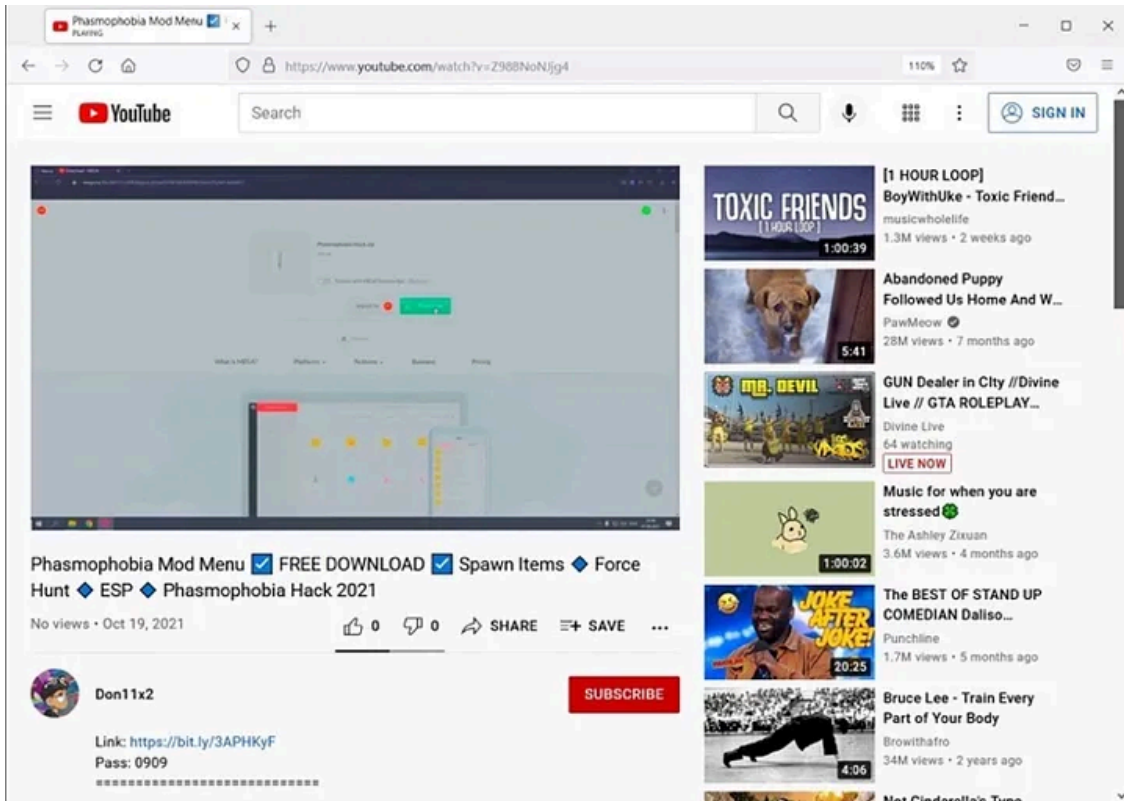
Press enter or click to view image in full size



Issues related to Redline Stealer

According to BleepingComputer released in October 2020, Redline Stealer was distributed through malicious links posted on YouTube Video Description related to free downloading of specific utility.

Press enter or click to view image in full size



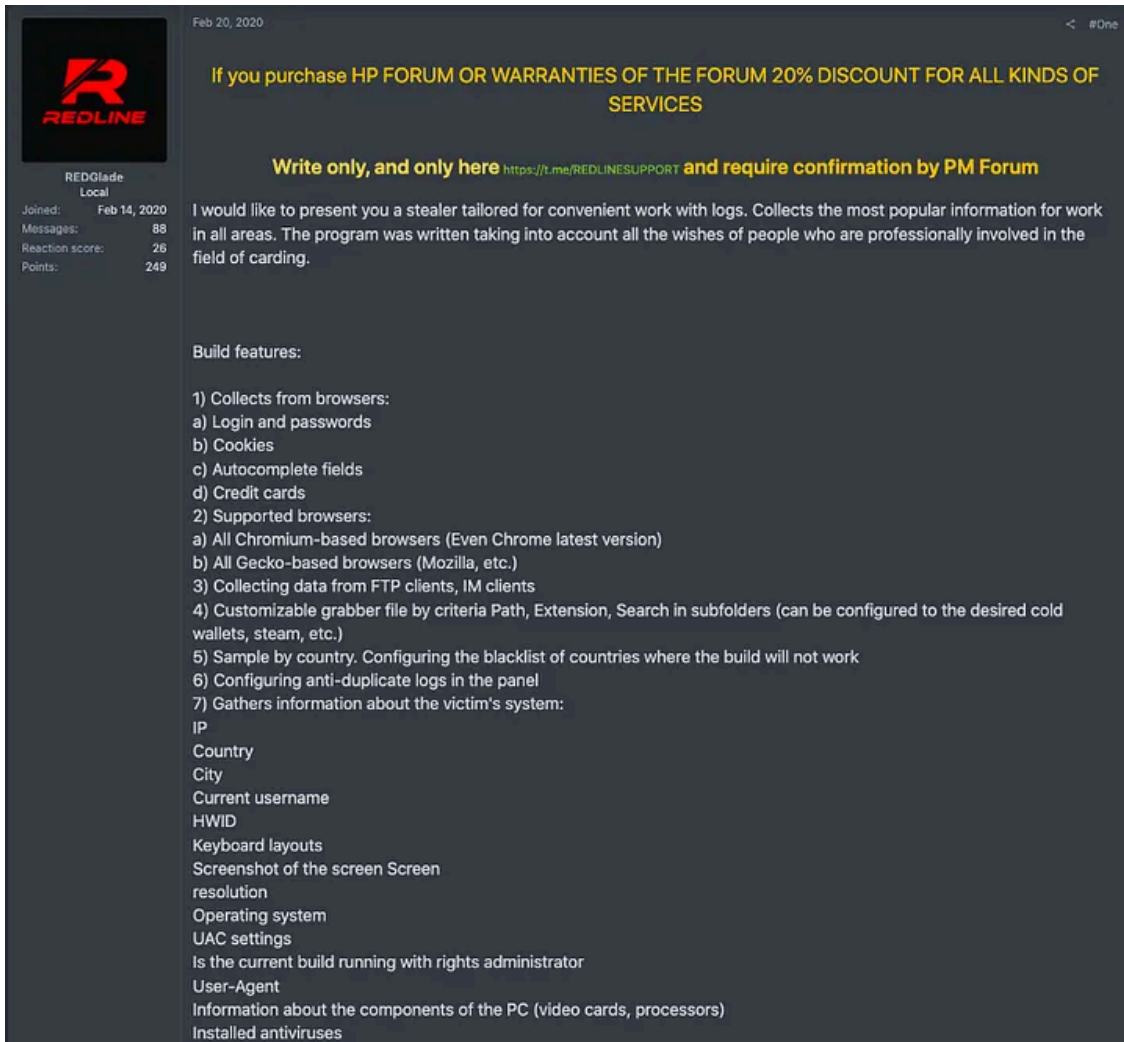
Redline via YouTube Video Description Link (Source: [BleepingComputer](#))

Redline Stealer in DDW

Redline Stealer first appeared in a Russian-based forum in February 2020. The user with the nickname “**REDGlade**” posted the promotion article and has been updating the version of Builder and Panel until at least January 2022. Redline Stealer is being rented for \$100 per month and sold for \$150 per month and \$800 for a lifetime. Additional services, such as scanner and crypto subscription, appear differently depending on the cost.

The builder program of Redline Stealer is sold by the official seller on the DDW forum, but also by other users who sell the cracked version of Redline Stealer. In addition, some users sell only the collected Redline Stealer Logs.

Press enter or click to view image in full size



Redline Stealer Promotion Article

Redline Stealer's Pricing Policy

RENT (\$100 / a month)

- 1 month of cryptor @spectrcrypt_bot (autocrypt + scanner)

LITE (\$150 / a month)

- 1 month of crypt subscription

PRO (\$200 / forever)

- 3 months of scanner subscription
- 3 months of cryptor @spectrcrypt_bot

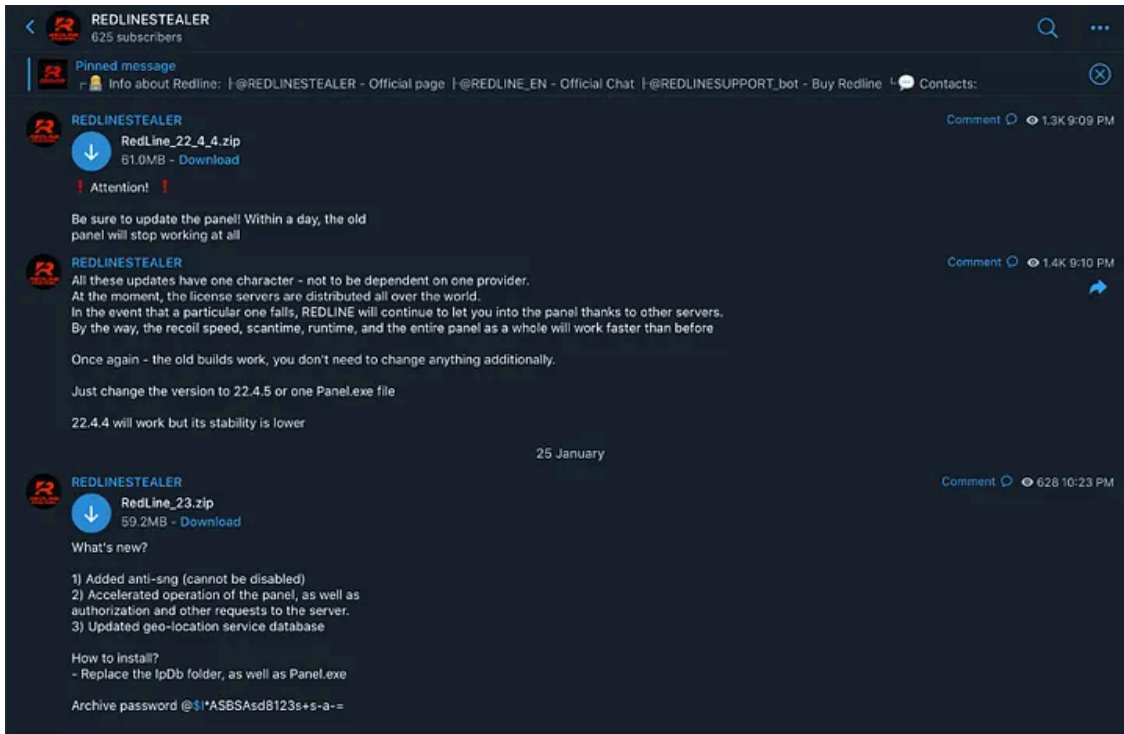
Channels operated by Redline Stealer Seller

Telegram channels operated by the Redline Stealer official seller are divided into 3 categories: **Official Page, Official Chat, and Buy Redline bot**. Announcement and updated information are posted on the Official Page channel, chat is freely available on the Official Chat channel, and Redline Stealer is sold on the Buy Redline bot.

Redline Stealer Telegram Channel

- @REDLINESTEALER — Official page
- @REDLINE_EN — Official Chat
- @REDLINESUPPORT_bot — Buy Redline bot

Press enter or click to view image in full size

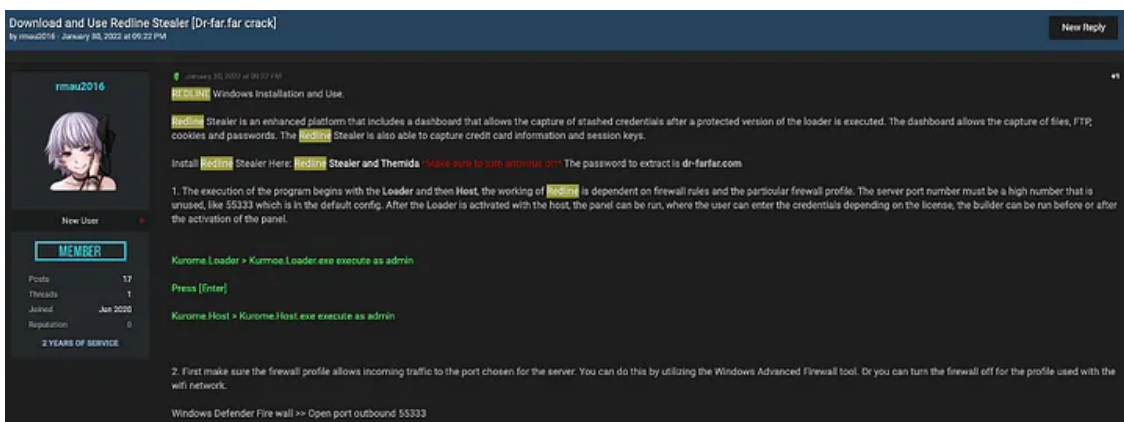


Redline Stealer Telegram Channel: Official Page

Cracked Redline Stealer & Log Seller

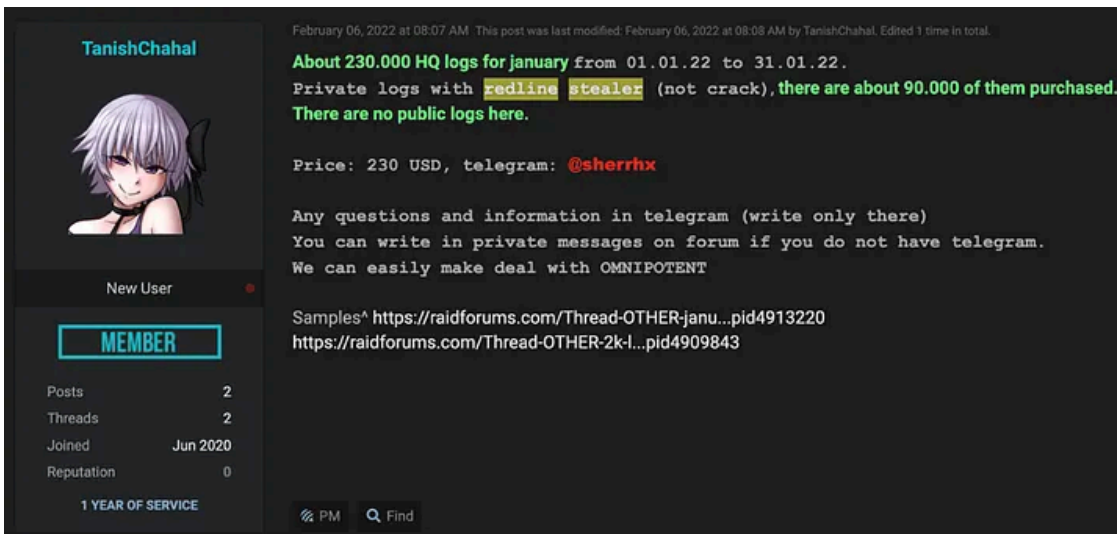
As Redline Stealer is an infostealer malware used by attackers a lot, there are several cracked versions, and other stealers derived from it. In addition, stealer logs collected through Redline Stealer are sold on the DDW forums, and they account for the largest portion of infostealer logs.

Press enter or click to view image in full size



Cracked Redline Stealer Sales Post

Press enter or click to view image in full size



Redline Stealer Log Sales Post

Redline Stealer Update Information

Redline Stealer Seller notifies update information on Telegram channel. As of January 2022, it has been updated to Builder v23, Panel v3.3.4. The main update information posted so far is shown in the table below.

| Date | Description |
|---------|---|
| 2020.03 | - Added Anti-VM - Added Cold Wallets |
| 2020.06 | - Added choice of target platform for build x86 / x64 - Added support for collection browser from Asian OS - Improved file grabber |
| 2020.08 | - Collect Telegram Files - Collect Nord/Open/Proton VPN Files - Collect Steam Files |
| 2021.05 | - Support *.scr extension files for creating builds |
| 2021.06 | - Collect tokens of Discords - Added search for extensions in all browsers based on chrome - Added "Visible" log fields - Added Wallets: Browser Extension |
| 2021.08 | - Communication Protocol: HTTP → Net.Tcp - Added Wallets: Browser Extension - Select Option: Send Log by Part / Full |

Redline Stealer Major Update

Especially among the updates in May 2020 to June 2020, supporting ***.scr extension** and added **Browser Extension Wallet information** were also applied to issues related to NFT hacking that occurred in June 2021. At that time, most of the victims infected by Redline Stealer had *.scr extension. Also, the victims' stolen crypto wallets were leaked by Redline Stealer.

Malware analysis

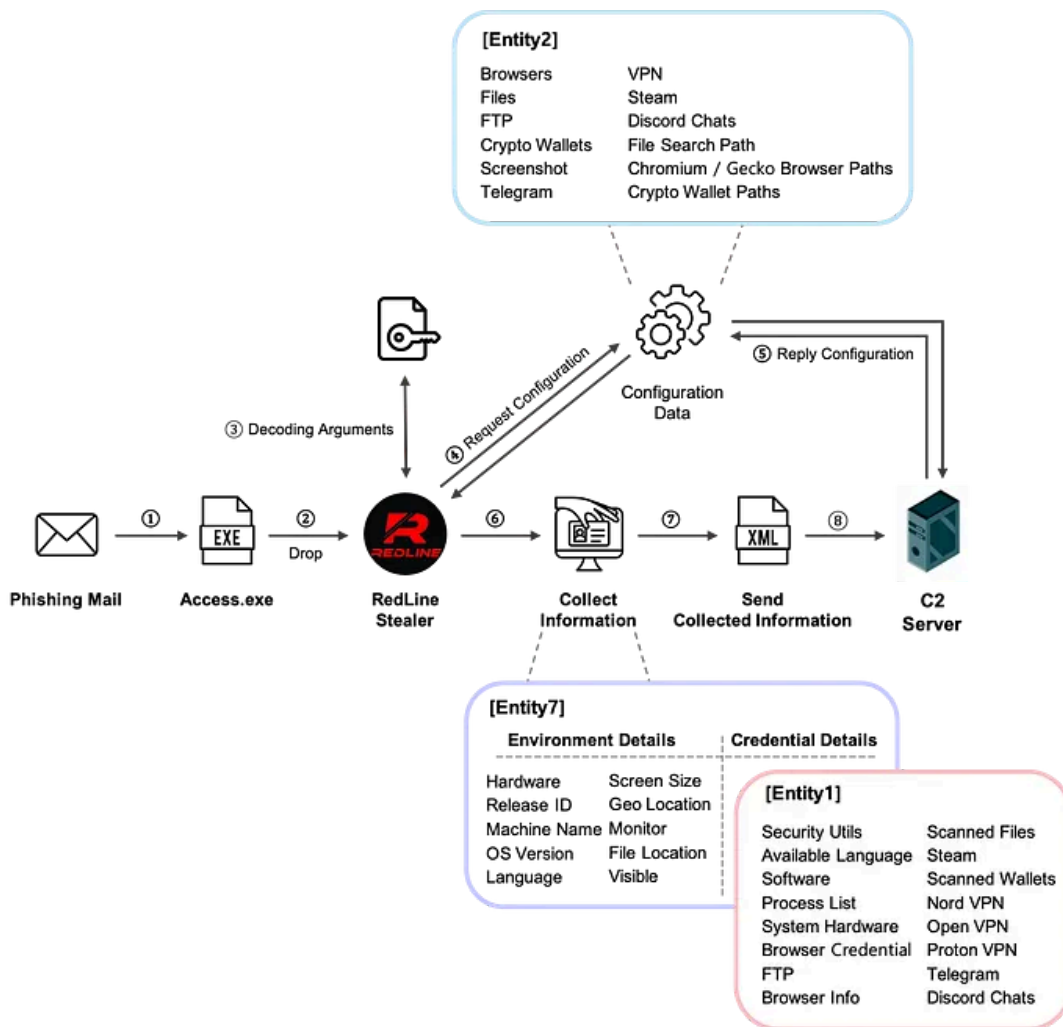
Sample Information

- File Name: 9882_1643998124_6086.exe
- File Type: PE32 executable (GUI) Intel 80386 Mono/.Net assembly, for MS Windows
- Malware Type: Redline Stealer v22
- MD5: d81d3c919ed3b1aaa2dc8d5f9cf382
- SHA256: cd3f0808ae7fc8aa5554192ed5b0894779bf88a9c56a7c317ddc6a4d7c249e0e

Redline Stealer Execution Flow

1. The attachment in phishing mail contains cracked software with Redline Stealer.
2. When the cracked software is executed, Redline Stealer is also executed in the background.
3. Encoded data such as C2 Server IP and Unique ID are decoded along with the XOR key and used for C2 communication.
4. After finishing the decoding process, Redline Stealer requests configuration data from the C2 Server.
 - Entity2: a structure that stores configuration data.
5. The C2 Server transmits configuration data to the infected PC.
6. Information is collected from the infected PC referring to stored configuration data.
 - Entity7: a structure that stores collected results. (Environment Details + Entity1)
 - Entity1: a structure that stores Credential Details.
7. The information is leaked twice.
 - Environment Details including infected PC information
 - Credential Details including crypto wallet, accounts, and user data information
8. Collected information is converted into XML format and transmitted to the C2 Server through SOAP Message.

Press enter or click to view image in full size



Redline Stealer Execution Flow

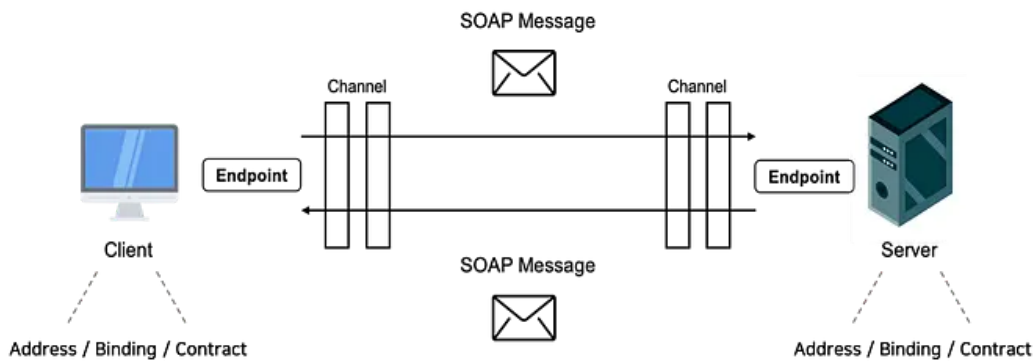
Configuration of C2 Communication

Redline Stealer with WCF

The framework Redline Stealer uses for C2 communication is WCF(Windows Communication Foundation). WCF is a system that allows endpoints to exchange messages and communicate across multiple computers connected to the network.

At least one endpoint must be configured to use WCF. When configuring the endpoint, three elements are required: **Address**, **Binding**, and **Contract**. ‘Address’ is the address providing the service, ‘Binding’ is the information related to the communication protocol used to access the service, and ‘Contract’ defines the service interface. The WCF Client can call the service defined as Service Contract, and when a specific method is called, a method of the same name implemented in the server is called. [ServiceContract] keyword, a service interface, is used to define the contract, [DataContract] keyword is used to define a data structure to be transmitted, and [OperationContract] keyword is used to define the function of the service.

Press enter or click to view image in full size

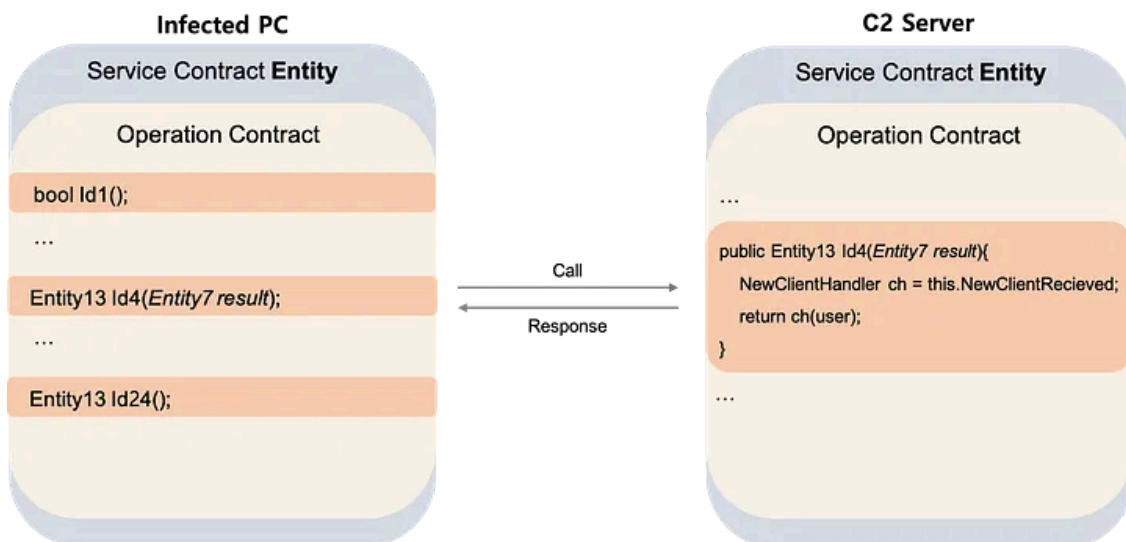


WCF Communication

The previous Redline Stealer used `BasicHTTPBinding()` for communication. However, from Redline Stealer v22 updated in August 2020, the communication protocol was changed to `NetTcpBinding()`. `NetTcpBinding()` has a performance advantage compared to `BasicHTTPBinding()` because SOAP messages are binary encoded and delivered.

Redline Stealer collects information by specifying a Service Contract named **Entity** and defines 24 Operation Contracts and 17 Data Contracts. When a method defined as Operation Contract is called from an infected PC to the C2 Server, a method of the same name implemented on the C2 Server is called. At this time, 'Entity7 result' is delivered to the C2 Server.

Press enter or click to view image in full size



WCF Service Call/Response

Decoding C2 Server and Unique ID

Get S2W's stories in your inbox

Join Medium for free to get updates from this writer.

Remember me for faster sign in

In Redline Stealer, the encoded C2 Server address and Unique ID are hard-coded. Therefore, when the malware is executed, they are decoded and used for C2 communication.

- Hardcoded data

C2 Server address: Dw0oGCQnJh4tByxCDjRVWScZLIUvOTwJDDZcUA

Unique ID: DyMgXCcJKlcvBwJB

Message: ""

Version: 1

- Decoding Process

FromBase64 → XOR → FromBase64

XOR Key: Agamis

- Decoding Result

C2 Server address: **62.182.159.86:65531**

Unique ID: **405794696**

Message: ""

Version: 1

Press enter or click to view image in full size

```
public static string Read(string b64, string stringKey)
{
    string result;
    try
    {
        if (string.IsNullOrEmpty(b64))
        {
            result = string.Empty;
        }
        else
        {
            result = StringDecrypt.FromBase64(StringDecrypt.Xor(StringDecrypt.FromBase64(b64), stringKey));
        }
    }
    catch
    {
        result = b64;
    }
    return result;
}
```

Decoding Method: Read()

Communication Method

As mentioned, Redline Stealer uses WCF for C2 communication.

- Endpoint Configuration: Address & Binding

Address: net.tcp//62.182.159.86:65531/

Binding: NetTcpBinding()

- Endpoint Configuration: Contract

Redline Stealer has a Service Contract named **Entity**, 17 Data Contracts that define the structure to store information, and 24 Operation Contracts that define the functionality of the service. Among them, the description of the Data Contract storing information is shown in the table below.

(*Functional descriptions for each Operation Contract are described in Appendix.)

Press enter or click to view image in full size

| [ServiceContract] Namespace | [DataContract] Name | Stored information |
|--------------------------------|------------------------|---|
| Entity | Entity2 | A structure that stores configuration data related information |
| | Entity7 | A structure that stores collected result (Environment Details + Entity1) |
| | Entity1 | A structure that stores Credential Details information |
| | Entity3 | A structure that stores System Hardware related information |
| | Entity4 | A structure that stores Browser installation related information |
| | Entity5 | A structure that stores File related information |
| | Entity8 | A structure that stores AutoFill related information |
| | Entity9 | A structure that stores Browser Credential related information |
| | Entity10 | A structure that stores Cookie related information |
| | Entity11 | A structure that stores CC related information |
| | Entity12 | A structure that stores Login Data related information |
| | Entity13 | A structure that stores Server Response |
| | Entity6 | A structure that stores Task-related information |
| | Entity14 | A structure that stores Hardware Type |
| | Entity16 | A structure that stores File Search information |
| | Entity17 | A structure that stores Crypto Wallet related information |
| | Entity15 | Update Action |

Data Contract List

- Try to connect

After configuring the endpoint, Redline Stealer tries to connect with the C2 Server and receives the response. Redline Stealer periodically checks whether it maintains a connection with the C2 Server during execution.

Configuration Data Request/Receive

Request configuration data

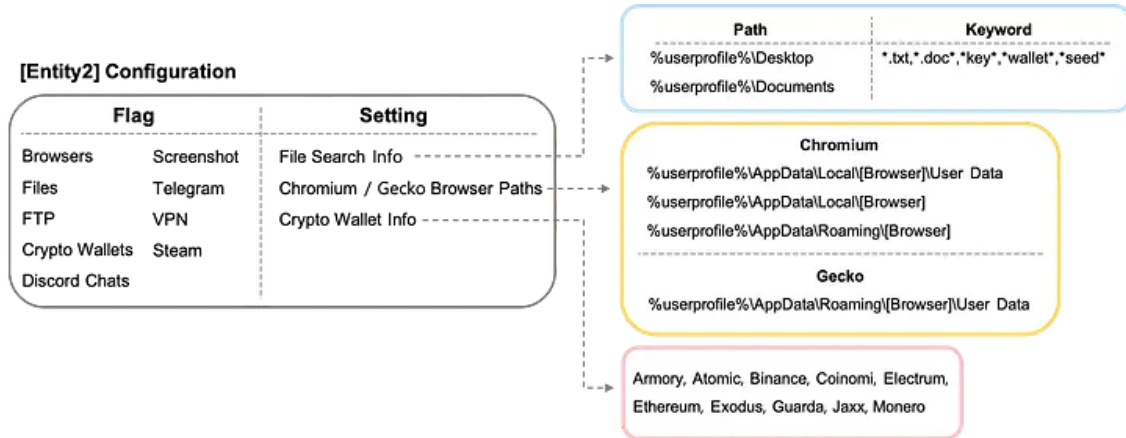
The Redline Stealer requests configuration data that specifies what information to collect, and includes the paths and keywords required to collect browser and local file information, and the name of crypto wallets to be

explored.

Response configuration data

The configuration data is stored in **Entity2** and used to collect information to be leaked. The configuration data consists of **Flag** indicating whether each item is collected and **Setting** indicating paths and keywords for collecting files.

Press enter or click to view image in full size

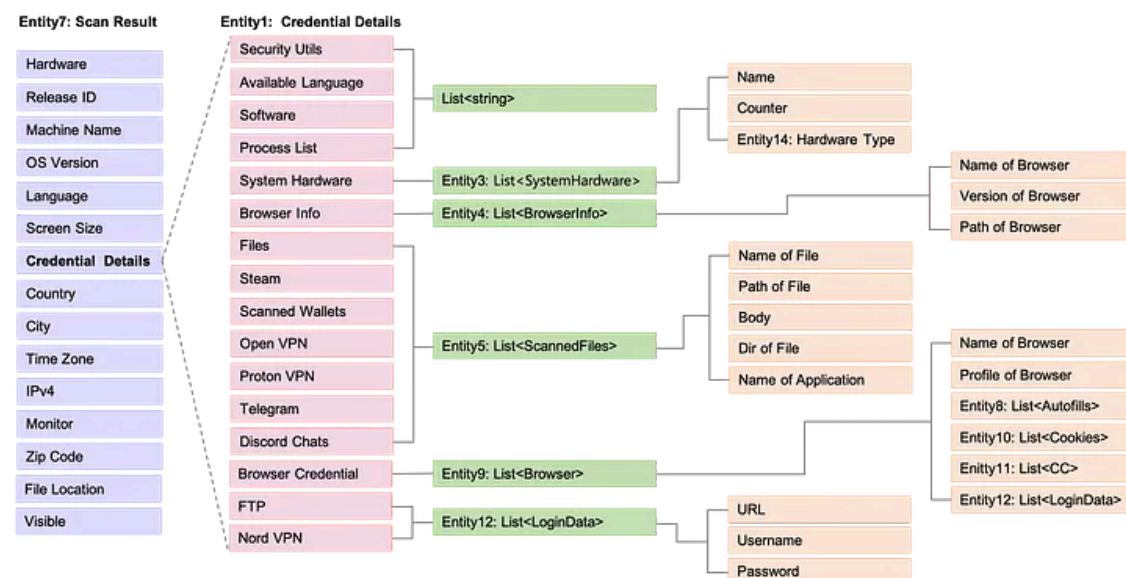


Collected information

Way to collect and store information

The information collected from the infected PC is stored in **Entity7**. Entity7 includes Environment Details and Entity1, and Entity1 separately stores Credential Details information. Each item in Entity1 uses the structure of Entity3~Entity5, Entity8~Entity12, and Entity14 to store related information. At this time, Entity1 may or may not be used depending on Redline Stealer's information leakage mode.

Press enter or click to view image in full size



A structure of Entity7

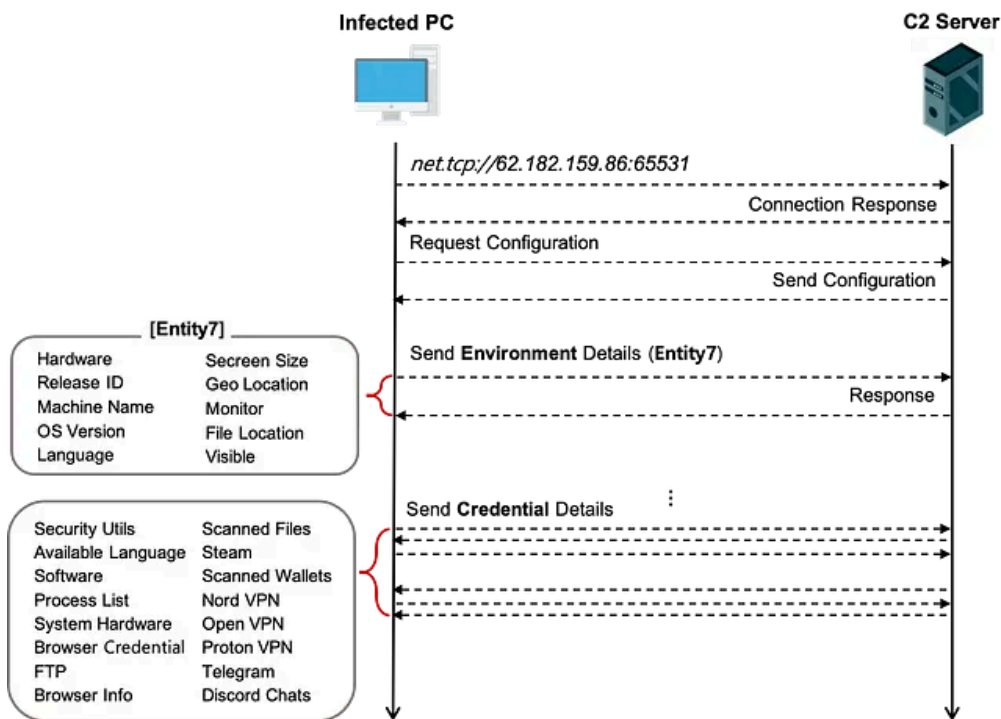
Way to leak information

Redline Stealer defines **two ways** to leak information.

- **Send Log by Parts** (Default)

The “Send Log by Parts” is a method of collecting information from the infected PC and then partially leaking it. That is, the collected ‘Environment Details’ are first leaked to the C2 Server by putting it in Entity7. In this case, each item of Entity1 is stored empty. After that, ‘Credential Details’ are collected, but not stored in Entity1 and immediately leaked after being collected by the item.

Press enter or click to view image in full size

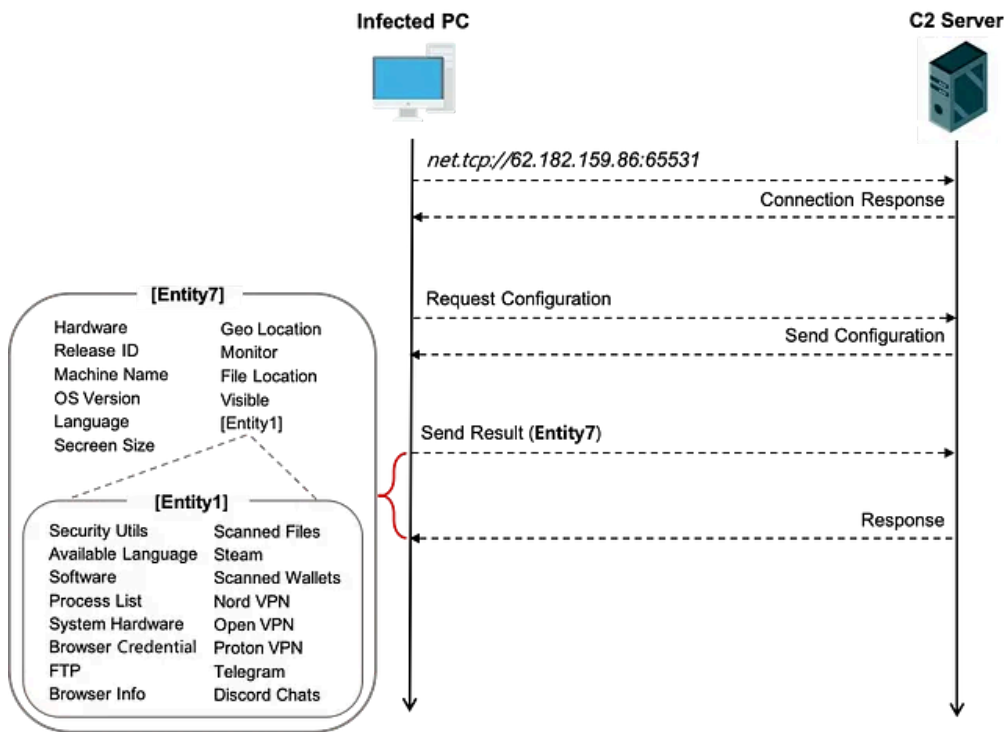


Send Log by Parts Flow

- **Send Log by Full**

This method stores all the collected information in Entity7 and leaks it. First, ‘Environment Details’ are collected and stored in Entity7. Credential Details are then collected and stored in Entity1. If Environment Details and Entity1 are filled in Entity7, it is leaked to the C2 Server.

Press enter or click to view image in full size



Send Log by Full-Flow

The biggest difference between the two methods is whether Entity1 is used or not. Environment Details and Entity1 collected from the infected PC are stored in Entity7, while Entity1 stores Credential Details. In the “Send Log by Full” method, Entity1 is used to leak information at once, but in the “Send Log by Parts” method, Entity1 is not used and each item of Credential Details is leaked as soon as it is collected.

What method Redline Stealer uses can be checked through the “Version” value among hard-coded data. If the version is 1, “Send Log by Parts” method is used, and in other cases, “Send Log by Full” method is used. In the case of the sample, since the version is set to 1, the “Send Log by Parts” method can be seen, which partially leaks the collected information to be used. Therefore, among the collected information, Credential Details is collected for each item and then leaked immediately.

Collect Environment Details

Device information of the infected PC is collected and stored in Entity7.

Entity7 includes hardware information, Unique ID, machine name, OS information, available languages, monitor information, IPv4, the malware file location, Redline Stealer infection history, and monitor screenshots where each item of Credential Details (Entity1) excluding monitor screenshots is stored empty.

Leak Environment Details

Environment Details stored in *Entity7 result* prepares to access service via *Id6()* method. Thereafter, the collected information is leaked by calling the defined *[OperationContract] Id4()* method. Upon receiving the leaked information, the C2 Server sends a response to the infected PC, which is stored and delivered in Entity13. The response type can be divided into *Unknown(Entity13.Id1)*, *Success(Entity13.Id2)*, *RepeatPart(Entity13.Id3)*, *NotFound(Entity13.Id4)*

Collect & Leak Credential Details

‘Credential Details’ are leaked whenever one item is collected. The information leakage process is the same as ‘Environment Details’ leakage process, but there is a difference in the information delivered to the C2 Server. Each item of Credential Details is leaked by calling the matched [OperationContract] Id#() method. When the C2 Server receives information, it sends a response to the infected PC, which is the same type of response it receives when Environment Details is leaked.

Result: Collected Information

Targets collected by Redline Stealer are largely divided into infection device information, installation information, crypto wallet information, account information, User Data information, and local file information. In the case of crypto wallet information, in addition to the crypto wallet list specified in configuration data, the installed Browser Extension Wallet list is checked to collect related information. The table summarizing the collected information by type is as follows.

| Type | Collected information | | |
|-------------------------|---|--|--------------------------------------|
| Infection Device | - Username - Monitor Size - OS version | - Language - Malware File Location - Process | - HW Serial - Time zone - IPv4 |
| Hardware | - Processor - Graphic - Memory | | |
| Installation | [Browser] | [SW] | [Anti-Virus] |
| | - Name - Version - Path | - Name - Version | - Name |
| Crypto Wallets | - *wallet* file - wallet.dat file | | |
| Accounts | [FTP] | [Browser] | [VPN] |
| | - Port - Username - Password | - Name - Profile - Login - Autofill - CC - Cookie | - URL - Username - Password |
| User Data | [Telegram] | [Discord] | [Steam] |
| | - All files in tdata folder | - Token.txt file | - *ssf* files - *.vdf files |
| Local Files | - Files in Desktop / Documents (keyword extension: *.txt, *.doc*, *key*, *wallet*, *seed*) | | |

Summarizing of collected information by type

Conclusion

- Redline Stealer is one of the most popular infostealers along with Vidar, Raccoon, and Ficker.
- Logs stolen through Redline Stealer are the most traded logs on DDW Forums.

- Redline Stealer has been updating versions until recently, and continuous analysis is needed in that the structure of Redline Stealer is gradually changing according to major updates.

Appendix

Description of each Operation Contract function.

| Name | Description | Name | Description |
|--------|---------------------------------|--------|--------------------------|
| Id1() | Connect to the C2 Server | Id13() | Send Browser Info |
| Id2() | Get Configuration data | Id14() | Send Files |
| Id3() | Send Entity7 (Send Log by Full) | Id15() | Send Scanned Wallets |
| Id4() | Send Environment Details | Id16() | Send Stream |
| Id5() | Send Screenshot File | Id17() | Send Nord VPN |
| Id6() | Send Security Utils | Id18() | Send Open VPN |
| Id7() | Send Available Language | Id19() | Send Proton VPN |
| Id8() | Send Installed Software | Id20() | Send Telegram |
| Id9() | Send Process List | Id21() | Send Discord |
| Id10() | Send System Hardware | Id22() | Connect to the C2 Server |
| Id11() | Send Browsers | Id23() | Connect to the C2 Server |
| Id12() | Send FTP | Id24() | Connect to the C2 Server |

Chromium-based Browser List

Battle.net, Chromium, Chrome, Opera Software, ChromePlus, Iridium, 7Star, CentBrowser, Chedot, Vivald

Gecko-based Browser List

Firefox, Waterfox, K-Meleon, Thunderbird, Comodo, Cyberfox, BlackHaw, Pale Moon

Browser Extension Wallet List

YoroiWallet, Tronlink, NiftyWallet, MetaMask, Coinbase, BinanceChain, BraveWallet, GuardaWallet, Eq

Methods collecting Environment Details

| Field/Method | Description |
|--------------------------------|--|
| ReleaseID | Set Redline Stealer Unique ID value |
| GetDefaultIPv4Address() | Get IPv4 of the infected PC |
| Visible() | Check Redline Stealer infection history |
| EntityResolver.First | Store Client-related information of the infected PC |
| GetUsername() | Get Username logged on OS |
| GetMonitorSize() | Get Monitor Size: {Weight x Height} |
| GetLang_OSver() | Get language and OS version currently used by the system |
| GetAssemblyLocation() | Get the location of the assembly containing the currently running code |
| GetHWSerial | Domain + Username + HW Serial |
| GetTZ() | Get the time zone currently used by the system |
| EntityResolver.Main | Store Credential Details information |
| GetScreenshots() | Take screenshots of the monitor |

Methods collecting Credential Details

| Method | Description |
|-----------------------|---|
| GetHardware() | Collect and leak Processor, Graphic, Memory related information |
| GetInstalledBrowser() | Collect and leak installed browsers' name, path, and version stored in Registry |
| GetInstalledSW() | Collect and leak installed software's' name, version stored in Registry |
| GetAnti() | Collect and leak installed Antivirus Products' name |
| GetProcessList() | Collect and leak the process information currently running |
| GetLang() | Collect and leak the available languages installed in the system |
| GetTelegramFiles() | Collect and leak Telegram setting, cache files |
| GetBrowserData() | Collect and leak Chromium / Gecko Browsers' Credential |
| GetFiles() | Collect and leak file information located in Desktop / Documents folder |
| GetFTPFiles() | Collect and leak FTP access history and administrator information |
| GetCryptoWallets() | Collect and leak Crypto Wallet address and private key |
| GetDiscordTokens() | Collect and leak Discord Token file |
| GetSteamFiles() | Collect Steam user auth and config related information |
| GetVPN() | Collect Nord / Open / Proton VPN's user login information |

Redline Stealer IoCs

- d81d3c919ed3b1aaa2dc8d5f9cf382 | cd3f0808ae7fc8aa5554192ed5b0894779bf88a9c56a7c317ddc6a4d7c249e0e
- af90600728c9d3d1270dd4da39a0f9e5 | 38a5b96fd07f03041f6eff913b85fc621fa314e1de87326accb00ee218c37756
- d6e630749bdd4f16c37ca15886fc6bdc | 020fbe48b4da34a90d3422f211aa0338681a7cb9e99292b2b9d738a354ed97de
- ce70574f6c90835076d9b195e90cd275 | c6d48514031cc6e83445b95f9ed4e975f2cdcebc2e9cc1914605058ff7af7764
- 10adb0969eb2b385d6bb8ad8e91bb0c4 | 9ac01cc861cfe9e340c66a5cd527ab8a7e3de345b851ebcf07a7ca08eeee2f88

Source: <https://medium.com/s2wblog/deep-analysis-of-redline-stealer-leaked-credential-with-wcf-7b31901da904>