باسمه تعالی

تحلیل فنی باج افزار CryptConsole-۲۰۱۸

مقدمه :

مشاهده و رصد فضای سایبری در روزهای اخیر در زمینه باج‌افزار، از شروع فعالیت نمونه جدیدی ب نام CryptConsole-۲۰۱۸ خبر می‌دهد. این باج‌افزار به نام CryptConsole-Sequre نیز شناخته می‌شود. بررسی‌ها نشان می‌دهد فعالیت این باج‌افزار در نیمه‌ی دوم ماه آوریل سال ۲۰۱۸ میلادی شروع شده و به نظر می‌رسد تمرکز آن بیشتر بر روی کاربران انگلیسی زبان می‌باشد. این باج‌افزار که هم اکنون در حال توسعه می‌باشد، ویژگی جالبی دارد که آن را از دیگر باج‌افزارها متمایز می‌کند، بدین صورت که پس از نفوذ به سیستم قربانی، به جای اجرای مستقیم یک پردازه، باج افزار یک کد رمزگذاری شده به زبان C# را کامپایل می‌کند و آن را به طور مستقیم در حافظه به اجرا در می‌آورد.

مشخصات فایل اجرایی :

نام فایل	smsss.exe
MD۵	c۳۵۵۰۶bd۳fedad۵۷e۷f۱ea۹۷۵ebcaec۵
SHA-۱	۰۹۷۷۶۷۶ae۸c۸۷۱۶۸۲٤a۱۳۰۳۷c۷eb٤c۷b۹۵c۵۸ae۷
SHA-۲۵۶	۲۰۸cca۱۲٤ddafe۳۵a۱۲۲f٦bdd۳٦۱۹۱۱۵۱a۲۷۳۰b٤e۱۰۵۱۸۰٤d۵f٦۸d۰cb٤b٤٤۱٤۵
اندازه فایل	۱۱۰.۵ KB
کامپایلر	Microsoft visual C# v۷.۰ / Basic .NET

فایل اجرایی این باج افزار دارای سه بخش است :

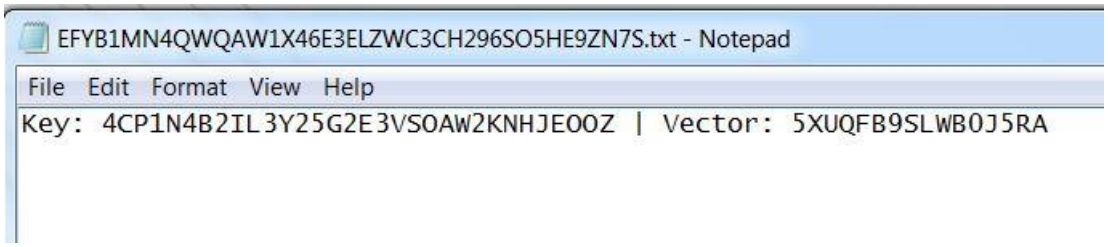
نام بخش	آنتروپی	آدرس مجازی	اندازه مجازی	اندازه خام
.text	۳.۲۹	۸۱۹۲	۱۱۰۵۰۰	۱۱۰۵۹۲
.rsrc	۳.٦۸	۱۲۲۸۸۰	۱۲۳۲	۱۵۳٦
.reloc	۰.۱	۱۳۱۰۷۲	۱۲	۵۱۲

تحلیل پویا :

برای بررسی عمیق‌تر باج‌افزار CryptConsole-۲۰۱۸، فایل اجرایی آن را در محیط آزمایشگاهی اجرا کردیم تا عملکرد باج‌افزار را از نزدیک مورد بررسی قرار دهیم. نتایج حاصل از این بررسی نشان داد که باج‌افزار مورد اشاره، پس از اجرا، دو فایل بر روی دسکتاپ قرار می دهد :

۱. فایل HOW DECRIPT FILES.hta که مربوط به پیغام باج‌خواهی می‌باشد. این فایل در تمامی پوشه‌ها نیز ایجاد می‌شود.

۲. فایل TFGV٥KV٣T٢V٢SXD٤V٩E٣O٤١JA٠MM٠٠٢٥BUDWDGZB.txt که این نام این فایل، کد شناسایی سیستم قربانی را نشان می دهد. محتوای این فایل در تصویر زیر قابل مشاهده است :



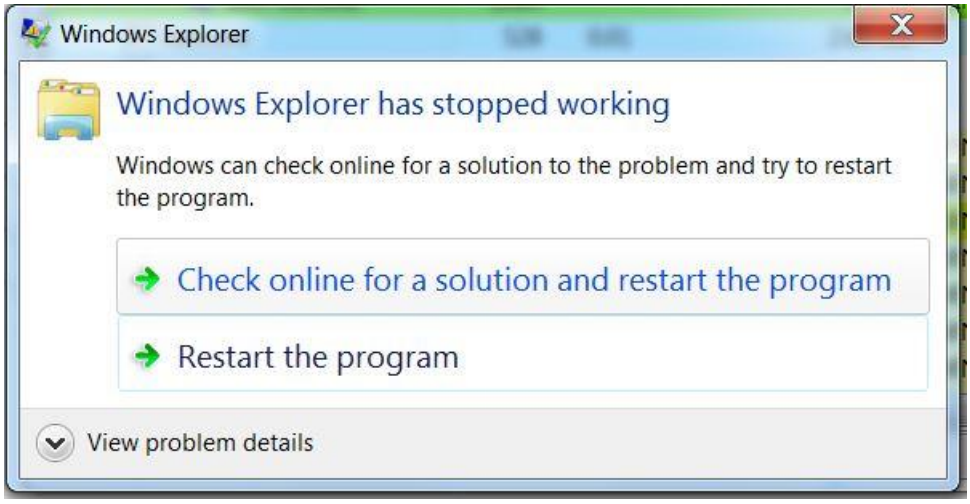
پس از آن، باج‌افزار به فعالیت خود جهت رمزگذاری فایل‌ها ادامه می‌دهد. باج‌افزار برای رمزگذاری فایل‌ها از الگوریتم رمزنگاری AES استفاده می‌کند و پس از رمزگذاری، نام تمام فایل‌های رمزگذاری شده را به شکل زیر تغییر می‌دهد :

sequre@tuta.io_ [random_۰-۹A-Z]

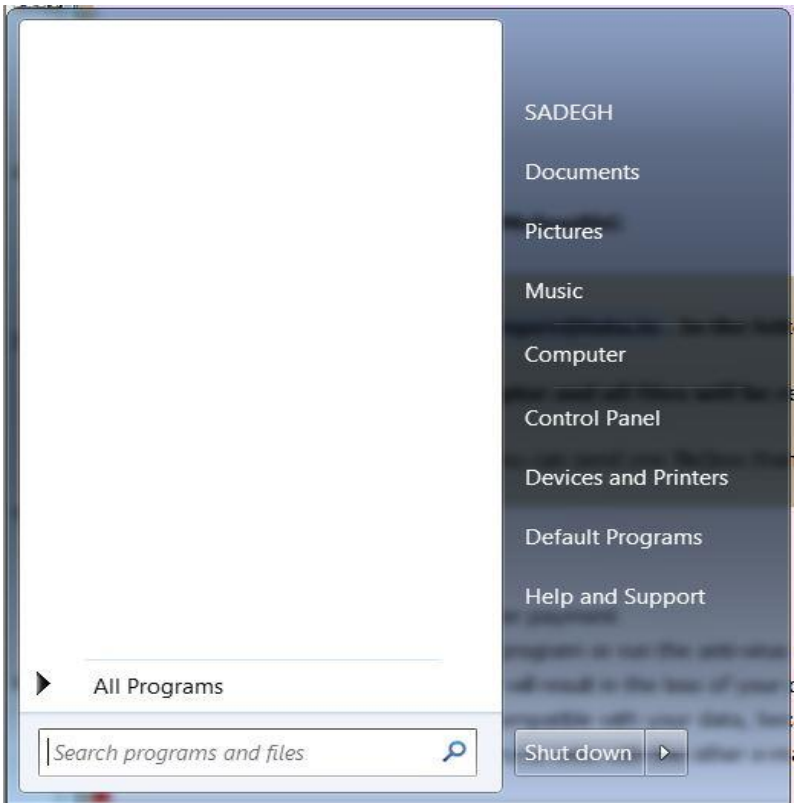
بررسی ها نشان می دهد که این باج‌افزار به جز فایل‌های مربوط به ویندوز که در پوشه‌های زیر وجود دارند تمامی انواع فایل‌های دیگر که بر روی سیستم موجود می‌باشند را رمزگذاری می‌کند :

Common Files\Services, Common Files\SpeechEngines, DVD Maker, internet explorer, Reference Assemblies, Windows Defender, Windows Journal, Windows Mail, Windows Media Player, windows NT, Windows Photo Viewer, Windows Portable Devices, Windows Sidebar

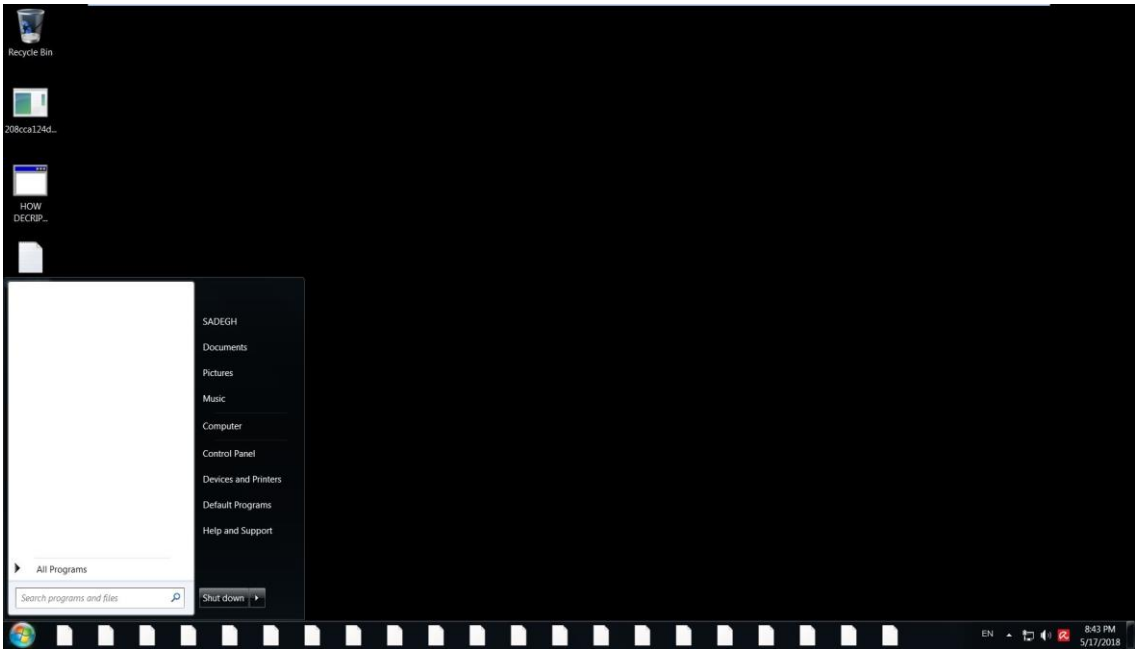
باج افزار پس از اجرا، فعالیت Windows Explorer را متوقف کرده و پیغام زیر به نمایش در می آید:



همچنین تمام پوشه‌ها و فایل‌های موجود در مسیرهای Libraries و Recycle Bin و ابزارهای کاربردی مربوط به ویندوز نیز حذف می‌شوند. پس از اتمام فرایند رمزگذاری فایل‌ها، اجرای باج‌افزار پایان می‌یابد و پیغام باج‌خواهی به نمایش در می‌آید. به دلیل رمزگذاری دایرکتوری مربوط به نرم‌افزارهای نصب شده بر روی سیستم قربانی هیچ یک از آن‌ها دیگر قابل استفاده نخواهند بود تصاویر زیر مربوط به اثرات باج‌افزار بر روی سیستم قربانی می‌باشد :

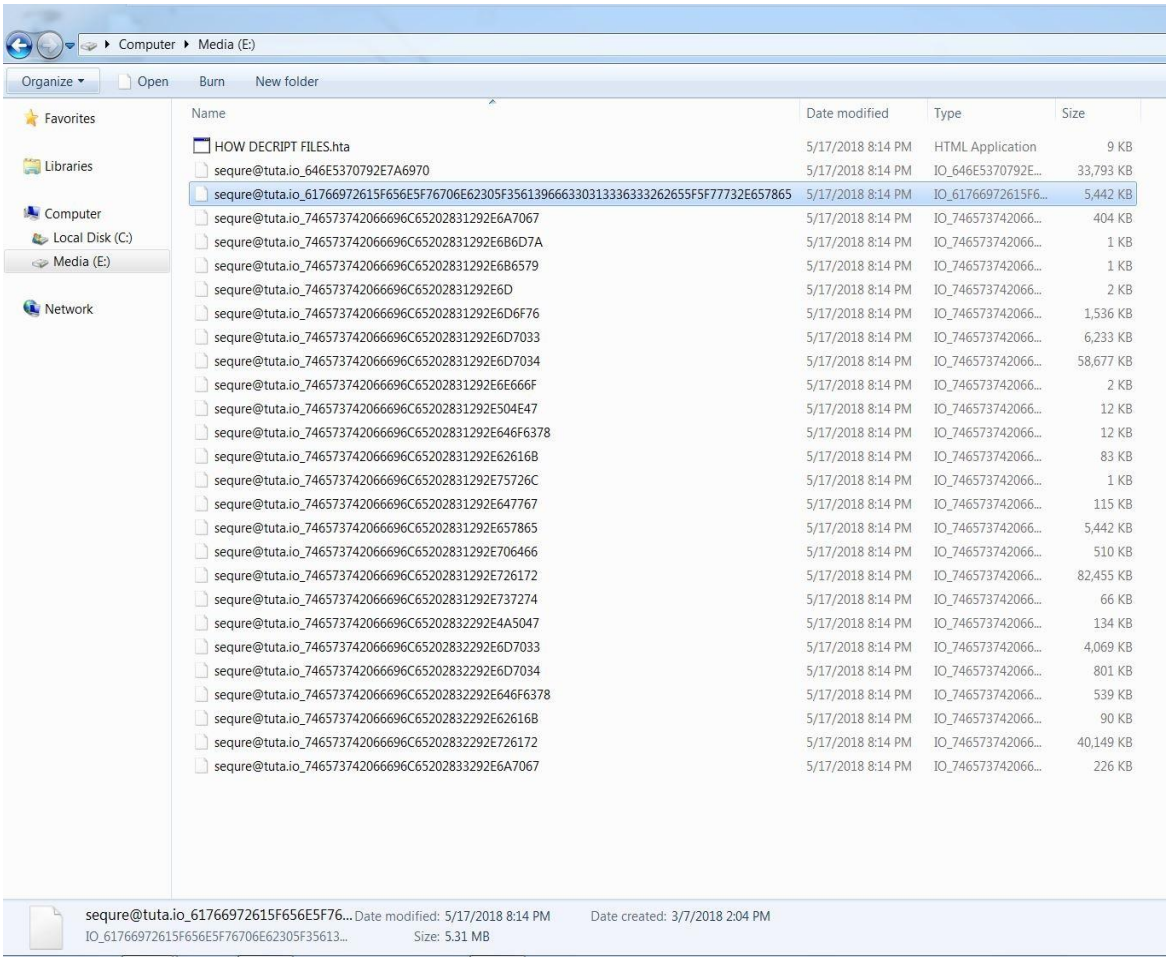


تصویر ۱: حذف تمام ابزارهای کاربردی مربوط به ویندوز توسط باج‌افزار

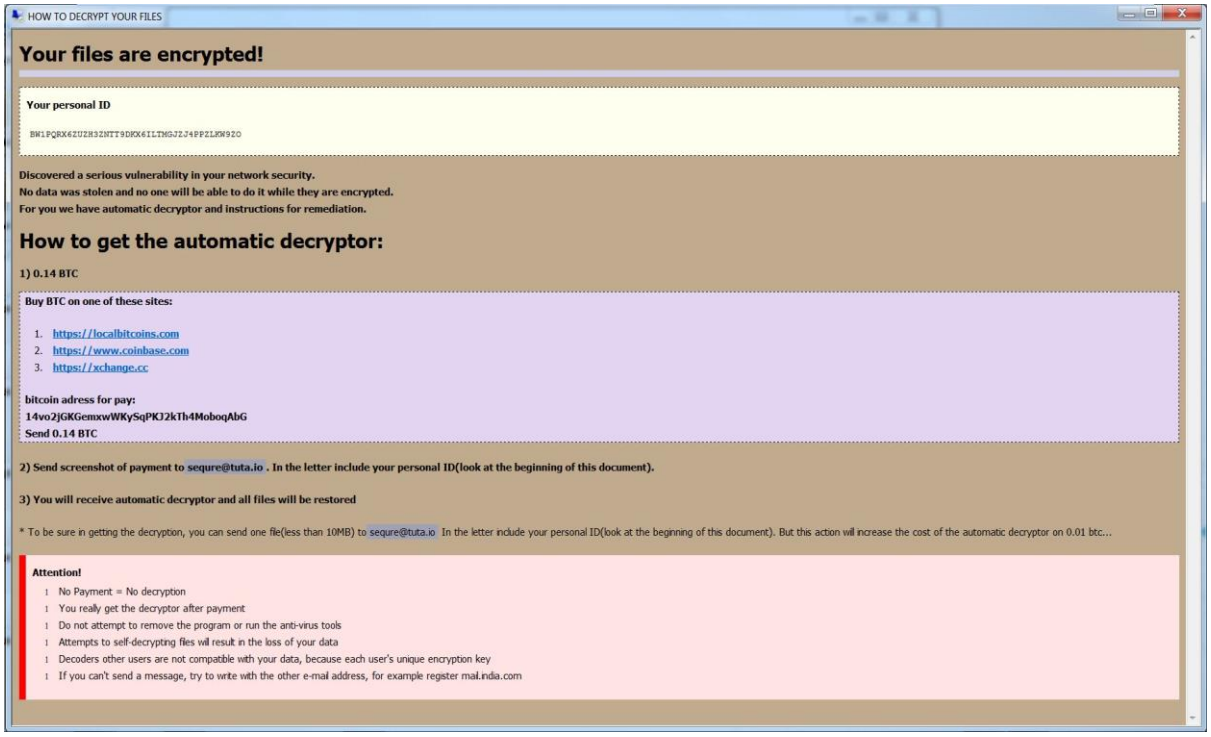


تصویر ۲: در صورت ری‌استارت ویندوز، محیط آن به شکل بالا تغییر می‌کند.

تصویر زیر نشان‌دهنده فایل‌های رمزگذاری شده توسط این باج‌افزار می‌باشد.



همانطور که در تصویر فوق قابل مشاهده است، نام فایل‌ها طبق الگویی که اشاره شد، تغییر پیدا می‌کند و یک فایل به نام HOW DECRIPT FILES.hta در کنار فایل‌های رمزگذاری شده ایجاد می‌شود که شامل پیغام باج‌خواهی می‌باشد. در تصویر زیر پیغام باج‌خواهی باج‌افزار CryptConsole-۲۰۱۸ را مشاهده می‌کنید.



بر اساس پیغام باج‌خواهی، مهاجمین اعلام نموده‌اند که فایل‌ها رمزگذاری شده‌اند و قربانیان برای رمزگشایی آن‌ها باید مبلغ ۰.۱٤ بیت‌کوین را از طریق کیف پول بیت‌کوین، به آدرس ۱٤vo۲jGKGemxwWKySqPKJ۲kTh٤MoboqAbG برای مهاجمین ارسال نمایند. قربانیان پس از پرداخت مبلغ باج، می بایست یک تصویر از مبلغ پرداخت شده و کد شناسایی خود را از طریق آدرس ایمیل sequre@tuta.io برای مهاجمین ارسال نمایند. پس از تایید مبلغ پرداختی، ابزار رمزگشایی برای قربانیان ارسال خواهد شد. ضمناً برای جلب اعتماد قربانیان، امکان رمزگشایی چند فایل قبل از پرداخت مبلغ باج‌خواهی نیز فراهم شده که در صورت تمایل به استفاده از این روش، ۰.۰۱ به مبلغ باج‌خواهی افزوده خواهد شد. مهلت پرداخت باج در پیغام باج‌خواهی مشخص نشده و در صورتی که هیچ پرداختی انجام نشود، رمزگشایی نیز در کار نخواهد بود و هر گونه تلاش برای رمزگشایی فایل‌ها، حذف نمودن باج‌افزار توسط ابزارهای امنیتی مانند آنتی‌ویروس‌ها و ... باعث حذف فایل‌ها خواهد شد. طبق بررسی‌های انجام شده کیف پول مربوط به این باج‌افزار تاکنون هیچ تراکنشی نداشته است.

Summary	
Address	14vo2jGKGemxwWKySqPKJ2kTh4MoboqAbG
Hash 160	2b14d8d9bc482ed9cc3d883489089b3f387e2f83
Tools	Related Tags - Unspent Outputs

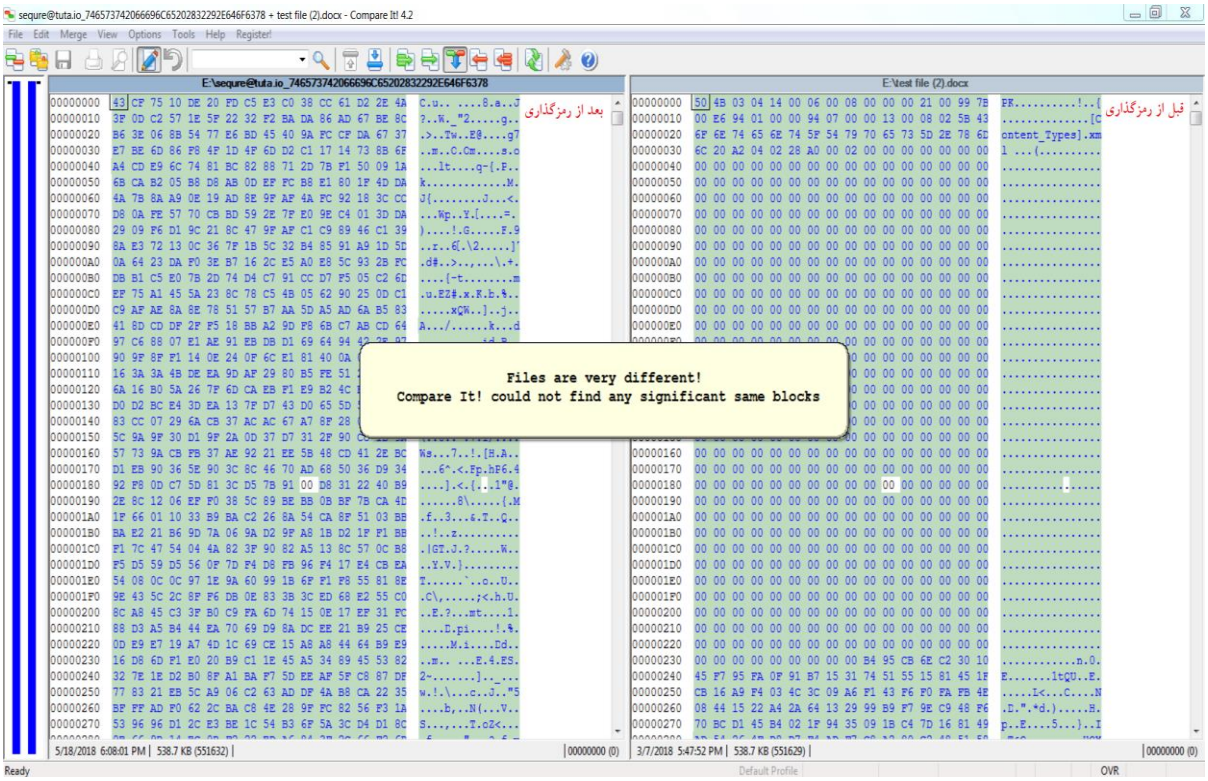
Transactions	
No. Transactions	0
Total Received	0 BTC
Final Balance	0 BTC

Request Payment Donation Button

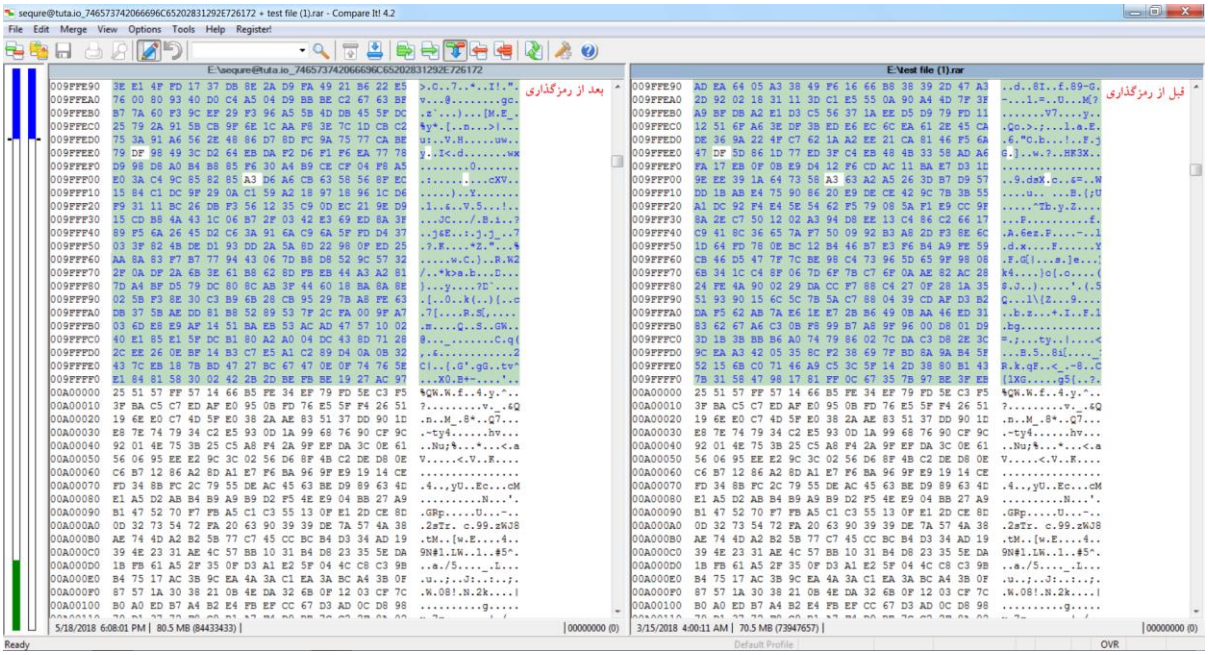
تحلیل ایستا:

پس از تحلیل کدمنبع باج‌افزار CryptConsole-۲۰۱۸ به نتایج زیر دست پیدا کردیم.

طبق بررسی‌هایی که بر روی فایل‌های مختلف قبل و بعد از رمزگذاری توسط باج‌افزار انجام دادیم شاهد این بودیم که باج‌افزار CryptConsole-۲۰۱۸ ساختار تمام فایل‌ها را به یک شکل رمزگذاری نمی‌کند و در مواجهه با فایل‌های مختلف رفتار متفاوتی از خود نشان می دهد. بدین صورت که ساختار بعضی از فایل‌ها را پس از رمزگذاری کاملا تغییر می‌دهد اما در مورد برخی از فایل‌ها فقط قسمتی از ساختار آن‌ها را تغییر می‌دهد، نتایج این بررسی‌ها در تصاویر زیر قابل مشاهده است :

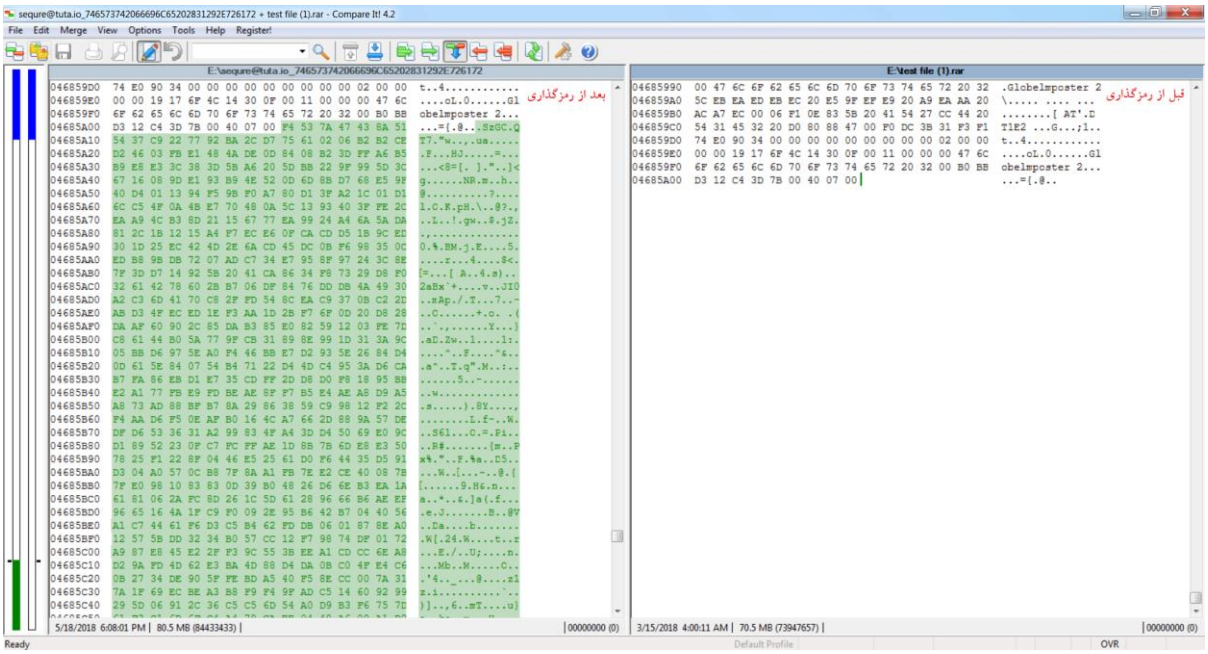


تصویر ۱ تمام ساختار فایل تغییر کرده است.



تصویر ۲ فقط بخشی از ساختار فایل تغییر کرده است.

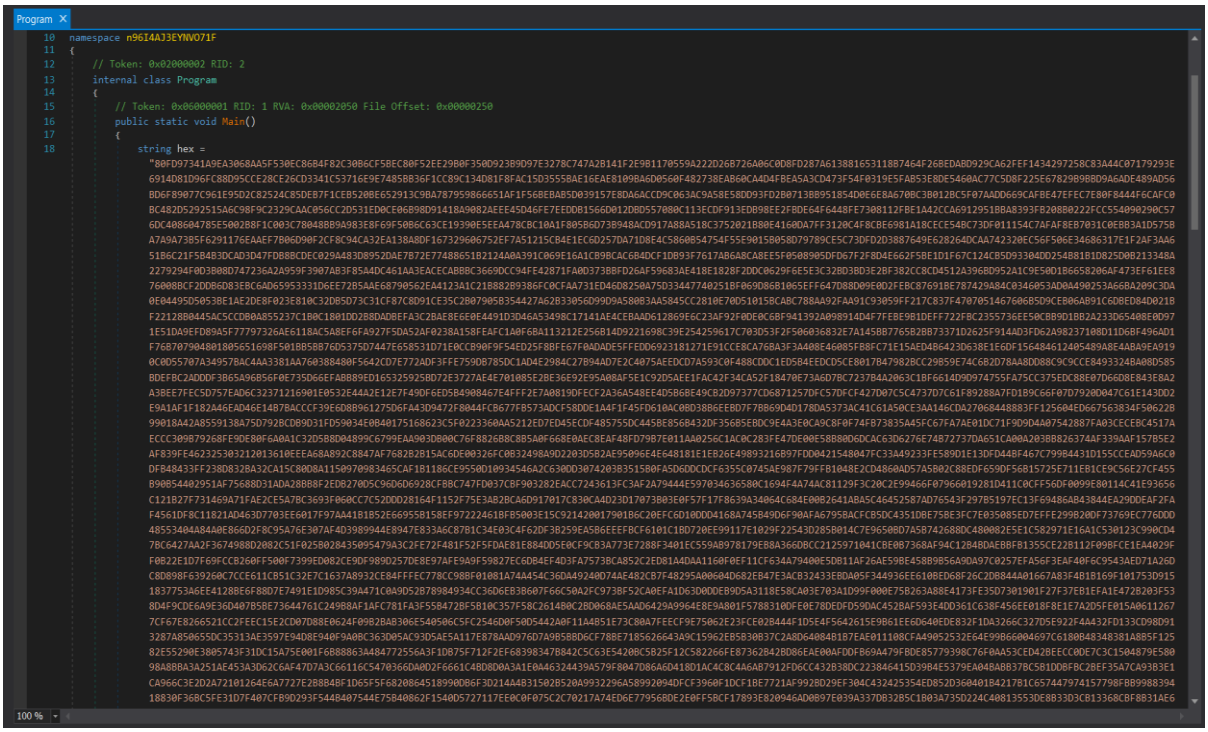
همچنین مشخص شد که پس از رمزگذاری به انتهای فایل‌ها پسوند [random_۰-۹A-Z] _io. اضافه می‌شود، این تغییر به خوبی در تصویر زیر قابل مشاهده است :



همچنین حجم برخی از فایل‌ها که ساختار آن‌ها به صورت کامل تغییر پیدا نمی‌کند، پس از رمزگذاری افزایش پیدا می‌کند.

طبق بررسی‌های انجام شده اکثر آنتی‌ویروس‌های معتبر، این باج‌افزار را به عنوان یک تروجان شناسایی نموده‌اند. لذا احتمال نفوذ باج‌افزار به سیستم از راه های متداول از جمله هرزنامه‌ها وجود دارد.

همان‌طور که اشاره شد این باج‌افزار شامل یک رشته رمزگذاری شده می‌باشد که در یک Dropper تعبیه شده که پس از حمله به سیستم قربانی، این رشته رمزگذاری شده کامپایل می‌شود و فایل‌ها را رمزگذاری می‌کند. تصویر زیر مربوط به بخشی از این کد رمزگذاری شده می‌باشد :



این رشته توسط یک کلید، رمزگشایی می‌شود که قطعه کد زیر مربوط به فرایند رمزگشایی و مقادیر کلید رمزگشایی، می‌باشد :

```
func7PLUOCHPZY017(byte[], byte[], byte[]...
1  // n96I4AJ3EYNVO71F.Program
2  // Token: 0x06000002 RID: 2 RVA: 0x000020B8 File Offset: 0x000002B8
3  private static byte[] func7PLUOCHPZY017(byte[] encrypted, byte[] key, byte[] vector)
4  {
5      byte[] result;
6      using (RijndaelManaged rijndaelManaged = new RijndaelManaged())
7      {
8          rijndaelManaged.Key = key;
9          rijndaelManaged.IV = vector;
10         result = Program.funcXOUUTM54E67A(rijndaelManaged, encrypted);
11     }
12     return result;
13 }
14
```

```
.cctor() : void
1  // n96I4AJ3EYNVO71F.Program
2  // Token: 0x06000009 RID: 9 RVA: 0x000024F0 File Offset: 0x000006F0
3  static Program()
4  {
5      // Note: this type is marked as 'beforefieldinit'.
6      Program.key = "B9Y2WUYKV1ZCAQQOTTKHSQ4ZRE39IX7J";
7      Program.vector = "D4P9IHJHLWXCLH3Y";
8  }
9
```

اکنون رشته رمزگشایی شده است و آماده کامپایل شدن می‌باشد. در تصاویر زیر قطعه کدهای مربوط به فرایند کامپایل شدن کد و دیگر فرایندهای مورد نیاز آمده است :

```
func8YVJMH5AR86A52(string[]) : void
1  // n96I4AJ3EYNVO71F.Program
2  // Token: 0x06000005 RID: 5 RVA: 0x00002298 File Offset: 0x00000498
3  private static void func8YVJMH5AR86A52(string[] code)
4  {
5      CompilerParameters compilerParameters = new CompilerParameters();
6      string currentDirectory = Directory.GetCurrentDirectory();
7      compilerParameters.GenerateInMemory = true;
8      compilerParameters.TreatWarningsAsErrors = false;
9      compilerParameters.GenerateExecutable = false;
10     compilerParameters.CompilerOptions = "/optimize";
11     string[] value = new string[]
12     {
13         "System.dll",
14         "System.Core.dll",
15         "mscorlib.dll"
16     };
17     compilerParameters.ReferencedAssemblies.AddRange(value);
18     CSharpCodeProvider csharpCodeProvider = new CSharpCodeProvider();
19     CompilerResults compilerResults = csharpCodeProvider.CompileAssemblyFromSource(compilerParameters, code);
20     if (compilerResults.Errors.HasErrors)
21     {
22         string text = "Compile error: ";
23         foreach (object obj in compilerResults.Errors)
24         {
25             CompilerError compilerError = (CompilerError)obj;
26             text = text + "rn" + compilerError.ToString();
27         }
28         throw new Exception(text);
29     }
30     Module module = compilerResults.CompiledAssembly.GetModules()[0];
31     Type type = null;
32     MethodInfo methodInfo = null;
33     if (module != null)
34     {
35         type = module.GetType("n96I4AJ3EYNVO71FC.Program");
36     }
37     if (type != null)
38     {
39         methodInfo = type.GetMethod("Main");
40     }
41     if (methodInfo != null)
42     {
43         methodInfo.Invoke(null, null);
44     }
45 }
100 %
```

تصویر ۱

ExpoloreAssembly(Assembly) : void

```
// n96I4AJ3EYNVO71F.Program
// Token: 0x06000006 RID: 6 RVA: 0x00002428 File Offset: 0x00000628
private static void ExpoloreAssembly(Assembly assembly)
{
    Console.WriteLine("Modules in the assembly:");
    foreach (Module module in assembly.GetModules())
    {
        Console.WriteLine("{0}", module);
        foreach (Type type in module.GetTypes())
        {
            Console.WriteLine("t{0}", type.Name);
            foreach (MethodInfo methodInfo in type.GetMethods())
            {
                Console.WriteLine("tt{0}", methodInfo.Name);
            }
        }
    }
}
```

تصویر ۲

funcXOUUTM54E67A(SymmetricAlgorith...

```
// n96I4AJ3EYNVO71F.Program
// Token: 0x06000003 RID: 3 RVA: 0x0000210C File Offset: 0x0000030C
private static byte[] funcXOUUTM54E67A(SymmetricAlgorithm alg, byte[] message)
{
    byte[] result;
    if (message == null || message.Length == 0)
    {
        result = message;
    }
    else
    {
        if (alg == null)
        {
            throw new ArgumentNullException("alg");
        }
        using (MemoryStream memoryStream = new MemoryStream())
        {
            using (ICryptoTransform cryptoTransform = alg.CreateDecryptor())
            {
                using (CryptoStream cryptoStream = new CryptoStream(memoryStream, cryptoTransform, CryptoStreamMode.Write))
                {
                    cryptoStream.Write(message, 0, message.Length);
                    cryptoStream.FlushFinalBlock();
                    result = memoryStream.ToArray();
                }
            }
        }
    }
    return result;
}
```

تصویر ۳

StringToByteArray(string) : byte[]

```
// n96I4AJ3EYNVO71F.Program
// Token: 0x06000004 RID: 4 RVA: 0x0000222C File Offset: 0x0000042C
private static byte[] StringToByteArray(string hex)
{
    return (from x in Enumerable.Range(0, hex.Length)
    where x % 2 == 0
    select Convert.ToByte(hex.Substring(x, 2), 16)).ToArray<byte>();
}
```

تصویر ٤

باج‌افزار CryptConsole-۲۰۱۸ فقط از کتابخانه ویندوزی زیر به همراه یک تابع از آن، استفاده می‌کند.

mscoree.dll
_CorExeMain

بر اساس تحلیل های صورت گرفته، این باج‌افزار فقط یک فرایند ایجاد می‌کند که آن هم به نام خود باج‌افزار می‌باشد :

HONESTSample_٥aec٨b٧f١٠٤١٩f٣٩٧٠٩٩d٣d٢.exe

- csc.exe /noconfig /fullpaths @"%TEMP%\etzpcjmo.cmdline"
- cvtres.exe /NOLOGO /READONLY /MACHINE:IX٨٦ "/OUT:%TEMP%\RESFA٢١.tmp" "%TEMP%\CSC٥٠٩BE٩٤١A٦E٧٤D٣١٩٩٧٨F٣C٨٦٣٢٢٠EA.TMP"

تحلیل ترافیک شبکه :

پس از بررسی ترافیک شبکه، متوجه هیچ گونه درخواست DNS و تلاش برای برقراری ارتباط با میزبان در نقطه‌ی جغرافیایی خاص توسط باج‌افزار CryptConsole-۲۰۱۸ نشدیم.

شناسایی :

در حال حاضر تعداد ۵۳ مورد از ٦٥ آنتی ویروس و آنتی بدافزار موجود در سامانه VirusTotal قادر به شناسایی این باج افزار بوده وآن را حذف یا غیرفعال می‌کنند.

Ad-Aware	Gen:Variant.Razy.178861	AegisLab	Troj.Ransom.W32!c
AhnLab-V3	Trojan/Win32.Ransom.C2477960	ALYac	Trojan.Ransom.CryptConsole
Antiy-AVL	Trojan[Ransom]/Win32.AGeneric	Arcabit	Trojan.Razy.D2BAAD
Avast	Win32:Malware-gen	AVG	Win32:Malware-gen
Avira	TR/Dropper.MSIL.Gen	AVware	Trojan.Win32.Generic!BT
Baidu	Win32.Trojan.WisdomEyes.16070401....	BitDefender	Gen:Variant.Razy.178861
CAT-QuickHeal	Trojan.IGENERIC	ClamAV	Win.Trojan.Agent-6520577-0
Comodo	UnclassifiedMalware	CrowdStrike Falcon	malicious_confidence_90% (W)
Cylance	Unsafe	Cyren	W32/Trojan.GMCD-6663
Emsisoft	Trojan.FileCoder (A)	Endgame	malicious (high confidence)
eScan	Gen:Variant.Razy.178861	ESET-NOD32	a variant of MSIL/GenKryptik.AIKC
F-Secure	Gen:Variant.Razy.178861	Fortinet	MSIL/GenKryptik.AIKC!tr
GData	Gen:Variant.Razy.178861	Ikarus	Trojan.MSIL.Krypt
Jiangmin	Trojan.Generic.cbxuv	K7AntiVirus	Trojan (00516d6a1)
K7GW	Trojan (00516d6a1)	Kaspersky	HEUR:Trojan-Ransom.Win32.Generic
Malwarebytes	Ransom.FileCryptor	MAX	malware (ai score=99)
McAfee	Generic.dsa	McAfee-GW-Edition	BehavesLike.Win32.Generic.cz
Microsoft	Ransom:Win32/Genasom	NANO-Antivirus	Trojan.Win32.GenKryptik.fakygk
Palo Alto Networks	generic.ml	Panda	Trj/GdSda.A
Qihoo-360	Win32/Trojan.0ad	Rising	Ransom.Generic!8.E315 (CLOUD)
SentinelOne	static engine - malicious	Sophos AV	Mal/Generic-S
Sophos ML	heuristic	Symantec	Trojan.Gen.2
Tencent	Win32.Trojan.Generic.Swkj	TrendMicro	Ransom_SEQUR.THDBGAH
TrendMicro-HouseCall	Ransom_SEQUR.THDBGAH	VBA32	TScope.Trojan.MSIL
VIPRE	Trojan.Win32.Generic!BT	ViRobot	Trojan.Win32.Z.Razy.113152.CW
Yandex	Trojan.GenKryptik!	Zillya	Trojan.GenKryptik.Win32.15891
ZoneAlarm	HEUR:Trojan-Ransom.Win32.Generic	Avast Mobile Security	Clean