


باسمه تعالی

تحلیل فنی باج افزار KeyPass

مقدمه :

مشاهده و رصد فضای سایبری در زمینه باج افزار، از شروع فعالیت نمونه جدیدی به نام KeyPass خبر می دهد. بررسی ها نشان می دهد که فعالیت این باج افزار در اوایل ماه آگوست سال ۲۰۱۸ میلادی شروع شده است و به نظر می رسد از خانواده باج افزار STOP باشد. این باج افزار از الگوریتم رمزنگاری AES-۲۵۶ برای رمزگذاری فایل ها استفاده می کند و پس از رمزگذاری، پسوند فایل ها را به "KEYPASS" تغییر می دهد و از قربانیان تقاضای پرداخت مبلغ ۳۰۰ دلار به عنوان باج می کند. در حال حاضر روش ورود یا انتشار این باج افزار دقیقاً مشخص نیست اما طبق گزارشات بدست آمده از کاربران در نقاط مختلف جهان، احتمالاً از طریق وبسایت هایی که نرم افزارهای کرک شده و یا قفل شکسته ارائه می دهند منتشر می شود. ضمناً باج افزار مورد در حال حاضر اشاره فاقد رمزگشا بوده و تنها راه مقابله با آن، بازگردانی اطلاعات از طریق فایل های پشتیبان می باشد.

مشخصات فایل اجرایی :

eev۴c۶۳faa۲eb۹۷۰۹b۱d۷۳۸۷۶۲e۲۸۰۷۲aece۲evb۹eeffc۵۹۱۳eb۶a۵fd۱۵۶۴۷۵۲.exe STOP.exe Keypass.exe vatup.exe ۵.exe	نام فایل
۹۰۱d۸۹۳f۶۶۵c۶f۹۷۴۱aa۹۴۰e۵f۲۷۵۹۵۲	MD۵
۳b۵۳۶۹c۰aeffe۵c۰d۰b۱۶۴a۳d۹۰ec۲۴۵b۰۹۳۶۷۴d	SHA-۱
eev۴c۶۳faa۲eb۹۷۰۹b۱d۷۳۸۷۶۲e۲۸۰۷۲aece۲evb۹eeffc۵۹۱۳eb۶a۵fd۱۵۶۴۷۵۲	SHA-۲۵۶
۲.۸۲ MB	اندازه فایل
VCL -> Microsoft Corporation	کامپایلر
	آیکن فایل اجرایی

فایل اجرایی این باج افزار دارای پنج بخش است :

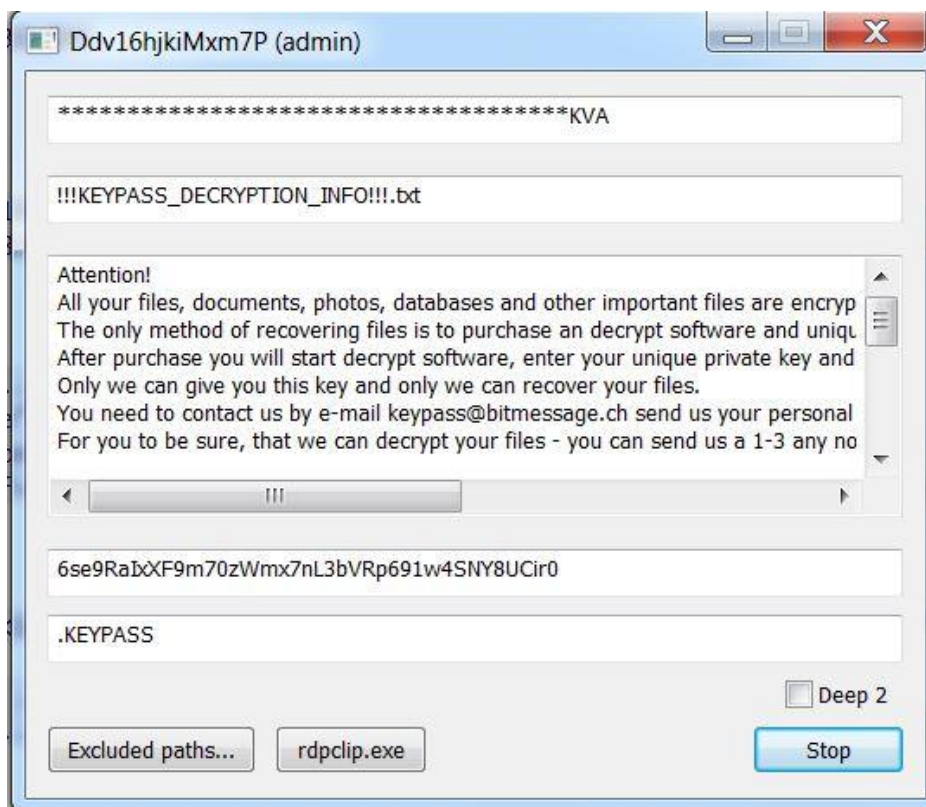
نام بخش	آنتروپی	آدرس مجازی	اندازه مجازی	اندازه خام
.text	۶.۶۳	۴۰۹۶	۲۰۸۱۲۴۹	۲۰۸۱۲۸۰
.rdata	۵.۱۱	۲۰۸۸۹۶۰	۴۸۸۸۹۴	۴۸۸۹۶۰
.data	۴.۹۸	۲۵۸۰۴۸۰	۸۷۵۹۶	۵۰۶۸۸
.rsrc	۴.۵۱	۲۶۷۰۵۹۲	۱۷۶۹۹۲	۱۷۷۱۵۲
.reloc	۶.۵	۲۸۵۰۸۱۶	۱۵۹۲۵۲	۱۵۹۷۴۴

تحلیل پویا :

برای بررسی عمیق‌تر باج‌افزار KeyPass، فایل اجرایی آن را در محیط آزمایشگاهی اجرا کردیم تا عملکرد باج‌افزار را از نزدیک مورد بررسی قرار دهیم. نتایج بدست آمده حاکی از آن است که فرآیند اجرای این باج‌افزار نسبت به سایر باج‌افزارها اندکی متفاوت و پیچیده می‌باشد. باج‌افزار KeyPass پس از ورود به سیستم، ارتباط با سرور فرمان و کنترل بررسی محیط سیستم، اقدام به رمزگذاری فایل‌ها با استفاده از الگوریتم رمزنگاری خود می‌کند. از آنجایی که تمام فایل‌های موجود، رمزگذاری نمی‌گردند، به نظر می‌رسد استثنائاتی در فرآیند رمزگذاری این باج‌افزار وجود دارد. این باج‌افزار حتی فایل اجرایی نرم‌افزارها و میانبرهای آن‌ها را هم رمزگذاری و غیرقابل دسترس می‌نماید.

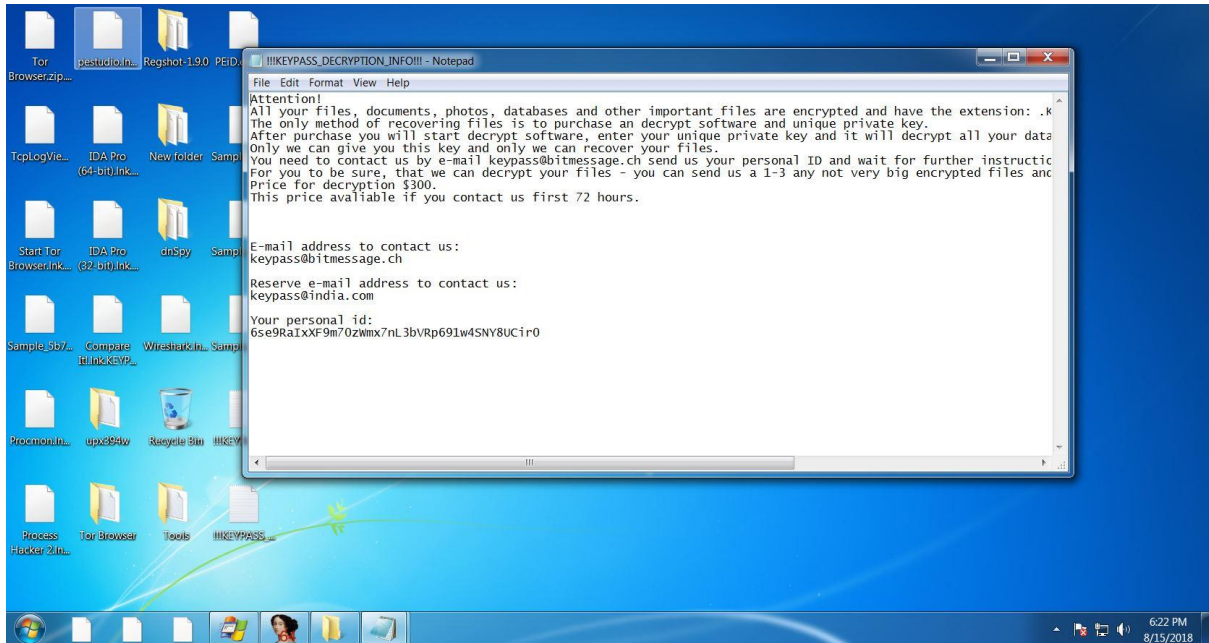
باج‌افزار KeyPass ابتدا فایل اجرایی خود را از مسیر جاری حذف نموده و یک نسخه از آن را در مسیر C:\Users\user\AppData\Local قرار می‌دهد که فرآیند اجرای باج‌افزار از آن مسیر ادامه می‌یابد. سپس فایل حاوی پیغام باج‌خواهی خود که به صورت یک فایل متنی و با عنوان Program Files !!!KEYPASS_DECRYPTATION_INFO!!!.txt می‌باشد را در دایرکتوری‌های مختلفی از جمله Startup ویندوز نیز قرار می‌گیرد تا با هر بار راه‌اندازی ویندوز، مجدداً اجرا گردد. این تغییرات در رجیستری نیز اعمال می‌گردند.

001FFCF8	UNICODE	Software\Microsoft\Windows\CurrentVersion\Policies\ComdIlg32
001FFC08	UNICODE	Software\Microsoft\Windows\CurrentVersion\Policies\Explorer
001FFC80	UNICODE	Software\Microsoft\Windows\CurrentVersion\Policies\Network
0023F736	UNICODE]Software\Microsoft\Windows\CurrentVersion\Run

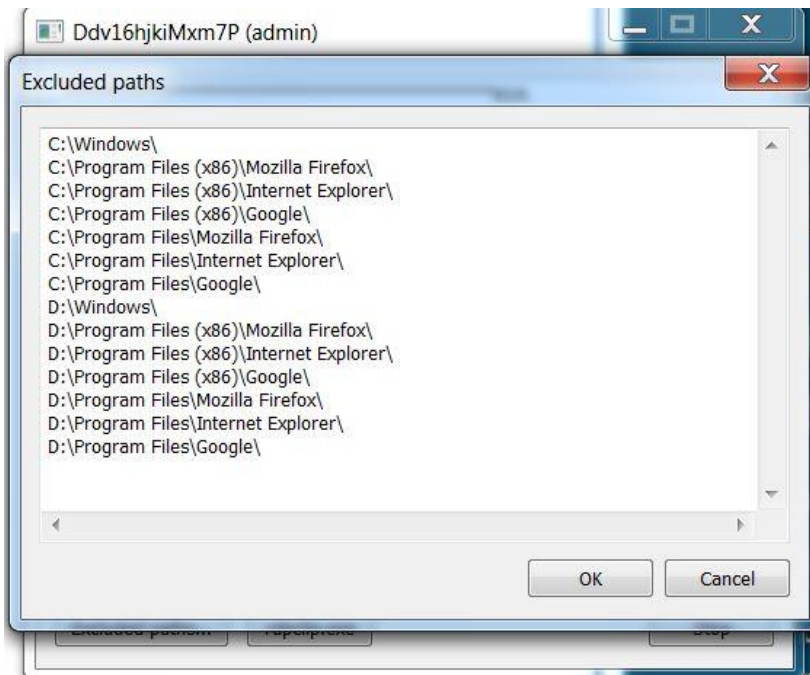


لازم به ذکر است که این باج افزار پس از اتمام فرآیند رمزگذاری، پسوند KEYPASS را به انتهای نام فایل های رمز شده اضافه می کند.

همانطور که قابل ملاحظه است در پیام باج خواهی این باج افزار به این مطلب اشاره شده که همه فایل های مهم سیستم رمز شده اند و قربانی به منظور بازگرداندن فایل های خود، می بایست رمزگشا را از مهاجم خریداری نماید. در ادامه نیز با اشاره به اینکه قربانی می تواند ۱ الی ۳ فایل که حجم زیادی نداشته باشند را همراه با شناسه ای که باج افزار به قربانی اختصاص داده است، به یکی از ایمیل های keypass@bitmessage.ch یا keypass@india.com ارسال نموده و به صورت رایگان فایل اصلی و غیر رمز شده خود را دریافت کند. همچنین مهاجم، مبلغ درخواستی باج را ۳۰۰ دلار اعلام نموده است. بدیهی است که ایمیل دوم به عنوان ایمیل رزرو در نظر گرفته شده است و اگر قربانی در برقراری ارتباط با ایمیل اول موفق نبود، می تواند ایمیل دوم را امتحان کند. تصویر زیر پیغام خواهی باج افزار KeyPass را نشان می دهد :



همانطور که پیشتر اشاره شد، این باج افزار استثنائاتی را در فرآیند رمز گذاری خود رعایت می کند و این استثنائات دایرکتوری های مربوط به مرورگرهای نصب شده بر روی سیستم و نیز دایرکتوری Windows می باشند. نکته ای که می توان یادآور شد آن است که این باج افزار دایرکتوری های مذکور را در درایوهای C و D در نظر گرفته است. بنابراین شرایطی در آن قربانی ویندوز خود را در درایو D نصب کرده باشد نیز در نظر گرفته شده است.

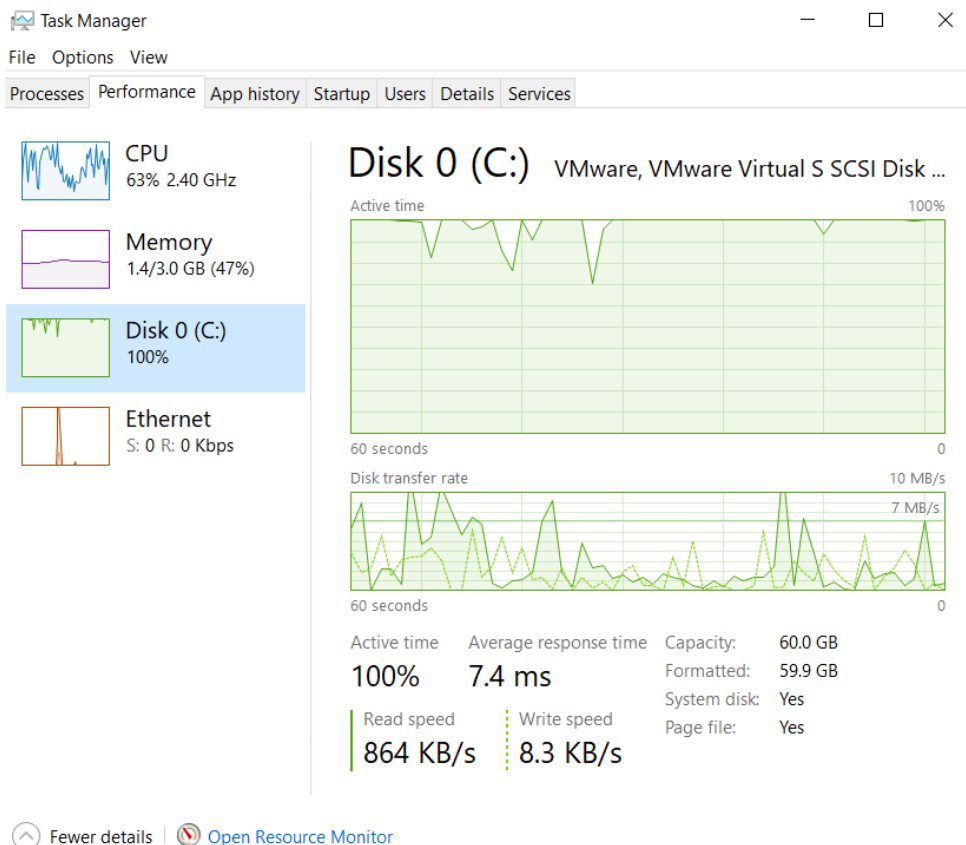


بررسی فعالیت های این باج افزار در سیستم، نحوه انجام فرآیند رمز گذاری برای هر فایل را آشکار می سازد.

Time o...	Process Name	PID	Operation	Path	Result	Detail
5:26:20...	Sample_5b732...	1900	CreateFile	C:\Python27\Lib\binhex.py	SUCCESS	Desired Access: Generic Read/Write, Disposition: Open, Options: Synchronous IO Non-Alert, Non-Directory File, Attri...
5:26:20...	Sample_5b732...	1900	QueryStandardInformationFile	C:\Python27\Lib\binhex.py	SUCCESS	AllocationSize: 16,384; EndOfFile: 14,984; NumberOfLinks: 1; DeletePending: False; Directory: False
5:26:20...	Sample_5b732...	1900	ReadFile	C:\Python27\Lib\binhex.py	SUCCESS	Offset 0; Length: 14,984; Priority: Normal
5:26:20...	Sample_5b732...	1900	ReadFile	C:\Python27\Lib\binhex.py	SUCCESS	Offset 0; Length: 14,984; I/O Flags: Non-cached, Paging I/O, Priority: Normal
5:26:20...	Sample_5b732...	1900	WriteFile	C:\Python27\Lib\binhex.py	SUCCESS	Offset 0; Length: 14,984; Priority: Normal
5:26:20...	Sample_5b732...	1900	CloseFile	C:\Python27\Lib\binhex.py	SUCCESS	
5:26:20...	Sample_5b732...	1900	CreateFile	C:\Python27\Lib\binhex.py	SUCCESS	Desired Access: Read Attributes, Delete, Synchronize, Disposition: Open, Options: Synchronous IO Non-Alert, Open R...
5:26:20...	Sample_5b732...	1900	QueryAttributeTagFile	C:\Python27\Lib\binhex.py	SUCCESS	Attributes: A: Reparse Tag (d)
5:26:20...	Sample_5b732...	1900	QueryBasicInformationFile	C:\Python27\Lib\binhex.py	SUCCESS	CreationTime: 3/8/2011 8:43:12 AM; LastAccessTime: 7/6/2018 10:51:53 PM; LastWriteTime: 8/15/2018 5:26:20 PM; Cha...
5:26:20...	Sample_5b732...	1900	CreateFile	C:\Python27\Lib	SUCCESS	Desired Access: Write Data/Add File, Synchronize, Disposition: Open, Options: . Attributes: n/a, ShareMode: Read, Wr...
5:26:20...	Sample_5b732...	1900	SelfRenameInformationFile	C:\Python27\Lib	SUCCESS	ReplaceIfExists: False; FileName: C:\Python27\Lib\binhex.py\KEYPASS
5:26:20...	Sample_5b732...	1900	CloseFile	C:\Python27\Lib	SUCCESS	
5:26:20...	Sample_5b732...	1900	CloseFile	C:\Python27\Lib\binhex.py\KEYPASS	SUCCESS	

همانطور که در تصویر بالا قابل ملاحظه است، باج افزار فایل هدف را با دسترسی خواندن و نوشتن گشوده و اندازه فایل که در حافظه اشغال شده است را پرسیده و سپس انتهای فایل را معین نموده است. در ادامه، باج افزار از ابتدای همان فایل تا انتهای فایل را می خواند و به همان اندازه نیز در فایل می نویسد که این بدین معنی است که فرآیند رمزگذاری محتوای فایل انجام شده است. سپس فایل را بسته و آخرین اطلاعات مربوط به آن که به روز شده است را می پرسد و پسوند KEYPASS را به انتهای همان فایل اضافه می کند و سپس سراغ فایل بعدی می رود و این عمل را بر روی تمام فایل های هدف تکرار می کند.

طبق بررسی های صورت گرفته، سرعت رمزگذاری فایل ها ارتباط مستقیمی با منابع سیستم قربانی دارد. در آزمایش های صورت گرفته، طبق تصویر زیر، باج افزار KeyPass بیش از همه، دیسک را درگیر می کند. لذا هرچه سرعت خواندن/نوشتن دیسک بالاتر باشد، فرآیند رمزگذاری سریعتر اتفاق می افتد. اما بطور میانگین این مدت زمان برای دیسکی با ظرفیت ۱۰۰ گیگابایت بین ۲ تا ۴ دقیقه است.



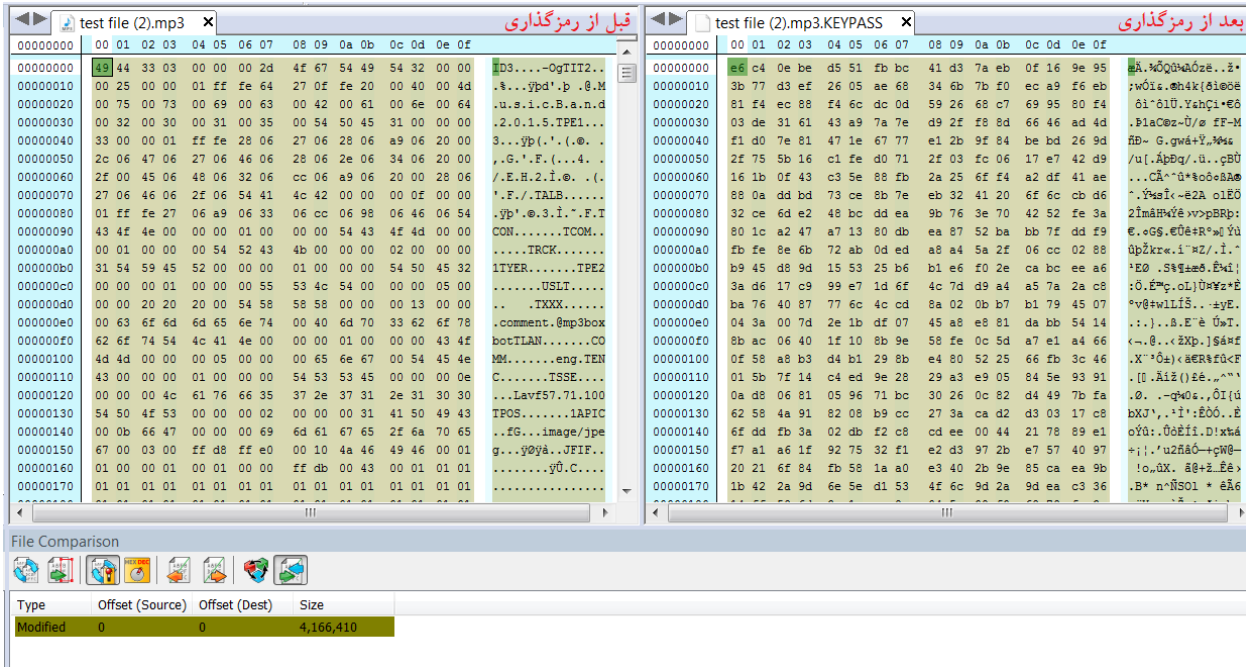
تحلیل ایستا:

با بررسی بیشتر کد های باج افزار به نتایج زیر دست یافتیم.

طبق بررسی هایی که بر روی فایل های مختلف قبل و بعد از رمزگذاری انجام دادیم شاهد این بودیم که باج افزار KEYPASS تنها ساختار فایل هایی را که حجم آن ها کمتر از ۵.۲۴۲.۸۸۰ بایت است را به طور کامل تغییر می دهد و فایل هایی که حجم آن ها از این مقدار بیشتر است، را بدین صورت رمزگذاری می کند که ۵.۲۴۲.۸۸۰ بایت ابتدای فایل را تغییر می دهد و سپس در صورت حجیم بودن فایل، ۵.۲۴۲.۸۸۰ بایت ادامه ی ساختار آن را تغییر نمی دهد و دوباره ۵.۲۴۲.۸۸۰ بایت دیگر را تغییر می دهد، این روش رمزگذاری ساختار فایل تا انتهای آن ادامه می یابد. تصاویر زیر نمونه ای از تغییرات ساختار فایل ها را نشان می دهد :

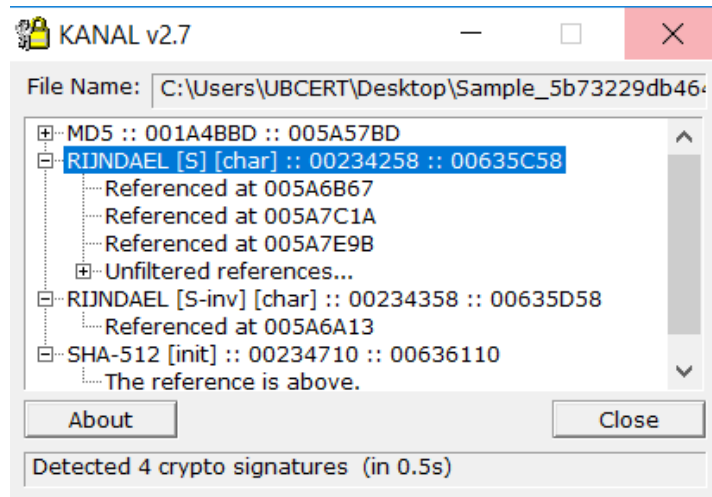
Type	Offset (Source)	Offset (Dest)	Size
Modified	0	0	5,242,880
Matched	5,242,880	5,242,880	52,428,800
Modified	57,671,680	57,671,680	5,242,880
Matched	62,914,560	62,914,560	52,428,800
Modified	115,343,360	115,343,360	5,242,880
Matched	120,586,240	120,586,240	52,428,800
Modified	173,015,040	173,015,040	5,242,880
Matched	178,257,920	178,257,920	52,428,800
Modified	230,686,720	230,686,720	5,242,880
Matched	235,929,600	235,929,600	52,428,800
Modified	288,358,400	288,358,400	5,242,880
Matched	293,601,280	293,601,280	52,428,800
Modified	346,030,080	346,030,080	5,242,880
Matched	351,272,960	351,272,960	52,428,800

تصویر ۱: فایل با حجم بیشتر از ۵.۲۴۲.۸۸۰ بایت که طبق روشی که اشاره نمودیم ساختار فایل ها را تغییر کرده است.

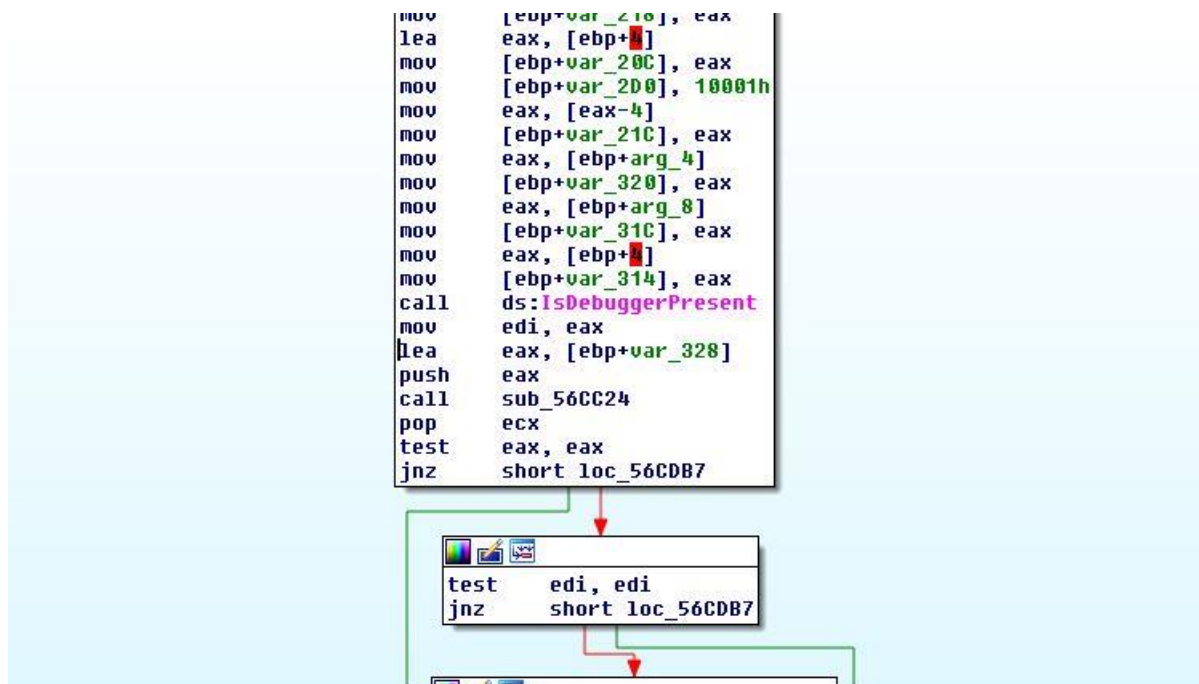


تصویر ۲: فایل با حجم کمتر از ۵.۲۴۲.۸۸۰ بایت که تمام ساختار آن تغییر کرده است.

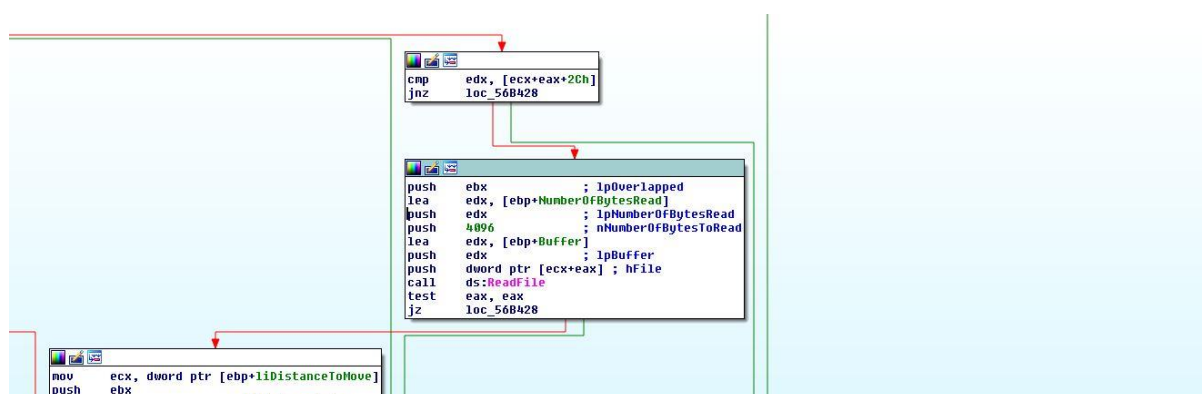
همانطور که در مقدمه اشاره شد باج افزار KeyPass از الگوریتم رمزنگاری ۲۵۶ – AEC (Rijndael) برای رمزگذاری فایل‌ها استفاده می‌کند.



همچنین نتایج بدست آمده نشان می‌دهد که این باج‌افزار از تکنیک آنتی دیباگ (ضد مهندسی معکوس) استفاده می‌کند و در صورتی که دیباگر در سیستم در حالت اجرا باشد این باج‌افزار، اجرا نخواهد شد.



همانطور که گفته شد باج افزار در هنگام رمزگذاری فایل‌ها، میزان فضای اشغال شده توسط فایل را در حافظه مشخص می‌کند. بررسی کد منبع باج‌افزار گویای این مطلب است که در حالت عادی ۴۰۹۶ کیلوبایت فضای اشغال شده توسط فایل هدف در نظر گرفته می‌شود و در شرایط خاصی نیز این عدد تغییر می‌کند که احتمالاً بسته به پارامترهای دیگر مربوط به فایل هدف از جمله حجم بالای آن است.



در ادامه به رشته‌های جالب توجهی برخوردیم که در تصویر زیر نمایان هستند:

```

.rdata:00642428          unicode 0, <Encryption>,0
.rdata:0064243E          align 10h
.rdata:00642440          ; wchar_t aLog
.rdata:00642440          aLog:                                ; DATA XREF: sub_412200+56C70
.rdata:00642440          unicode 0, <--Log>,0
.rdata:00642440          ; wchar_t aAdmin
.rdata:00642440          aAdmin:                              ; DATA XREF: sub_412200:loc_412795f0
.rdata:00642440          ; sub_412200+99F70
.rdata:00642440          unicode 0, <--Admin>,0
.rdata:00642450          ; wchar_t aFornetres
.rdata:00642450          aFornetres:                          ; DATA XREF: sub_412200:loc_4127BAf0
.rdata:00642450          unicode 0, <--ForNetRes>,0
.rdata:00642474          ; wchar_t aAutostart
.rdata:00642474          aAutostart:                          ; DATA XREF: sub_412200:loc_412812f0
.rdata:00642474          unicode 0, <--AutoStart>,0
.rdata:00642480          ; wchar_t aService
.rdata:00642480          aService:                            ; DATA XREF: sub_412200:loc_412868f0
.rdata:00642480          unicode 0, <--Service>,0
.rdata:006424A0          ; DATA XREF: sub_412200+9C670
.rdata:006424A0          ; sub_412200+A6270
.rdata:006424A0          unicode 0, <--Log>,0
.rdata:006424A0          align 10h
.rdata:006424B0          aRunas:                              ; DATA XREF: sub_412200+A2D70
.rdata:006424B0          unicode 0, <Runas>,0
.rdata:006424BC          align 10h
.rdata:006424C0          ; CHAR MultiByteStr[]
.rdata:006424C0          MultiByteStr [ db 'http://cosonar.mcdi.ru/get.php',0
00240AC0 00000000006424C0: .rdata:MultiByteStr
  
```

همانطور که در تصویر بالا قابل ملاحظه است، رشته‌هایی نظیر --Log ، --Admin ، --ForNetRes و ... و در انتها نیز آدرس <http://cosonar.mcdi.ru/get.php> جلب توجه می‌کنند.

با توجه به بررسی فرآیندهای اجرا شده در سیستم توسط این باج‌افزار، مشخص می‌گردد که این رشته‌های متنی برای باج‌افزار همانند یک سوئیچ خط فرمان کاربرد دارند و هر کدام از موارد بالا کارایی خاص خود را دارند.

- [Sample 0b73229db464a96eb6f650.c.exe](#) ۱۴۷۲
 - [Sample 0b73229db464a96eb6f650.c.exe](#) ۸۶۴
 - [Sample 0b73229db464a96eb6f650.c.exe](#) ۲۲۴۸ --Admin
 - [Sample 0b73229db464a96eb6f650.c.exe](#) ۲۴۸۸ --ForNetRes
x 01 7 4 v 4 h 0 0 3 x J 0 i y h U f H Q 1 W 7 0 0 R D S i c m S f g 7 2 K V A
7 s e 9 R a l x X F 9 m 7 0 z W m x 7 n L 7 b V R p 7 9 1 w 4 S N Y 1 U C i r 0
 - [Sample 0b73229db464a96eb6f650.c.exe](#) ۲۶۳۲ --Service
2 4 8 8 x 0 1 7 4 v 4 h 0 0 3 x J 0 i y h U f H Q 1 W 7 0 0 R D S i c m S f g 7 2 K V A
7 s e 9 R a l x X F 9 m 7 0 z W m x 7 n L 7 b V R p 7 9 1 w 4 S N Y 1 U C i r 0
 - [Sample 0b73229db464a96eb6f650.c.exe](#) ۲۵۲۰ --Service ۲۲۴۸
x 0 1 7 4 v 4 h 0 0 3 x J 0 i y h U f H Q 1 W 7 0 0 R D S i c m S f g 7 2 K V A
7 s e 9 R a l x X F 9 m 7 0 z W m x 7 n L 7 b V R p 7 9 1 w 4 S N Y 1 U C i r 0
 - [cmd.exe](#) ۱۵۱۲ `cmd /c ""C:\Users\user\AppData\Local\Temp\delfself.bat""`

بنابراین، می‌توان نتیجه گرفت که آدرس موجود مربوط به سرور فرمان و کنترل باج‌افزار می‌باشد و هر کدام از رشته‌های موجود به مثابه یک دستور به باج‌افزار می‌باشند.

همچنین شایان ذکر است که باج‌افزار با انداختن فایل `delfself.bat` در دایرکتوری `Temp` فایل اجرایی خود را از سیستم قربانی حذف می‌کند.

تحلیل ترافیک شبکه :

ترافیک شبکه ایجاد شده توسط باج افزار KeyPass نکات جالب توجهی دارد که از جمله می توان به اتصال این باج افزار به سایت خدمات هاستینگ در روسیه به نشانی <https://mhost.ru> از طریق پورت ۸۰ اشاره نمود. البته گفتنی است که باج افزار به میزبانی که این شرکت ارائه می دهد، متصل می گردد. تصویر زیر صفحه اول وبسایت مورد اشاره را نشان می دهد.

Махост — лидер авторитетных рейтингов



بررسی فعالیت باج افزار در سیستم و خروجی Wireshark و TCPLogView این موضوع را تایید می کنند. همانطور که در تصاویر زیر نمایان است، این باج افزار با سروری از همین شرکت خدمات دهنده به آدرس s9.h.mchost.ru و آی پی ۱۷۸.۲۰۸.۸۳.۱۳ در ارتباط است.

Time	Process Name	PID	Operation	Path	Result	Detail
5:22:54	Sample_5b732	1900	TCP Connect	WIN-78QSPHQ1OK49566 -> s9.h.mchost.ru	SUCCESS	Length: 0, mss: 1390, sackopt: 0, tsopt: 0, wsopt: 0, rcvwin: 65330, rcvwinscale: 0, sndwinscale: 0, seqnum: 0, connid: 0
5:22:54	Sample_5b732	1900	TCP Send	WIN-78QSPHQ1OK49566 -> s9.h.mchost.ru	SUCCESS	Length: 23, starttime: 179616, endtime: 179617, seqnum: 0, connid: 0
5:22:54	Sample_5b732	1900	TCP Send	WIN-78QSPHQ1OK49566 -> s9.h.mchost.ru	SUCCESS	Length: 24, starttime: 179616, endtime: 179619, seqnum: 0, connid: 0
5:22:54	Sample_5b732	1900	TCP Send	WIN-78QSPHQ1OK49566 -> s9.h.mchost.ru	SUCCESS	Length: 13, starttime: 179616, endtime: 179619, seqnum: 0, connid: 0
5:22:54	Sample_5b732	1900	TCP Receive	WIN-78QSPHQ1OK49566 -> s9.h.mchost.ru	SUCCESS	Length: 21, starttime: 179616, endtime: 179619, seqnum: 0, connid: 0
5:22:54	Sample_5b732	1900	TCP Receive	WIN-78QSPHQ1OK49566 -> s9.h.mchost.ru	SUCCESS	Length: 504, seqnum: 0, connid: 0
5:22:54	Sample_5b732	1900	TCP Receive	WIN-78QSPHQ1OK49566 -> s9.h.mchost.ru	SUCCESS	Length: 796, seqnum: 0, connid: 0
5:22:54	Sample_5b732	1900	TCP Disconnect	WIN-78QSPHQ1OK49566 -> s9.h.mchost.ru	SUCCESS	Length: 0, seqnum: 0, connid: 0
5:23:00	Sample_5b732	1900	TCP Connect	WIN-78QSPHQ1OK49569 -> s9.h.mchost.ru	SUCCESS	Length: 0, mss: 1390, sackopt: 0, tsopt: 0, wsopt: 0, rcvwin: 65330, rcvwinscale: 0, sndwinscale: 0, seqnum: 0, connid: 0
5:23:00	Sample_5b732	1900	TCP Send	WIN-78QSPHQ1OK49569 -> s9.h.mchost.ru	SUCCESS	Length: 23, starttime: 179670, endtime: 179673, seqnum: 0, connid: 0
5:23:00	Sample_5b732	1900	TCP Send	WIN-78QSPHQ1OK49569 -> s9.h.mchost.ru	SUCCESS	Length: 24, starttime: 179670, endtime: 179673, seqnum: 0, connid: 0
5:23:00	Sample_5b732	1900	TCP Send	WIN-78QSPHQ1OK49569 -> s9.h.mchost.ru	SUCCESS	Length: 13, starttime: 179670, endtime: 179673, seqnum: 0, connid: 0
5:23:00	Sample_5b732	1900	TCP Receive	WIN-78QSPHQ1OK49569 -> s9.h.mchost.ru	SUCCESS	Length: 21, starttime: 179670, endtime: 179673, seqnum: 0, connid: 0
5:23:00	Sample_5b732	1900	TCP Receive	WIN-78QSPHQ1OK49569 -> s9.h.mchost.ru	SUCCESS	Length: 504, seqnum: 0, connid: 0
5:23:00	Sample_5b732	1900	TCP Receive	WIN-78QSPHQ1OK49569 -> s9.h.mchost.ru	SUCCESS	Length: 796, seqnum: 0, connid: 0
5:23:00	Sample_5b732	1900	TCP Disconnect	WIN-78QSPHQ1OK49569 -> s9.h.mchost.ru	SUCCESS	Length: 360, seqnum: 0, connid: 0
5:23:05	Sample_5b732	1900	TCP Connect	WIN-78QSPHQ1OK49570 -> s9.h.mchost.ru	SUCCESS	Length: 0, mss: 1390, sackopt: 0, tsopt: 0, wsopt: 0, rcvwin: 65330, rcvwinscale: 0, sndwinscale: 0, seqnum: 0, connid: 0
5:23:05	Sample_5b732	1900	TCP Send	WIN-78QSPHQ1OK49570 -> s9.h.mchost.ru	SUCCESS	Length: 23, starttime: 179725, endtime: 179726, seqnum: 0, connid: 0
5:23:05	Sample_5b732	1900	TCP Send	WIN-78QSPHQ1OK49570 -> s9.h.mchost.ru	SUCCESS	Length: 24, starttime: 179725, endtime: 179727, seqnum: 0, connid: 0
5:23:05	Sample_5b732	1900	TCP Send	WIN-78QSPHQ1OK49570 -> s9.h.mchost.ru	SUCCESS	Length: 13, starttime: 179725, endtime: 179727, seqnum: 0, connid: 0
5:23:05	Sample_5b732	1900	TCP Receive	WIN-78QSPHQ1OK49570 -> s9.h.mchost.ru	SUCCESS	Length: 21, starttime: 179725, endtime: 179727, seqnum: 0, connid: 0
5:23:05	Sample_5b732	1900	TCP Receive	WIN-78QSPHQ1OK49570 -> s9.h.mchost.ru	SUCCESS	Length: 504, seqnum: 0, connid: 0
5:23:05	Sample_5b732	1900	TCP Receive	WIN-78QSPHQ1OK49570 -> s9.h.mchost.ru	SUCCESS	Length: 796, seqnum: 0, connid: 0
5:23:05	Sample_5b732	1900	TCP Receive	WIN-78QSPHQ1OK49570 -> s9.h.mchost.ru	SUCCESS	Length: 360, seqnum: 0, connid: 0
5:23:05	Sample_5b732	1900	TCP Disconnect	WIN-78QSPHQ1OK49570 -> s9.h.mchost.ru	SUCCESS	Length: 0, seqnum: 0, connid: 0
5:23:10	Sample_5b732	1900	TCP Connect	WIN-78QSPHQ1OK49571 -> s9.h.mchost.ru	SUCCESS	Length: 0, mss: 1390, sackopt: 0, tsopt: 0, wsopt: 0, rcvwin: 65330, rcvwinscale: 0, sndwinscale: 0, seqnum: 0, connid: 0
5:23:10	Sample_5b732	1900	TCP Send	WIN-78QSPHQ1OK49571 -> s9.h.mchost.ru	SUCCESS	Length: 23, starttime: 179779, endtime: 179780, seqnum: 0, connid: 0
5:23:11	Sample_5b732	1900	TCP Send	WIN-78QSPHQ1OK49571 -> s9.h.mchost.ru	SUCCESS	Length: 24, starttime: 179779, endtime: 179781, seqnum: 0, connid: 0
5:23:11	Sample_5b732	1900	TCP Send	WIN-78QSPHQ1OK49571 -> s9.h.mchost.ru	SUCCESS	Length: 13, starttime: 179779, endtime: 179781, seqnum: 0, connid: 0
5:23:11	Sample_5b732	1900	TCP Receive	WIN-78QSPHQ1OK49571 -> s9.h.mchost.ru	SUCCESS	Length: 21, starttime: 179779, endtime: 179781, seqnum: 0, connid: 0
5:23:11	Sample_5b732	1900	TCP Receive	WIN-78QSPHQ1OK49571 -> s9.h.mchost.ru	SUCCESS	Length: 504, seqnum: 0, connid: 0
5:23:11	Sample_5b732	1900	TCP Receive	WIN-78QSPHQ1OK49571 -> s9.h.mchost.ru	SUCCESS	Length: 796, seqnum: 0, connid: 0
5:23:11	Sample_5b732	1900	TCP Receive	WIN-78QSPHQ1OK49571 -> s9.h.mchost.ru	SUCCESS	Length: 360, seqnum: 0, connid: 0
5:23:11	Sample_5b732	1900	TCP Disconnect	WIN-78QSPHQ1OK49571 -> s9.h.mchost.ru	SUCCESS	Length: 0, seqnum: 0, connid: 0

Showing 36 of 2,108,209 events (0.0017%) Backed by virtual memory

Event Time	Event Type	Local Address	Remote Address	Remote Host Name	Local Port	Remote Port	Process ID	Process Name
8/15/2018 6:55:2...	Open	192.168.1.4	178.208.83.13	s9.h.mchost.ru	49564	80	3868	Sample_5b
8/15/2018 6:55:2...	Close	192.168.1.4	178.208.83.13	s9.h.mchost.ru	49564	80	3868	Sample_5b
8/15/2018 6:55:2...	Open	192.168.1.4	178.208.83.13	s9.h.mchost.ru	49565	80	3868	Sample_5b
8/15/2018 6:55:2...	Close	192.168.1.4	178.208.83.13	s9.h.mchost.ru	49565	80	3868	Sample_5b
8/15/2018 6:55:3...	Open	192.168.1.4	178.208.83.13	s9.h.mchost.ru	49566	80	3868	Sample_5b
8/15/2018 6:55:3...	Close	192.168.1.4	178.208.83.13	s9.h.mchost.ru	49566	80	3868	Sample_5b
8/15/2018 6:55:3...	Open	192.168.1.4	178.208.83.13	s9.h.mchost.ru	49567	80	3868	Sample_5b
8/15/2018 6:55:4...	Close	192.168.1.4	178.208.83.13	s9.h.mchost.ru	49567	80	3868	Sample_5b

8 item(s) NirSoft Freeware. <http://www.nirsoft.net>

No.	Time	Source	Destination	Protocol	Length	Info
589	146.620528	192.168.1.4	178.208.83.13	TCP	66	49567 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
590	146.752573	178.208.83.13	192.168.1.4	TCP	60	80 → 49567 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1390
591	146.752749	192.168.1.4	178.208.83.13	TCP	54	49567 → 80 [ACK] Seq=1 Ack=1 Win=65330 Len=0
592	146.752909	192.168.1.4	178.208.83.13	TCP	77	49567 → 80 [PSH, ACK] Seq=1 Ack=1 Win=65330 Len=23 [TCP segment of a reassembled PDU]
593	146.888563	178.208.83.13	192.168.1.4	TCP	60	80 → 49567 [ACK] Seq=1 Ack=24 Win=26000 Len=0
594	146.888632	192.168.1.4	178.208.83.13	HTTP	112	GET /get.php HTTP/1.0
595	147.024441	178.208.83.13	192.168.1.4	TCP	60	80 → 49567 [ACK] Seq=1 Ack=82 Win=26000 Len=0
596	147.025737	178.208.83.13	192.168.1.4	TCP	1354	80 → 49567 [ACK] Seq=1 Ack=82 Win=26000 Len=1300 [TCP segment of a reassembled PDU]
597	147.025740	178.208.83.13	192.168.1.4	HTTP	414	HTTP/1.1 404 Not Found (text/html)
598	147.025890	192.168.1.4	178.208.83.13	TCP	54	49567 → 80 [ACK] Seq=82 Ack=1662 Win=65330 Len=0

```

0000  00 0c 29 d9 62 fe 3c 1e 04 e6 ec b3 08 00 45 88  ..).b.<. ....E.
0010  00 28 52 da 40 00 31 06 2e e4 b2 d0 53 0d c0 a8  .(R.@.1. ...S...
0020  01 04 00 50 c1 9f 44 67 ca 10 54 18 7b 1f 50 10  ..P..Dg ..T.{.P.
0030  65 90 e3 1a 00 00 00 19 49 72 61 6e                e.....Iran
  
```

نکته قابل توجهی در بسته‌های دریافتی باج‌افزار KeyPass از سرور مذکور دیده می‌شود که آن، نام ایران در چند بسته دریافتی می‌باشد. در تصویر زیر نیز، نظاره گر یک Conversation بین باج‌افزار و سرور مزبور هستید.

```


GET /get.php HTTP/1.0
Host: cosonar.mcdir.ru
Accept: */*
Connection: close

HTTP/1.1 404 Not Found
Server: nginx
Date: Wed, 15 Aug 2018 14:25:22 GMT
Content-Type: text/html
Content-Length: 1390
Connection: close
Vary: Accept-Encoding
Last-Modified: Tue, 02 Jan 2018 12:36:34 GMT
ETag: "4940106-56e-561ca595b5880"
Accept-Ranges: bytes

<HTML><HEAD>
<title>404 Not Found - .....</title>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
<link rel="icon" href="/favicon5536fc9343e9ab0aca071354084dd3cc.ico" type="image/x-icon" />
<link rel="shortcut icon" href="/favicon5536fc9343e9ab0aca071354084dd3cc.ico" type="image/x-icon" />
</HEAD>
<BODY>
<center>
<a href="https://mchost.ru/"></a><br><br>
<H1>404 Not Found<br><C.....</H1><br>
... <a href="https://mchost.ru/">.....</a>
</BODY>
</HTML>
<!--
- Unfortunately, Microsoft has added a clever new
- "feature" to Internet Explorer. If the text of
  
```

خروجی سامانه VirusTotal :

در حال حاضر تعداد ۴۳ مورد از ۶۷ آنتی ویروس و آنتی بدافزار موجود در سامانه VirusTotal قادر به شناسایی این باج افزار بوده و آن را حذف یا غیرفعال می کنند.



43 / 67

43 engines detected this file


SHA-256 ee74c63faa2eb9709b1d738762e28072aece2e7b9eefc5913eb6a5fd1564752

File name 5.exe

File size 2.82 MB

Last analysis 2018-08-15 03:09:14 UTC

Community score -363















































Detection

Details

Relations

Behavior

Community 5

Ad-Aware	 Trojan.GenericKD.31166491	AhnLab-V3	 Trojan/Win32.Ransom.R233970
ALYac	 Trojan.Ransom.Filecoder	Arcabit	 Trojan.Generic.D1DB901B
Avast	 Win32:Trojan-gen	AVG	 Win32:Trojan-gen
Avira	 TR/FileCoder.pfzqxh	BitDefender	 Trojan.GenericKD.31166491
CAT-QuickHeal	 Trojan.IGENERIC	ClamAV	 Win.Trojan.Agent-6644902-0
Comodo	 UnclassifiedMalware	CrowdStrike Falcon	 malicious_confidence_100% (W)
Cylance	 Unsafe	Cyren	 W32/Trojan.YATI-8672
Emsisoft	 Trojan.GenericKD.31166491 (B)	Endgame	 malicious (moderate confidence)
eScan	 Trojan.GenericKD.31166491	ESET-NOD32	 Win32/Filecoder.NRR
F-Secure	 Trojan.GenericKD.31166491	Fortinet	 W32/Filecoder.NRR!tr
GData	 Trojan.GenericKD.31166491	Jiangmin	 Trojan.Encoder.a
K7AntiVirus	 Trojan (0053a0921)	K7GW	 Trojan (0053a0921)
Kaspersky	 Trojan-Ransom.Win32.Encoder.n	Malwarebytes	 Ransom.FileCryptor
MAX	 malware (ai score=100)	McAfee	 Artemis!901D893F665C
McAfee-GW-Edition	 BehavesLike.Win32.BadFile.vh	Microsoft	 Trojan:Win32/Occamy.C
Palo Alto Networks	 generic.ml	Panda	 Trj/GdSda.A
Qihoo-360	 Win32/Trojan.Ransom.873	Rising	 Ransom.FileCryptor!8.1A7 (CLOUD)
Sophos AV	 Mal/Ransom-FS	Symantec	 Trojan.Gen.2
TACHYON	 Ransom/W32.Encoder.2958848	Tencent	 Win32.Trojan.Encoder.Hoxx
TrendMicro	 Ransom_ENCODER.THHADAH	TrendMicro-HouseCall	 Ransom_ENCODER.THHADAH
Webroot	 W32.Ransom.Gen	Zillya	 Trojan.GenericKD.Win32.145894
ZoneAlarm	 Trojan-Ransom.Win32.Encoder.n	AegisLab	 Clean

خروجی سامانه ویروس کاو مرکز ماهر :

در حال حاضر یعنی در زمان نگارش این گزارش تعداد ۴۳ مورد از ۶۷ آنتی ویروس و آنتی بدافزار موجود در سامانه ویروس کاو مرکز ماهر قادر به شناسایی این باج افزار بوده و آن را حذف یا غیرفعال می کنند.