باسمه تعالی

تحلیل فنی باج‌افزار Mega Cryptorr

مقدمه :

مشاهده و رصد فضای سایبری در زمینه باج‌افزار، از شروع فعالیت نمونه جدیدی از خانواده‌ی InsaneCrypt به نام Mega Cryptorr خبر می‌دهد که پس از رمزگذاری فایل‌ها، به انتهای آن‌ها پسوند .bip را اضافه می‌کند. بررسی‌ها نشان می‌دهد که فعالیت این باج‌افزار در تاریخ ۲۹ نوامبر سال ۲۰۱۸ میلادی شروع شده و به نظر می‌رسد تمرکز آن بیشتر بر روی کاربران انگلیسی زبان می‌باشد. این باج‌افزار از الگوریتم رمزنگاری AES برای رمزگذاری فایل‌ها استفاده می‌کند و تنها فایل‌هایی با پسوندهای مشخص را که در ادامه به آن‌ها اشاره خواهیم نمود، رمزگذاری می‌کند. طبق بررسی‌های انجام شده ریشه‌یابی این خانواده از باج‌افزار به صورت زیر می‌باشد :

desuCrypt (Trial) > InsaneCrypt >> GusCrypter > Mega Cryptorr

مشخصات فایل اجرایی :

نام فایل	cryptor_dyn.exe
MD۵	۰d٦ec٤٦b٢٥١٤١٧db٦٢٤٤١٠٣d٧٥٥٩e٤٠c
SHA-١	٤e٢٥da٩٦b٣d٤٣٠٨٢٨da٧e٣bae٠٥٤٣٨b٠da٦f٦٥٤٧
SHA-٢٥٦	٩٠١٧b٠ce٧٩٣٥٥٥e٢٧٣٥c٥٠e٠e٠٢f٢cfd٢١a٢٩c٧٤٥b٦٩١٧٥f٢٨٧١٢١٩b٢٤٤٦٥١٣٨
اندازه فایل	٤١٤.٥ KB
کامپایلر	VC٨ -> Microsoft Corporation

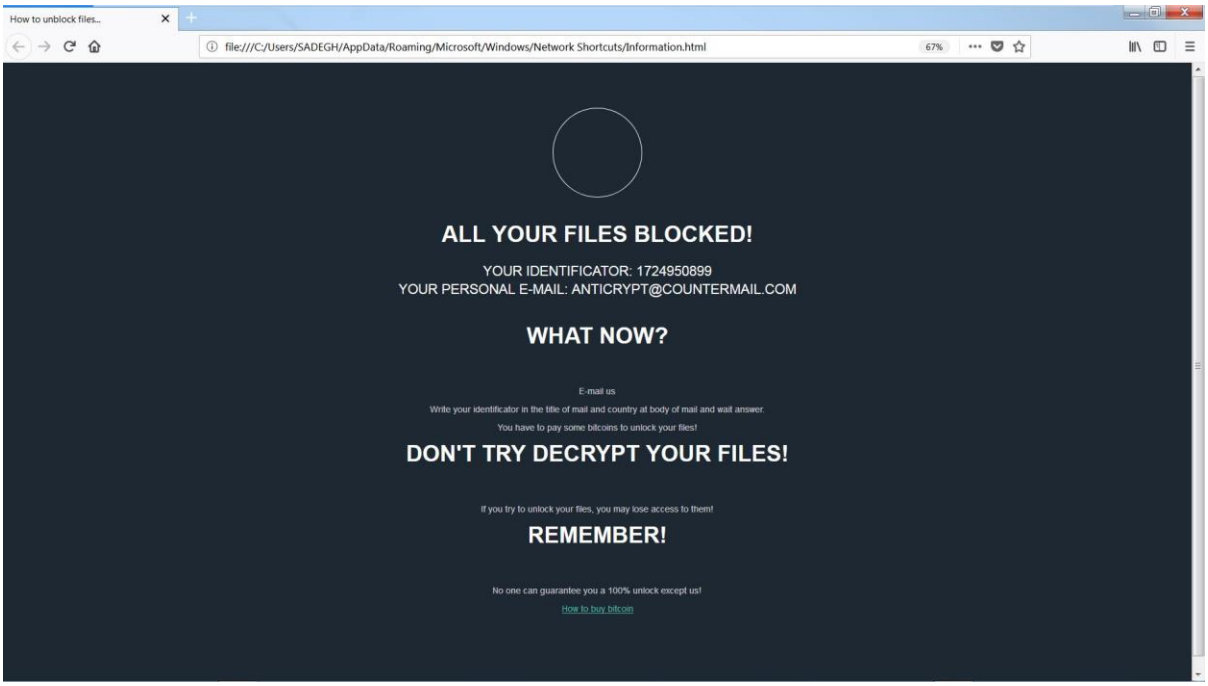
فایل اجرایی این باج‌افزار دارای پنج بخش است :

نام بخش	آنتروپی	آدرس مجازی	اندازه مجازی	اندازه خام
.text	6.64	4096	275664	275968
.rdata	5.71	282624	115608	115712
.data	3.62	401408	16844	12288
.rsrc	3.52	421888	1205	1536
.reloc	6.6	425984	17848	17920

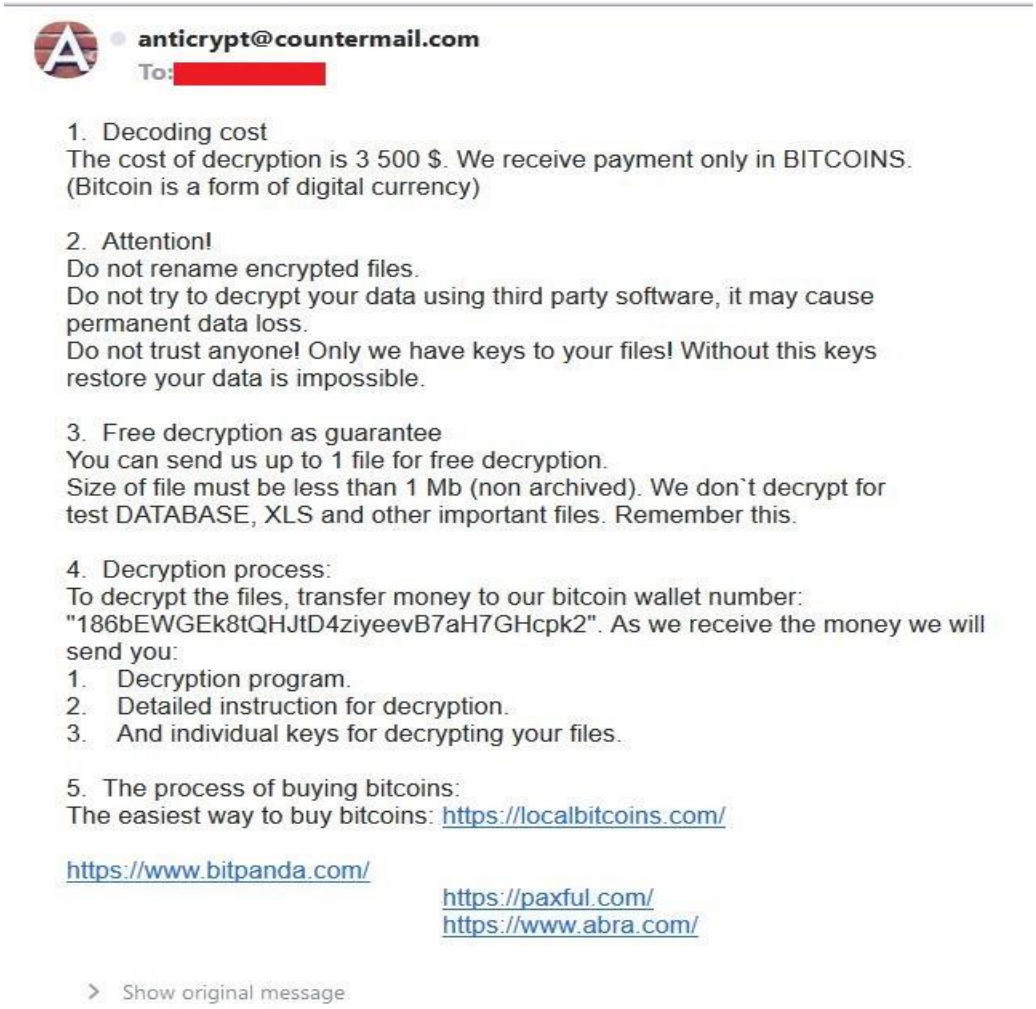
تحلیل پویا :

برای بررسی عمیق‌تر باج‌افزار Mega Cryptorr، 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، شروع به اسکن تمام درایوهای موجود بر روی سیستم قربانیان می‌کند و بدین ترتیب فرایند رمزگذاری فایل‌ها را آغاز می‌شود. همچنین این باج‌افزار در طول اجرای خود یک فایل HTML تحت عنوان Information.html که محتوای آن شامل پیغام باج‌خواهی می‌باشد را بر روی Desktop و در دایرکتوری‌های مختلف ایجاد می‌کند. در انتها فرایند مربوط به فایل اجرایی باج‌افزار خاتمه پیدا می‌کند و فایل اجرایی آن نیز حذف می‌شود.

تصویر زیر پیغام باج‌خواهی باج افزار Mega Cryptorr را نشان می‌دهد.



بر اساس پیغام باج‌خواهی مهاجمین به قربانیان اعلام نموده‌اند که تمام فایل‌های شما رمزگذاری شده‌اند و قربانیان برای رمزگشایی آن‌ها بایستی از طریق آدرس ایمیل DECRYPTHELPFILES@PROTONMAIL.COM با مهاجمین ارتباط برقرار نمایند و آن‌ها بایستی در ایمیل ارسالی علاوه بر کدشناسایی منحصربفرد خود، نام کشور خود را نیز ذکر نمایند. پس از برقراری ارتباط به صورت ناشناس با مهاجمین پیام زیر از سوی آن‌ها برای ما ارسال گردید:



anticrypt@countermail.com
To: [redacted]

1. Decoding cost
The cost of decryption is 3 500 $. We receive payment only in BITCOINS.
(Bitcoin is a form of digital currency)

2. Attention!
Do not rename encrypted files.
Do not try to decrypt your data using third party software, it may cause permanent data loss.
Do not trust anyone! Only we have keys to your files! Without this keys restore your data is impossible.

3. Free decryption as guarantee
You can send us up to 1 file for free decryption.
Size of file must be less than 1 Mb (non archived). We don`t decrypt for test DATABASE, XLS and other important files. Remember this.

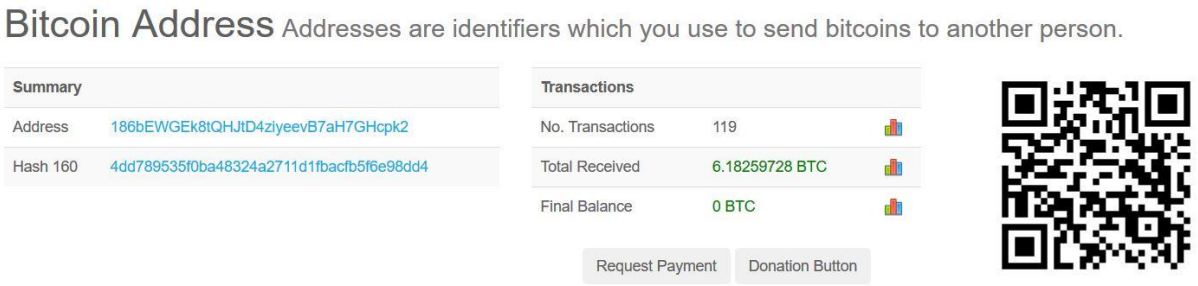
4. Decryption process:
To decrypt the files, transfer money to our bitcoin wallet number: "186bEWGEk8tQHJtD4ziyeevB7aH7GHcpk2". As we receive the money we will send you:
1. Decryption program.
2. Detailed instruction for decryption.
3. And individual keys for decrypting your files.

5. The process of buying bitcoins:
The easiest way to buy bitcoins: https://localbitcoins.com/

https://www.bitpanda.com/
https://paxful.com/
https://www.abra.com/

> Show original message

بر اساس این پیغام مهاجمین اعلام نموده‌اند که قربانیان معادل مبلغ ۳۵۰۰ دلار به بیت‌کوین را به کیف پـول بیت‌کوین به آدرس ۱۸٦bEWGEk۸tQHJtD٤ziyeevB۷aH۷GHcpk۲ ارسال نمایند و بعد از تایید پرداخـت مبلغ مورد نظر، ابزار رمزگشایی فایل‌ها به همراه دستورالعمل‌های لازم برای رمزگشایی، برای قربانیان ارسال خواهد شد. طبق بررسی‌های صورت گرفته کیف پول مربوط به این باج‌افـزار تـاکنون تعـداد ۱۱۹ تـراکنش برابر با ٦.۱۸۲٥۹۷۲۸ BTC داشته است.



Bitcoin Address Addresses are identifiers which you use to send bitcoins to another person.

Summary	
Address	186bEWGEk8tQHJtD4ziyeevB7aH7GHcpk2
Hash 160	4dd789535f0ba48324a2711d1fbacfb5f6e98dd4

Transactions	
No. Transactions	119
Total Received	6.18259728 BTC
Final Balance	0 BTC

Request Payment | Donation Button

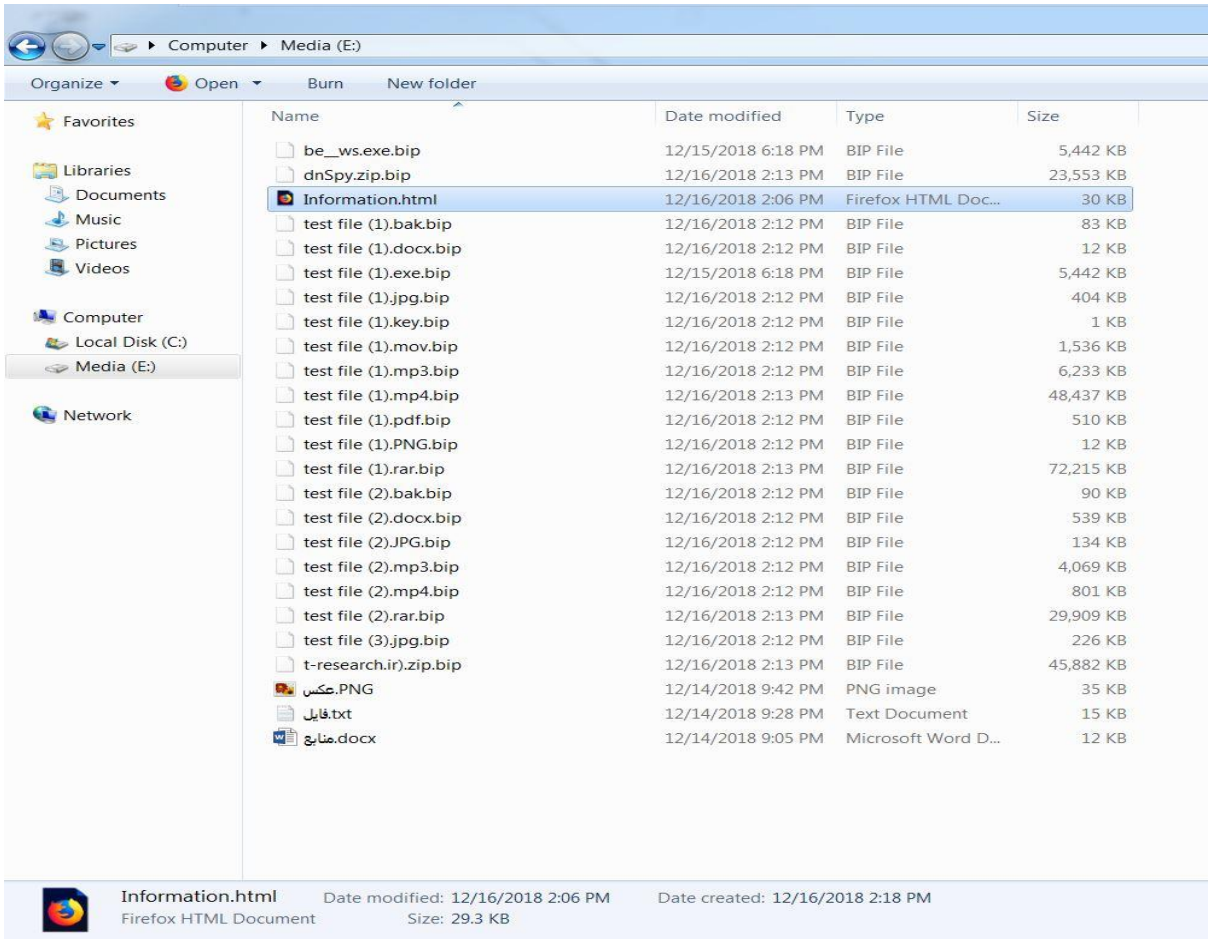
طبق بررسی‌های انجام شده باج‌افزار Mega Cryptorr فایل‌های موجود در دایرکتوری‌های زیر را رمزگذاری نمی‌کند :

Windows, Program Files, Program Files (x٨٦), All Users, $Recycle.Bin, Intel, ProgramData, Windows.old

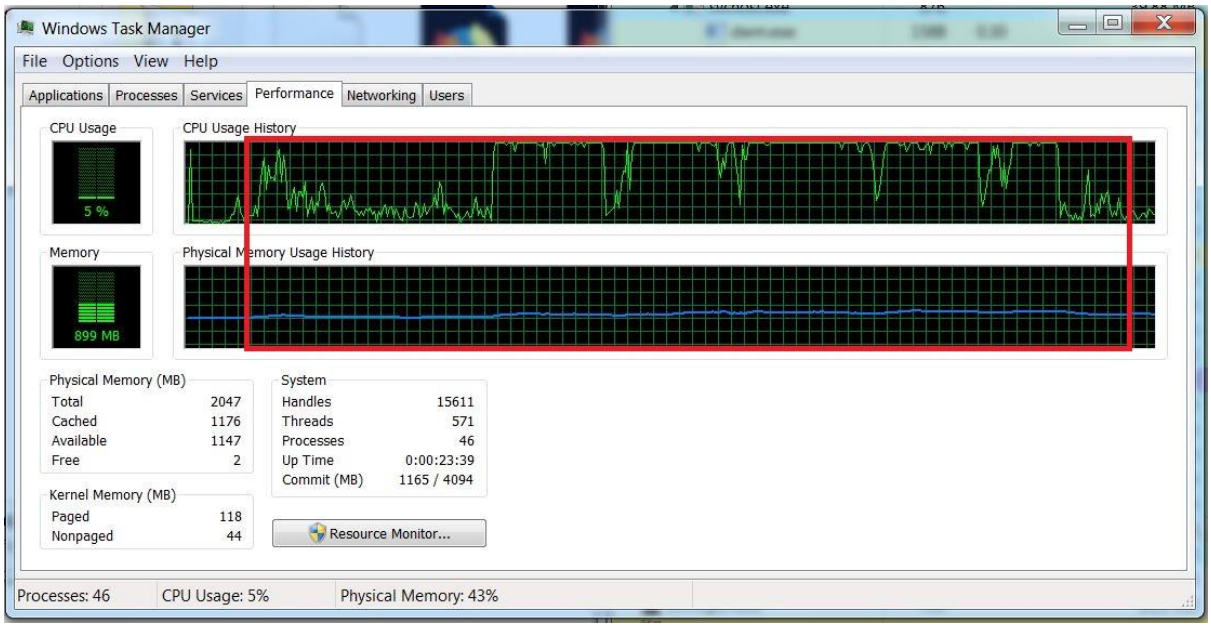
همانطور که اشاره شد این باج‌افزار فایل‌هایی با پسوندهای مشخص را مورد هدف حمله‌ی خود قرار می‌دهد که در زیر لیست پسوندهای این فایل‌ها قابل مشاهده است :

.gdb, .ldf, .mdf, .mdb, .pdf, .fdb, .mde, .txt, .png, .jpg, .jpeg, .bmp, .gif, .zip, .rar, .٧z, .١cd, .sql, .bak, .back, .cab, .log, .ico, .old, .rtf, .lnk, .doc, .docx, .xls, .xlsx, .tif, .vsc, .mkv, .flac, .der, .sch, .crt, .pem, .pbix, .hbk, .epx, .dpl, .bpl, .htm, .csv, .mp٤, .mp٣, .myi, .myd, .xml, .r١١, .vsl, .newdb, .srf, .pst, .ods, .dt, .cf, .erf, .html, .php,. asp, .aspx, .js, .ppt, .pptx, .java, .cpp, .css, .h, .c, .jar, .swift, .cs, .shtml, .less, .sass, .dat, .json, .key, .avi, .٣gp, .wmv, .py, .accdb, .rmvb, .mpg, .mov, .vob, .flv, .swf, .wma, .aac, .mmf, .amr, .m٤a, .m٤r, .ogg, .mp٢, .wav, .pcx, .tga, .tiff, .odt, .dll, .exe, .lrf, .glf, .msi, .vhdx, .vhd, .bin, .vsv, .tib, .vue

همانطور که پیشتر اشاره کردیم، این باج‌افزار فایل‌ها را با استفاده از الگوریتم رمزنگاری AES رمزگذاری می‌کند و در صورتی که نام فایل‌ها به زبان فارسی باشد، باج‌افزار قادر به رمزگذاری آن‌ها نیست. همانطور که پیش‌تر نیز اشاره شد پسوند فایل‌ها پس از رمزگذاری به .bip تغییر پیدا می‌کند. تصویر زیر نشان‌دهنده فایل‌های رمزگذاری شده توسط این باج‌افزار می‌باشد :



طبق مشاهدات صورت گرفته، در صورت بالا بودن ظرفیت منابع سیستم قربانی، سرعت رمزگذاری فایل‌ها نیز بالاتر خواهد بود. هنگام اجرای باج‌افزار Mega Cryptorr شاهد بودیم که این باج‌افزار به طور میانگین از ٦٥ الی ٧٠ درصد ظرفیت CPU و کمتر از ١٠ درصد ظرفیت حافظه (RAM) استفاده می‌کند. همچنین مدت زمان رمزگذاری فایل‌ها با توجه به اینکه باج‌افزار فایل‌هایی با پسوندهای مشخص را رمزگذاری می‌کند بستگی به حجم داده‌های موجود بر روی سیستم قربانیان دارد. به طور مثال طبق بررسی‌های صورت گرفته در محیط آزمایشگاه، مدت زمان لازم جهت رمزگذاری یک هارد دیسک با ظرفیت ٢٥ گیگابایت، ٦ دقیقه بود. تصویر زیر مربوط به نمودار مصرف منابع سیستم توسط باج‌افزار، از لحظه‌ی شروع تا انتهای فرایند رمزگذاری می‌باشد :

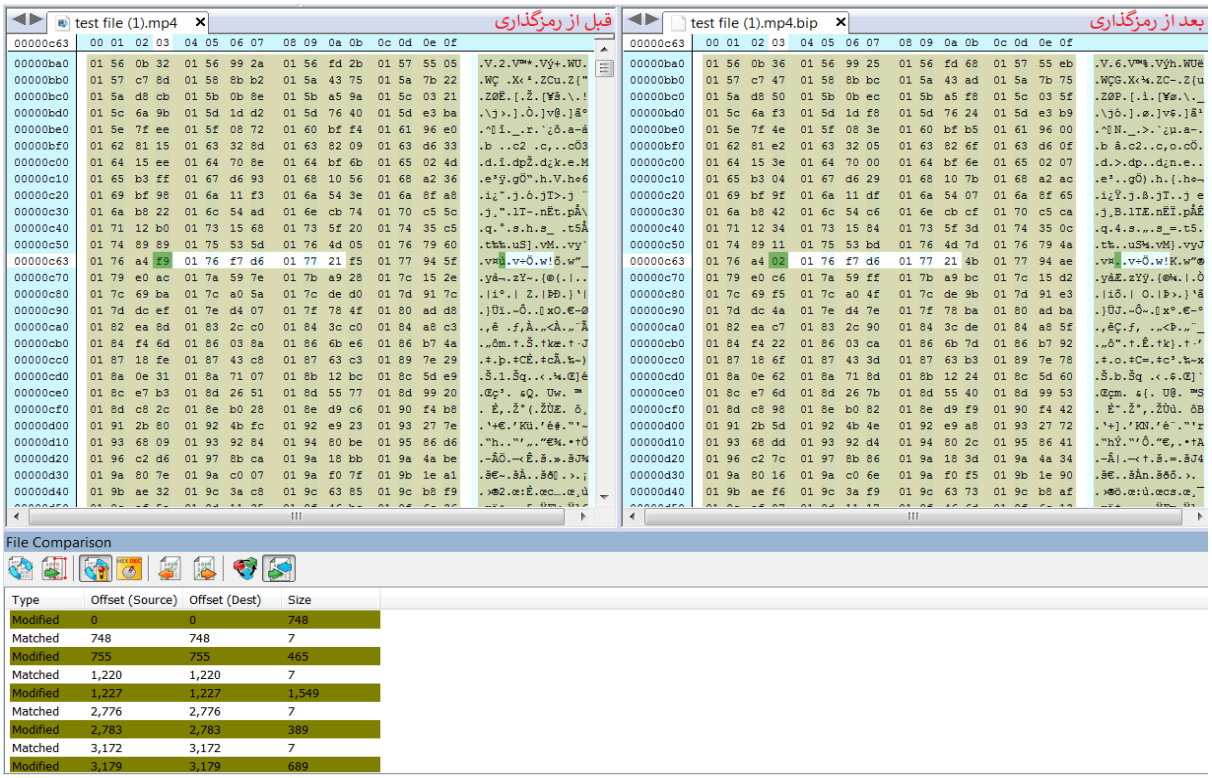


بر اساس بررسی‌های انجام شده اکثر آنتی‌ویروس‌های معتبر، این باج‌افزار را به عنوان یک تروجان شناسایی نموده‌اند. لذا احتمال نفوذ باج‌افزار به سیستم از راه‌های متداول از جمله هرزنامه‌ها وجود دارد. بنابراین توصیه می‌گردد از باز نمودن هرگونه ایمیل حاوی پیوست مشکوک جداً خودداری نمایند.

تحلیل ایستا:

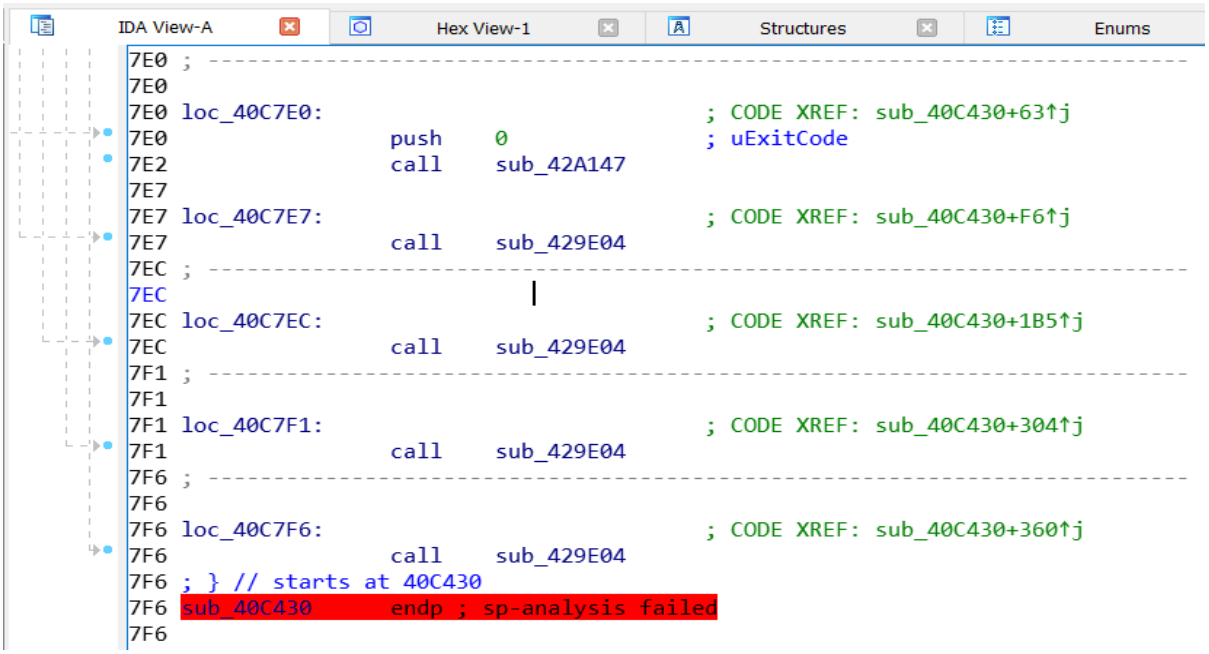
پس از تحلیل کد باج‌افزار Mega Cryptorr به نتایج زیر دست پیدا کردیم.

طبق بررسی‌هایی که بر روی فایل‌های مختلف قبل و بعد از رمزگذاری انجام دادیم شاهد این بودیم که باج‌افزار Mega Cryptorr ساختار فایل‌ها را به صورت کامل تغییر نمی‌دهد، تصویر زیر نمونه‌ای از تغییرات ساختار فایل‌ها را نشان می‌دهد :

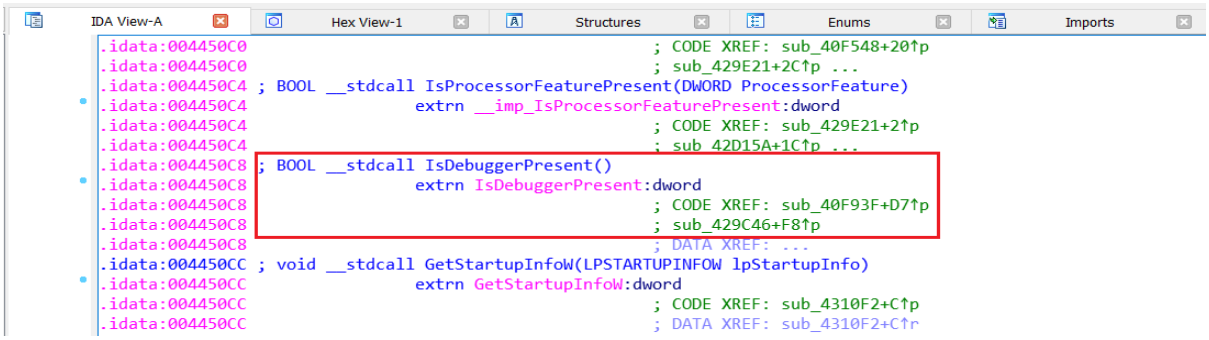


همانطور که در تصویر نیز قابل مشاهده است بخش‌های قهوه‌ای رنگ مربوط به قسمت‌هایی از ساختار فایل می‌باشد که در طول فرایند رمزگذاری، به طور کامل تغییر نموده‌اند و بخش‌های سفید رنگ نیز مربوط به قسمت‌هایی از ساختار فایل می‌باشند که تغییر نکرده‌اند.

قطعه کد زیر مربوط به استفاده از روش‌های مختلف ضد دیس‌اسمبل جهت جلوگیری از بررسی و تحلیل‌های بیشتر توسط محققین می‌باشد :



تابع IsDebuggerPresent() که از توابع کتابخانه Kernel۳۲ می‌باشد برای جلوگیری از اجرای باج‌افزار در محیط‌های دیباگر استفاده می‌شود تا در هنگام تحلیل با ایجاد خطا در دیباگرها مانع فعالیت گردد. قطعه کد زیر مربوط به این فرایند می‌باشد :



در قطعه کدهای زیر با استفاده از تابع IsDebuggerPresent() اقدام به بررسی محیط دیباگر می‌کند. اگر نتیجه به دست آمده مثبت باشد (یعنی محیط اجرا دیباگر باشد)، با استفاده از تابع SetUnhandledExceptionFilter() باعث ایجاد خطا می‌شود و از ادامه‌ی فعالیت جلوگیری می‌نماید.

cryptor_dyn.c

```
17476    int *v24; // [esp+CCh] [ebp-260h]
17477    __int16 v25; // [esp+D0h] [ebp-25Ch]
17478    int v26; // [esp+2D4h] [ebp-58h]
17479    int v27; // [esp+2D8h] [ebp-54h]
17480    int v28; // [esp+2E0h] [ebp-4Ch]
17481    struct _EXCEPTION_POINTERS ExceptionInfo; // [esp+324h] [ebp-8h]
17482    int savedregs; // [esp+32Ch] [ebp+0h]
17483    int retaddr; // [esp+330h] [ebp+4h]
17484
17485    if ( IsProcessorFeaturePresent(0x17u) )
17486      __fastfail(a4);
17487    sub_40FAEE();
17488    v19 = sub_426770(v9, 0, 0x2CCu);
17489    v18 = v4;
17490    v17 = v5;
17491    v16 = a1;
17492    v15 = a3;
17493    v14 = a2;
17494    v25 = __SS__;
17495    v22 = __CS__;
17496    v13 = __DS__;
17497    v12 = __ES__;
17498    v11 = __FS__;
17499    v10 = __GS__;
17500    v6 = __readeflags();
17501    v23 = v6;
17502    v21 = retaddr;
17503    v24 = &retaddr;
17504    v9[0].m128i_i32[0] = 65537;
17505    v20 = savedregs;
17506    sub_426770((__m128i *)&v26, 0, 0x50u);
17507    v26 = 1073741845;
17508    v27 = 1;
17509    v28 = retaddr;
17510    v7 = IsDebuggerPresent();
17511    ExceptionInfo.ExceptionRecord = (PEXCEPTION_RECORD)&v26;
17512    ExceptionInfo.ContextRecord = (PCONTEXT)v9;
17513    v8 = v7 == 1;
17514    SetUnhandledExceptionFilter(0);
17515    if ( !UnhandledExceptionFilter(&ExceptionInfo) && !v8 )
17516      sub_40FAEE();
17517  }
```

قطعه کد زیر مربوط به پیغام باج‌خواهی در کدمنبع باج‌افزار می‌باشد :

cryptor_dyn.c

```
9275    sub_406A00((int)v21, (int)&v52, (int)v8, v22, 0);
9276    sub_40B430((int)&v37, aDoctypeHtmlHtm);
9277    sub_40B430((int)&v37, aMRow320025Marg);
9278    sub_40B430((int)&v37, aGinLeft9166667);
9279    v23 = &dword_462A98;
9280    if ( (unsigned int)dword_462AAC >= 0x10 )
9281      v23 = (int *)dword_462A98;
9282    sub_40BF80((int)v23, &v37, dword_462AA8);
9283    sub_40B430((int)&v37, " <br>\n\nYour personal e-mail: ");
9284    v24 = &dword_462A80;
9285    if ( (unsigned int)dword_462A94 >= 0x10 )
9286      v24 = (int *)dword_462A80;
9287    sub_40BF80((int)v24, &v37, dword_462A90);
9288    sub_40B430(
9289      (int)&v37,
9290      " <br>\n"
9291      "\n"
9292      "<h2>What now?</h2><br>\n"
9293      "E-mail us<br>\n"
9294      "\n"
9295      "Write your identificator in the title of mail and country at body of mail and wait answer.<br>\n"
9296      "\n"
9297      "You have to pay some bitcoins to unlock your files!<br>\n"
9298      "\n"
9299      "<h2>Don't try decrypt your files!</h2><br>\n"
9300      "If you try to unlock your files, you may lose access to them!<br>\n"
9301      "\n"
9302      "<h2>Remember!</h2><br>\n"
9303      "No one can guarantee you a 100% unlock except us!<br>\n"
9304      "\n"
9305      "<a href=\"https://bitcoin.org/en/buy\">How to buy bitcoin</a></p>\n"
9306      "\t\t\t\t\n"
9307      "\n"
9308      "\t\n"
9309      "\n"
9310      "\t\t\n"
9311      "\t</body>\n"
9312      "</html>");
```

قطعه کد زیر مربوط به تابع ایجاد فایل مربوط به پیغام باج‌خواهی می‌باشد :

cryptor_dyn.c

```
9179    v49 = 0;
9180    sub_4056D0(&v42, (int)&a2);
9181    LOBYTE(v49) = 1;
9182    v8 = sub_402E70(&v42, (unsigned int *)&v29, "\\");
9183    LOBYTE(v49) = 2;
9184    v9 = v8[5];
9185    v10 = v8[4];
9186    if ( v8[5] - v10 < 0x10 )
9187    {
9188      LOBYTE(v35) = 0;
9189      v8 = sub_402FC0(v8, 0x10u, v35, (unsigned int)"Information.html", 0x10u);
9190    }
9191    else
9192    {
9193      v8[4] = v10 + 16;
9194      v11 = (unsigned int)v8;
9195      if ( v9 >= 0x10 )
9196        v11 = *v8;
9197      v12 = v11 + v10;
9198      sub_4261F0(v11 + v10, (unsigned int)"Information.html", 0x10u);
9199      *(_BYTE *)(v12 + 16) = 0;
9200    }
9201    v31 = *(_OWORD *)v8;
9202    v13 = *((_QWORD *)v8 + 2);
9203    v8[4] = 0;
9204    v8[5] = 15;
9205    v32 = v13;
9206    *(_BYTE *)v8 = 0;
9207    sub_402DF0((int)&v42, (int)&v52, (int)&v31);
9208    if ( HIDWORD(v32) >= 0x10 )
9209    {
9210      v14 = (void *)v31;
9211      if ( (unsigned int)(HIDWORD(v32) + 1) >= 0x1000 )
9212      {
9213        v14 = *(void **)(v31 - 4);
9214        if ( (unsigned int)(v31 - (_DWORD)v14 - 4) > 0x1F )
9215          sub_429E04((int)&v52, (int)v8);
9216      }
9217      sub_40ECDC(v14);
9218    }
```

قطعه کد زیر مربوط به دستور حذف فایل اجرایی باج‌افزار پس از پایان فرایند رمزگذاری فایل‌ها می‌باشد :

cryptor_dyn.c

```
13929   sub_4057A0(
13930     (unsigned int *)&v18,
13931     (unsigned int)&v21,
13932     (char *)&v21 + strlen((const char *)&v21) + 1 - ((char *)&v21 + 1));
13933   v26 = 0;
13934   v2 = sub_40B2E0(" /C timeout /T 15 /NOBREAK && del \"", (unsigned int *)&v12, (unsigned int *)&v18);
13935   LOBYTE(v26) = 1;
13936   v3 = v2[5];
13937   v4 = v2[4];
13938   if ( v2[5] - v4 < 4 )
13939   {
13940     v17 = 0;
13941     v2 = sub_402FC0(v2, 4u, *(int *)&v17, (unsigned int)"\" /F", 4u);
13942   }
13943   else
13944   {
13945     v2[4] = v4 + 4;
13946     v5 = (unsigned int)v2;
13947     if ( v3 >= 0x10 )
13948       v5 = *v2;
13949     v6 = v5 + v4;
13950     sub_4261F0(v5 + v4, (unsigned int)"\" /F", 4u);
13951     *(_BYTE *)(v6 + 4) = 0;
13952   }
13953   v15 = 0;
13954   v7 = (const CHAR *)&v14;
13955   v16 = 0;
13956   *(_OWORD *)&v14 = *(_OWORD *)v2;
13957   *(_QWORD *)&v15 = *((_QWORD *)v2 + 2);
13958   v2[4] = 0;
13959   v2[5] = 15;
13960   *(_BYTE *)v2 = 0;
13961   if ( v16 >= 0x10 )
13962     v7 = v14;
13963   result = ShellExecuteA(0, "open", "cmd.exe", v7, 0, 0);
13964   if ( v16 >= 0x10 )
13965   {
13966     v9 = (CHAR *)v14;
13967     if ( v16 + 1 >= 0x1000 )
13968     {
13969       v9 = (CHAR *)*((_DWORD *)v14 - 1);
13970       if ( (unsigned int)(v14 - v9 - 4) > 0x1F )
```

قطعه کد زیر مربوط به تابع IsProcessorFeaturePresent() می‌باشد که باج‌افزار با استفاده از آن بررسی می‌نماید که ویژگی‌های پردازنده موردنظر باج‌افزار با سیستم قربانی یکسان است یا خیر.

IDA View-A | Hex View-1 | Structures | Enums

```
15A
15A ; =============== S U B R O U T I N E =======================================
15A
15A ; Attributes: noreturn
15A
15A sub_42D15A      proc near               ; CODE XREF: sub_40E93E+A↑j
15A                                         ; sub_425BE0:loc_425BF6↑p ...
15A                 call    sub_4363BE
15F                 test    eax, eax
161                 jz      short loc_42D16B
163                 push    16h
165                 call    sub_43640E
16A                 pop     ecx
16B
16B loc_42D16B:                             ; CODE XREF: sub_42D15A+7↑j
16B                 test    byte_462274, 2
172                 jz      short loc_42D196
174                 push    17h             ; ProcessorFeature
176                 call    ds:__imp_IsProcessorFeaturePresent
17C                 test    eax, eax
17E                 jz      short loc_42D185
180                 push    7
182                 pop     ecx
183                 int     29h             ; Win8: RtlFailFast(ecx)
185 ; ---------------------------------------------------------------------------
```

همانطور که اشاره نمودیم باج‌افزار Mega Cryptorr فایل‌های موجود در برخی از دایرکتوری‌ها را رمزگذاری نمی‌کند، قطعه کد زیر مربوط به بخشی از این فرایند می‌باشد :

cryptor_dyn.c

```
6380          if ( v23 >= 0x10 )
6381            v52 = v32;
6382          if ( v268 == 7 )
6383          {
6384            v53 = 3;
6385            v54 = "Windows";
6386            if ( *(_DWORD *)v52 == *(_DWORD *)"Windows" )
6387            {
6388              v52 += 4;
6389              v54 = "ows";
6390              v53 = -1;
6391            }
6392            v55 = (unsigned __int8)*v52 < (unsigned __int8)*v54;
6393            if ( *v52 != *v54
6394              || (v56 = v52[1], v55 = v56 < (unsigned __int8)v54[1], v56 != v54[1])
6395              || (v57 = v52[2], v55 = v57 < (unsigned __int8)v54[2], v57 != v54[2])
6396              || v53 != -1 && (v58 = v52[3], v55 = v58 < (unsigned __int8)v54[3], v58 != v54[3]) )
6397            {
6398              v59 = -v55 | 1;
6399            }
6400            else
6401            {
6402              v59 = 0;
6403            }
6404            v32 = v244;
6405            if ( !v59 )
6406              goto LABEL_438;
6407            v23 = v269;
6408          }
6409          v60 = &v267;
6410          if ( v23 >= 0x10 )
6411            v60 = v32;
6412          if ( v268 == 13 )
6413          {
6414            v61 = "Program Files";
6415            v62 = 9;
6416            do
6417            {
6418              if ( *(_DWORD *)v60 != *(_DWORD *)v61 )
6419                break;
6420              v60 += 4;
6421              v61 += 4;
```

همچنین لیست کامل دایرکتوری‌های مورد اشاره در قطعه کد زیر قابل مشاهده است :

IDA View-A | Hex View-1 | Structures | Enums

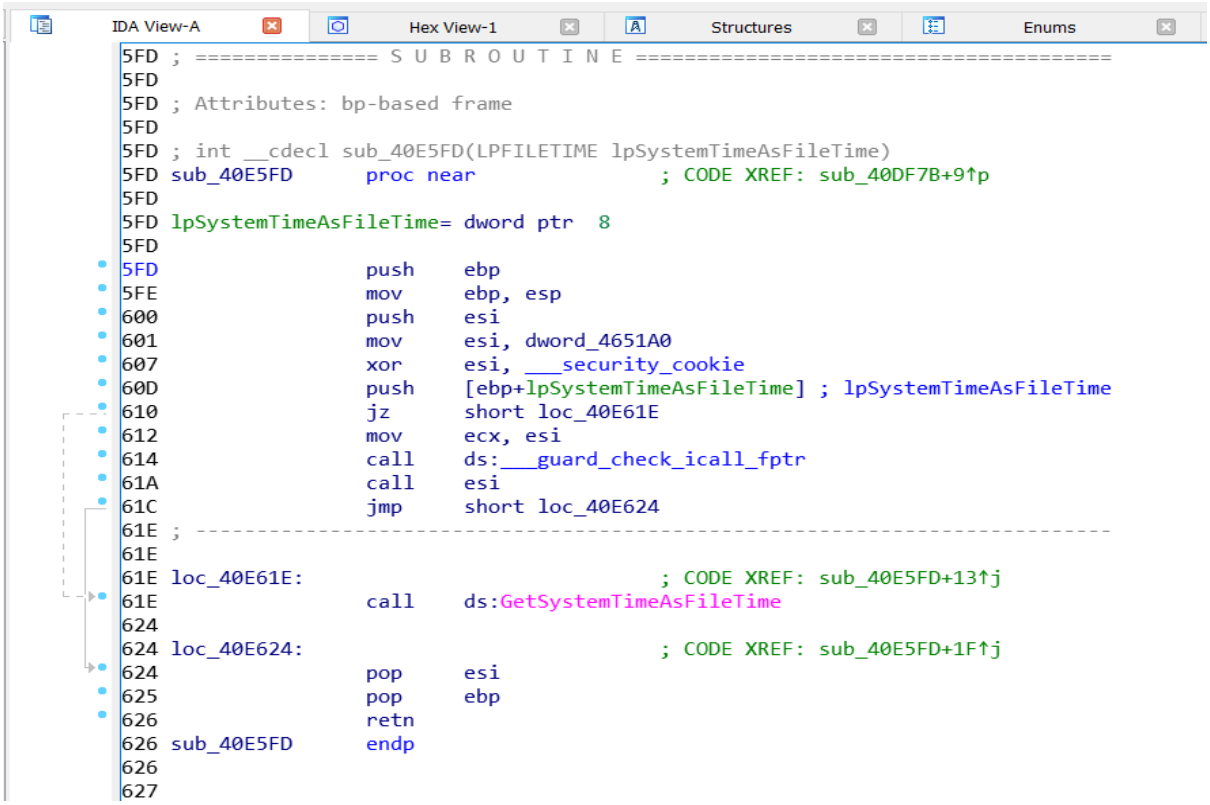
```
.rdata:004538D1                 align 4
.rdata:004538D4 aWindows        db 'Windows',0          ; DATA XREF: sub_4033A0+408↑o
.rdata:004538D4                                         ; sub_404A30+21D↑o
.rdata:004538DC aProgramFiles   db 'Program Files',0    ; DATA XREF: sub_4033A0+46B↑o
.rdata:004538DC                                         ; sub_404A30+274↑o
.rdata:004538EA                 align 4
.rdata:004538EC aProgramFilesX8 db 'Program Files (x86)',0
.rdata:004538EC                                         ; DATA XREF: sub_4033A0+4D5↑o
.rdata:004538EC                                         ; sub_404A30+2D6↑o
.rdata:00453900 aWindowsOld     db 'Windows.old',0      ; DATA XREF: sub_4033A0+542↑o
.rdata:00453900                                         ; sub_404A30+336↑o
.rdata:0045390C aAllUsers       db 'All Users',0        ; DATA XREF: sub_4033A0+5B2↑o
.rdata:0045390C                                         ; sub_404A30+396↑o
.rdata:00453916                 align 4
.rdata:00453918 aIntel          db 'Intel',0            ; DATA XREF: sub_4033A0+627↑o
.rdata:00453918                                         ; sub_404A30+3FB↑o
.rdata:0045391E                 align 10h
.rdata:00453920 aInformationHtm db 'Information.html',0 ; DATA XREF: sub_4033A0+68C↑o
.rdata:00453920                                         ; sub_404A30+454↑o ...
.rdata:00453931                 align 4
.rdata:00453934 aRecycleBin     db '$Recycle.Bin',0     ; DATA XREF: sub_4033A0+6F3↑o
.rdata:00453934                                         ; sub_404A30+4B3↑o
.rdata:00453941                 align 4
.rdata:00453944 aRecycleBin_0   db 'Recycle.Bin',0      ; DATA XREF: sub_4033A0+75F↑o
.rdata:00453944                                         ; sub_404A30+513↑o
.rdata:00453950 aRecycleBin_1   db '$RECYCLE.BIN',0     ; DATA XREF: sub_4033A0+7C9↑o
.rdata:00453950                                         ; sub_404A30+576↑o
.rdata:0045395D                 align 10h
.rdata:00453960 aProgramdata    db 'ProgramData',0      ; DATA XREF: sub_4033A0+830↑o
.rdata:00453960                                         ; sub_404A30+5D3↑o
```

همانطور که اشاره نمودیم این باج‌افزار تنها برخی فایل‌ها با پسوندهای مشخص را رمزگذاری می‌کند قطعه کد زیر مربوط به فرایند بررسی فایل‌های مورد هدف باج‌افزار می‌باشد :

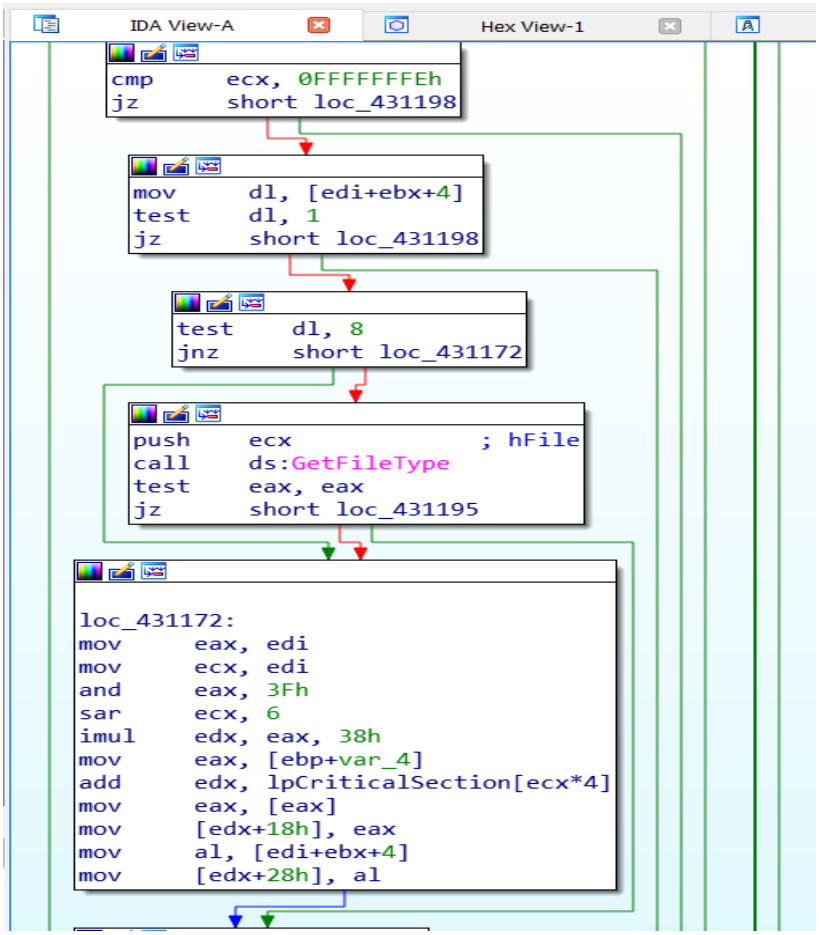
cryptor_dyn.c

```
4316  //----- (00401190) --------------------------------------------------------
4317  int sub_401190()
4318  {
4319    sub_4057A0((unsigned int *)&unk_462AB0, (unsigned int)".gdb", 4u);
4320    dword_462AD8 = 0;
4321    dword_462ADC = 15;
4322    LOBYTE(byte_462AC8[0]) = 0;
4323    sub_4057A0(byte_462AC8, (unsigned int)".ldf", 4u);
4324    dword_462AF0 = 0;
4325    dword_462AF4 = 15;
4326    byte_462AE0 = 0;
4327    sub_4057A0((unsigned int *)&byte_462AE0, (unsigned int)".mdf", 4u);
4328    dword_462B08 = 0;
4329    dword_462B0C = 15;
4330    LOBYTE(byte_462AF8[0]) = 0;
4331    sub_4057A0(byte_462AF8, (unsigned int)".mdb", 4u);
4332    dword_462B20 = 0;
4333    dword_462B24 = 15;
4334    byte_462B10 = 0;
4335    sub_4057A0((unsigned int *)&byte_462B10, (unsigned int)".pdf", 4u);
4336    dword_462B38 = 0;
4337    dword_462B3C = 15;
4338    LOBYTE(byte_462B28[0]) = 0;
4339    sub_4057A0(byte_462B28, (unsigned int)".fdb", 4u);
4340    dword_462B50 = 0;
4341    dword_462B54 = 15;
4342    byte_462B40 = 0;
4343    sub_4057A0((unsigned int *)&byte_462B40, (unsigned int)".mde", 4u);
4344    dword_462B68 = 0;
4345    dword_462B6C = 15;
4346    LOBYTE(byte_462B58[0]) = 0;
4347    sub_4057A0(byte_462B58, (unsigned int)".txt", 4u);
4348    dword_462B80 = 0;
4349    dword_462B84 = 15;
4350    byte_462B70 = 0;
4351    sub_4057A0((unsigned int *)&byte_462B70, (unsigned int)".png", 4u);
4352    dword_462B98 = 0;
4353    dword_462B9C = 15;
4354    LOBYTE(byte_462B88[0]) = 0;
4355    sub_4057A0(byte_462B88, (unsigned int)".jpg", 4u);
4356    dword_462BB0 = 0;
4357    dword_462BB4 = 15;
```

قطعه کد زیر مربوط به تابع GetSystemTimeAsFileTime() می‌باشد که باج‌افزار با استفاده از آن تاریخ و زمان سیستم قربانی را بازیابی می‌کند که به نظر می‌رسد باج‌افزار تنها کاربران خاصی در نقاط مختلف دنیا را مورد هدف خود قرار می‌دهد :



قطعه کد زیر مربوط به تابع GetFileType() می‌باشد که باج‌افزار با استفاده از آن نوع فایل‌ها را بررسی می‌کند:



قطعه کد زیر مربوط به تابع GetLogicalProcessorInformation() می‌باشد که باج‌افزار با استفاده از این تابع اطلاعات مربوط به پردازنده و سخت‌افزار سیستم قربانی را بازیابی می‌کند :

cryptor_dyn.c

```
19891  //------ (004117C5) --------------------------------------------------------
19892  struct _SYSTEM_LOGICAL_PROCESSOR_INFORMATION *__cdecl sub_4117C5(PDWORD ReturnedLength)
19893  {
19894    struct _SYSTEM_LOGICAL_PROCESSOR_INFORMATION *v1; // eax
19895    struct _SYSTEM_LOGICAL_PROCESSOR_INFORMATION *v2; // edi
19896    int v3; // eax
19897    char v5; // [esp+8h] [ebp-10h]
19898    char v6; // [esp+Ch] [ebp-Ch]
19899
19900    GetLogicalProcessorInformation(0, ReturnedLength);
19901    if ( GetLastError() != 122 )
19902    {
19903      v3 = GetLastError();
19904      if ( v3 > 0 )
19905  LABEL_9:
19906        v3 = (unsigned __int16)v3 | 0x80070000;
19907  LABEL_10:
19908      sub_419AA3(&v5, v3);
19909      sub_425BFC(&v5, &_TI2_AVscheduler_resource_allocation_error_Concurrency__);
19910      goto LABEL_11;
19911    }
19912    v1 = (struct _SYSTEM_LOGICAL_PROCESSOR_INFORMATION *)sub_429E55(*ReturnedLength);
19913    v2 = v1;
19914    if ( !v1 )
19915    {
19916      sub_40CABB(&v6);
19917      sub_425BFC(&v6, &_TI2_AVbad_alloc_std__);
19918  LABEL_11:
19919      __debugbreak();
19920      JUMPOUT(*(_DWORD *)sub_41184C);
19921    }
19922    if ( !GetLogicalProcessorInformation(v1, ReturnedLength) )
19923    {
19924      v3 = GetLastError();
19925      if ( v3 > 0 )
19926        goto LABEL_9;
19927      goto LABEL_10;
19928    }
19929    return v2;
19930  }
```

همانطور که اشاره شد باج‌افزار Mega Cryptorr پس از رمزگذاری فایل‌ها، به انتهای آن‌ها پسوند .bip را اضافه می‌کند، قطعه کد زیر مربوط به این فرایند می‌باشد :

cryptor_dyn.c

```
5189
5190    v45 = 0;
5191    v9 = (const CHAR *)&lpFileName;
5192    if ( (unsigned int)a8 >= 0x10 )
5193      v9 = lpFileName;
5194    SetFileAttributesA(v9, 0x80u);
5195    v43 = 0;
5196    v44 = 15;
5197    LOBYTE(v42) = 0;
5198    LOBYTE(v45) = 1;
5199    v10 = sub_402E70(&lpFileName, (unsigned int *)&v39, ".bip");
5200    sub_402DF0((int)&v42, a1, (int)v10);
5201    if ( v41 >= 0x10 )
5202    {
5203      v11 = v39;
5204      if ( v41 + 1 >= 0x1000 )
5205      {
5206        v11 = (_BYTE *)*((_DWORD *)v39 - 1);
5207        if ( (unsigned int)(v39 - v11 - 4) > 0x1F )
5208          sub_429E04(a1, a2);
5209      }
5210      sub_40ECDC(v11);
5211    }
5212    v40 = 0;
5213    v41 = 15;
5214    LOBYTE(v39) = 0;
5215    LOBYTE(v45) = 2;
5216    v12 = &dword_4635C0;
5217    if ( (unsigned int)dword_4635D4 >= 0x10 )
5218      v12 = (int *)dword_4635C0;
5219    v36 = dword_4635D0;
5220    v13 = 0;
5221    v31 = v12;
5222    v37 = 0;
5223    v38 = 0;
5224    do
5225    {
5226      v35[v13] = v13;
5227      ++v13;
5228    }
5229    while ( v13 < 256 );
```

قطعه کد زیر مربوط به بخشی از فرایند بررسی درایوهای مختلف سیستم قربانیان و اسکن آن‌ها جهت رمزگذاری فایل‌های می‌باشد :

cryptor_dyn.c

```
7313    v6 = 15;
7314    v1 = 0;
7315    sub_4057A0((unsigned int *)&v1, (unsigned int)"A:\\", 3u);
7316    sub_404A30(*(_DWORD *)&v1, v2, v3, v4, v5, v6);
7317    v5 = 0;
7318    v6 = 15;
7319    v1 = 0;
7320    sub_4057A0((unsigned int *)&v1, (unsigned int)"B:\\", 3u);
7321    sub_404A30(*(_DWORD *)&v1, v2, v3, v4, v5, v6);
7322    v5 = 0;
7323    v6 = 15;
7324    v1 = 0;
7325    sub_4057A0((unsigned int *)&v1, (unsigned int)"C:\\", 3u);
7326    sub_404A30(*(_DWORD *)&v1, v2, v3, v4, v5, v6);
7327    v5 = 0;
7328    v6 = 15;
7329    v1 = 0;
7330    sub_4057A0((unsigned int *)&v1, (unsigned int)"D:\\", 3u);
7331    sub_404A30(*(_DWORD *)&v1, v2, v3, v4, v5, v6);
7332    v5 = 0;
7333    v6 = 15;
7334    v1 = 0;
7335    sub_4057A0((unsigned int *)&v1, (unsigned int)"E:\\", 3u);
7336    sub_404A30(*(_DWORD *)&v1, v2, v3, v4, v5, v6);
7337    v5 = 0;
7338    v6 = 15;
7339    v1 = 0;
7340    sub_4057A0((unsigned int *)&v1, (unsigned int)"F:\\", 3u);
7341    sub_404A30(*(_DWORD *)&v1, v2, v3, v4, v5, v6);
7342    v5 = 0;
7343    v6 = 15;
7344    v1 = 0;
7345    sub_4057A0((unsigned int *)&v1, (unsigned int)"G:\\", 3u);
7346    sub_404A30(*(_DWORD *)&v1, v2, v3, v4, v5, v6);
7347    v5 = 0;
7348    v6 = 15;
7349    v1 = 0;
7350    sub_4057A0((unsigned int *)&v1, (unsigned int)"H:\\", 3u);
7351    sub_404A30(*(_DWORD *)&v1, v2, v3, v4, v5, v6);
7352    v5 = 0;
7353    v6 = 15;
7354    v1 = 0;
```

قطعه کد زیر مربوط به تابع CreateFileW() می‌باشد که باج‌افزار برای ایجاد و یا باز نمودن فایل‌های مدنظر خود آن را فراخوانی می‌کند :

IDA View-A | Hex View-1 | Structures | Enums

```
50B
50B ; =============== S U B R O U T I N E =======================================
50B
50B ; Attributes: bp-based frame
50B
50B ; int __cdecl sub_43D50B(LPCWSTR lpFileName, LPSECURITY_ATTRIBUTES lpSecurityAttributes,
50B sub_43D50B      proc near               ; CODE XREF: sub_43D858+A4↓p
50B                                         ; sub_43D858+E9↓p ...
50B
50B lpFileName      = dword ptr  8
50B lpSecurityAttributes= dword ptr  0Ch
50B dwDesiredAccess = dword ptr  14h
50B dwCreationDisposition= dword ptr  18h
50B dwShareMode     = dword ptr  1Ch
50B arg_18          = dword ptr  20h
50B arg_1C          = dword ptr  24h
50B
50B                 mov     edi, edi
50D                 push    ebp
50E                 mov     ebp, esp
510                 mov     eax, [ebp+arg_18]
513                 or      eax, [ebp+arg_1C]
516                 push    0               ; hTemplateFile
518                 push    eax             ; dwFlagsAndAttributes
519                 push    [ebp+dwCreationDisposition] ; dwCreationDisposition
51C                 push    [ebp+lpSecurityAttributes] ; lpSecurityAttributes
51F                 push    [ebp+dwShareMode] ; dwShareMode
522                 push    [ebp+dwDesiredAccess] ; dwDesiredAccess
525                 push    [ebp+lpFileName] ; lpFileName
528                 call    ds:CreateFileW
52E                 pop     ebp
52F                 retn
52F sub_43D50B      endp
52F
```

قطعه کد زیر مربوط به تابع GetVolumeInformationA() می‌باشد که باج‌افزار آن را جهت بازیابی اطلاعات مربوط به سیستم فایل فراخوانی می‌کند :

IDA View-A | Hex View-1 | Structures | Enums

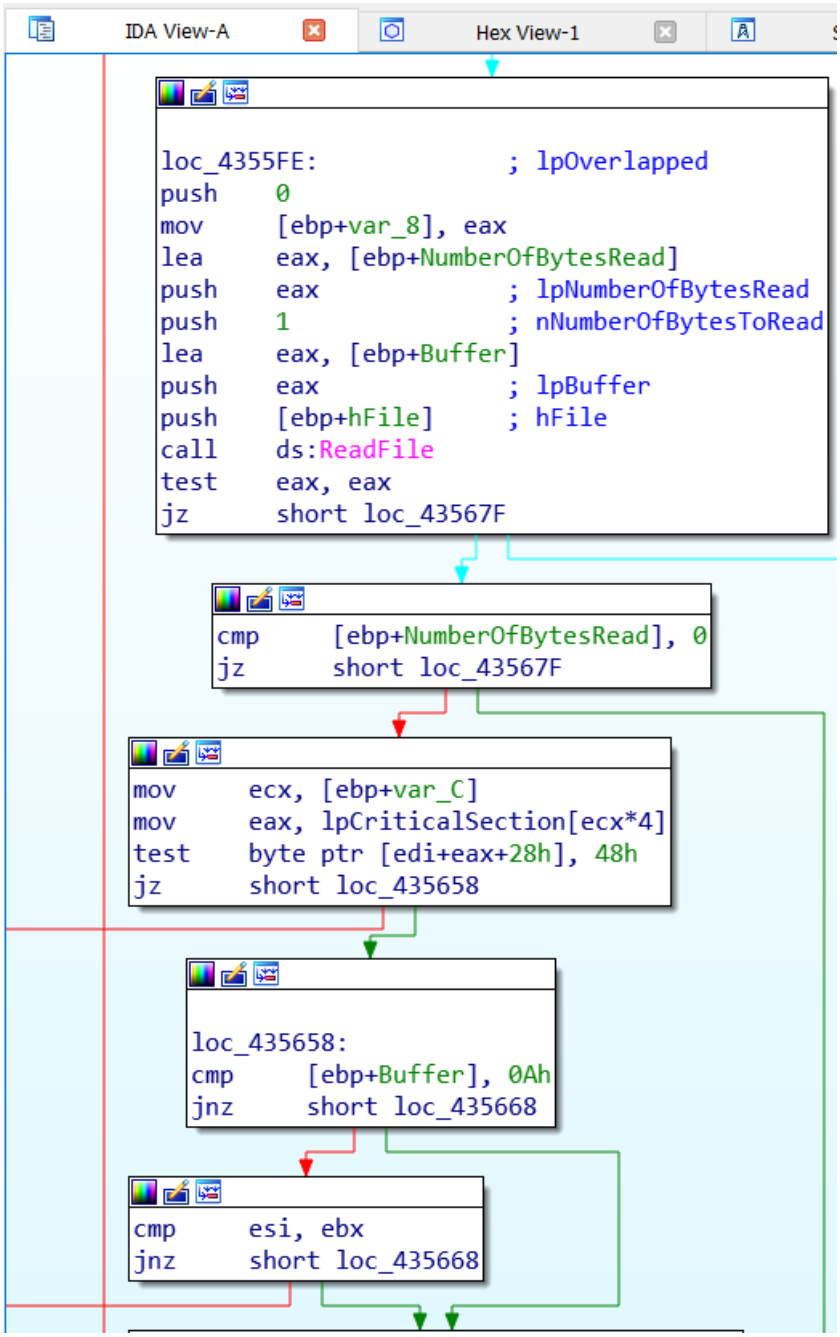
```
010                 push    eax
011                 sub     esp, 2F8h
017                 mov     eax, ___security_cookie
01C                 xor     eax, ebp
01E                 mov     [ebp+var_10], eax
021                 push    eax
022                 lea     eax, [ebp+var_C]
025                 mov     large fs:0, eax
02B                 push    104h            ; nFileSystemNameSize
030                 lea     eax, [ebp+FileSystemNameBuffer]
036                 mov     [ebp+var_2FC], 0
040                 push    eax             ; lpFileSystemNameBuffer
041                 lea     eax, [ebp+FileSystemFlags]
047                 push    eax             ; lpFileSystemFlags
048                 lea     eax, [ebp+MaximumComponentLength]
04E                 push    eax             ; lpMaximumComponentLength
04F                 lea     eax, [ebp+VolumeSerialNumber]
055                 push    eax             ; lpVolumeSerialNumber
056                 push    104h            ; nVolumeNameSize
05B                 lea     eax, [ebp+VolumeNameBuffer]
061                 push    eax             ; lpVolumeNameBuffer
062                 push    offset RootPathName ; "C:\\"
067                 call    ds:GetVolumeInformationA
06D                 push    0B0h
072                 lea     eax, [ebp+var_2F8]
078                 push    0
07A                 push    eax
07B                 call    sub_426770
```

قطعه کد زیر مربوط به تابع GetFileSizeEx() می‌باشد که جهت بازیابی حجم فایل‌ها توسط باج‌افزار فراخوانی می‌شود :

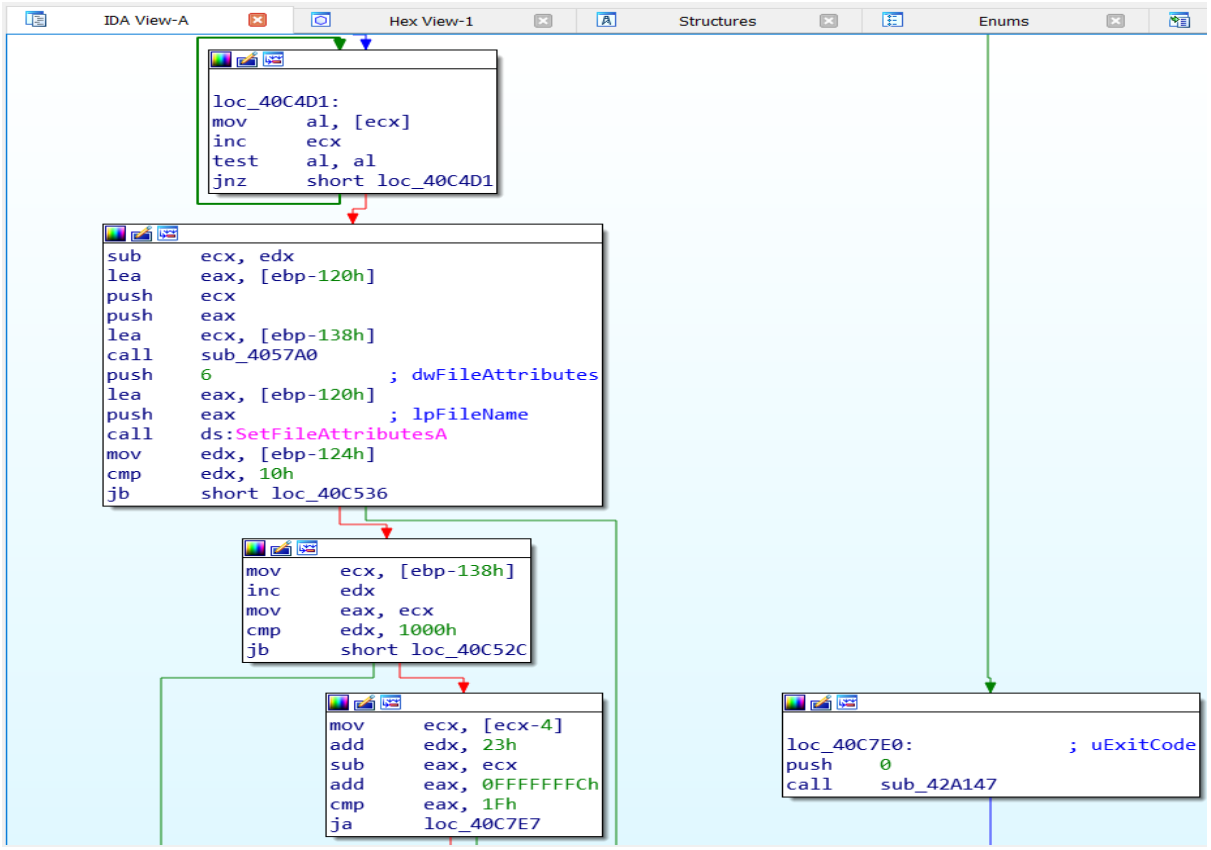
IDA View-A | Hex View-1 | Structures | Enums

```
091
091 loc_431091:                             ; CODE XREF: sub_431060+28↑j
091                 mov     eax, [eax+10h]
094                 nop
095                 push    eax
096                 call    sub_43A1F6
09B                 mov     esi, eax
09D                 pop     ecx
09E                 cmp     esi, 0FFFFFFFFh
0A1                 jz      short loc_4310DF
0A3                 xor     ebx, ebx
0A5                 lea     eax, [ebp+NewFilePointer]
0A8                 inc     ebx
0A9                 push    ebx             ; dwMoveMethod
0AA                 push    eax             ; lpNewFilePointer
0AB                 push    0
0AD                 push    0               ; liDistanceToMove
0AF                 push    esi             ; hFile
0B0                 call    ds:SetFilePointerEx
0B6                 test    eax, eax
0B8                 jz      short loc_4310DF
0BA                 lea     eax, [ebp+FileSize]
0BD                 push    eax             ; lpFileSize
0BE                 push    esi             ; hFile
0BF                 call    ds:GetFileSizeEx
0C5                 test    eax, eax
0C7                 jz      short loc_4310DF
0C9                 mov     eax, dword ptr [ebp+NewFilePointer]
0CC                 cmp     eax, dword ptr [ebp+FileSize]
0CF                 jnz     short loc_4310D9
0D1                 mov     eax, dword ptr [ebp+NewFilePointer+4]
0D4                 cmp     eax, dword ptr [ebp+FileSize+4]
0D7                 jz      short loc_4310DB
0D9
0D9 loc_4310D9:                             ; CODE XREF: sub_431060+6F↑j
0D9                 xor     bl, bl
0DB
0DB loc_4310DB:                             ; CODE XREF: sub_431060+77↑j
0DB                 mov     al, bl
```

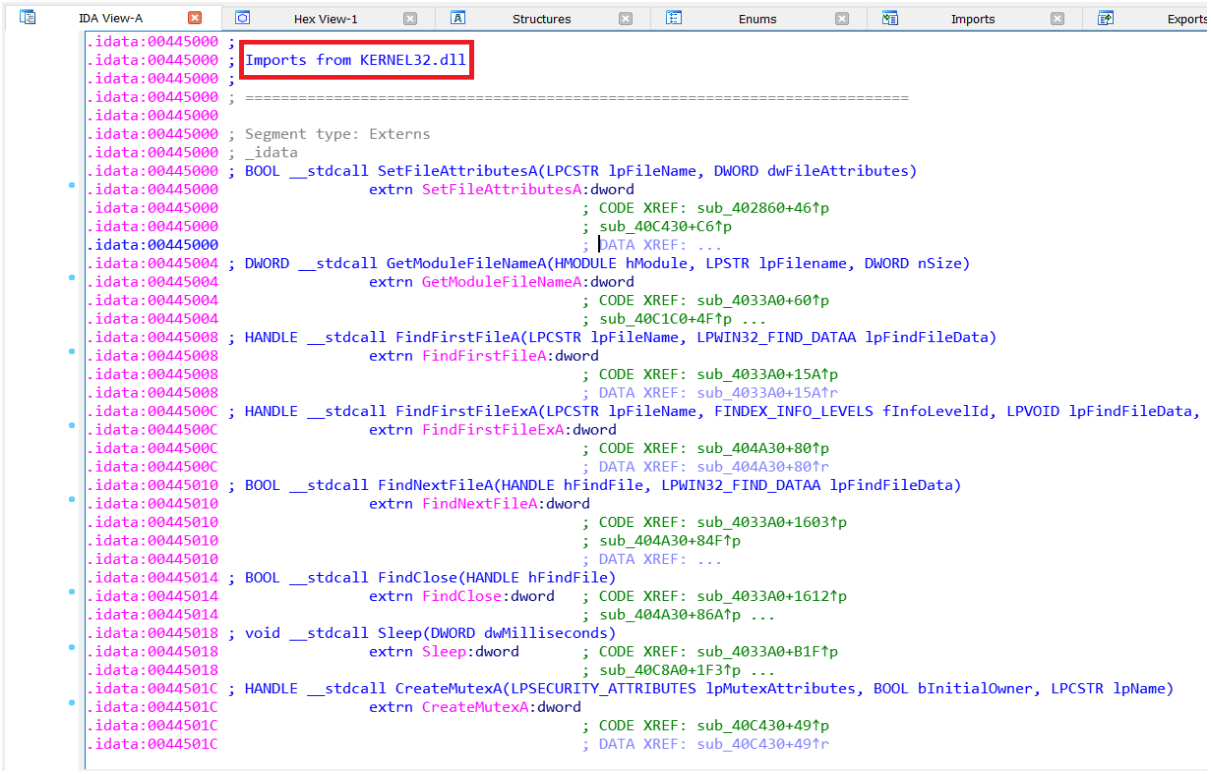
قطعه کد زیر مربوط به تابع ReadFile() می‌باشد و باج‌افزار این تابع را جهت خواندن فایل‌ها، فراخوانی می‌کند :



قطعه کد زیر مربوط به تابع SetFileAttributesA() می‌باشد که با فراخوانی آن توسط باج‌افزار یک ویژگی خاص برای یک فایل یا دایرکتوری تعیین می‌شود :



این باج‌افزار از کتابخانه‌های ویندوزی به همراه توابعی از هرکدام از کتابخانه‌ها استفاده می‌کند، در تصویر، استفاده از این کتابخانه‌ها به خوبی قابل مشاهده است، همچنین لیست کامل این کتابخانه‌ها به همراه توابع مورد استفاده نیز در ادامه‌ی متن آمده است.



SHELL۳۲.dll
ShellExecuteA

KERNEL۳۲.DLL	KERNEL۳۲.dll	KERNEL۳۲.dll	KERNEL۳۲.dll
GetStdHandle	GetProcessAffinityMask	SetFilePointerEx	CreateTimerQueueTimer
InterlockedPopEntrySList	CreateEventW	DeleteTimerQueueTimer	FindNextFileA
WaitForSingleObject	CreateFileW	CreateMutexA	IsValidLocale
SignalObjectAndWait	GetFileType	RegisterWaitForSingleObject	DuplicateHandle
CreateTimerQueue	TlsSetValue	CreateThread	FindFirstFileExW
DeleteCriticalSection	HeapAlloc	SetEnvironmentVariableW	GetUserDefaultLCID
GetCurrentProcess	LeaveCriticalSection	InterlockedFlushSList	ReadConsoleW
GetConsoleMode	GetLastError	SetUnhandledExceptionFilter	ExitProcess
EnumSystemLocalesW	LCMapStringW	IsProcessorFeaturePresent	GetModuleFileNameA
FreeEnvironmentStringsW	GetConsoleCP	ExitThread	GetVolumeInformationA
InitializeSListHead	UnregisterWaitEx	DecodePointer	SetThreadPriority
GetLocaleInfoW	FindNextFileW	TerminateProcess	UnhandledExceptionFilter
SetStdHandle	GetEnvironmentStringsW	GetModuleHandleExW	LoadLibraryExW
GetCPInfo	WaitForSingleObjectEx	ChangeTimerQueueTimer	MultiByteToWideChar
WriteFile	SwitchToThread	SetEndOfFile	DeleteFileW
GetSystemTimeAsFileTime	UnregisterWait	GetCurrentThreadId	GetProcAddress
GetCommandLineA	GetCurrentProcessId	WriteConsoleW	GetProcessHeap
GetThreadTimes	GetCommandLineW	InitializeCriticalSectionAndSpi	QueryDepthSList
HeapReAlloc	WideCharToMultiByte	nCount	CompareStringW
GetStringTypeW	HeapSize	HeapFree	GetFileSizeEx
SetFileAttributesA	SetThreadAffinityMask	EnterCriticalSection	FindFirstFileExA
FreeLibrary	GetCurrentThread	LoadLibraryW	FindFirstFileA
GetThreadPriority	RaiseException	GetVersionExW	GetNumaHighestNodeNu
InterlockedPushEntrySList	ReleaseSemaphore	SetEvent	mber
FindClose	TlsFree	QueryPerformanceCounter	IsValidCodePage
TlsGetValue	GetModuleHandleA	GetTickCount	VirtualFree
EncodePointer	ReadFile	TlsAlloc	Sleep
FreeLibraryAndExitThread	CloseHandle	VirtualProtect	VirtualAlloc
SetLastError	GetModuleHandleW	FlushFileBuffers	GetOEMCP

GetModuleFileNameW TryEnterCriticalSection	GetLogicalProcessorInfor mation	RtlUnwind GetStartupInfoW	GetACP IsDebuggerPresent

بر اساس بررسی‌های صورت گرفته، این باج‌افزار پس از اجرا فرایندهای زیر را ایجاد می‌کند :

cryptor_dyn.exe

- cmd.exe /c reg Add "HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" /v "inf" /t REG_SZ /d "%USERPROFILE%\Information.html" /f
  - reg.exe reg Add "HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" /v "inf" /t REG_SZ /d "%USERPROFILE%\Information.html" /f
- cmd.exe /c reg Add "HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" /v "inf" /t REG_SZ /d "%USERPROFILE%\Information.html" /f
  - reg.exe reg Add "HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" /v "inf" /t REG_SZ /d "%USERPROFILE%\Information.html" /f

باج‌افزار Mega Cryptorr با اجرای فرایندهای فوق پیغام باج‌خواهی را در قسمت Run رجیستری ثبت می‌کند، لذا در هنگام آغاز به کار مجدد سیستم، پیغام باج‌خواهی به قربانیان نمایش داده می‌شود. قطعه کد زیر مربوط به این فرایند می‌باشد :

cryptor_dyn.c

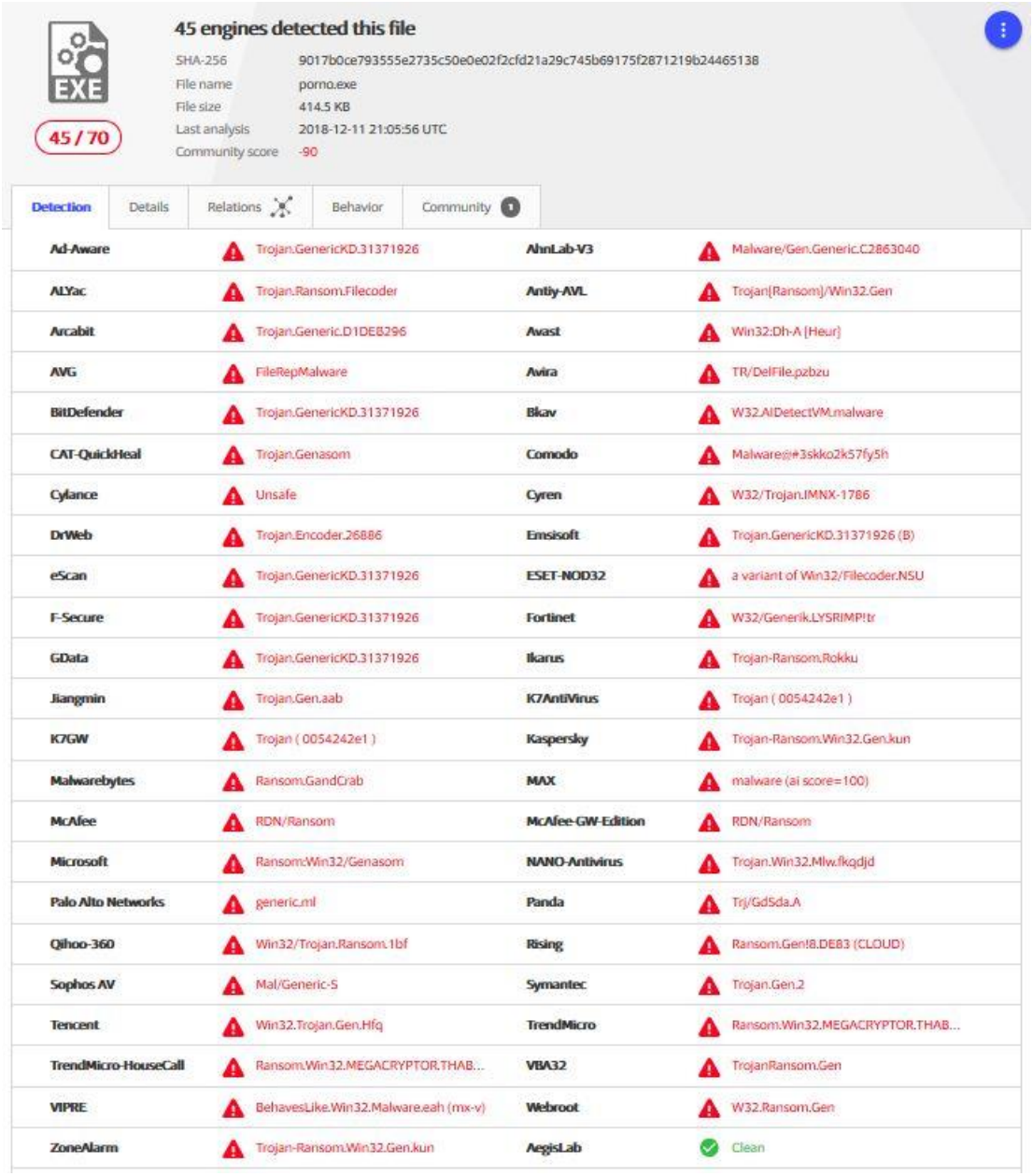
```
8822  v3 = (const char *)sub_42C7B9((unsigned __int8 *)&v50, a2, (__m128i *)"USERPROFILE");
8823  sub_4057A0((unsigned int *)&v40, (unsigned int)v3, strlen(v3));
8824  v47 = 0;
8825  if ( v42 - v41 < 0x11 )
8826  {
8827    v37 = 0;
8828    sub_402FC0((unsigned int *)&v40, 0x11u, *(int *)&v37, (unsigned int)"\\Information.html", 0x11u);
8829  }
8830  else
8831  {
8832    v4 = &v40;
8833    if ( v42 >= 0x10 )
8834      v4 = *(char **)&v40;
8835    v5 = (unsigned int)&v4[v41];
8836    v41 += 17;
8837    sub_4261F0(v5, (unsigned int)"\\Information.html", 0x11u);
8838    *(_BYTE *)(v5 + 17) = 0;
8839  }
8840  v6 = sub_40B2E0(
8841         "reg Add \"HKLM\\SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Run\" /v \"inf\" /t REG_SZ /d \"",
8842         (unsigned int *)&v31,
8843         (unsigned int *)&v40);
8844  LOBYTE(v47) = 1;
8845  v7 = v6[5];
8846  v8 = v6[4];
8847  if ( v6[5] - v8 < 5 )
8848  {
8849    v37 = 0;
8850    v6 = sub_402FC0(v6, 5u, *(int *)&v37, (unsigned int)"\"  /f", 5u);
8851  }
8852  else
8853  {
8854    v6[4] = v8 + 5;
8855    v9 = (unsigned int)v6;
8856    if ( v7 >= 0x10 )
8857      v9 = *v6;
8858    v10 = v9 + v8;
8859    sub_4261F0(v9 + v8, (unsigned int)"\"  /f", 5u);
8860    *(_BYTE *)(v10 + 5) = 0;
8861  }
```

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 Mega Cryptorr نشدیم.

خروجی سامانه VirusTotal :

در حال حاضر تعداد ٤٥ مورد از ٧٠ آنتی ویروس و آنتی بـدافزار موجـود در سـامانه VirusTotal قـادر بـه شناسایی این باج افزار بوده وآن را حذف یا غیرفعال می‌کنند.



45 engines detected this file

SHA-256	9017b0ce793555e2735c50e0e02f2cfd21a29c745b69175f2871219b24465138
File name	porno.exe
File size	414.5 KB
Last analysis	2018-12-11 21:05:56 UTC
Community score	-90

45 / 70

Detection | Details | Relations | Behavior | Community 1

Ad-Aware	Trojan.GenericKD.31371926	AhnLab-V3	Malware/Gen.Generic.C2863040
ALYac	Trojan.Ransom.Filecoder	Antiy-AVL	Trojan[Ransom]/Win32.Gen
Arcabit	Trojan.Generic.D1DEB296	Avast	Win32:Dh-A [Heur]
AVG	FileRepMalware	Avira	TR/DelFile.pzbzu
BitDefender	Trojan.GenericKD.31371926	Bkav	W32.AIDetectVM.malware
CAT-QuickHeal	Trojan.Genasom	Comodo	Malware@#3skko2k57fy5h
Cylance	Unsafe	Cyren	W32/Trojan.IMNX-1786
DrWeb	Trojan.Encoder.26886	Emsisoft	Trojan.GenericKD.31371926 (B)
eScan	Trojan.GenericKD.31371926	ESET-NOD32	a variant of Win32/Filecoder.NSU
F-Secure	Trojan.GenericKD.31371926	Fortinet	W32/Generik.LYSRIMP!tr
GData	Trojan.GenericKD.31371926	Ikarus	Trojan-Ransom.Rokku
Jiangmin	Trojan.Gen.aab	K7AntiVirus	Trojan (0054242e1)
K7GW	Trojan (0054242e1)	Kaspersky	Trojan-Ransom.Win32.Gen.kun
Malwarebytes	Ransom.GandCrab	MAX	malware (ai score=100)
McAfee	RDN/Ransom	McAfee-GW-Edition	RDN/Ransom
Microsoft	Ransom:Win32/Genasom	NANO-Antivirus	Trojan.Win32.Mlw.fkqdjd
Palo Alto Networks	generic.ml	Panda	Trj/Gd5da.A
Qihoo-360	Win32/Trojan.Ransom.1bf	Rising	Ransom.Gen!8.DE83 (CLOUD)
Sophos AV	Mal/Generic-S	Symantec	Trojan.Gen.2
Tencent	Win32.Trojan.Gen.Hfq	TrendMicro	Ransom.Win32.MEGACRYPTOR.THAB...
TrendMicro-HouseCall	Ransom.Win32.MEGACRYPTOR.THAB...	VBA32	TrojanRansom.Gen
VIPRE	BehavesLike.Win32.Malware.eah (mx-v)	Webroot	W32.Ransom.Gen
ZoneAlarm	Trojan-Ransom.Win32.Gen.kun	AegisLab	Clean

خروجی سامانه ویروس‌کاو مرکز ماهر :

در حال حاضر تعداد ۱۲ 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: ۲۴ آذر ۱۳۹۷ - ۳:۳۰

MD5: 0d6ec46b251417db6244103d7559e40c

SHA1: 4e25da96b3d430828da7e3bae05438b0da6f6547

SHA256: 9017b0ce793555e2735c50e0e02f2cfd21a29c745b69175f2871219b24465138

وضعیت:

نتایج اسکن:

آنتی ویروس	نتیجه اسکن	
avast	✓	Clean
comodo	ii	Dangerous
mcafee	ii	Dangerous RDN/Ransom trojan
avira	ii	Dangerous TR/Dell`ile.pzbzu
sophos	ii	Dangerous Troj/Ransom-FDK
escan	ii	Dangerous Trojan.GenericKD.31371926(DB)
symantec	ii	Dangerous Trojan.Gen.2
fprot	✓	Clean
clamav	✓	Clean
bitdefender	ii	Dangerous
drweb	ii	Dangerous Trojan.Encoder.26886
fsecure	ii	Dangerous Trojan.GenericKD.31371926
gdata	ii	Dangerous Trojan.GenericKD.31371926
kaspersky	ii	Dangerous
avg	✓	Clean
پادویش	✓	Clean
eset	ii	Dangerous Win32/Filecoder.NSU trojan