ATTRIBUTION AND BIAS: MY TERRIBLE MISTAKES IN THREAT INTELLIGENCE ATTRIBUTION

Seongsu Park,

Lead security researcher @ GReAT

GREAT

AUG 2022

Seongsu Park

- Global Research and Analysis Team
- Lead security researcher
- Tracking targeted attacks focused on APAC
- Tracking Korean-speaking actors

Focus Area

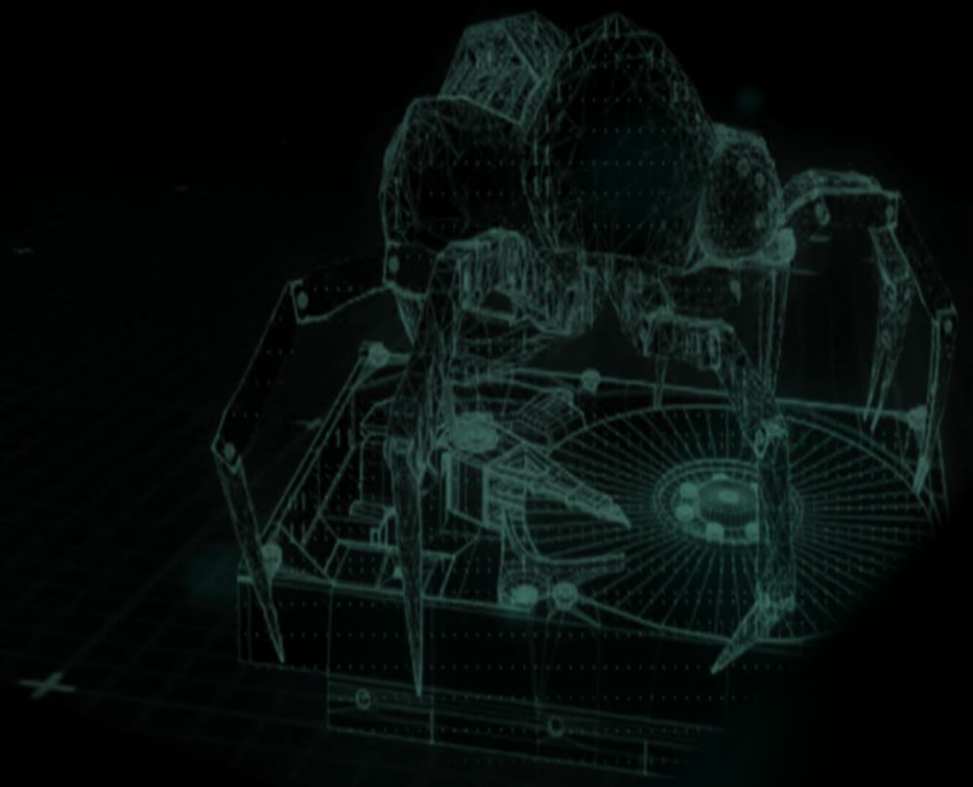
- Investigative Research
- Reversing Malware
- Digital Forensics
- Threat Intelligence

Threat intelligence and Attribution

Threat intelligence is **evidence-based knowledge**, including context, mechanisms, indicators, implications and **actionable advice**, about an existing or emerging menace or hazard to assets to that can be used to inform decisions regarding the subject's response to that menace or hazard.

Attribution is the **process of tracking, identifying and laying blame on the perpetrator** of a cyberattack or other hacking exploit.

Failure #1: Bias perception



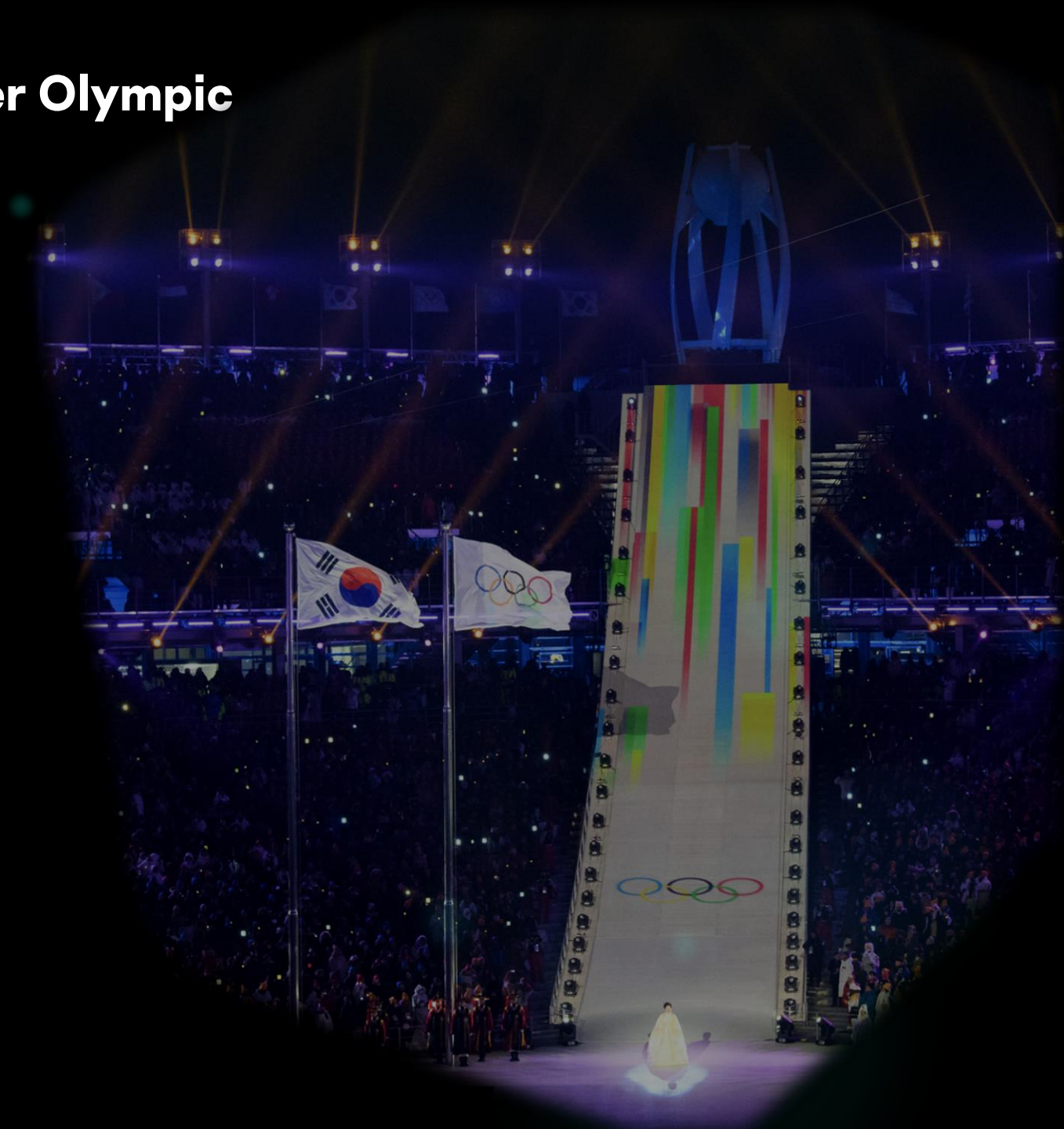
Cyber attack on PyeongChang Winter Olympic

WHEN?

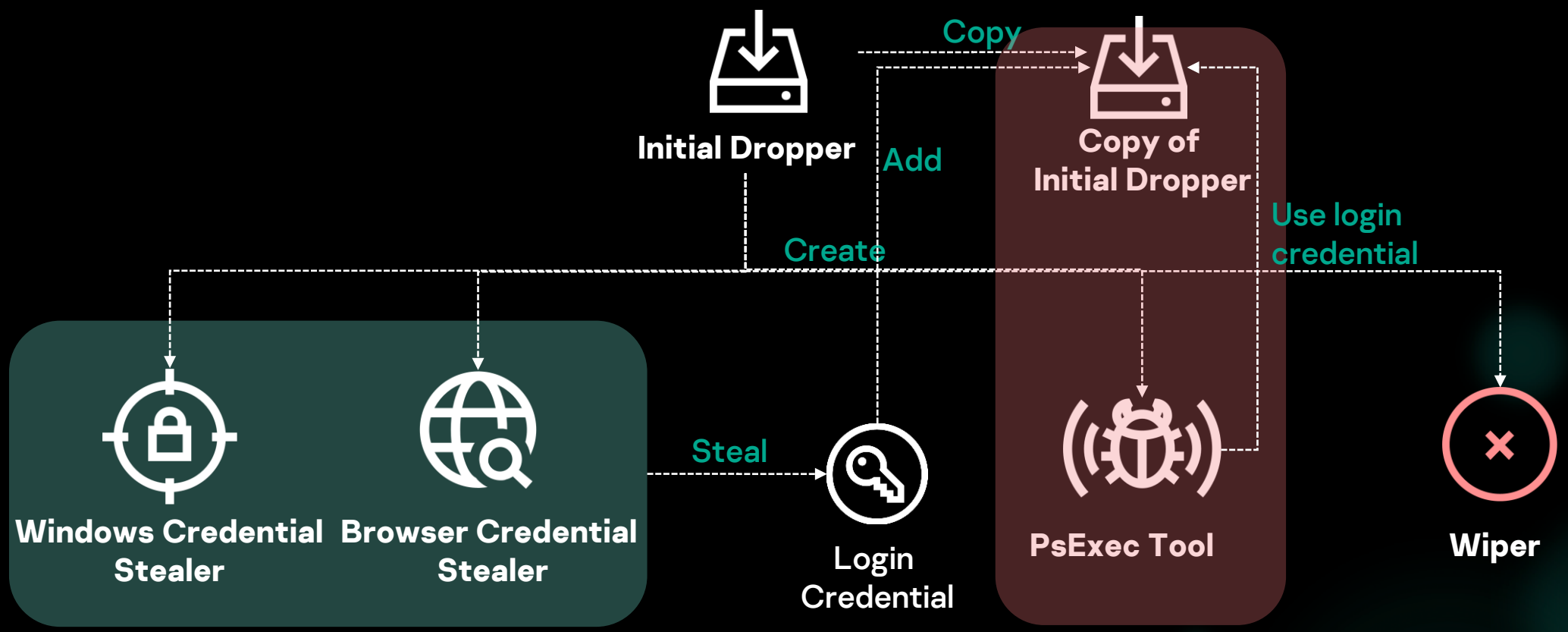
February 09, 2018 20:00 (KST)

DAMAGE OF CYBER ATTACK

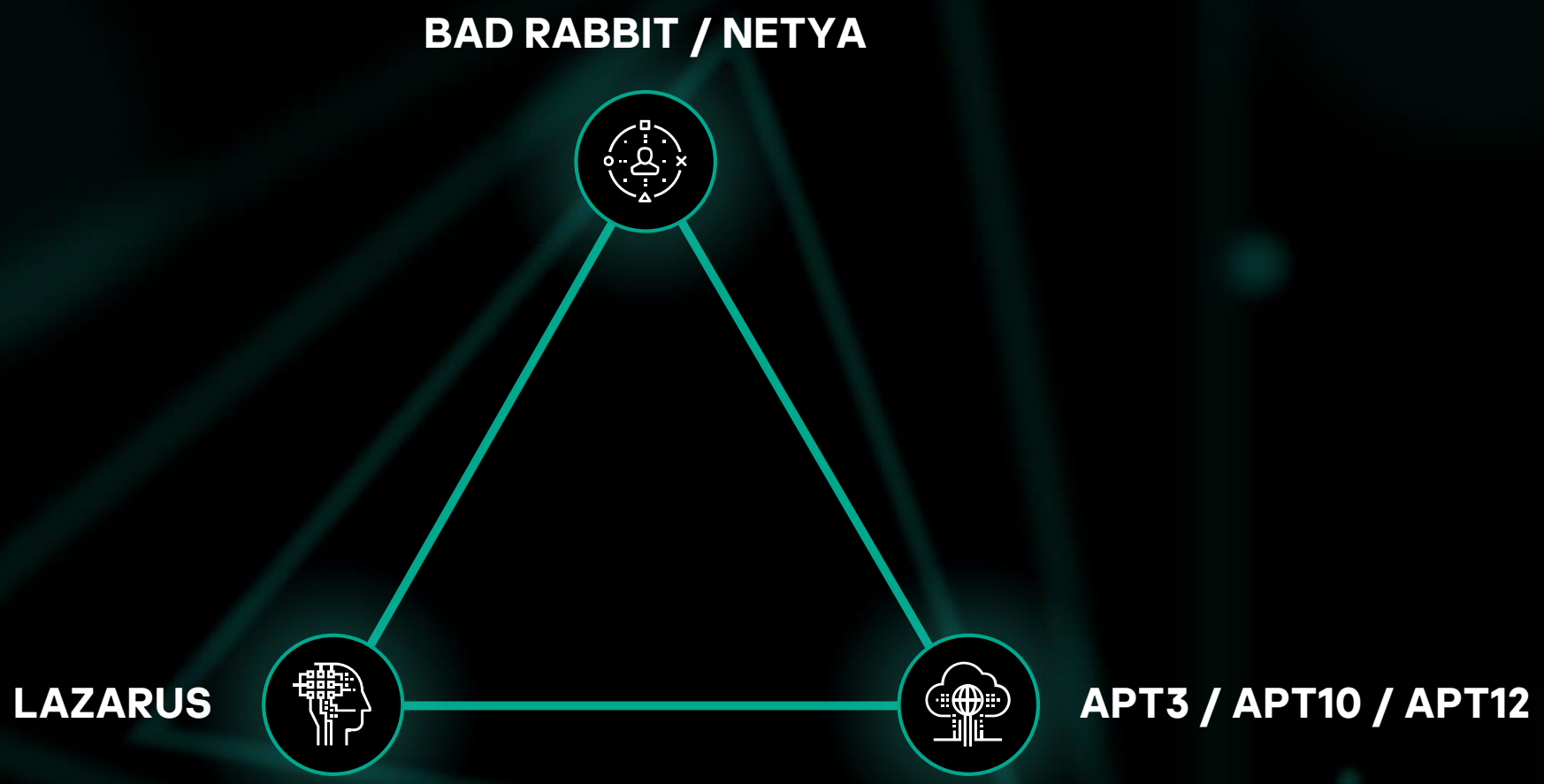
- 50 servers (33 of committee, 17 of partner)
- More 300 servers were affected
- Failure of Wi-Fi, IPTV, Email system
- 4 category, 52 service was stopped
  (Transport, Accommodation, Management of Olympic village, Distribution of uniform..)



Components of OlympicDestroyer



Hell of attribution



Possibly Bluenoroff?

```
Buffer = 0;
memset(&v17, 0, 0xFFCu);
v18 = 0;
v19 = 0;
v1 = CreateFileA(lpFileName, 0x40000000u, 0, 0, 3u, 0x80u, 0);
v2 = v1;
if ( v1 == (HANDLE)-1 )
  return GetLastError();
SetFilePointer(v1, -1, 0, 2u);
WriteFile(v2, &Buffer, 1u, &NumberOfBytesWritten, 0);
FlushFileBuffers(v2);
FileSize.QuadPart = 0i64;
GetFileSizeEx(v2, &FileSize);
SetFilePointer(v2, 0, 0, 0);
v4 = FileSize.HighPart;
v5 = FileSize.LowPart;
v6 = 0;
v7 = 0;
if ( FileSize.HighPart >= 0 && (FileSize.HighPart > 0 || FileSize.LowPart > 0) )
{
  while ( 1 )
  {
    v8 = __OFSUB__(__PAIR__(v4, v5), __PAIR__(v7, v6));
    v11 = v5 - v6;
    v9 = (__PAIR__(v4, v5) - __PAIR__((unsigned int)v7, v6)) >> 32;
    v10 = v5 - v6;
    if ( v9 < 0 || (unsigned __int8)((v9 < 0) ^ v8) | (v9 == 0) && v11 <= 0x1000 )
    {
      v15 = v9;
    }
    else
    {
      v10 = 0x1000;
      v15 = 0;
    }
    if ( !WriteFile(v2, &Buffer, v10, &NumberOfBytesWritten, 0) || !NumberOfBytesWritten )
      break;
    v4 = FileSize.HighPart;
    v12 = NumberOfBytesWritten + v6;
```

```
17  NumberOfBytesWritten = 0;
18  v11 = 0i64;
19  memset(&Buffer, 0, 0x1000u);
20  v2 = CreateFileW(v1, 0x40000000u, 0, 0, 3u, 0x80u, 0);
21  v3 = v2;
22  if ( v2 == (HANDLE)-1 )
23    return GetLastError();
24  SetFilePointer(v2, -1, 0, 2u);
25  if ( WriteFile(v3, &Buffer, 1u, &NumberOfBytesWritten, 0) )
26    FlushFileBuffers(v3);
27  GetFileSizeEx(v3, &FileSize);
28  SetFilePointer(v3, 0, 0, 0);
29  v5 = FileSize.HighPart;
30  v6 = FileSize.LowPart;
31  if ( FileSize.HighPart >= 0 || FileSize.LowPart > 0 )
32  {
33    while ( 1 )
34    {
35      v7 = (__PAIR__((unsigned int)v5, v6) - v11) >> 32;
36      v8 = v6 - v11;
37      if ( __PAIR__(v7, v8) > 0x1000 )
38        v8 = 0x1000;
39      if ( !WriteFile(v3, &Buffer, v8, &NumberOfBytesWritten, 0) || !NumberOfBytesWritten )
40        break;
41      v5 = FileSize.HighPart;
42      v11 += NumberOfBytesWritten;
43      if ( HIDWORD(v11) < FileSize.HighPart )
44      {
45        v6 = FileSize.LowPart;
46      }
47      else
48      {
49        if ( HIDWORD(v11) > FileSize.HighPart )
50          break;
51        v6 = FileSize.LowPart;
52        if ( (unsigned int)v11 > FileSize.LowPart )
53          break;
54      }
55    }
```

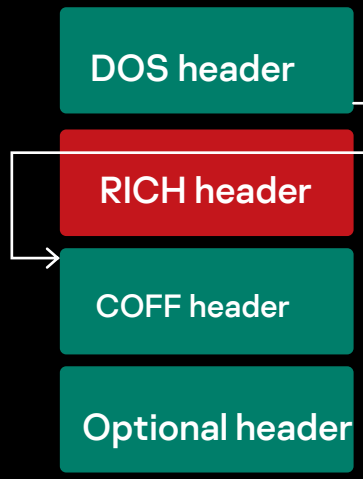
Wiping code similarity with Bluenoroff wiper malware

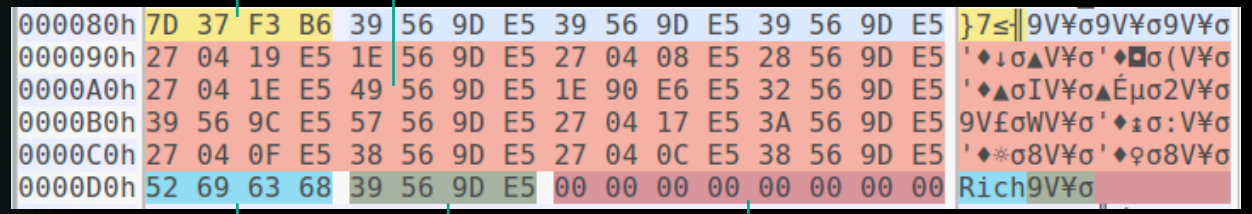
About RICH header

Undocumented header in PE file

- Obfuscated unpublicized part of PE file
- Maybe included from Visual studio 6 (1998)

- **mCV : Miner version of compiler**
- **ProdID : Unique identifier about type of object/tool**
- **Count : How often above values were used by linker**





Source : https://infocon.hackingand.coffee/Hacktivity/Hacktivity%202016/Presentations/George_Webster-and-Julian-Kirsch.pdf

Devil in the RICH header

Extract RICH header of wiper and hunting

Making yara rule and run it to our sample set

```
rule apt_ZZ_Pyeongchang_Olympic_attack_RICH_header {
meta:
        copyright = "Kaspersky Lab"
        description = "Rule to detect Pyeongchang_Olympic_attack samples"
        last_modified = "2018-02-13"
        hash = "3c0d740347b0362331c882c2dee96dbf"
        hash = "5D0FFBC8389F27B0649696F0EF5B3CFE"
        version = "1.0"

strings:
        $c ={00 D3 1E 27 79 97 7F 49 2A 97 7F 49 2A 97 7F 49 2A EC
        63 45 2A 96 7F 49 2A F8 60 43 2A 9C 7F 49 2A 14 63 47
        2A 92 7F 49 2A F8 60 4D 2A 93 7F 49 2A 54 70 14 2A 90
        7F 49 2A 97 7F 48 2A DA 7F 49 2A A1 59 42 2A 94 7F 49
        2A 52 69 63 68 97 7F 49 2A 00}

condition:
        uint16(0) == 0x5A4D and
        filesize < 5000000 and
        $c
}
```

Only 4 Bluenoroff wiper detected!

Devil in the RICH header

Cafully look into Olympic Destroyer wiper RICH header

RICH header in Olympic Destroyer wiper

: Binary created with Visual Studio 6

Raw data	Type	Count	Produced by
000C 1C7B 00000001	oldnames	1	12 build 7291
000A 1F6F 0000000B	cobj	11	VC 6 (build 8047)
000E 1C83 00000005	masm613	5	MASM 6 (build 7299)
0004 1F6F 00000004	stdlibdll	4	VC 6 (build 8047)
005D 0FC3 00000007	sdk/imp	7	VC 2003 (build 4035)
0001 0000 0000004D	imports	77	imports (build 0)
000B 2636 00000003	c++obj	3	VC 6 (build 9782)

mscoree.dll reference of VS6 compiled binary

```
CorExitProcess  m s c o r e e . d l l  r u n t i m e  e r r o r
 T L O S S  e r r o r  S I N G  e r r o r  D O M A I
r  R 6 0 3 3  A t t e m p t  t o  u s e  M S I L
```

___ tmainCRTStartup function of Olympic Destroyer

```
00401822        ___tmainCRTStartup proc near          ; CODE XREF: start+5↓j
00401822        StartupInfo     = _STARTUPINFOW ptr -68h
00401822        var_24          = dword ptr -24h
00401822        var_20          = dword ptr -20h
00401822        var_1C          = dword ptr -1Ch
00401822        ms_exc          = CPPEH_RECORD ptr -18h
00401822 6A 58                  push    58h
```

Olympic Destroyer wiper **compiled on "2018:02:09 10:42:19"** has original RICH header

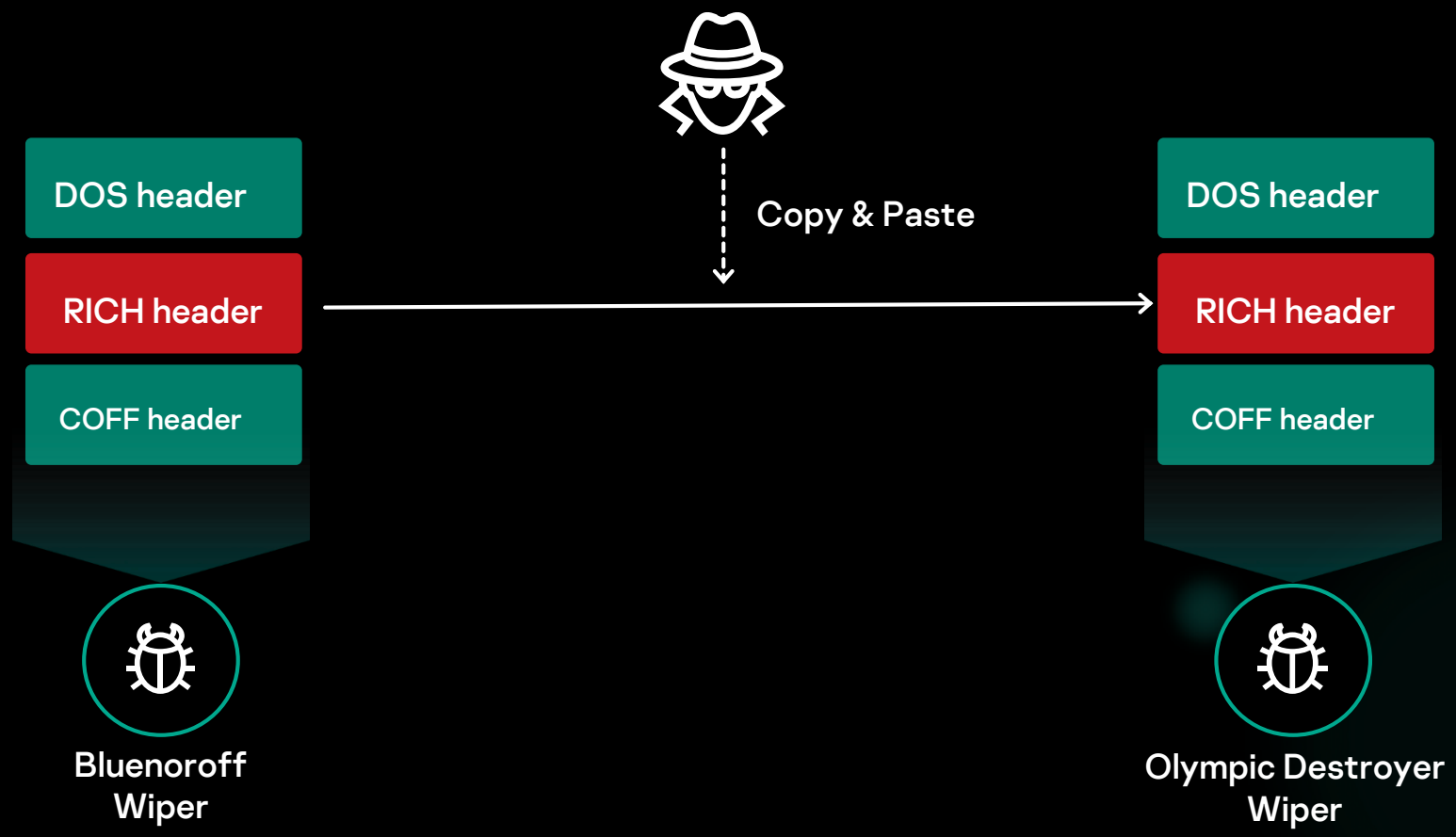
MS Internal Name	Visual Studio Release
prodidUtc1600_CPP	Visual Studio 2010 (10.00)
prodidMasm1000	Visual Studio 2010 (10.00)
prodidUtc1600_C	Visual Studio 2010 (10.00)
prodidImplib900	Visual Studio 2008 (09.00)
prodidImport0	Visual Studio (00.00)
prodidUtc1600_LTCG_CPP	Visual Studio 2010 (10.00)
prodidLinker1000	Visual Studio 2010 (10.00)

Actual version is Visual Studio 2010 (MSVC 10)!

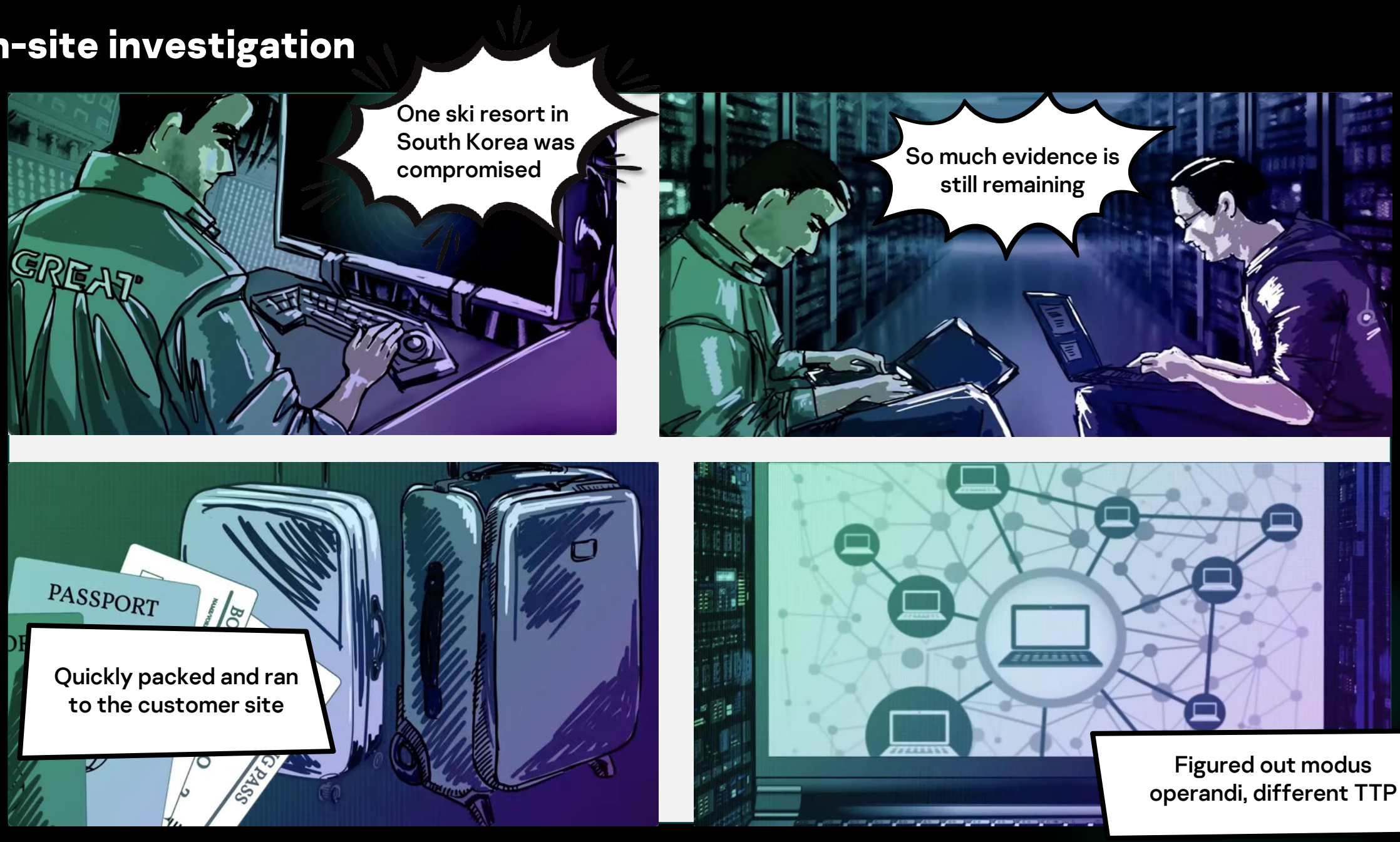
Devil in the RICH header

Malware author copy and paste RICH header from Bluenoroff wiper

Complex false flag operation designed to attribute this attack to Bluenoroff group



On-site investigation



On-site investigation

Initial infection:

Patient 0 was infected before a week of the incident, possibly from a third-party s/w vendor who manages internal s/w

Tools:

Hevily rely on Powershell
Powershell Empier
PowerSploit

Lateral movement:

PsExec, stolen credential, meterpreter, Powershell

Infrastructure:

Additional C2, attacker's server manage Teamviewer

"

Tools, Techniques, and Procedures are totally different from Bluenoroff group

What were the failure factors?

Impatient conclusion with inadequate evidence

Perception of Bias

: Sabotage attack against South Korea = North Korean actors

Failure #2: Over-reliance

Scarcruft VS Darkhotel

Scarcruft

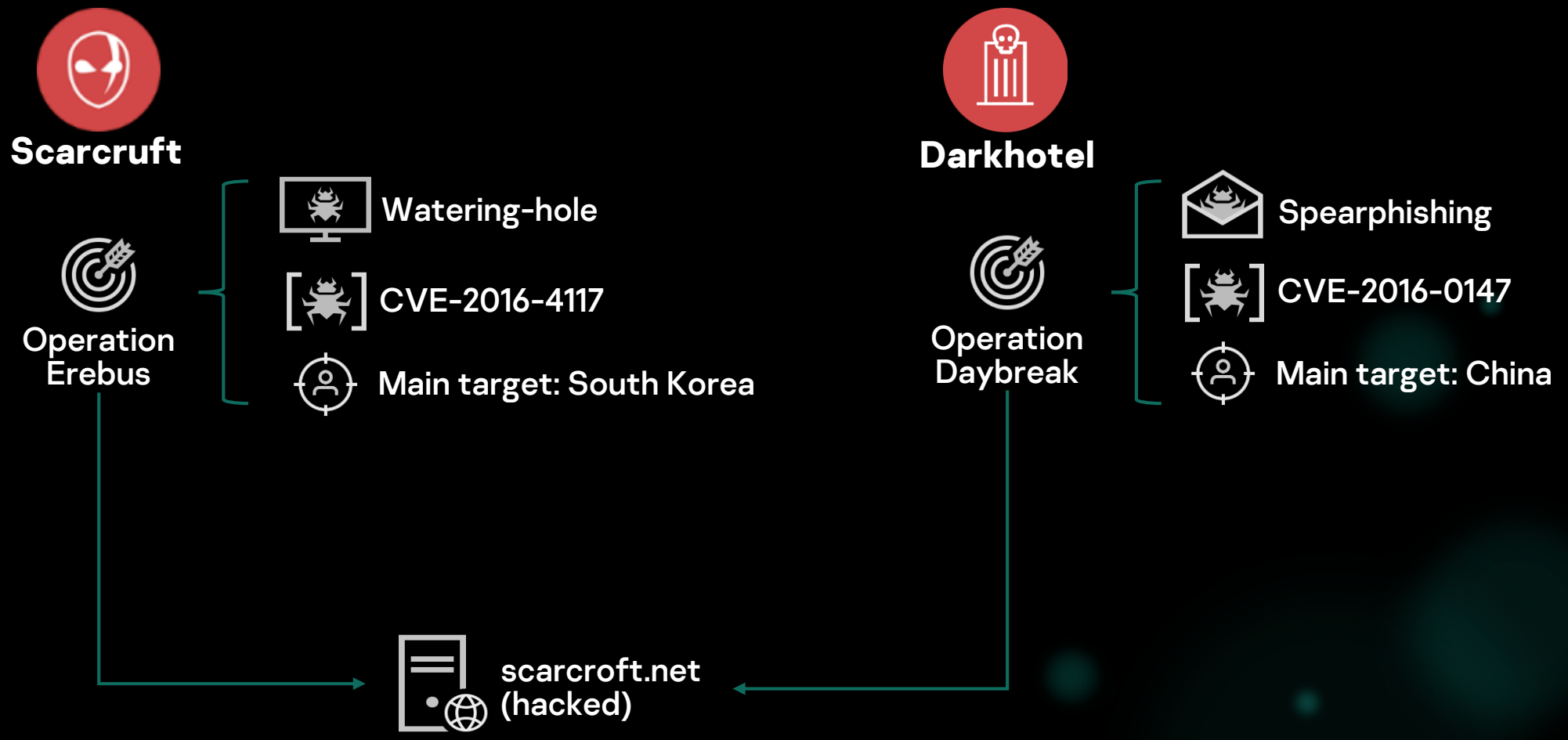
- Published by Kaspersky in 2016
- Targeting the public and private sectors primarily in South Korea
- a.k.a. APT 37, Group 123, InkySquid

Darkhotel

- Published by Kaspersky in 2014
- Compromised hotel networks and hit selected targets
- a.k.a DUBNIUM, Fallout Team

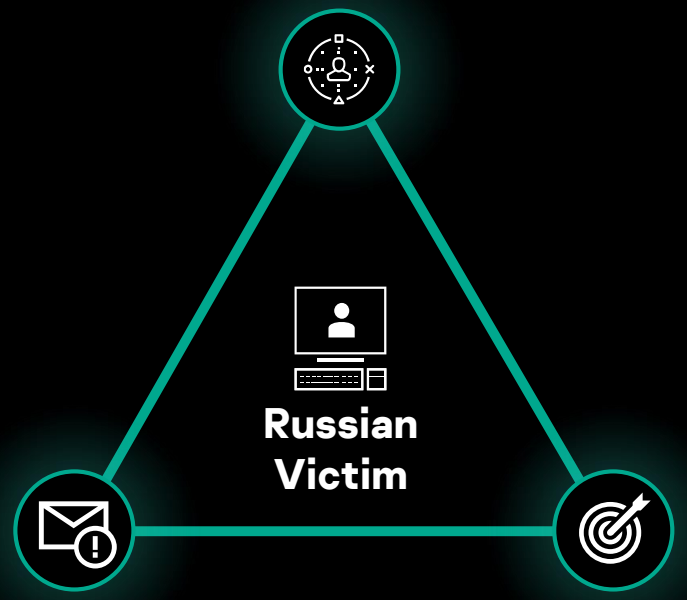
- Korean-speaking actors
- Similar victimology
- Similar TTPs(using 0-day occasionally)
- One group want to hide the other group

Scarcruft VS Darkhotel
: The first conflict of them



Scarcruft VS Darkhotel
: Different actors from the same victim

2018-09-21 ScarCruft
Downloader infected



2018-03-26 DarkHotel
GreezeBackdoor infected

2018-05-01 Download additional file from update.drp.su/drpdump.exe

2018-04-03 Konni
File: Why North Korea slams South Korea's recent defense talks with U.S-Japan.zip

My story: Operation Soundcheck

We uncovered an IE 0day vulnerability has been embedded in malicious MS Office document, targeting limited users by a known APT actor.Details reported to MSRC @msftsecresponse

b48ddad351dd16e4b24f3909c53c8901 Search

1 files found

File	Ratio
10ceb5916cd90e75f8789881af40287c655831c5086ae1575b327556b63cdb24b48ddad351dd16e4b24f3909c53c8901 exploit rtf cve-2018-8174 cve-2017-0199 ole-autolink	36 / 57

similar-to:10ceb5916cd90e75f8789881af40287c655831c5086ae1575 Search

File	Ratio	First sub.
10ceb5916cd90e75f8789881af40287c655831c5086ae1575b327556b63cdb24b48ddad351dd16e4b24f3909c53c8901 exploit rtf cve-2018-8174 cve-2017-0199 ole-autolink	36 / 57	2018-04-18 06:50:30
de1409ccd869153ab444de9740b1733e50f182beea5daea7a9b77e56bd354aa98fe644a70e8d9524c5f71b8c004740fb exploit rtf cve-2018-8174 cve-2017-0199 ole-autolink	31 / 59	2018-05-09 21:00:51
6e2a271f9e137bc8c62fa304ede3b5bac046f4957d3f8249dde60357463e651dbad9b4b415a395650194b3f2081932aa cve-2012-0158 exploit rtf cve-2017-0199 ole-autolink	31 / 59	2018-05-15 08:33:34
45a86012cb99762d57d0fe1626d5cdc9046751e26eac7d9ef0e8adedb03b8661e52c0d49f37bba5cc910fc0f56738720 rtf cve-2012-0158 cve-2018-8174 exploit cve-2017-0199 ole-autolink	21 / 59	2018-07-11 03:34:18
dd7c3564c13536ecd00707abe914b3d5e13971fd1fc3b601c12375e1f46fd15feb4cd469ccaa9f45416377568cc811a6 exploit rtf cve-2017-0199 ole-autolink	23 / 59	2018-01-02 02:22:49
3e652b613182254897cd17203e2f4e97977d0a54389b6a1c0eb32a9289b20cb1b0df2371aa7ff2524ce7524e50b97a7a exploit rtf cve-2017-0199 ole-autolink	36 / 57	2017-07-14 06:28:39
e883a310b0d35fe6932bff0175b236ebda39153a9019950079d1681d5246757b4b290e3be760d0c3cb7ef1a0b64cbd87 cve-2012-0158 exploit rtf cve-2017-0199 ole-autolink	29 / 59	2018-03-13 09:09:48
f1419cde4dd4e1785d6ec6d33afb413e938f6aece2e8d55cf6328a9d2ac3c2d067507ba3f892739ec3d87c6a6e3e0a65 exploit rtf cve-2017-0199 ole-autolink	38 / 57	2017-05-23 07:25:21
4e8257418b0480daeb90f564bd441fe97bc283c1f4a532959935458e2f51dedc2f19acd7f3feff14f4cf8865c73386e5 exploit rtf cve-2017-0199 ole-autolink	34 / 56	2017-08-02 01:56:10
1b988660549a69d34a6be16f28357a1b899b26e65305fc4b46daf25f28ffd56f	36 / 59	2017-04-30

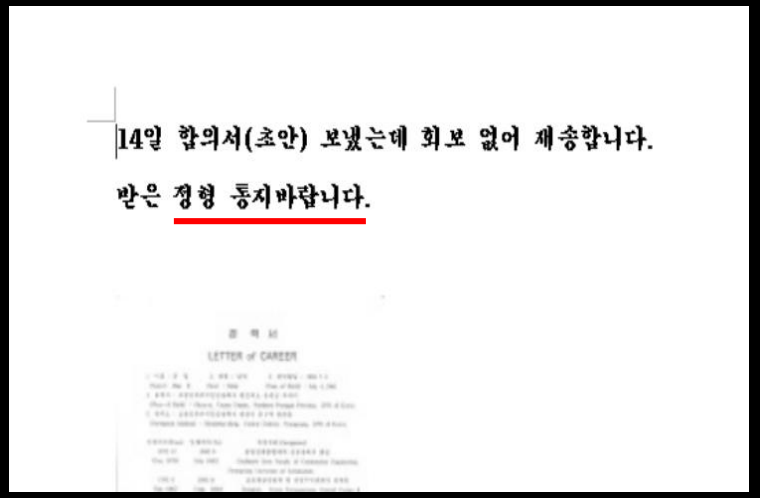
My story: Operation Soundcheck

Falling into the (wrong) rabbit hole

Evidence#1. Other vendors linked these IOCs to the Scarcruft

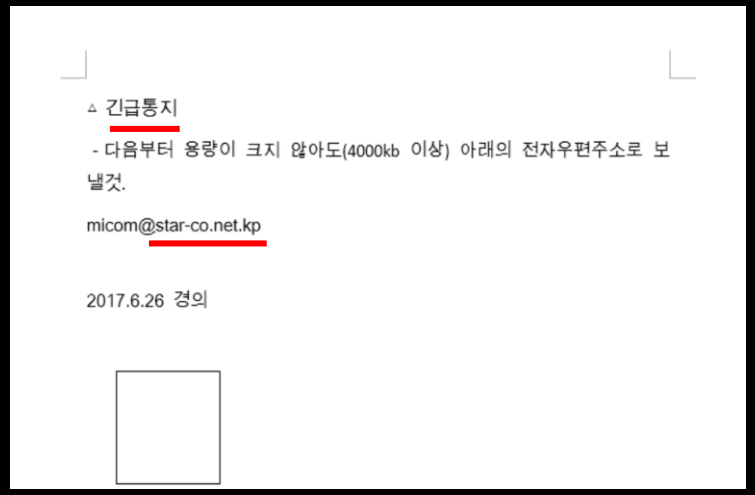
Virustotal similar-to function			
67507ba3f892739ec3d87c6a6e3e0a65	Potter	Potter	old.jrchina[.]com/btob_asiana/udel_calcel.php?fdid=
b0df2371aa7ff2524ce7524e50b97a7a	Potter	Potter	foodforu.heliohost[.]org/blog/udel_calcel.php?fdid=U
4b290e3be760d0c3cb7ef1a0b64cbd87	Windows User	Windows User	livemail-promotion[.]com/bezhan-7/ofac.php
eb4cd469ccaa9f45416377568cc811a6	dunwoody	dunwoody	43.252.37.186/mbitcoin/bitcoinmining.png
2f19acd7f3feff14f4cf8865c73386e5	Potter	Potter	radioideal.com[.]ve/rtn/bbs_view.php?mid=10201706
ce0620a21b0ae4c5a527c5379b9d6664	Lazy	Potter	samiltech[.]com/admin/data/bbs/bbs_in.php

Evidence #2. Lots of North Korean expressions from decoy documents


14일 합의서(초안) 보냈는데 회보 없어 재송합니다.

받은 정형 통지바랍니다.

LETTER of CAREER


△ 긴급통지

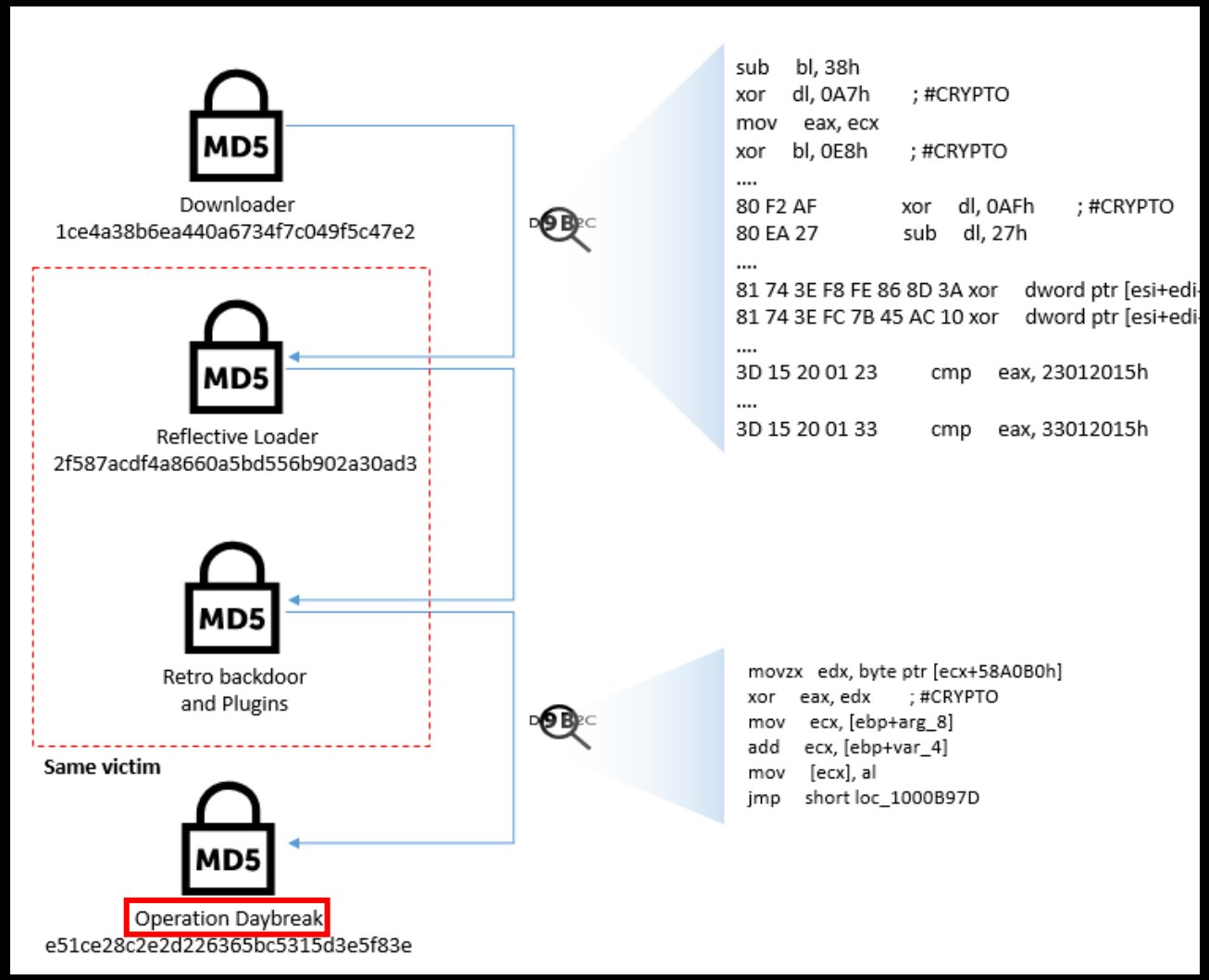
- 다음부터 용량이 크지 않아도(4000kb 이상) 아래의 전자우편주소로 보낼것.

micom@star-co.net.kp

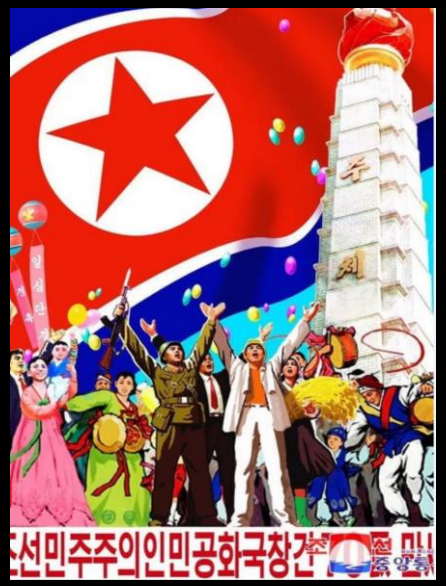
2017.6.26 경의

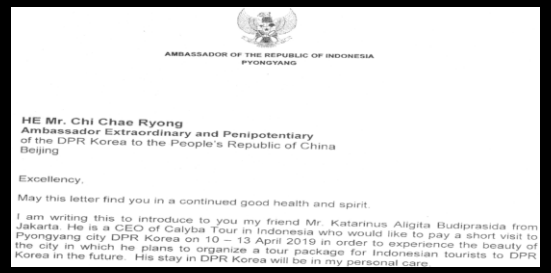
My story: Operation Soundcheck

Operation Soundcheck linked to the Operation DayBreak



Another similar case: Darkhotel vs CoughingDown/Higaisa





AMBASSADOR OF THE REPUBLIC OF INDONESIA
PYONGYANG

HE Mr. Chi Chae Ryong
Ambassador Extraordinary and Penipotentiary
of the DPR Korea to the People's Republic of China
Beijing

Excellency,

May this letter find you in a continued good health and spirit.

I am writing this to introduce to you my friend Mr. Katarinus Aligita Budiprasida from Jakarta. He is a CEO of Calyba Tour in Indonesia who would like to pay a short visit to Pyongyang city DPR Korea on 10 – 13 April 2019 in order to experience the beauty of the city in which he plans to organize a tour package for Indonesian tourists to DPR Korea in the future. His stay in DPR Korea will be in my personal care.

Campaign of Darkhotel?

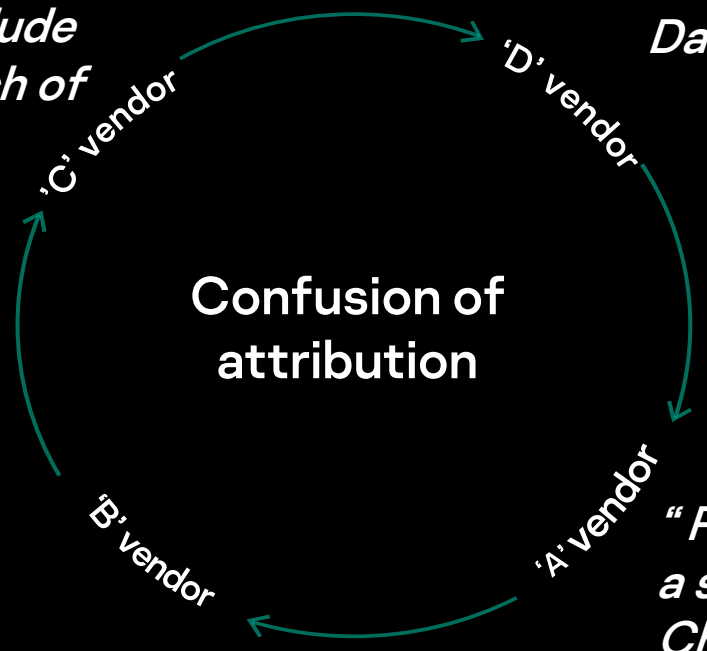
- Victimology
- Uncommon user-agent: Mozilla/5.0 (Macintosh; U; Intel Mac OS X OLE2A; kp);
- Working timezone: GMT+8/9
- Font: 천리마체(Chonrima font)
- (!) Other vendor mentioned similar decoy employed by DarkHotel

Darkhotel was behind in this op

Another similar case: Darkhotel vs CoughingDown/Higaisa

" We did not find any overlap with Darkhotel, But, we do not exclude that this actor may be a branch of Darkhotel

" Gh0st RAT variant is not in Darkhotel arsenal



" APT actor with government background from the Korean Peninsula.

" Polish government attacked with a spearphisng disguised as a Chinese govenrment

The biggest failure factor of my case: Overtrust other's research without verification

What were the failure factors?

- Sophisticate false flag
- Stereotype of language characteristics
- Over-reliance of tools
- Overtrust of other reports

Failure #3: Impatient conclusion

Operation Applejeus: Beginning of saga

Opeation Applejeus

- Campaign of Lazarus
- Targeting cryptocurrency industry with fake company
- First macOS malware of Lazarus group

www.celasllc.com

```
POST /checkupdate.php HTTP/1.1
Accept: image/gif, ..
Host: www.celasllc.com
--jeus
Content-Disposition: form-data; name="api";
get_config
--jeus
Content-Disposition: form-data; name="upload";
filename="temp.gif"
GIF89a...
```

To: SUPPORT@celasllc.com
Date: Jul 20, 2018, 4:45 PM
Subject: Question about Celas Trade Pro

Hi,
This is Seongsu Park from Kaspersky Lab(www.kaspersky.com). Recently, we investigating specific cyber attack, and we suspect the Celas Trade Pro trading program is compromised by a threat actor.

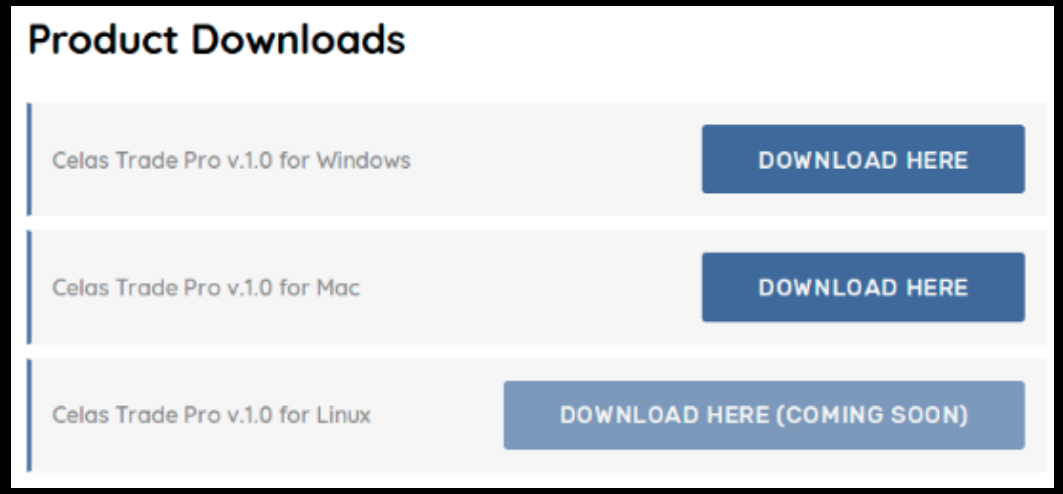
celastradepro_win_installer_1.00.00.msi

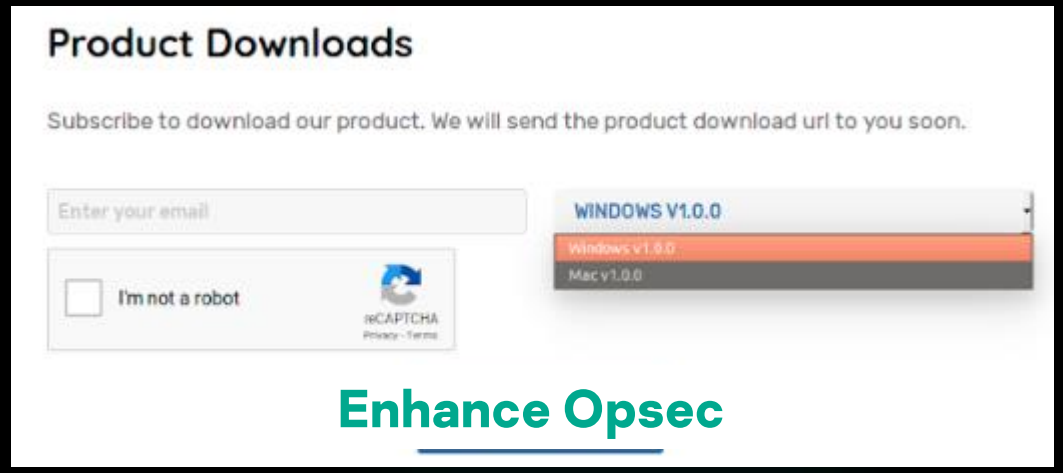
Known Lazarus malware

Cryptocurrency Exchange

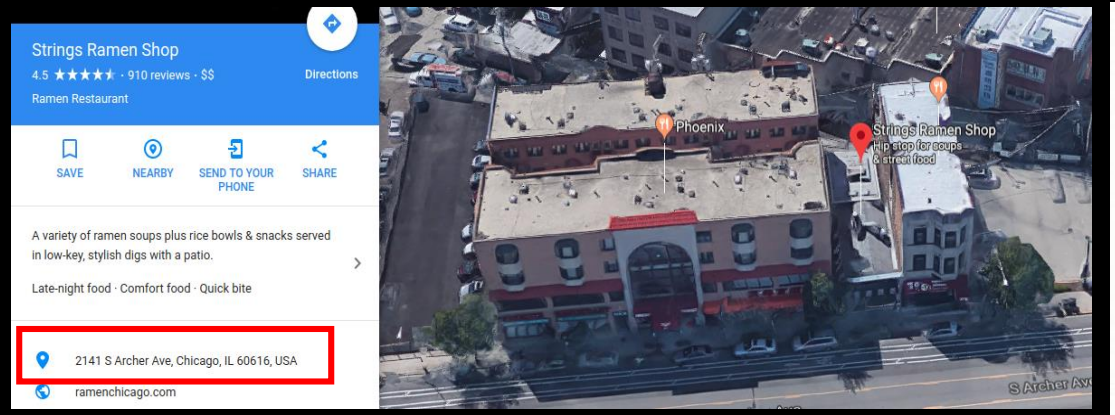
Kind researcher

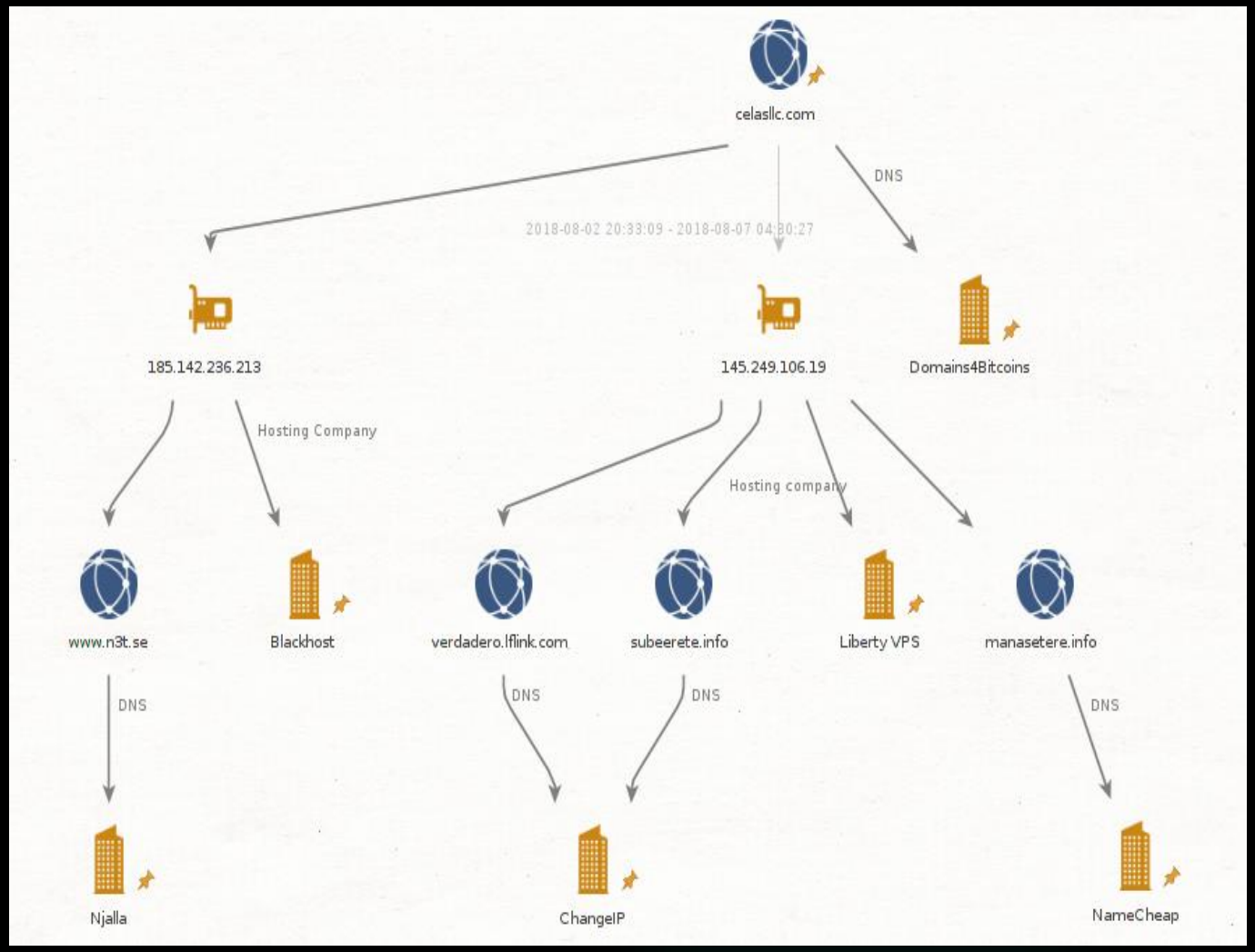
Operation Applejeus: Helping badguy's security enhancement





Operation Applejeus: Infrastructure





What were the failure factors?

Mocked by sophisticated preparation by threat actor:

- manages fake websites with SSL certificate
- develops fake trading application
- operates 24x7 support center

Impatient conclusion

Takeaways

- **Attribution is matter, but VERY HARD.**
- **As human beings, we have lots of stereotypes and perceptions of bias.**
- **Concludes the attribution with as much as evidence we can have.**
- **VERIFY, VERIFY, VERIFY.**

Question?

@unpacker

seongsu.park@kaspersky.com