



INTEZER

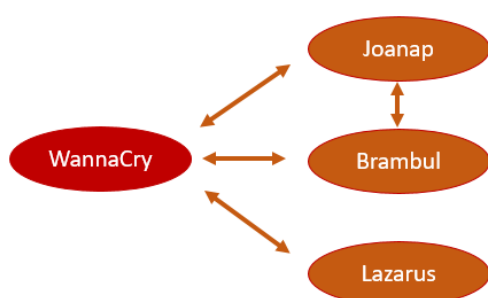
WannaCry Ransomware:
Potential Link to North Korea

Overview

On Friday, 12 May 2017, a large cyber-attack using WannaCry ransomware was launched, infecting more than 230,000 computers in 150 countries, demanding ransom payments in the cryptocurrency bitcoin in 28 languages. WannaCry used the leaked EternalBlue exploit from the NSA, in order to spread itself throughout Windows networks.

Potential Link to North Korea

Using Intezer's Code Intelligence™ technology, we were able to find strong links to other malware families, believed to be developed by North Korean hackers, or known to be used in attacks against South Korean organizations. In this document, we will share some of the details that led us to connect this large-scale ransomware attack to these malware families.



Unique Unzip Library

By extracting thousands of code-pieces (“genes”) from WannaCry samples and identifying them in our Global Genome Database, which contains billions of code pieces of both malware and legitimate applications, we have found several pieces of code from a rare version of a known library. The original, more common library is “unzip 0.15 Copyright 1998 Gilles Vollant”, as we can see in WannaCry’s strings:

```
['s'] .rdata:00410B45 00000006 C Qkkbal
['s'] .rdata:00410DA1 00000005 C wn>Jj
['s'] .rdata:00410E05 0000002A C unzip 0.15 Copyright 1998 Gilles Vollant
['s'] .rdata:004115D6 00000006 unic... @7
['s'] .rdata:0041748R 0000000D C =i& &l 766IA??~
```

But this was actually a modified version of this popular library. The modified version of the library was seen only rarely – mostly in malicious programs. Here is the unique code (“gene”) of this library version:



Code Intelligence™ technology was able to identify this piece of code as extremely unique and mostly seen in malware (e.g. Explosive Malware). Searching for clues, we have conducted a hunt in VirusTotal to find files using the same piece of code.

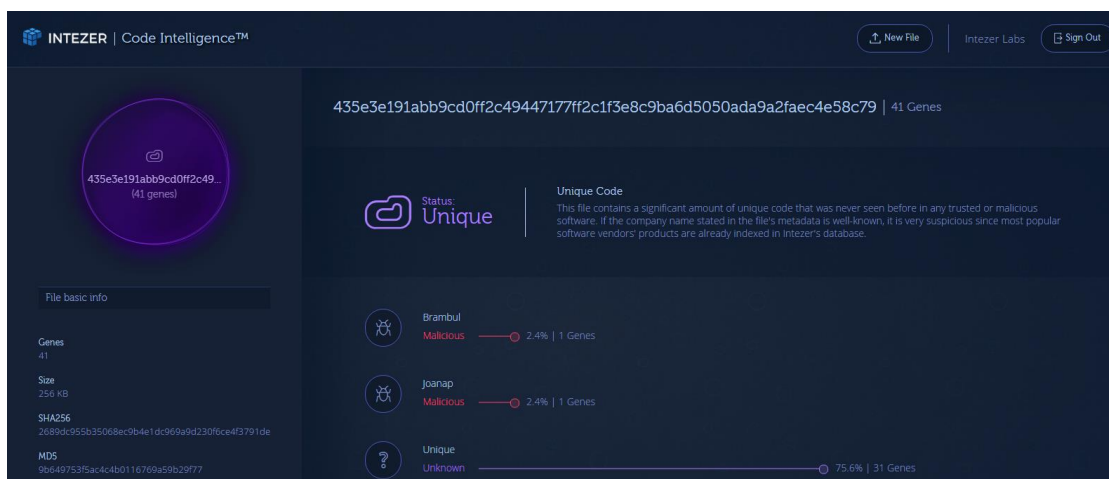
The search resulted in finding a relatively small set of files, many of them were versions of the WannaCry malware. Some of the WannaCry old versions, as of today (May-16, 2017), **are not detected by any Anti Virus (0 / 60 detections)**. These versions are most likely to be modules dumped from memory in a Dionaea (<https://github.com/rep/dionaea>) honeypot server, due to the special submission paths in VirusTotal, such as:

“/var/dionaea/binaries/smb-gse1z2_a.tmp”

The oldest sample uploaded by the honeypot was:

435e3e191abb9cd0ff2c49447177ff2c1f3e8c9ba6d5050ada9a2faec4e58c79 (SHA256)

We believe this is an older version of WannaCry due to its similar use of the rare library, the usage of TOR as C&C, and it’s SMB Worm capability. A quick analysis using Code Intelligence gives the following results:



It is clear to see that this early version of WannaCry, first seen in 4/27/2017 17:06, shares code with Joanap and Brambul malware families, **which are known to attack South Korean organizations**. For example, this is a code piece found similar to Joanap in this sample:

WannaCry
(old version)

Joanap

```

; WannaCry (old version)
argp= dword ptr -120h
timeout= timeval ptr -11Ch
name= sockaddr ptr -114h
writefds= fd_set ptr -104h
arg_0= dword ptr 4
hostshort= word ptr 8
arg_8= dword ptr 0Ch

Sub esp, 120h
mov ecx, dword ptr [esp+120h+hostshort]
mov eax, [esp+120h+arg_0]
push esi
push ecx ; hostshort
mov [esp+128h+argp], 1
mov dword ptr [esp+128h+name.sa_data+2], eax
call htons
push 6 ; protocol
push 1 ; type
push 2 ; af
mov word ptr [esp+130h+name.sa_data], ax
mov [esp+130h+name.sa_family], 2
call socket
mov esi, eax
cmp esi, 0FFFFFFFFh
jz short loc_40423C

; JoanaP
var_120= dword ptr -120h
var_11C= dword ptr -11Ch
var_118= word ptr -118h
var_114= word ptr -114h
var_112= word ptr -112h
var_110= dword ptr -110h
var_104= dword ptr -104h
var_100= dword ptr -100h
arg_0= dword ptr 4
arg_4= dword ptr 8
arg_8= dword ptr 0Ch

sub esp, 120h
mov ecx, [esp+120h+arg_4]
mov eax, [esp+120h+arg_0]
push esi
push ecx
mov [esp+128h+var_120], 1
mov [esp+128h+var_110], eax
call duord_100109D4
push 6
push 1
push 2
mov [esp+130h+var_112], ax
mov [esp+130h+var_114], 2
call duord_10010A00
mov esi, eax
cmp esi, 0FFFFFFFFh
jz short loc_100033C2

; Shared Code
lea edx, [esp+124h+argp]
push edx ; argp
push 8004667Eh ; cnd
push esi ; s
call ioctlsocket
mov eax, [esp+124h+arg_8]
lea ecx, [esp+124h+name]
push 10h ; namelen
push ecx ; name
push esi ; s
mov [esp+130h+writefds.fd_array], esi
mov [esp+130h+writefds.fd_count], 1
mov [esp+130h+timeout.tv_sec], eax
mov [esp+130h+timeout.tv_usec], 0
call connect
lea edx, [esp+124h+timeout]
lea eax, [esp+124h+writefds]
push edx ; timeout
push 0 ; exceptfds
push eax ; writefds
push 0 ; readfds
push 0 ; nfd
call select
test eax, eax
jle short loc_404236

; Shared Code (continued)
mov eax, esi
pop esi
add esp, 120h
ret 0Ch ; call closesocket
    
```

In addition, it seems that this sample contains an SMB brute-force capability and hardcoded password strings, which is very similar to the behavior of Brambul worm, as described in Symantec's blog (<https://www.symantec.com/connect/blogs/duuzer-back-door-trojan-targets-south-korea-take-over-computers>):

['s'] .data:00416653	00000007	C	master
['s'] .data:0041665B	00000007	C	dragon
['s'] .data:00416663	00000009	C	1qaz2wsx
['s'] .data:0041666F	00000007	C	111111
['s'] .data:00416677	00000007	C	abc123
['s'] .data:0041667F	0000000B	C	1234567890
['s'] .data:0041668B	00000008	C	welcome
['s'] .data:00416693	00000008	C	welcome
['s'] .data:0041669B	00000009	C	baseball
['s'] .data:004166A7	00000008	C	1234567
['s'] .data:004166AF	00000005	C	1234
['s'] .data:004166B7	00000009	C	football
['s'] .data:004166C3	0000000A	C	1234567890
['s'] .data:004166CF	00000006	C	12345
['s'] .data:004166D7	00000007	C	qwerty
['s'] .data:004166DF	00000009	C	12345678
['s'] .data:004166EB	00000009	C	password

Clear Code Connections to Lazarus and JoanaP

Intezer's Code Intelligence™ technology immediately found connections between the JoanaP malware and recent WannaCry samples, as shown in the report below:

The screenshot displays the Intezer Code Intelligence interface. At the top, it shows the file hash `b9c5d4339809e0ad9a00d4d3dd26fdf44a32819a54abf846bb9b560d81391c25` and `326 Genes`. A red circle highlights the file's icon and hash. The status is **Malicious**, with a note: "This file is a known malware and exists in Intezer's blacklist." Below this, a list of genes is shown with their respective percentages and gene counts:

- WannaCry: Malicious, 72.4% | 236 Genes
- JoanaP: Malicious, 3.4% | 11 Genes
- Common: Neutral, 21.5% | 70 Genes
- DameWare: Neutral, 2.8% | 9 Genes

On the left, under "File basic info", the following details are listed:

- Genes: 326
- Size: 240 KB
- Company: Microsoft Corporation
- Product: Microsoft® Windows® Operating System
- SHA256: b9c5d4339809e0ad9a00d4d3dd26fdf44a32819a54abf846bb9b560d81391c25
- MD5: 7b72b572a205768755c07238fb32cc

Moreover, Intezer's Code Intelligence™ technology immediately found connection between the Lazarus group, a hacking group associated with North Korea, and previous versions of WannaCry. These versions of WannaCry were found by our search for the unique library described in the previous section.

This is a piece of code from the WannaCry previous version, which according to our Global Genome Database, was seen before **only in the Lazarus group malware**. This specific code comparison analysis was already published by Neel Mehta.

```
var_h= dword ptr -4
arg_0= dword ptr -4

push ecx
push ebx
push ebp
mov ebp, [esp+0Ch+arg_0]
push esi
push edi
push 20h
mov eax, [ebp+0]
lea esi, [ebp+h]
and al, 1
or al, 1
inc esi
mov [ebp+0], eax
mov byte ptr [esi-1], 3
mov byte ptr [esi], 1
inc esi
push esi
call sub_400130 ; time
push 0
add esp, 0Ch ; hostlong
push eax
call ds:tan1
mov [esi], eax
add esi, 20h
mov byte ptr [esi], 0
inc esi
call ds:rand
cdq
mov ecx, 5
xor edi, edi
ldiv ecx
lea eax, [esi+2]
add edx, 2
lea ebx, [edx+edx*2]
shl ebx, 1
test ebx, ebx
jle short loc_402630

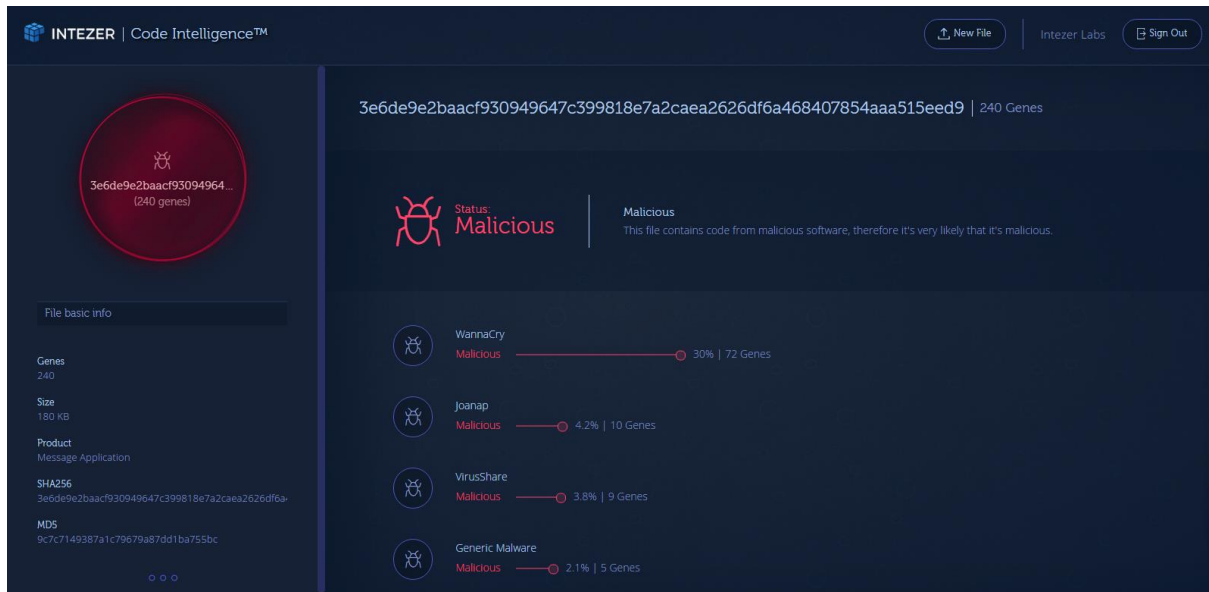
mov [esp+14h+arg_0], eax

loc_4025D5:
call ds:and
xor edx, edx
mov ecx, 4Bh
div ecx
xor eax, eax
test edi, edi
mov [esp+14h+var_h], edx
short loc_4025FA
```

Neel Mehta
@neelmehta Following

9c7c7149387a1c79679a87dd1ba755bc @
0x402560, 0x40F598
ac21c8ad899727137c4b94458d7aa8d8 @
0x10004ba0, 0x10012AA4
[#WannaCryptAttribution](#)

The exact same sample had even stronger connections to the Joana family, with 10 genes appearing in 5 different functions:



Summary

Intezer has seen clear code connections between previously unrelated malware families: WannaCry, Lazarus, Joana and Brambul. This evidence strongly suggests that these hacking tools were written or modified by the same author. In numerous publications, some of the families are already attributed to North Korean hackers, thus it is highly probable that WannaCry ransomware was written or used by North Korean cyber attackers.

About Intezer

Intezer provides disruptive cyber security solutions based on its novel technology, Code Intelligence™. Code Intelligence™ is like “DNA Mapping” for software, able to identify the nature and origins of any unknown file or binary code. Intezer Labs Inc. was founded in 2015 by a unique team of cyber security professionals, including the founder and former CEO of CyberArk, and the former head of the Israeli Military CERT.

Code Intelligence™ Cloud Service

Intezer’s Code Intelligence™ Cloud Service is an online service for rapid File Investigation and Malware analysis that provides a fast, in-depth understanding of any file by mapping its code DNA. Providing a simple interface and API, it functions as a plug-and-play solution for any process within your organization’s incident response plans or daily cyber security monitoring.

This product is extremely effective in reducing false positives, accelerating incident response, and to deal with sophisticated attacks including Fileless malware or any other type of memory samples.

CONTACT US FOR AN INVITE CODE: contact@intezer.com

By detecting code reuse from both legit and malicious software, Code Intelligence™ provides a full understanding of the DNA of any unknown suspect, effectively transforming files and threats into an open book – as if an experienced Reverse Engineer has analyzed the whole assembly code.

Code Intelligence™ dissects any given file or binary into thousands of small fragments, and then compares them to a huge database to check in which software or malware these fragments were seen before, providing a full DNA mapping of the file.

