“Tick” Group Continues Attacks

By Kaoru Hayashi
July 24, 2017 at 6:00 PM
Category: Unit 42
Tags: 9002, Daserf, Datper, Gh0st, HomamDownloader, JAPAN KOREA, Minzen, NamelessHdoor, Tick

The “Tick” group has conducted cyber espionage attacks against organizations in the Republic of Korea and Japan for several years. The group focuses on companies that have intellectual property or sensitive information like those in the Defense and High-Tech industries. The group is known to use custom malware called Daserf, but also employs multiple commodity and custom tools, exploit vulnerabilities, and use social engineering techniques.

Regarding the command and control (C2) infrastructure, Tick previously used domains registered through privacy protection services to keep their anonymity, but have moved to compromised websites in recent attacks. With multiple tools and anonymous infrastructure, they are running longstanding and persistent attack campaigns. We have observed that the adversary has repeatedly attacked a high-profile target in Japan using multiple malware families for the last three years.

Tick Tools

Symantec was first to publicly report on Tick, followed by JPCERT/CC in 2016. These reports discussed the group’s malware, Daserf (a.k.a Muirim or Nioupale) and some additional downloader programs. Though Daserf wasn’t a popular attack tool at the time of publishing the two reports, it dates back to at least 2011. Using AutoFocus, we were able to identify the link among Daserf and two other threats, 9002 and Invader. These threats shared infrastructure between July 2012 and April 2013.

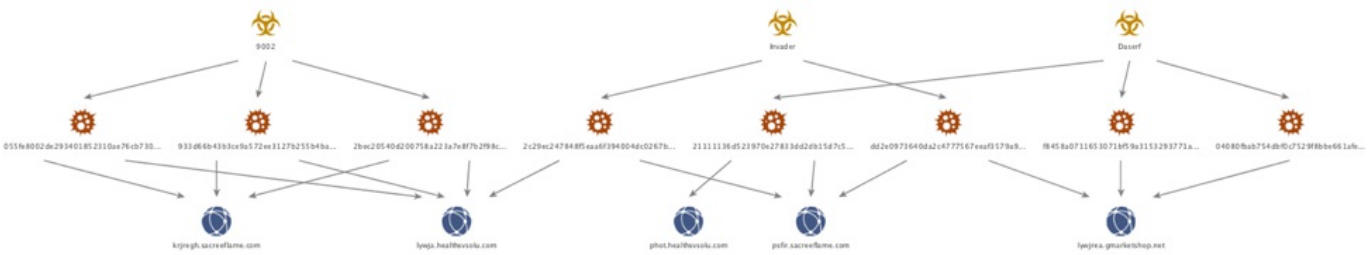

Figure 1 Sharing C2 servers among threats

Invader (a.k.a Kickesgo) is a backdoor that injects its main code into a legitimate process, such as explorer.exe, and has following functions:

- Logs keystrokes and mouse movement
- Captures screenshots
- Opens cmd.exe shell
- Enumerates processes
- Executes programs
- Removes itself
- Enumerates all opening TCP and UDP ports

9002 is the infamous RAT frequently seen in targeted attacks reported by various security vendors, including Palo Alto Networks. Interestingly, the C2 servers linking 9002 to Daserf were described in the report of an Adobe Flash Zero-day attack from FireEye in 2013. These domains were registered through the privacy protection services in 2008 and 2011.

- krjregh.sacreeflame[.]com
- lywja.healthsvsolu[.]com

Though we don't know the targets of these malware samples at the time of writing this article, we suspect the same group is behind these threats for a number of reasons. The samples of Daserf that shared infrastructure were submitted to VirusTotal only from Japan multiple times in 2013. As noted in a later section, another Invader sample shared different C2 servers with Daserf. Symantec reported that Tick exploited additional Adobe Flash and Microsoft Office vulnerabilities. SecureWorks said the adversary group is abusing a previously undisclosed vulnerability in Japanese Software Asset Management system on endpoints. Therefore, Tick or their digital quartermaster is capable of deploying new and unique exploits.

Minzen and Nameless Backdoor

In July 2016, we identified a compromised website in Japan that was hosting a Daserf variant. The web server was also a C2 server for another threat, Minzen (a.k.a, XXMM, Wali, or ShadowWali). The threat often uses compromised web servers in Japan and the Republic of Korea.

As Kaspersky and Cybereason recently posted, Minzen is a modular malware that has both 32-bit and 64-bit components in its resource section or configuration data in its body. One of the Minzen samples (SHA256: 9374040a9e2f47f7037edaac19f21ff1ef6a999ff98c306504f89a37196074a2) found in the Republic of Korea in December 2016 installs simple backdoor module as a final payload on a compromised computer. It opens a TCP port and receives commands from a remote attacker. According to the debug path in the body, the author of the tool called it "NamelessHdoor," and its internal version is identified as "V1.5."

```
0040D6D8                db 'C:¥Users¥123¥Documents¥Visual Studio 2010¥Projects¥shadowDoor¥Rel' ; PdbFileName
0040D6D8                db 'ease¥NamelessHdoor.pdb',0
```

Figure 2 Debug path left in the backdoor module in Minzen

The payload is based on "Nameless Backdoor" which has been publicly available for more than ten years. The oldest code we could identify was hosted on a famous Chinese source code sharing site since 2005. The author of the NamelessHdoor appears to have created additional versions of the Nameless Backdoor by removing unnecessary functions, and added open-source DLL injection code from ReflectiveDLLLoader.

Figure 3 Nameless Backdoor posted on Chinese Source File Sharing Site

There is minimal public information regarding the Nameless Backdoor, except for the interesting report from Cyphort in 2015. The researcher of the company analyzed multiple threats, including Invader, Nioupale(Daserf) and Hdoor found in an attack against an Asian financial institution. We examined the sample described in the report as Hdoor and found it's a previous version of the NamelessHdoor we discovered in the Minzen sample, but without support for DLL injection.

```
.text:00401649                 lea     edx, [esp+43Ch+LibFileName]
.text:0040164D                 push    offset aZf0za   ; "zF((0za"
.text:00401652                 push    edx
.text:00401653                 mov     edi, eax
.text:00401655                 call    simple_substitution_cipher
.text:0040165A                 mov     ebx, ds:GetProcAddress
.text:00401660                 add     esp, 8
.text:00401663                 lea     eax, [esp+43Ch+LibFileName]
.text:00401667                 push    eax             ; lpProcName
.text:00401668                 push    edi             ; hModule
.text:00401669                 call    ebx ; GetProcAddress
```

Figure 4 Strings in NamelessHdoor sample found in 2015

Shared Infrastructure and Cipher Code with Custom Gh0st

Other interesting samples in the report are dllhost.exe and Shell64.dll. We don't have the same files but found possible variants close to their description in the article. These include the following:

- Executable files that connect to the same remote server, blog.softfix.co[.]kr:80, download a DLL file and execute the 'lowmain' export function.
- DLL files have 'lowmain' and 'main' exports.

It turned out that the DLL files we found are a custom variant of Gh0st RAT, and the EXE files download the RAT. Since the source code is publicly available, Gh0st RAT has been used by

multiple actors for years.

The domain, softfix.co[.]kr was registered in 2014. One of subdomains, news.softfix.co[.]kr was the C2 server of Daserf (SHA256: 9c7a34390e92d4551c26a3feb5b181757b3309995acd1f92e0f63f888aa89423). Another subdomain, bbs.softfix.co[.]kr was hosted on same IP address as bbs.gokickes[.]com, which was reported as the C2 server of Invader by Cyphort. We also identified www.gokickes[.]com was the C2 of another Invader variant (SHA256: 57e1d3122e6dc88d9eb2989f081de88a0e6864e767281d509ff58834928895fb).

In addition to the infrastructure, the attacker also shared code. The Gh0st downloaders employ simple substitution ciphers for hiding strings.

```
.text:00401649                  lea     edx, [esp+43Ch+LibFileName]
.text:0040164D                  push    offset aZf0za   ; "zF((0za"
.text:00401652                  push    edx
.text:00401653                  mov     edi, eax
.text:00401655                  call    simple_substitution_cipher
.text:0040165A                  mov     ebx, ds:GetProcAddress
.text:00401660                  add     esp, 8
.text:00401663                  lea     eax, [esp+43Ch+LibFileName]
.text:00401667                  push    eax             ; lpProcName
.text:00401668                  push    edi             ; hModule
.text:00401669                  call    ebx ; GetProcAddress
```

Figure 5 Decryption code in Gh0st Downloader

The cipher converts one character to another based on a substitution table, which can be seen below. As an example, the character ‘K’ in plain text is changed to ‘5’ in cipher text, ‘h’ is converted to ‘j’ and so on. The string ‘connect’ was encoded to ‘zF((0za’ using this table.

Text	characters
plain text	KhL9V1ds5Z”QnfNC&Fb8xGr-()<>[]{}|+THce;0%7Oiz#W DE6qS? aw./BJlk,yUPjgl `^@$*tumYA'p2RoX=v_:M43`
cipher text	5j2Cnx^@$*(){}|+mX k3DK’LGchHNPgZ,z0T8_sRU7)<>”[lBpdfI#%bu;yt-YeoW?4vAMQVa.6qJi:=wFO9&/1ESr

Table 1 Substitution Table used in Gh0st Downloader

The following Python script can decipher the encoded string.

```
plaintext = "KhL9V1ds5Z\"QnfNC&amp;Fb8xGr-()&lt;&gt;[]{}|+THce;0%7Oiz#W DE6qS?aw./BJlk,yU
ciphertext = "5j2Cnx\`^@$*(]{}|+mX k3DK'LGchHNPgZ,z0T8_sRU7)&lt;&gt;\"[lBpdfI#%bu;yt-YeoW

enc_string = "zF((0za"

dec_strings = ''
for c in enc_string:
 dec_strings += plaintext[ciphertext.find(c)]

print dec_strings
```

The exact same table for simple substitution cipher is used in a variant of Daserf (SHA256: 01d681c51ad0c7c3d4b320973c61c28a353624ac665fd390553b364d17911f46). We also found a very similar table in other Tick tools. Since the strings are unique to these threats, we

believe a developer linked to the group built these tools. Because of the shared domains and code, we believe the incident reported by Cyphort have ties to Tick. The following tables were identified for their associated malware samples:

Minzen (SHA256:26727d139b593486237b975e7bdf93a8148c52d5fb48d5fe540a634a16a6ba82):

```
plain text  = "5j2Cnx`^@$*(]{}|+mX k3DK'LGc!hHNPgZ,z0T8_sRU7)&lt;&gt;"[lBpdfI#%bu;yt-YeoW?
cipher text = "KhL9V1ds5Z"QnfNC&amp;Fb8xGr-()&lt;&gt;[]{}|+THce;0%7O!iz#W DE6qS?aw./BJlk,yL
```

Datper (SHA256: 7d70d659c421b50604ce3e0a1bf423ab7e54b9df361360933bac3bb852a31849):

```
plain text  = "KhL9V1ds5Z"QnfNC&amp;Fb8xGr-()&lt;&gt;[]{}|+THce;0%7O!iz#W DE6qS?aw./BJlk,yL
cipher text = "5j2Cnx`^@$*(]{}|+mX k3DK'LGc!hHNPgZ,z0T8_sRU7)&lt;&gt;"[lBpdfI#%bu;yt-YeoW?
```

Spearphishing Email with Patched File Encryption Program

We also identified another malware family, HomamDownloader, sharing some servers with Daserf. An overview of the connections among these threats is discussed in below.

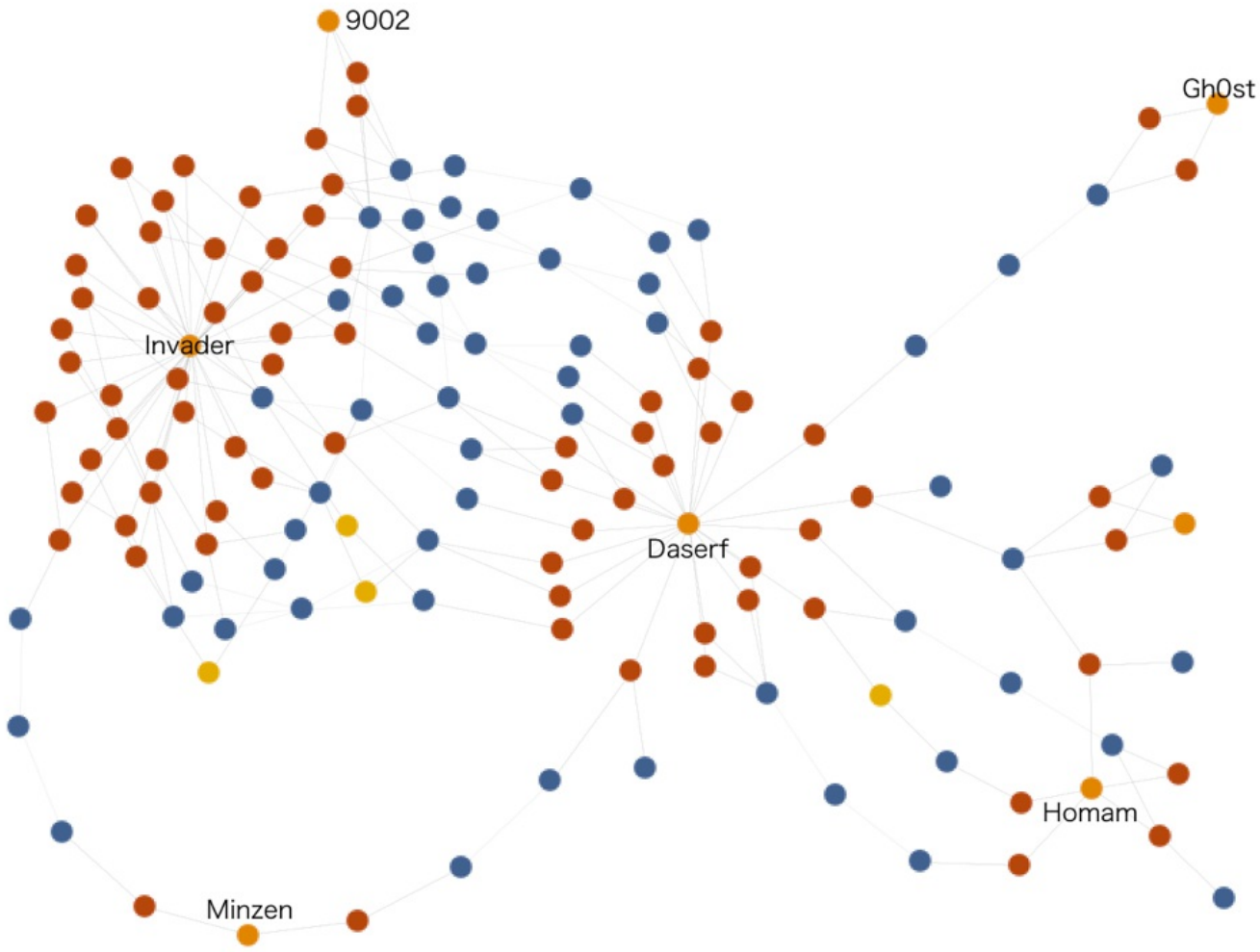

Figure 6 Links among threats and servers

HomamDownloader is a small downloader program with minimal interesting characteristics from a technical point of view. HomamDownloader was discovered to be delivered by Tick via a spearphishing email. The adversary crafted credible email and attachment after understanding the targets and their behavior.

The email below was sent from a personal email account with a subject line of “New Year

Wishes on January 1st". The message asked the recipient to rename the attachment extension from "._X_" to ".exe" and opening it with the password specified in the email to view the Happy New Year eCard in the correct and polite language.

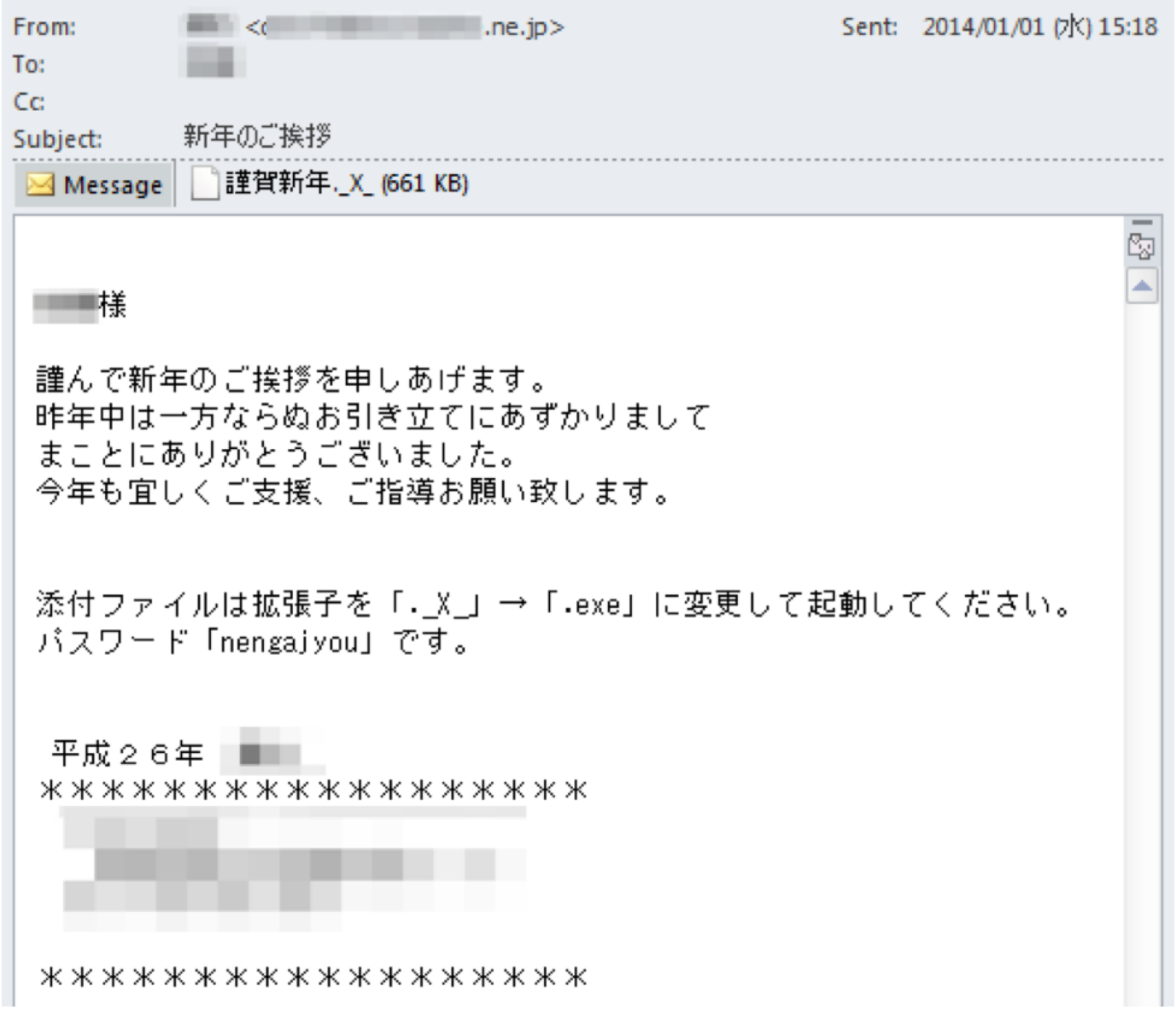

Figure 7 Spearphishing Email with HomamDownloader

The image above is translated to the following in English:

Dear XXXX,

Heartfelt Greetings for the New Year.
Thank you very much for your support over the past year.
I will greatly appreciate your further guidance and encouragement.

Would you please change the file extension of the attachment from "._X_" to ".exe" and open it?
Password is "nengajyo".

For those who are not familiar with Japanese companies, the email must look suspicious, especially given that the executable file attachment has the incorrect file extension. However, this may look legitimate in some cases. Many Japanese companies introduced a file encryption system for secure data exchange over email. The system encrypts documents with a user-specified password and often creates a self-extracting (SFX) file for ease of decrypting the file to recipients. When sending the SFX file with a password by email, senders

usually rename the file extension from .exe to something else to avoid blocking or detecting the attachment by an email gateway or security product. The adversary may know Japanese enterprise users exchange these emails in such a way and crafts the spearphishing email in the same manner.

In addition to the social engineering email technique, the attacker also employs a trick to the attachment. The actor embedded malicious code to a resource section of the legitimate SFX file created by a file encryption tool, and modified the entry point of the program for jumping to the malicious code soon after the SFX program starts. The malicious code drops HomamDownloader, then jumps back to the regular flow in the CODE section, which in turn asks the user the password and decrypts the file. Therefore, once a user executes the attachment and sees the password dialog on SFX, the downloader dropped by the malicious code starts working even if the user chooses the Cancel on the password window. Should the user become aware of the infection later, it may be difficult to find the cause due to the fact that the original embedded file contained within the SFX is benign.

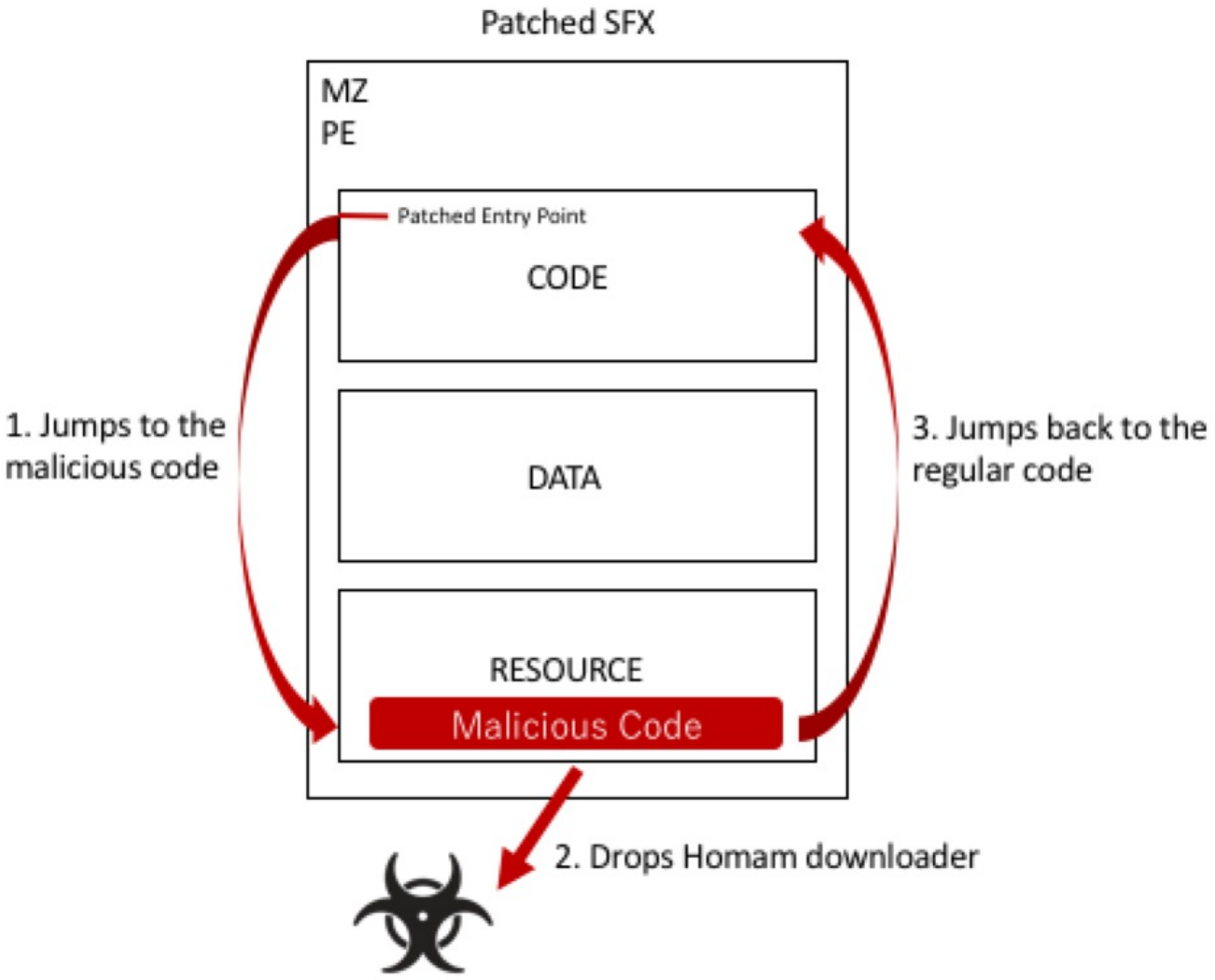

Figure 8 Execution flow of Patched SFX file

Conclusion

Tick was spotted last year, but they are actively and silently attacking various organizations in South Korea and Japan for a number of years. While some of the group's tools, tactics, and procedures (TTPs) have been covered within this article, it is likely there is much that still remains uncovered.

Palo Alto Networks customers are protected by these threats in the following ways:

1. All samples discussed are classified as malicious by the WildFire sandbox platform.
2. All identified domains have been classified as malicious.
3. AutoFocus users can track the malware described in this report using Tick campaign tag and various malware tags.
4. Customers running Traps are protected from the discussed threats.

Indicator of compromise

SHA256

Daserf

- 04080fbab754dbf0c7529f8bbe661afef9c2cba74e3797428538ed5c243d705a
- f8458a0711653071bf59a3153293771a6fb5d1de9af7ea814de58f473cba9d06
- e8edde4519763bb6669ba99e33b4803a7655805b8c3475b49af0a49913577e51
- 21111136d523970e27833dd2db15d7c50803d8f6f4f377d4d9602ba9fbd355cd
- 9c7a34390e92d4551c26a3feb5b181757b3309995acd1f92e0f63f888aa89423

Invader

- 0df20ccd074b722d5fe1358b329c7bdebcd7e3902a1ca4ca8d5a98cc5ce4c287
- e9574627349aeb7dd7f5b9f9c5ede7faa06511d7fdf98804526ca1b2e7ce127e
- 57e1d3122e6dc88d9eb2989f081de88a0e6864e767281d509ff58834928895fb

9002

- 933d66b43b3ce9a572ee3127b255b4baf69d6fdd7cb24da609b52ee277baa76e
- 2bec20540d200758a223a7e8f7b2f98cd4949e106c1907d3f194216208c5b2fe
- 055fe8002de293401852310ae76cb730c570f2037c3c832a52a79b70e2cb7831

Minzen

- 797d9c00022eaa2f86ddc9374f60d7ad92128ca07204b3e2fe791c08da9ce2b1
- 9374040a9e2f47f7037edaac19f21ff1ef6a999ff98c306504f89a37196074a2
- 26727d139b593486237b975e7bdf93a8148c52d5fb48d5fe540a634a16a6ba82

NamelessHdoor

- dfc8a6da93481e9dab767c8b42e2ffbcd08fb813123c91b723a6e6d70196636f

Gh0stRAt Downloader

- ce47e7827da145823a6f2b755975d1d2f5eda045b4c542c9b9d05544f3a9b974
- e34f4a9c598ad3bb243cb39969fb9509427ff9c08e63e8811ad26b72af046f0c

Custom Gh0st

- 8e5a0a5f733f62712b840e7f5051a2bd68508ea207e582a190c8947a06e26f40

Datper

- 7d70d659c421b50604ce3e0a1bf423ab7e54b9df361360933bac3bb852a31849

HomamDownloader

- a624d2cd6dee3b6150df3ca61ee0f992e2d6b08b3107f5b00f8bf8bcfe07ebe7

C2
lywjrea.gmarketshop[.]net
krjregh.sacreeflame[.]com
psfir.sacreeflame[.]com
lywja.healthsvsolu[.]com
phot.healthsvsolu[.]com
blog.softfix.co[.]kr
news.softfix.co[.]kr
www.gokickes[.]com
log.gokickes[.]com
sansei.jpn[.]com

Got something to say?

Leave a comment...

Notify me of followup comments via e-mail

Name (required)

Email (required)

Website

SUBMIT

SUBSCRIBE TO NEWSLETTERS

Email SUBSCRIBE

COMPANY

Company

Careers

Sitemap

Report a Vulnerability

LEGAL NOTICES

Privacy Policy

Terms of Use

ACCOUNT

Manage Subscription

SALES > 888.704.5196 »

SEE A DEMO »

TAKE A TEST DRIVE