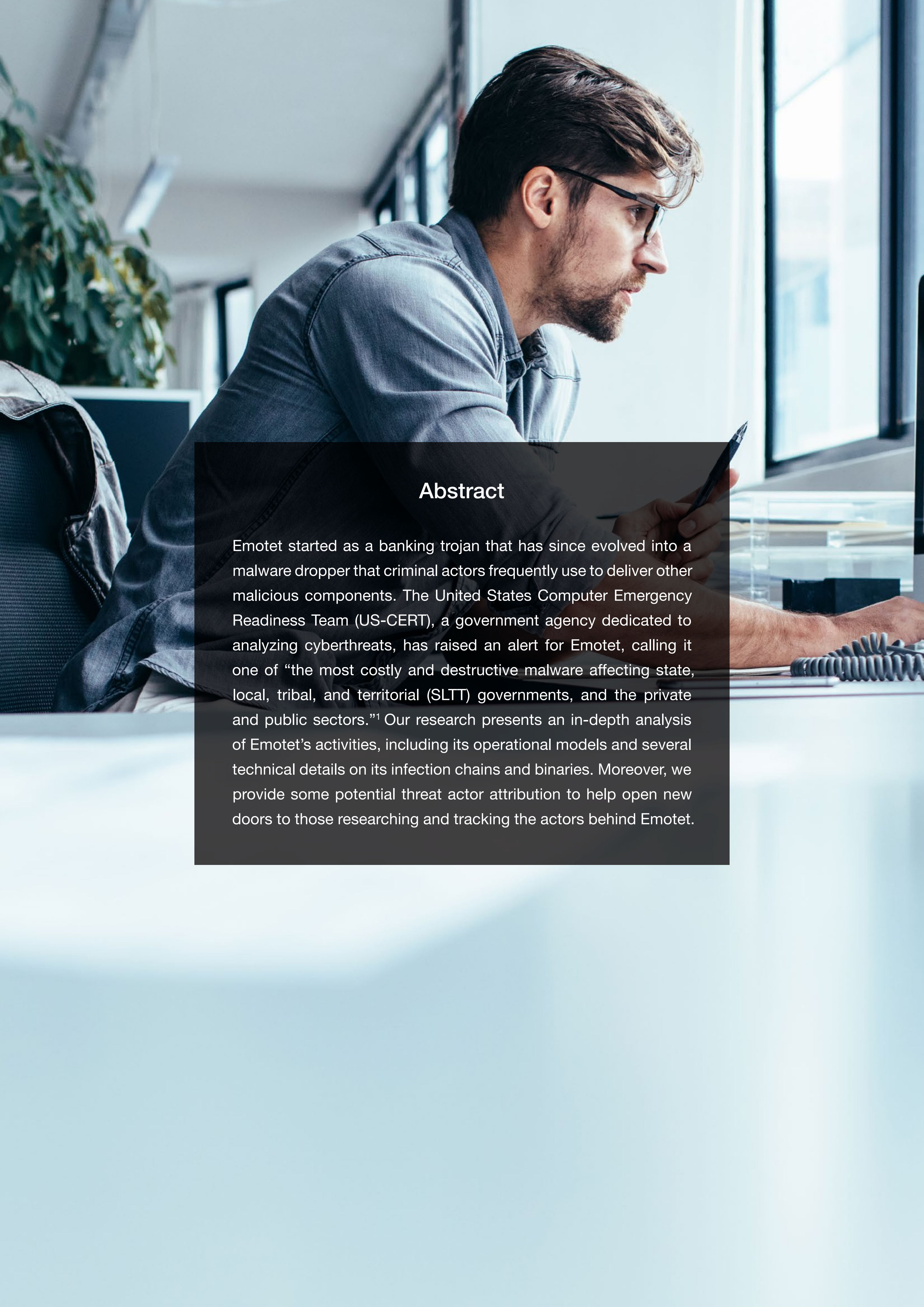
Exploring Emotet's Activities

Published by:

Trend Micro Research

Contents



Abstract

Emotet started as a banking trojan that has since evolved into a malware dropper that criminal actors frequently use to deliver other malicious components. The United States Computer Emergency Readiness Team (US-CERT), a government agency dedicated to analyzing cyberthreats, has raised an alert for Emotet, calling it one of "the most costly and destructive malware affecting state, local, tribal, and territorial (SLTT) governments, and the private and public sectors."[1] Our research presents an in-depth analysis of Emotet's activities, including its operational models and several technical details on its infection chains and binaries. Moreover, we provide some potential threat actor attribution to help open new doors to those researching and tracking the actors behind Emotet.

Introduction

Emotet, a modular trojan that Trend Micro discovered in June 2014,[2] has recently evolved into a global threat distributor by carrying and distributing other several infamous banking trojans. This development has massively affected victims.[3, 4] According to the US-CERT, Emotet costs up to $1 million per incident to remediate.[5] To fight against the infamous cybercrime group, we provide a comprehensive research on Emotet's artifacts collected from June 1, 2018, to September 15, 2018, comprising 8,528 unique URLs, 5,849 document droppers, and 573 executable samples.

The data we used includes data collected by Trend Micro and public malware research repositories without any hack-back activities. We applied some heuristic signatures on email gateway solutions to source possible Emotet artifacts and then utilized a self-built system to track the infection chains and analyze Emotet's executables. We also employed YARA signatures on public malware research repositories to source potential objects. We cannot guarantee the extensiveness of the coverage of Emotet's artifacts, but we have done our best to ensure that the research is sound.

This research presents an in-depth analysis of Emotet's infection chains, binaries, and configurations in executable files. We provide detailed documentation on the obfuscation techniques used on Emotet's executables as well as an investigation of all of the malware's related artifacts. The data gathered from our analysis revealed the infrastructures behind Emotet as well as possible attribution. Here is a summary of our findings:

1. Emotet's business is tied to Russian-speaking actors, according to open-source intelligence records.
2. Its operators use at least two independent infrastructures running parallel to one another to support the Emotet botnet. By grouping the C&C servers and the RSA keys, we were able to get two distinct infrastructure groups. We also saw that those behind the Emotet malware switched its RSA keys on a monthly basis. The next-stage payloads pushed by the two groups did not show any major differences in terms of purpose or targets, which means the differing infrastructure of the two groups may have been designed to make it more difficult to track Emotet and minimize the possibility of failure.
3. Multilayer operating mechanisms might have been adopted in the creation of Emotet's artifacts. The inconsistency between the activity patterns shows that the infrastructure used to create and spread document droppers is different from the infrastructure used to pack and deploy Emotet

executables. The creation of document droppers stops during the non-working hours between 1:00 to 6:00 (UTC). Meanwhile, there might be three sets of machines that are used to pack and deploy Emotet's executable payloads, two of which are probably set to the UTC +0 and UTC +7 time zones, respectively.

4. The author of the Emotet malware likely resides somewhere in the UTC +10 time zone or further east. After we grouped the executable samples by their unpacked payloads' compilation timestamps, we found two sample groups that showed an inconsistency between the compilation timestamps and the corresponding first-seen records in the wild. This leads to the possibility that the compilation timestamps point to the local time on the malware author's machine. If the local time is accurate, it's possible that the malware author is located somewhere in the UTC +10 time zone or further east.

Infection Chain

Emotet is known for using social engineering tricks to launch its attacks. The infection chain starts with a spam email — one that could have been sent from legitimate email addresses that have been compromised, since Emotet's spamming module allows it to log into remote email service providers and spam emails from infected accounts. After a user clicks on the malicious URL or downloads the attachment, a dropper, usually a document file with a malicious macro, will be downloaded. Once the dropper successfully runs, it downloads an Emotet executable. In a recent Emotet campaign we've observed, the malware's payload is a loader that will install next-stage payloads or execute commands received from C&C servers.

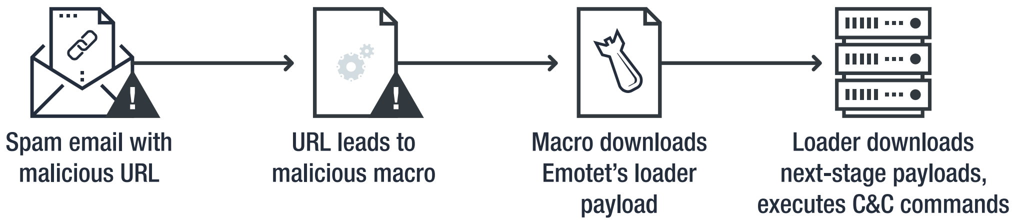


Figure 1. An overview of Emotet's infection chain

Social Engineering

The majority of Emotet's spam emails adopt business-related topics to lure victims into clicking the URLs or downloading attachments. These spam emails have been reported to be written in English, German, and French. For example, aside from a fake invoice, we also found an Emotet spam campaign disguised as an Independence Day e-Card sent on July 4.

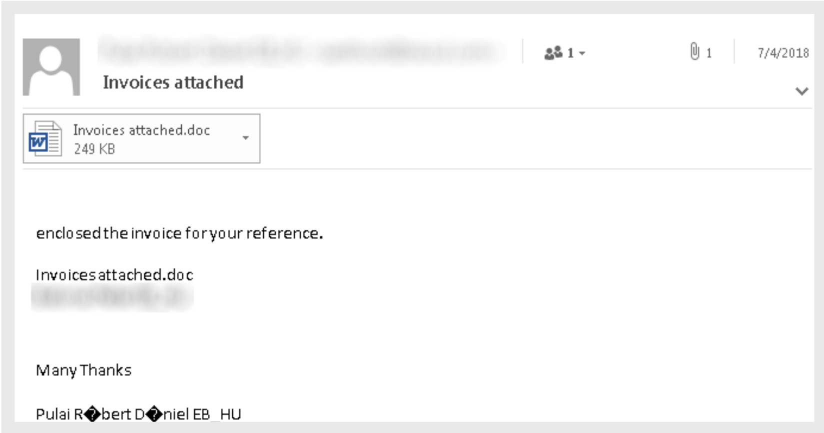


Figure 2. Emotet spam sample under the guise of a fake invoice in MS word format

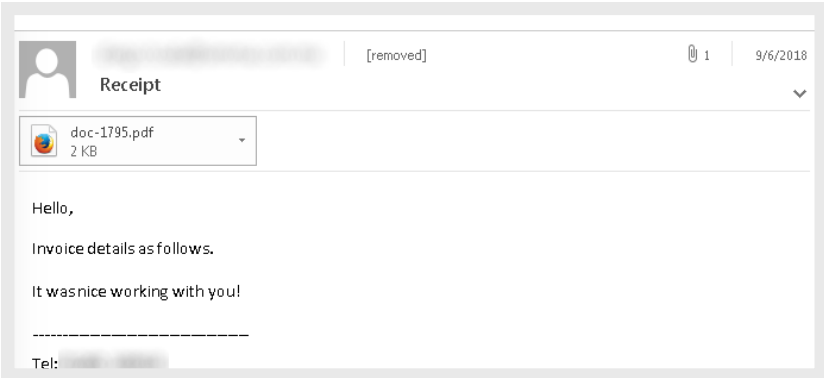


Figure 3. Emotet spam sample disguised as a fake invoice in PDF format

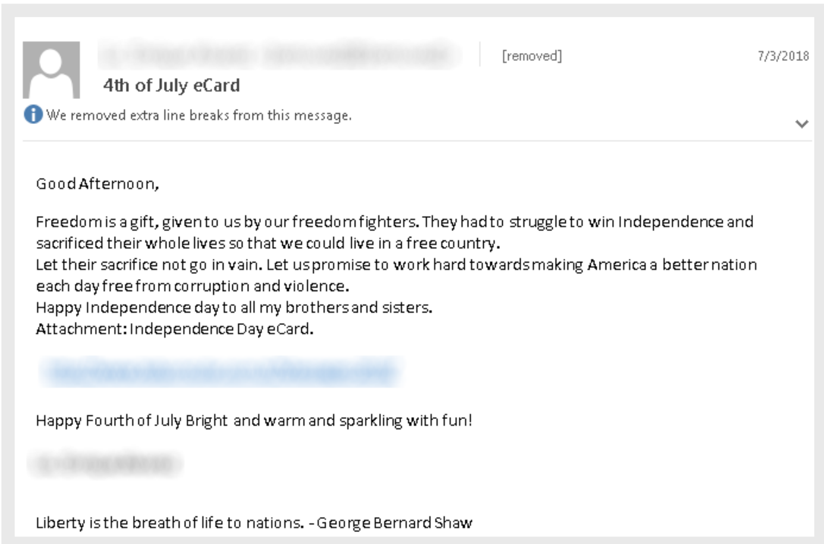


Figure 4. Emotet spam sample with an embedded malicious URL

Malicious URL Analysis

We collected 8,528 unique URLs that led to our collection of Emotet's document droppers — 922 of which lead to Emotet's executables. A manual check revealed that most of the URLs we found seemed to be compromised sites. During our analysis, we discovered that one of the URLs was visited 4,627 times. We recorded roughly 400 of Emotet's URLs on the same day, and were able to estimate the new visit number of Emotet's URLs — 1,680,000 visitors — if we assume all of the URLs had the same click volume. Note that the actual number should be lower since a part of the URL visits might be from security vendors or their products, various automated internet crawlers, and security or threat researchers.

Document Dropper Analysis

We gathered 2,258 document droppers from which we extracted executable files. After studying these droppers, we realized that the code page identifiers and the metadata versions could provide us with some useful information about the malware. The code page identifier was used to interpret the way in which the text were encoded,[6] and the metadata version indicated the current version Microsoft Word automatically saved. We extracted the code page identifiers and the versions from the samples and found that these can be grouped into two. The first group has version = 917504 and code page = 1251 (Windows Cyrillic-Slavic, mostly used by Russians, Bulgarians, Serbians and Macedonians), while the second group has version = 1048576 and code page = 1252 (Windows Latin 1 ANSI, mostly used in Western Europe including France, but also for any Windows system using the English language). We performed a retro-hunt on these signatures and found 3,591 extra samples, expanding our collection to a total of 5,849 document droppers.

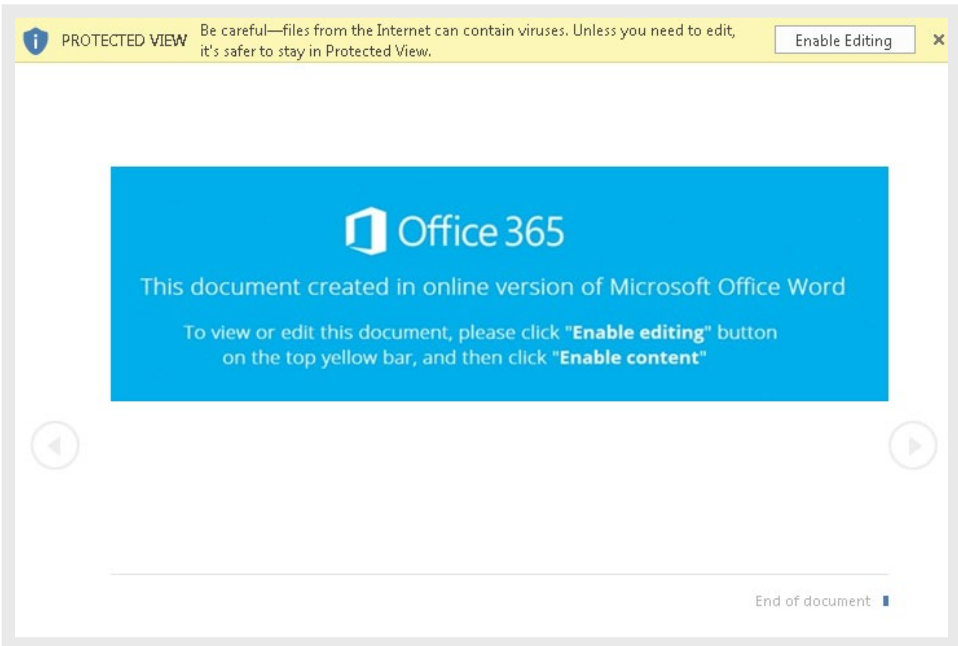


Figure 5. An example of Emotet's malicious document dropper

The macro in the document is obfuscated, and uses AutoOpen() to trigger the payload. The goal of the embedded Visual Basic code is to get Emotet's executable hosted on the remote server by running Windows PowerShell.[7, 8]

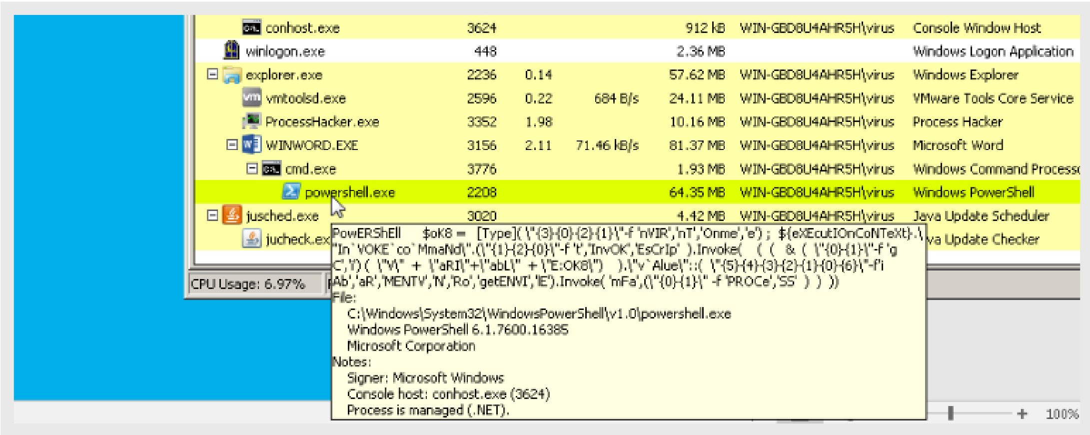


Figure 6. An example of an obfuscated PowerShell command run by an Emotet document dropper in a sandbox environment[9]

Each document dropper is embedded with about five URLs that host Emotet executable payloads. The format of the URLs hosting the executables is different from the format of the URLs that drop documents. The following are some examples:

URLs used to host Emotet executables:

– hxxp://websitedesigngarden[.]com/e6vTCit
– hxxp://emicontrol[.]com/85a
– hxxp://grupoembatec[.]com/zHVN
– hxxp://stevebrown[.]nl/3YA1kb/

URLs used to host Emotet documents:

– hxxp://167[.]99[.]81[.]74/433650Z/PAYROLL/Smallbusiness
– hxxp://3music[.]net/DOC/US_us/New-order
– hxxp://acb-blog[.]com/7gwg7ySK/de_DE/Firmenkunden/
– hxxp://aesbusiness[.]ru/02385XSGEBFE/PAY/Commercial

Additionally, the obfuscation method used on Emotet's macro attachments was found to be the same as the one used in some Ursnif campaigns. However, the format of the URLs that hosted the executables was different. For instance:

URLs used to host Emotet executables:

- hxxp://websitedesigngarden[.]com/e6vTCit
- hxxp://emicontrol[.]com/85a
- hxxp://grupoembatec[.]com/zHVN
- hxxp://stevebrown[.]nl/3YA1kb/

URLs used to host Ursnif executables:

- hxxp://d792jssk19usnskdxnsw[.]com/MXE/lodpos[.]php?l=yows2[.]xt2

We have only seen this obfuscation method being used by the cybercriminal groups that push these two malware families. We found that one of the groups that pushed Ursnif used some tools written in the Russian language.

Binary Analysis

The following sections provide a detailed technical analysis of the Emotet samples we collected, including the obfuscation methods used, the packing techniques employed, as well as its behaviors.

Obfuscation

Emotet's executables adopt a unique anti-analysis method. The executables' obfuscation method moves the first few instructions from a specific location inside the payload to another specific memory region (which we named "jump table") allocated by the loader. The missing instructions are frequently found at the beginning of each function. The jump table is located outside of the module; therefore, simply dumping the module will lead to a broken executable. The following figures illustrate the obfuscation technique.

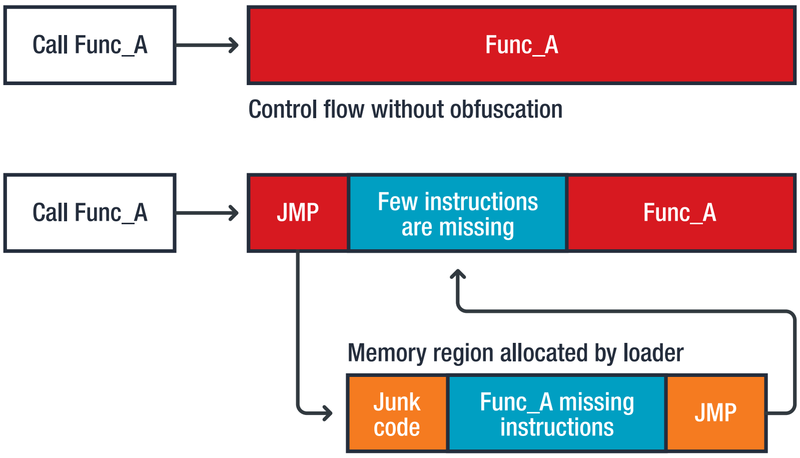


Figure 7. An overview of Emotet's obfuscation technique

The following is a concrete example: The binary calls a function located at 0x1DA1030. The function prologue and a few instructions of that function disappear, and the first instruction is replaced by a JMP in the memory region located at 0x52001D. The memory region is not in the same module as Emotet's payload itself. After all of the missing instructions in the jump table are executed, the binary jumps back to its original function.

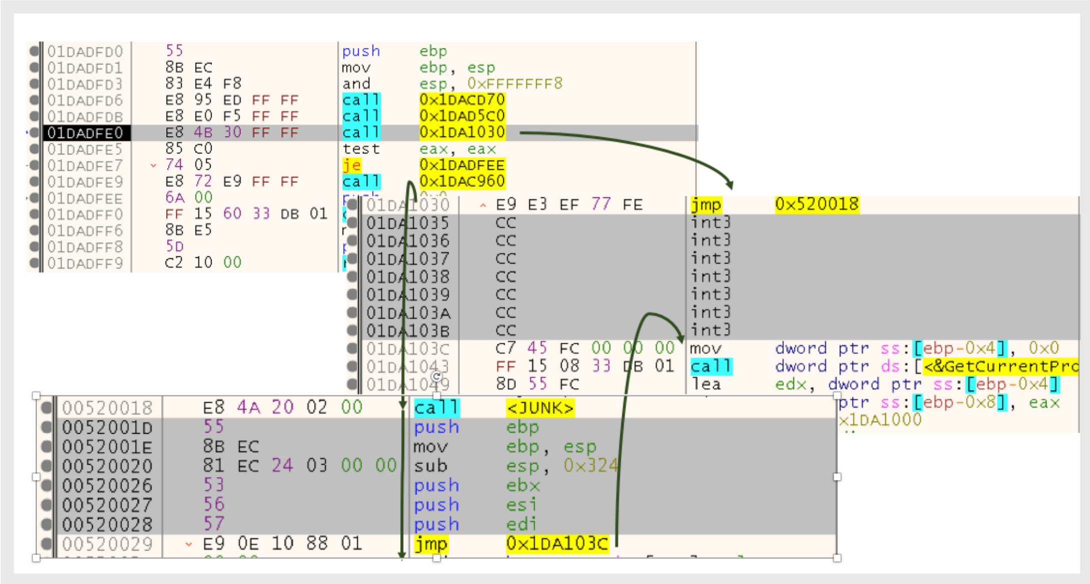


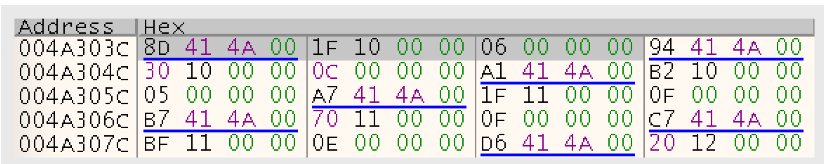
Figure 8. Emotet's obfuscated control flow explanation

Packing Technique

Emotet's samples were packed using customized packers. These packers first decrypt the loaders, and the loaders then decrypt and load Emotet's main payloads. The obfuscation jump tables will be installed before the main payloads run.

The obfuscation installation process is in the loader's procedures. We document the structure used by the loaders to install the obfuscation jump tables as:

```
{
PBYTE instructions;     pointer to the missing instructions
DWORD offset;           offset (RVA) in payload where should be patched
DWORD size;             size of missing instructions
}
```



Address	Hex			
004A303C	8D 41 4A 00	1F 10 00 00	06 00 00 00	94 41 4A 00
004A304C	30 10 00 00	0C 00 00 00	A1 41 4A 00	B2 10 00 00
004A305C	05 00 00 00	A7 41 4A 00	1F 11 00 00	0F 00 00 00
004A306C	B7 41 4A 00	70 11 00 00	0F 00 00 00	C7 41 4A 00
004A307C	BF 11 00 00	0E 00 00 00	D6 41 4A 00	20 12 00 00

Figure 9. A table of jump table entries

The current ImageBase of the payload is at 0x4E0000, while the ImageBase of the loader is at 0x4A0000. As the above figure shows, the first entry on the table is instruction = 0x4A418D, offset = 0x101F, size = 0x6.

For the first entry, the 6-byte instructions at 0x4A418D is:

```
$ ==>     004A418D    B8 01 00 00 00    mov    eax, 0x1
$+5       004A4192    5D                pop    ebp
```

Figure 10. The first entry's instruction

ImageBase + offset = 0x4E101F. The memory content is:

```
$ ==>     004E101F    CC    int3
$+1       004E1020    CC    int3
$+2       004E1021    CC    int3
$+3       004E1022    CC    int3
$+4       004E1023    CC    int3
$+5       004E1024    CC    int3
```

Figure 11. The memory content before the obfuscation technique is installed

After the loader finishes installing the obfuscation technique, the loader writes a jump to the jump table.

```
$ ==>     004E101F    E9 E0 EF FD FF    jmp    0x4C0004
$+5       004E1024    CC                int3
```

Figure 12. The memory content after the obfuscation technique is installed, which leads to a jump on the jump table

On the jump table (0x4C004), the missing instructions are executed and jump back to execute its original function at 0x4E1025.

```
004C0004    E8 5E 20 FE FF    call    <junk>
004C0009    B8 01 00 00 00    mov     eax, 0x1
004C000E    5D                pop     ebp
004C000F    E9 11 10 02 00    jmp     0x4E1025
```

Figure 13. Jumps back to execute missing instructions

Configuration

The Emotet samples embed C&C servers in the IP and Port formats. The RSA key for encrypting network traffic can also be found in the configuration part of each Emotet executable.

The structure for the C&C server is as follows:

```
#pragma pack(4)
{
DWORD IP;
USHORT PORT;
}
```

For instance, the first C&C server in the configuration is 173.11.129.38:80.

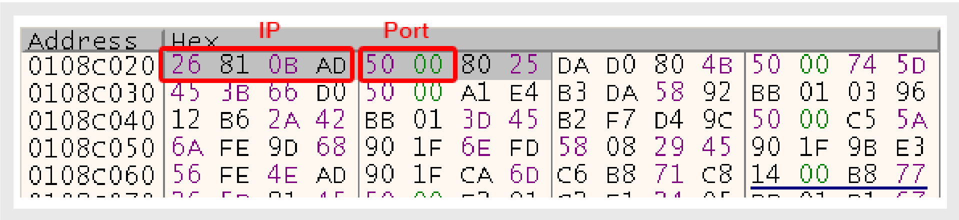


Figure 14. C&Cs in the configuration

The encoded RSA public key blob is also embedded in the configuration. It will be later decoded by CryptDecodeObjectEx() to a PublicKeyBlob object in Figure 15.[10]

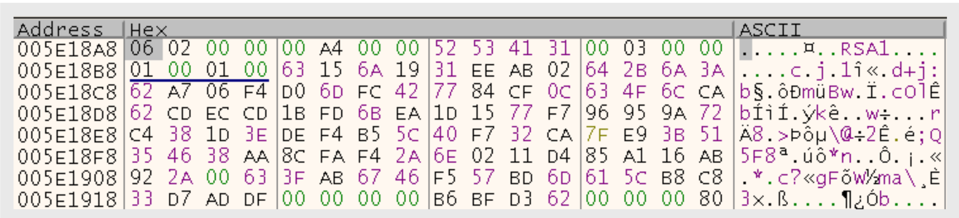


Figure 15. The decoded PublicKeyBlob object

Network Communication Protocol

The binaries of Emotet collect information related to operating systems, processes, and sometimes mail client information, which it sends back to its C&C servers. The protocol is based on Google ProtoBuf,[11] and all of the messages are encrypted by AES. If the packet does not exceed a certain length, it places information in a cookie and sends the GET requests. Otherwise, it uses POST to deliver data.

Paweł Srokosz has done amazing research on Emotet's network protocol, which is worth reading.[12]

Modules

Emotet has been found carrying the following legitimate tools or modules that are being abused by Emotet actors:[13]

– Mail PassView[14]

– WebBrowserPassView[15]

– Spam Module

Emotet is capable of sending out spam emails via its spam module. The module can abuse the compromised email accounts powered by Emotet's C&C servers, which makes blocking Emotet's spam emails harder.

Next-Stage Payload

Recent campaigns used Emotet as a loader, carrying other malware as next-stage payloads. Qakbot, IcedID, Zeus Panda, TrickBot and Dridex were all recorded to have been dropped by Emotet.

Emotet's Ties to Russian-Speaking Actors

Feodo (AKA Bugat or Cridex) is a banking trojan that emerged in 2010. At the time, Atif Mushtaq (then a FireEye researcher) wrote that the malware was not a toolkit and that a single cybercriminal group was pushing it.[16] Two succeeding branches soon appeared, the first of which is the infamous banking trojan Dridex that was found around September 2011. Nikita Slepogin, a Kaspersky researcher, has documented the evolution of this malware in detail.[17] The other branch is Geodo (AKA Emotet). While Geodo/Emotet inherits the design of Feodo's network infrastructure, its code is completely different from that of the original Feodo.[18] Geodo/Emotet was also found using the stolen SMTP credentials from compromised computers in the Feodo botnet.[19] The successor of Geodo, which some researchers named Heodo or Emotet v4, followed years after, appearing in March 2017. In summary, Emotet is the successor of Feodo, which can be traced back to 2011.

It has recently delivered several payloads, including the Zeus Panda banker, TrickBot, IcedID, and AZORult. For this section, we referred to @malware_traffic's data on Malware Traffic Analysis and analyzed the configuration of TrickBot and Zeus Panda banker dropped by Emotet.[20]

TrickBot, the successor of Dyreza (or Dyre),[21] is a modularized banking trojan that uses a group tag (gtag) to keep track of its affiliates. So far, several gtags "arz1," "del77," "jim316," "lib316," "tot285," and "del34" have already been recorded. According to @malware_traffic, Emotet loaded TrickBot on the following occasions: June 14 and 15, July 6, 16, and 31, Aug. 8, Sep. 4, 14, 20, 21, and 26. He noticed that he got different TrickBot samples from the same Emotet malware whenever UK or US IPs were used to connect to the C&C servers.

On September 21, @malware_traffic got TrickBot samples with gtags "arz1," "del77," "jim317," and "lib316."[22] We successfully got the web injection configurations of these samples from the C&C servers. The targets were banks, cryptocurrency exchanges and financial institutions located in the US, Germany, UK, Canada, Austria, Spain, Ireland, Netherlands, Japan, Greek, South Africa, and Bulgaria. The target lists and the fake sites used for redirection are shared among different gtags, which might indicate that TrickBot serves as a cybercrime service that looks for business affiliations.

Zeus Panda is a banking trojan and a variant of Zeus. The malware will not infect victims who have Russian (0x419), Ukraine (0x422), Belarus (0x423) and Kazakhstan (0x43f) keyboard codes installed.

There are 17 samples of Zeus Panda recorded to have been dropped by Emotet during our research period. The RSA public key for traffic encryption is embedded in the configuration. We discovered that there is only one RSA public key among these samples, and it indicates that only one actor might have used those Zeus Panda samples during that period. Unfortunately, we failed to get the web injection from the C&C servers, so we cannot provide who the actor was targeting. However, we were able to find some open source intelligence recording that the majority of the targets were located in the U.S., Canada, and Germany.

IcedID is a banking trojan that IBM discovered in September 2017 and has collaborated with TrickBot, according to FlashPoint researchers.[23, 24] Emotet has been recorded dropping IcedID since June 2017. Its targets at that time were mostly located in the U.S.[25]

These malware families are associated with Russian-speaking actors. TrickBot is the successor of Dyre, which was frequently loaded by Upatre from a few years ago. Upatre also dropped GameOver Zeus (P2P Zeus) in the past. GameOver Zeus has been attributed to an experienced criminal organization led by Russian actors.[26] Besides, Trickbot has been used by a cybercriminal group associated with Dridex, Necurs, Locky, and CryptoLocker.[27] AZORult is a loader and an information stealer that was sold on several Russian-speaking underground forums by an actor who used the pseudonym “CrydBrox.” With this intelligence, it is safe to assume that the actor behind Emotet has some business relationships with Russian-speaking actors.

Shares the same weapon(s) with other infamous Russian-speaking gangs

Aside from Emotet’s apparent Russian-speaking business affiliations, technical evidence also shows that it might have the same weapon providers as other notorious cybercriminal gangs, including the actors behind Ursnif, Dridex and BitPaymer. A blog entry we recently published detailed how the executable loaders that these four groups used share identical payload decryption procedures and internal data structures.[28] This indicates that these four gangs use the same tool, which could support the inference that they share the same weapon providers or have some relationships with one another that allow them to exchange resources. While Ursnif’s ancestor, GOZI-ISFB, was created by a Russian, Dridex’s was once administrated by an arrested Russian-speaking background actor.[29] Meanwhile, BitPaymer shares similar code with Dridex, which could also point to attribution links.[30] This reveals that these Russian-speaking cybergangs may have fostered working relationships with one another.

Two Infrastructures Running in Parallel

We've collected and analyzed 573 Emotet executable samples. The configuration inside an executable includes a list of C&C servers and an RSA key for encrypting the connection.

RSA Keys

We extracted six unique RSA public keys from the Emotet executables. Each RSA key has a 768-bit modulus and exponent 65537. We calculated the CRC32 of each RSA key blob and gave each key a name to represent the specific keys in the article.

Key Name	CRC32	Infra
A	fcb2fb3b	1
B	86e9acef	1
C	ceff5362	1
D	fc8e8aaa	2
E	8f1eb5e	2
F	aef0def8	2

Table 1. The RSA keys extracted from Emotet executables

C&C Servers

Each Emotet sample contains multiple hardcoded C&C servers. We extracted 721 unique C&C servers in total. On average, one Emotet sample contains 39 C&C servers, with a maximum of 44 and a minimum of 14. Based on our observation, only a few C&C servers embedded in a single Emotet sample are actually active.

Most of the C&C servers are located in the U.S., Mexico, and Canada. The Top 3 ASN connected to Emotet's C&C servers are ASN7922, ASN8151, and ASN22773.

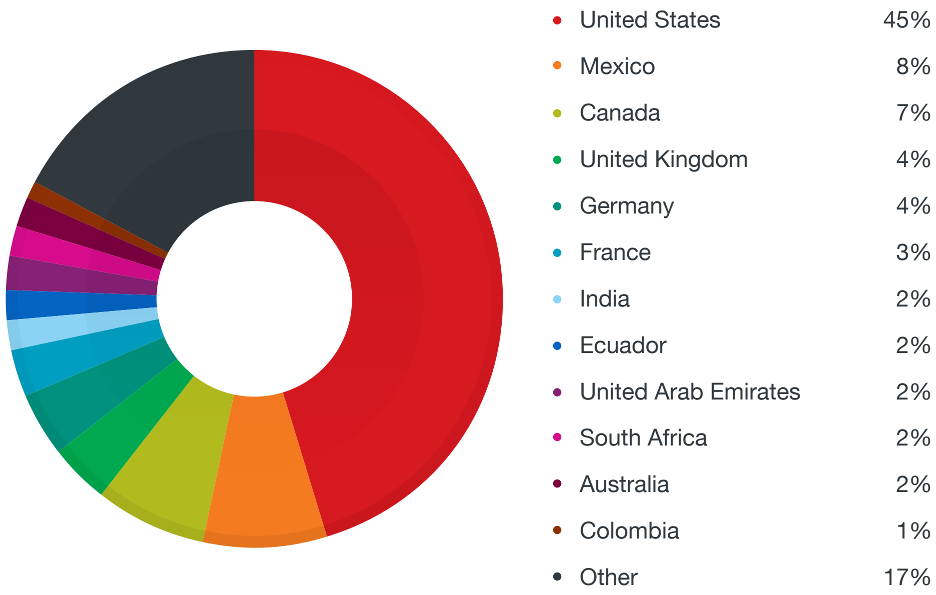


Figure 16. Countries where Emotet C&C servers are distributed

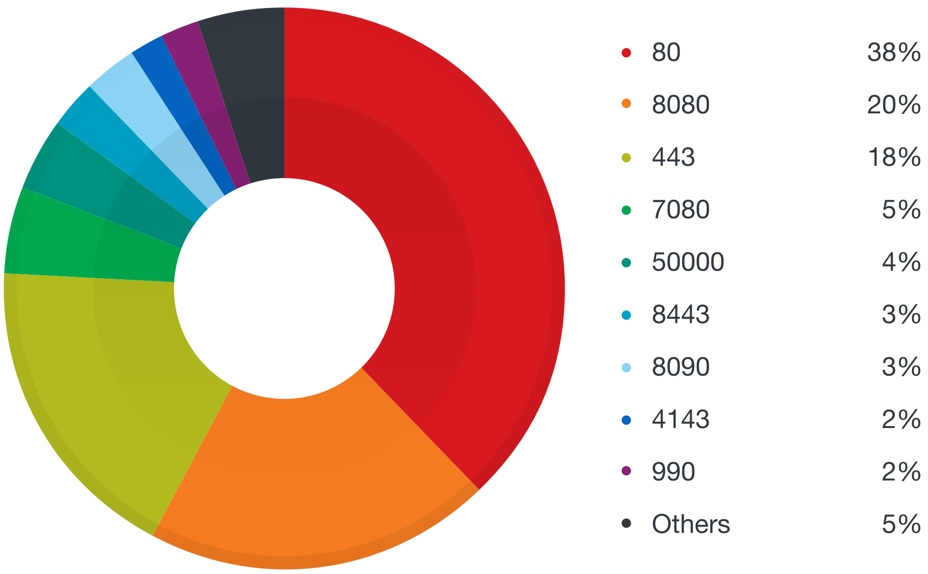


Figure 17. Distribution of Emotet C&C servers' ports

We visualized the relationship between each RSA key and its set of C&C servers and discovered two RSA groups. Keys A, B, and C were in one group (Group 1), and keys D, E, and F were in another (Group 2).

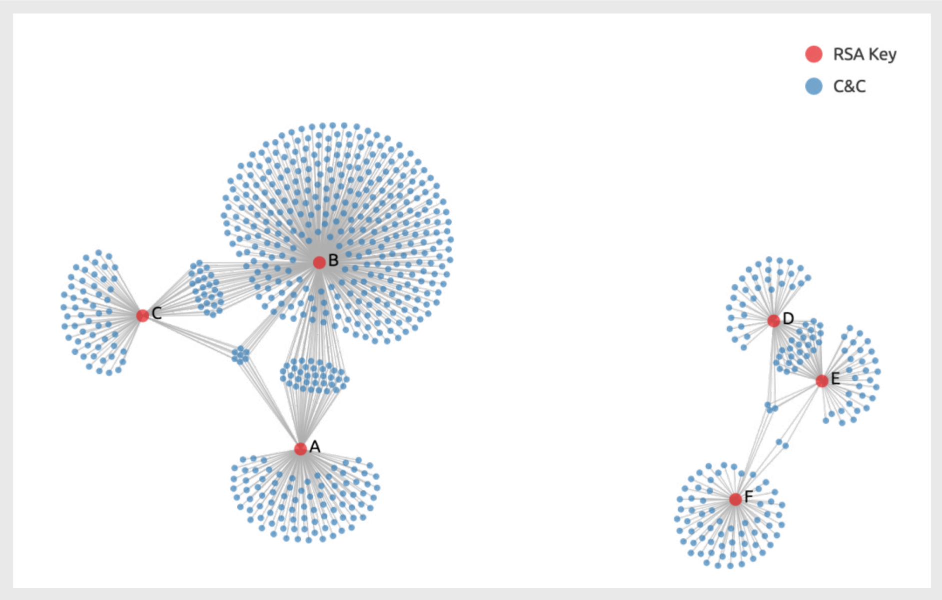


Figure 18. The relationship between RSA keys and C&C servers. Each blue dot represents a unique C&C server, while the red ones indicate RSA public keys

Figure 18 shows that the two groups did not share C&C servers. Our analysis did show a link between the dates the RSA keys were received and the two groups' activities: each RSA key was observed to have been used for one month before threat actors switched to another key on the first working day of the new month (Jul. 2, 2018 and Sep. 3, 2018, both fall on a Monday).

	June	July	August	September
Keys used by Group 1	fcb2fb3b (A)	86e9acef (B)		ceff5362 (C)
Keys used by Group 2	fc8e8aaa (D)	8f1eb5e (E)	–	aef0def8 (F)

Table 2. Two groups of RSA keys and their corresponding active months

Figure 19. The time the RSA keys were received. Each dot represents the time when the RSA key was found in the configuration of a new sample. The group 1 keys are in green, and orange is for group 2

We also observed that Group 1 had more artifacts than Group 2. Based on our data, we received 469 unpacked Emotet samples for Group 1 and 102 for Group 2. We also did not find any activity for Group 2 in August, as shown in Figure 4.

Two Different Emotet Groups, Two Different Agendas?

Our initial assumption was that the two Emotet groups were either created for different purposes or utilized by different operators. To prove this assumption, we referred to @malware_traffic and categorized the IoCs respectively. However, we did not find any major difference between the IoCs under these two groups. For instance, TrickBot with gtag arz1 was found to have been sent by Group 1 on September 20 and by Group 2 the next day. Without any strong evidence, we can only tell that the two might be different infrastructures designed to make tracking Emotet more difficult and help minimize the possibility of failure.

Date	Emotet Group	RSA Key	Next-stage Payload
2018-07-03	2	E	Panda Banker
2018-07-09	1	B	Panda Banker
2018-07-16	2	E	Panda Banker
2018-07-19	2	E	Panda Banker
2018-07-30	1	B	Panda Banker
2018-07-31	1	B	Panda Banker
2018-08-08	1	B	TrickBot

Date	Emotet Group	RSA Key	Next-stage Payload
2018-08-10	1	B	Panda Banker
2018-08-13	1	B	Panda Banker
2018-08-14	1	B	Panda Banker
2018-08-15	1	B	Panda Banker
2018-08-16	1	B	Panda Banker
2018-08-22	1	B	Panda Banker
2018-08-24	1	B	Panda Banker
2018-08-26	1	B	Panda Banker
2018-09-04	2	F	IcedID, TrickBot
2018-09-05	2	F	IcedID, AZORult
2018-09-06	1	C	IcedID, AZORult
2018-09-14	1	C	TrickBot gtag: del72
2018-09-20	1	C	TrickBot gtag: arz1
2018-09-21	2	F	TrickBot gtag: arz1, del77, jim316, lib316
2018-09-25	2	F	TrickBot gtat: arz1, IcedID, AZORult

Table 3. The next-stage payload delivered by Emotet's two infrastructures between July and September 2018

Compiling Emotet's Source Code for Each Infrastructure

Customized packers and obfuscators protect Emotet payloads. We studied the compilation timestamps against each sample before and after packing and saw that some of the timestamps in packed samples were forged, while some seemed legitimate. The samples with legitimate timestamps showed a difference of a few minutes between the time they were compiled and the time they were found in the wild. For example, sample SHA256: 648dce03ac4c32217ce5c0b279bc3775faf030cafb313c74009fe60ffde3c924 (Detected by Trend Micro as TSPY_EMOTET.NSFOGAH) was compiled at 2018-06-06 05:40:17 and was found in the wild four minutes later. However, sample SHA256: 07deb1b8a86d2a4c7a3015899383dcc4c15dfadcfafc3f2b8d1e3aa89a6c7ac4 (Detected by Trend Micro as TSPY_EMOTET.TTIBBJD) was compiled on 2035-07-30 21:36:11, which is obviously a fake timestamp. Since it is difficult to distinguish legitimate timestamps from forged ones, research on the packed files' timestamps may prove to be fruitless.

Even though the compilation timestamp might be bogus, we decided to analyze the unpacked Emotet samples and saw that their timestamps seemed legitimate. Out of 571 unpacked Emotet samples, only 11 distinct compilation timestamps were found. If the timestamp is forged during every compilation, the samples compiled with the same pieces of code should contain identical code sections but with different compilation timestamps. However, we found that the unpacked samples with the same timestamp shared the identical code section, while differences can be found among those with different timestamps. The changes between the different timestamps also seem to be new-version updates.

The data in Table 4 show that the actor might have used automatic tools or scripts to compile Emotet's source code for each infrastructure, since a number of unique samples shared the same compilation timestamp. The data also shows that the actors prepared the payload for Groups 1 and 2 sequentially. For example, on June 3, 2018, 46 Emotet samples were generated at 20:08 (UTC) using Group 1's RSA public key and C&C servers. Two minutes later, the 37 other Emotet samples were generated.

We noticed that the attackers tended to update Emotet samples on Monday or Wednesday (UTC). We also observed that the code section is the same among the samples with the same compilation timestamp. The only difference is in the C&C servers embedded in the data section. It is possible that each time a source code is compiled, several C&C servers on the attacker's control list were chosen to generate a new sample.

Emotet Group	RSA Key	Compilation Timestamp in Epoch	Payloads' Compilation Timestamp in UTC	Unique Sample Count
1	A	1528056487	2018-06-03 20:08:07	46
2	D	1528056680	2018-06-03 20:11:20	37
1	B	1530547690	2018-07-02 16:08:10	28
2	E	1530547815	2018-07-02 16:10:15	24
1	B	1531161666	2018-07-09 18:41:06	31
2	E	1531161732	2018-07-09 18:42:12	17
2	E	1531899206	2018-07-18 07:33:26	55
2	E	1531906587	2018-07-18 09:36:27	4
1	B	1532502303	2018-07-25 07:05:03	270
1	C	1536011873	2018-09-03 21:57:53	18
2	F	1536011945	2018-09-03 21:59:05	11

Table 4. Unique samples collected in the wild with our corresponding compilation timestamps

Emotet's Artifacts: Multi-group and Multilayer Operation

To gain an in-depth understanding of Emotet's operations, we studied two attributes: the creation time in the document droppers' metadata and the compilation time of the packed Emotet executable samples. We discovered that Emotet might have adopted different operating mechanisms for a) creating and spreading document droppers and b) packing and deploying PE payloads, as both mechanisms demonstrated their own notable characteristics. The former reveals that it stops creating and spreading document droppers during the non-working hours between 1:00 to 6:00 (UTC) roughly on a weekly basis. Meanwhile, the latter shows that at least three sets of machines are used to pack and deploy Emotet's executable payloads — two of which are possibly set to the time zones of UTC +0 and UTC +7, respectively.

Document Droppers: Weekly-Based Activity Patterns with Non-Working Hours from 1:00 to 6:00 (UTC)

We started our observation at the document droppers' creation time. Emotet's operators frequently use documents — of which there is a significant volume and easily collected — as executable droppers. We first found several documents of the campaigns that share the same timestamp, which indicates that those were generated automatically in batches using a tool. Next, we plotted the activity time pattern by the day and hour the tools were used to generate the documents (i.e., the unique timestamps, not on the actual volume of samples), shown in Figure 20 and Figure 21. Figure 20 reveals a weekly pattern of one or two days of inactivity, while Figure 21 shows inactivity between the non-working hours of 1:00 to 6:00 (UTC). Based on our data, we also observed that the actors behind Emotet used the tools more frequently, from a few times a day to more than twenty times a day, in September.

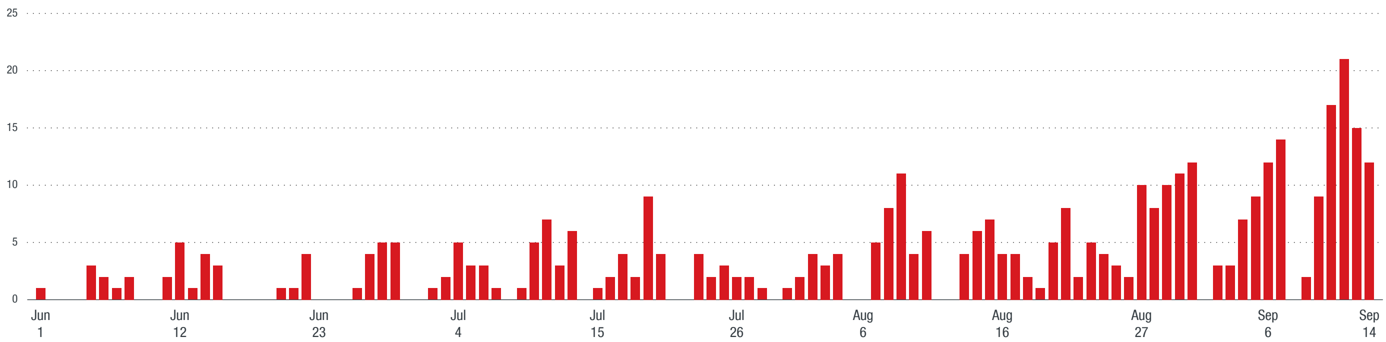


Figure 20. Daily activity pattern of Emotet's document droppers

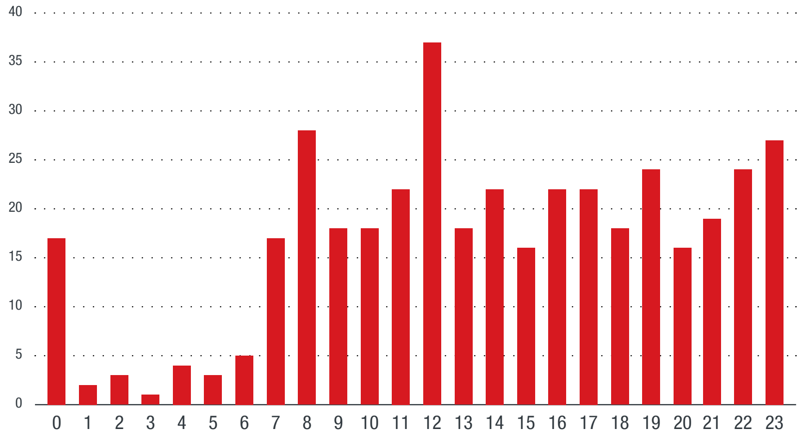


Figure 21. Hourly activity pattern of Emotet's document droppers

Executable Samples: At Least Three Sets of Operating Machines Working

Other artifacts that interest us are Emotet's executable samples, which are packed by a homebrew packer. Contemporary malware packers usually wipe or forge the compilation timestamps in the packed samples, which is the same case in some of the Emotet's samples. However, not all of these timestamps are bogus: there are still several possibly legitimate timestamps that can be found in these artifacts and should not be ignored. For example, the compilation timestamps of samples with SHA-256 30049dadda36afb0667765155aa8b3e9066511f47e017561bee7e456d4c0236d and 2f93c8c97f99c77880027b149d257268f45bce1255aeaefdc4f21f5bd744573f indicate that they appeared in the wild just a few minutes after they were compiled.

We used the following expression to calculate the time gap between a sample's first record of having appeared in the wild and its compilation timestamp:

```
delta = Math.Floor(record of first appearance - compilation timestamp)
```

Since plotting delta between -24 to +24 hours seems legitimate, we selected 371 samples from 571 samples (65%) with potentially legitimate timestamps. We further plotted *delta* by shorter intervals (by the minute) and surprisingly got two groups as shown in Figure 22. If the timestamps were randomly forged, we should have seen a uniform delta distribution.

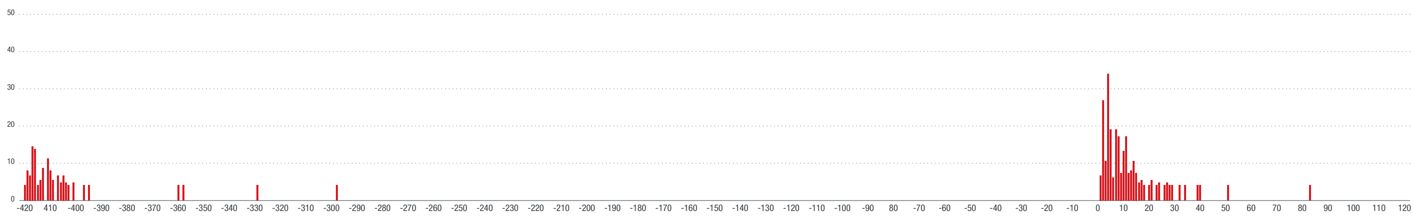


Figure 22. Delta distribution (packed compilation time)

101 samples were packed seven hours before they were found in the wild, while another 267 samples had a delta below 60 minutes. This indicates that the machines used to pack the first sample group might have been set to the UTC +7 time zone, while the machines used to pack the second group might have been set to the UTC +0 time zone. The machines used to pack samples in the third group smashed the compilation timestamps to fake ones, leaving us with insufficient information on them.

We observed that the two sets of machines (set to UTC +0 and UTC +7 time zones) seem to be used consecutively. Figure 23 shows the order in which the executable samples of these two sets of machines were first found in the wild. It demonstrates that the samples belonging to the two sets took turns appearing in the wild, lasting around 1 to 5 days each time. This might indicate that the two sets of operating machines were used in succession to produce and deploy packed executable samples.[32, 33]

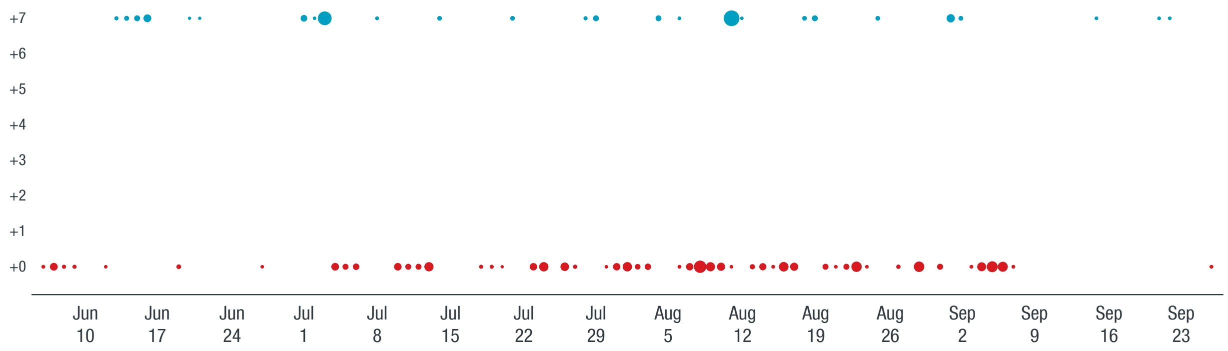


Figure 23. The first-seen dates of the executable samples found to have been set to the UTC+0 and UTC+7 time zones. The size of each circle represents the number of samples found that day

Multilayer Operating Mechanisms for Creating Document Droppers and Packed Executable Samples

After portraying the activity patterns of the document droppers and packed executable samples respectively (in Figure 24), we discovered the obvious inconsistencies between their rest days. There were some days that new documents emerged but no executables appeared, or vice versa. This could mean that there are multilayer operating mechanisms for creating document droppers and producing packed executable samples. Some researches have already illustrated the compartmentalized economy in the current Eastern Europe cybercriminal underground society.

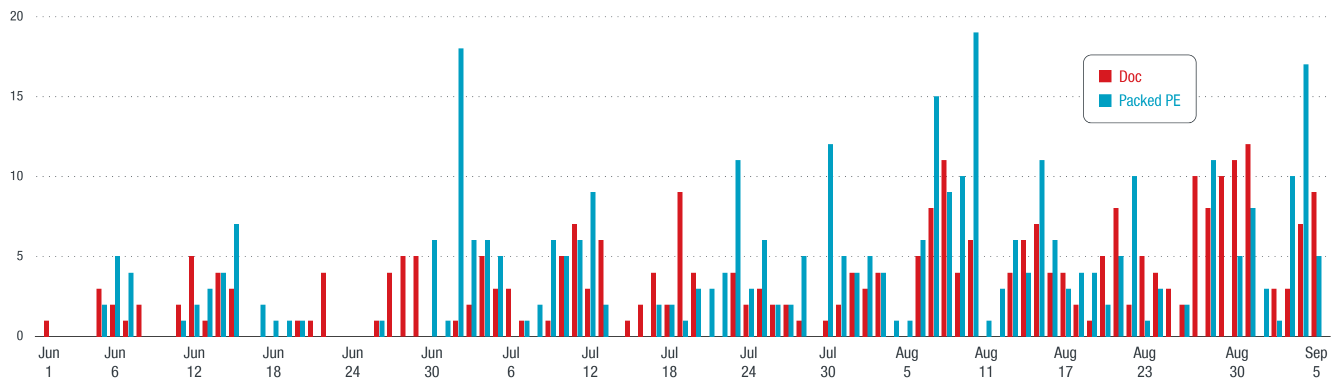


Figure 24. The activity patterns of the document droppers (red) and the packed executable samples (blue)

The Malware Author's Likely Location

Next, we study the compilation timestamps against the first time the malware sample appeared in the wild. We converted all of the compilation timestamps from Epoch timestamps to UTC time, shown in Table 5. We noticed that there are two groups of samples that were seen in the wild before its corresponding compilation timestamps, which means the compilation timestamps might have been generated using the machine's local time. If we assume that the time on the malware authors' machine is accurate, this indicates that the malware author compiled the source code in the UTC +9 time zone or in a time zone further east. Furthermore, the first sample with the Epoch compilation timestamp of 1536011873 was created at 21:57:53 and first appeared in the wild at 12:59. Based on the previous delta compilation time and the time it was first seen, we assume that two minutes is not long enough for compiled samples to appear in the wild, so we added another hour to the delta time. Based on this, we conclude that the malware authors might be located in the UTC +10 time zone or further east.

Infra	RSA Key	Compilation Timestamp in Epoch	Payloads' Compilation Timestamp (UTC)	Time First Seen in the Wild (UTC)	SHA-256
1	A	1528056487	2018-06-03 20:08:07	2018-06-05 03:35	8579ba1eb5cbfd09247e59312449a85fcef42205c075e0dac0d6ac490d0b972d
2	D	1528056680	2018-06-03 20:11:20	2018-06-06 09:10	5aebbb2aa8f76f49970deca34c5e9f8fd6adc5ee0fec1ec09a398e9832893bb5
1	B	1530547690	2018-07-02 16:08:10	2018-07-02 20:27	83f9194627c275b8b8508990fb3e77063a93c3387462c87dc1a1bfccd6e268cf
2	E	1530547815	2018-07-02 16:10:15	2018-07-02 20:34	fa26cce9318c4b1885a6f1e23d9756580a5994178b89ad8beaa889d9c81714aa
1	B	1531161666	2018-07-09 18:41:06	2018-07-09 21:15	537139ce2f4b572eb290d635842aa6335bc7906b3501891cf9852e817f0e6eb9
2	E	1531161732	2018-07-09 18:42:12	2018-07-09 22:09	a8c1e30c59b68348e96b597bb770a2bce88988d0f0c41d2398a8b475e13d41c2

Infra	RSA Key	Compilation Timestamp in Epoch	Payloads' Compilation Timestamp (UTC)	Time First Seen in the Wild (UTC)	SHA-256
2	E	1531899206	2018-07-18 07:33:26	2018-07-18 11:03	0396d6a58613b1e08237b2acfc92df4e1c6f41ce7eb47c0fc2b8bd8aa6c2f8f5
2	E	1531906587	2018-07-18 09:36:27	2018-07-19 09:31	8a2fe06612deef4aa0a6db145f69f5f3af6b9ea7e2f6e2e63d740ee0afb052b3
1	B	1532502303	2018-07-25 07:05:03	2018-07-25 08:10	d103ec149f8ff4a4828c26bc6c61716aa8b239e4e48a01c8ee324741b5f1f9cf
1	C	1536011873	2018-09-03 21:57:53	2018-09-03 12:59	719103e82e66a3b93daa96a4c8d9f1fd2e59978e1309762fbff098d8d781cc0c
2	F	1536011945	2018-09-03 21:59:05	2018-09-03 14:13	01658cd99bcc431e2b5d9116fb90f878bc1907077f0d16d91f7e8d7cc29605a4

Table 5. The compilation timestamp of the unpacked executable samples compared with the time the samples first appeared in the wild

The major cities and countries in the UTC +10 time zone or further east includes Vladivostok in Russia, Guam in the U.S., Melbourne, Brisbane, Sydney, and Canberra in Australia, and Auckland in New Zealand.

Prevention and Solutions

Emotet is mainly delivered via email, adopting social engineering tricks and possibly sent using legitimate email addresses. To avoid unknowingly downloading this malware, we advise users to avoid opening unknown or suspicious emails. It is also important to keep their operating systems updated since Emotet drops SMB exploits as a propagation method. Regularly changing passwords is also useful in combating Emotet. The malware has been observed to drop browser and email password grabber modules to steal users' credentials and has also used hacking tools, i.e., Mimikatz, to recover stored passwords.

We have provided the signatures that identify Emotet-related threats for the benefit of security experts and blue teamers.

Emotet authors often use financial-related subjects, content, and attachment file names in the emails they send to victims. For instance, the frequently used words Emotet actors use are "invoice," "payment," and "receipts" (in English, German. and other languages). If the email makes use of embedded URLs to deliver malicious document droppers, the URLs sometimes contain country information (e.g., US, DE) and some keywords such as "commercial," "small business," "payroll," invoice," "payment," and "personal," among others. The following are some URL examples:

- hxxp://arad-net.ir/files/**En_us**/**Invoice**-for-sent/Deposit/
- hxxp://arendaufa02.ru/files/**En_us**/Aug2018/**Invoice**-067831
- hxxp://cestenelles.jakobson.fr/521EHMUI/**BIZ**/**Personal**
- hxxp://checkout.spyversity.com/9iifVzAhH4pD3D/**BIZ**/Firmenkunden
- hxxp://challengerballtournament.com/9773605LDMSIR/identity/**Smallbusiness**

However, Emotet cannot be easily identified based on the contents of an email. Emotet utilizes social engineering tricks and sends emails from compromised accounts, but several other malware families are also capable of doing this.

The appearance of Emotet's document droppers, however, is almost similar to the screen shown in Figure 5. If there are obfuscated macros in the document, it is possible that the malware belongs to either the Emotet or Ursnif malware families. The metadata of a dropped document to is a good indicator to help determine if it belongs to the Emotet or Ursnif family. Emotet has metadata version 1048576 and a code page of 1252. In order to get the correct metadata from the dropped documents, analysts can submit the documents to public malware scanning engines (e.g., VirusTotal) or use olemeta to obtain the needed information.[33] Checking the embedded URLs is another way to determine if a document belongs to the Emotet malware family. Generally speaking, a document embeds around four URLs in the following format:

URLs that host Emotet's executables:

- hxxp://websitedesigngarden[.]com/e6vTCit
- hxxp://emicontrol[.]com/85a
- hxxp://grupoembatec[.]com/zHVN
- hxxp://stevebrown[.]nl/3YA1kb/

URL that hosts Ursnif's executables:

- hxxp://d792jssk19usnskdxnsw[.]com/MXE/lodpos[.]php?l=yows2[.]xt2

Since the macros are heavily obfuscated, analysts can use automatic sandboxing to log the URLs connected to the macros.

Emotet's executables are packed by custom packers, making it difficult to detect Emotet using static signatures. However, runtime information can provide some valuable data. Emotet's executable often has an obvious signature behavior — a series of network connections to hxxp://[ip]:[port] as disclosed in sandbox reports (in Figure 26). Emotet uses IPs instead of domains for its C&Cs. To get a report of Emotet's network information, analysts could make use of public sandbox services or scanning engines that provide sandbox reports.

http://183.82.101.78
http://63.142.32.242:8080
http://70.164.197.196:8080
http://76.175.26.109
http://199.119.78.23:443
http://77.146.69.15
http://70.105.162.74:443
http://222.214.218.192:4143
http://148.74.40.144:50000
http://95.141.175.240:443
http://207.47.71.46:7080
http://31.49.122.115:50000

Figure 26. Sample connection record of an Emotet executable as seen on VirusTotal

We have documented Emotet's packing and obfuscation methods in the previous section. The configuration inside an executable sample includes a set of C&C servers and an RSA key, which can help track the sample's corresponding infrastructure. We have listed the RSA keys and the C&C servers collected in our research with the infrastructure information in Appendix A.[34] The compilation timestamps against the unpacked payloads are also good indicators for distinguishing the version of Emotet executables.

Conclusion

In this research, we present a comprehensive study on Emotet. We uncover the malware's technical details and infrastructures, as well as provide context on possible attribution. The purpose of the research is to help communities and law enforcement better understand the crime group with only factual data.

According to previous research, the Emotet botnet tended to keep a low profile to avoid law enforcement's radar before 2016.[35, 36] However, it changed when Emotet started to carry several banking trojan families in 2017, turning it into one of the most infamous threats. While it gave communities a better look at Emotet's activities, several critical questions remain answered.

For starters, the Emotet botnet is so huge that it needs to be operated by several individuals. While we have no evidence of any kind supporting it, we suspect a large group of individuals runs Emotet.

Then, we found two clusters of Emotet exploitation based on RSA keys and C2 servers. Based on our long-term experience with botnets and familiarity with the way they work, we suspect that Emotet benefits only a few individuals. While several famous botnets can be rented out as a service by a great number of customers, this does not seem to be the case with Emotet. We believe that Emotet is either completely run and used by the same people or sold as a service to a limited number of individuals or groups. If Emotet is indeed sold or rented as a service, we expect those who buy or rent it to be highly skilled and trusted by Emotet's authors.

References

1. (20 July 2018). *US-CERT*. "Alert (TA18-201A) Emotet Malware." Last accessed on 20 November 2018 at https://www.us-cert.gov/ncas/alerts/TA18-201A.

2. Joie Salvio. (24 June 2014). *Trend Micro Security Intelligence Blog*. "New Banking Malware Uses Network Sniffing for Data Theft." Last accessed on 15 January 2019 at https://blog.trendmicro.com/trendlabs-security-intelligence/new-banking-malware-uses-network-sniffing-for-data-theft/.

3. Brad Duncan. (18 July 2018). *Unit 42*. "Malware Team Up: Malspam Pushing Emotet + Trickbot." Last accessed on 16 January 2019 at https://researchcenter.paloaltonetworks.com/2018/07/unit42-malware-team-malspam-pushing-emotet-trickbot/.

4. (18 July 2018). *Symantec Threat Intelligence Blog*. "The Evolution of Emotet: From Banking Trojan to Threat Distributor." Last accessed on 16 January 2019 at https://www.symantec.com/blogs/threat-intelligence/evolution-emotet-trojan-distributor.

5. (20 July 2018). *US-CERT*. "Alert (TA18-201A) Emotet Malware." Last accessed on 15 January 2019 at https://www.us-cert.gov/ncas/alerts/TA18-201A.

6. Windows code page. (n.d.) *Wikipedia*. Last accessed on 15 January 2019 at https://en.wikipedia.org/wiki/Windows_code_page.

7. Vishal Thakur. (31 May 2018). *Malwarebytes Labs*. "Malware analysis: decoding Emotet, part 1." Last accessed on 16 January 2019 at https://blog.malwarebytes.com/threat-analysis/2018/05/malware-analysis-decoding-emotet-part-1/.

8. Vishal Thakur. (8 June 2018). *Malwarebytes Labs*. "Malware analysis: decoding Emotet, part 2." Last accessed on 16 January 2019 at https://blog.malwarebytes.com/threat-analysis/2018/06/malware-analysis-decoding-emotet-part-2/.

9. Analysis of File-D3079439.doc. (22 September 2018). *Any run*. Last accessed on 15 January 2019 at https://app.any.run/tasks/f5a7c56a-e326-418f-a159-d62b4eb5446a.

10. Microsoft Documents. (30 March 2017). *Microsoft*. "Public KeyBlob Structure."Last accessed on 15 January 2019 at https://docs.microsoft.com/en-us/dotnet/framework/unmanaged-api/strong-naming/publickeyblob-structure.

11. Protocol Buffers' Home Page. (n.d.) *Google Developers*. Last accessed on 15 January 2019 at https://developers.google.com/protocol-buffers/.

12. Pawel Srokosz. (24 May 2017). *CERT Polska*. "Analysis of Emotet v4." Last accessed on 16 January 2019 at https://www.cert.pl/en/news/single/analysis-of-emotet-v4/.

13. (24 July 2018). *Check Point Research*. "Emotet: The Tricky Trojan that 'Git Clones'." Last accessed on 16 January 16 2019 at https://research.checkpoint.com/emotet-tricky-trojan-git-clones/.

14. Nir Sofer. (n.d.) *NirSoft*. "Mail PassView v1.86 - Extract lost email passwords." Last accessed on 15 January 2019 at https://www.nirsoft.net/utils/mailpv.html.

15. Nir Sofer. (n.d.) *NirSoft*. "WebBrowserPassView v1.86." Last accessed on 15 January 2019 at http://www.nirsoft.net/utils/web_browser_password.html.

16. Atif Mushtaq. (21 October 2010). *FireEye Blogs*. "Feodo - A new botnet on the rise." Last accessed on 15 January 2019 at https://www.fireeye.com/blog/threat-research/2010/10/feodosoff-a-new-botnet-on-the-rise.html.

17. Nikita Slepogin. (25 May 2017). *Kaspersky Lab SecureList*. "Dridex: A History of Evolution." Last accessed on 15 January 2019 at https://securelist.com/dridex-a-history-of-evolution/78531/.

18. Feodo Tracker's homepage. (n.d.) *Feodo Tracker*. Last accessed on 15 January 2019 at https://feodotracker.abuse.ch/.

19. Michael Mimoso. (1 July 2014). *Threat Post*. "Cridex Variant Geodo Part Trojan, Part Email Worm." Last accessed on 15 January 2019 at https://threatpost.com/cridex-variant-geodo-part-trojan-part-email-worm/106943/.

20. Traffic Analysis' homepage. (n.d.) *Traffic Analysis*. Last accessed on 15 January 2019 at http://www.malware-traffic-analysis.net/.

21. (24 October 2016). *MalwareBytes Lab*. "Introducing TrickBot, Dyreza's successor." Last accessed on 15 January 2019 at https://blog.malwarebytes.com/threat-analysis/2016/10/trick-bot-dyrezas-successor/.

22. (21 September 2018). *Malware Traffic Analysis*. "EMOTET INFECTIONS WITH TRICKBOT (UK AND US)." Last accessed on 15 January 2019 at http://www.malware-traffic-analysis.net/2018/09/21/index.html.

23. (n.d.) *IBM X-Force Exchange*. "Panda Banker." Last accessed on 15 January 2019 at https://exchange.xforce.ibmcloud.com/collection/Panda-Banker-7d63501790ab200d8a6852508d6f6863.

24. (30 May 2018). *Flashpoint Intel*. "Trickbot and IcedID Botnet Operators Collaborate to Increase Impact." Last accessed on 15 January 2019 at https://www.flashpoint-intel.com/blog/trickbot-icedid-collaborate-increase-impact/.

25. (17 November 2017). *Fidelis Cybersecurity ThreatGeek Blog*. "Tracking Emotet payload: IcedID." Last accessed on 15 January 2019 at https://www.fidelissecurity.com/threatgeek/threat-intelligence/emotet-payload-icedid.

26. Michael Sandee, Tillman Werner, Elliott Peterson. (5 August 2015). *Blackhat*. "Gameover Zeus – Bad Guys and Backends." Last accessed on 15 January 2019 at https://www.blackhat.com/docs/us-15/materials/us-15-Peterson-GameOver-Zeus-Badguys-And-Backends.pdf.

27. (27 September 2017). *Proofpoint*. "Threat Actor Profile: TA505, From Dridex to GlobeImposter." Last accessed on 15 January 2019 at https://www.proofpoint.com/us/threat-insight/post/threat-actor-profile-ta505-dridex-globeimposter.

28. Trend Micro. (18 December 2018). *Trend Micro Security Intelligence Blog*. "URSNIF, EMOTET, DRIDEX and BitPaymer Gangs Linked by a Similar Loader." Last accessed on 17 January 2019 at https://blog.trendmicro.com/trendlabs-security-intelligence/ursnif-emotet-dridex-and-bitpaymer-gangs-linked-by-a-similar-loader/.

29. (13 October 2015). *FBI*. "Bugat Botnet Administrator Arrested and Malware Disabled." Last accessed on 17 January 2019 at https://www.fbi.gov/contact-us/field-offices/pittsburgh/news/press-releases/bugat-botnet-administrator-arrested-and-malware-disabled.

30. Michal Poslušný. (26 January 2018). *We Live Security*. "FriedEx: BitPaymer ransomware the work of Dridex authors." Last accessed on 17 January 2019 at https://www.welivesecurity.com/2018/01/26/friedex-bitpaymer-ransomware-work-dridex-authors/.

31. Jonathan Lusthaus. (23 October 2018). *Harvard University Press*. Industry of Anonymity: Inside the Business of Cybercrime.

32. Winnona DeSombre and Dan Byrnes. (10 October 2018). *Recorded Future*. "Thieves and Geeks: Russian and Chinese Hacking Communities."Last accessed on 15 January 2019 at https://www.recordedfuture.com/russian-chinese-hacking-communities/.

33. (19 February 2018). *Github*. "Olemeta." Last accessed on 15 January 2019 at https://github.com/decalage2/oletools/wiki/olemeta.

34. Exploring Emotet's Activities - Appendix A. https://documents.trendmicro.com/assets/ExploringEmotet'sActivities_AppendixA_Final.pdf.

35. Alexey Shulmin. (9 April 2015). *Kaspersky Lab SecureList*. "The Banking Trojan Emotet: Detailed Analysis." Last accessed on 15 January 2019 at https://securelist.com/the-banking-trojan-emotet-detailed-analysis/69560/.

36. (n.d.) *Center for Internet Security*. "Emotet Changes TTPs and Arrives in United States." Last accessed on 15 January 2019 at https://www.cisecurity.org/blog/emotet-changes-ttp-and-arrives-in-united-states/.

TREND MICRO™ RESEARCH

Trend Micro, a global leader in cybersecurity, helps to make the world safe for exchanging digital information.

Trend Micro Research is powered by experts who are passionate about discovering new threats, sharing key insights, and supporting efforts to stop cybercriminals. Our global team helps identify millions of threats daily, leads the industry in vulnerability disclosures, and publishes innovative research on new threats techniques. We continually work to anticipate new threats and deliver thought-provoking research.

www.trendmicro.com