# Farseer: Previously Unknown Malware Family bolsters the Chinese armoury

By Alex Hinchliffe, Mike Harbison

Published: 2019-02-26 · Archived: 2026-04-05 12:47:02 UTC

Last year, Unit 42 wrote about a newly discovered espionage Android malware family, HenBox, which had countless features for spying on their victims – primarily the Uyghur population – including interaction with Xiaomi IoT devices, and the Chinese consumer electronics manufacturer's smart phones.

Through investigations into infrastructure used by HenBox malware, Unit 42 has discovered another malware family built for the more frequently-targeted Microsoft Windows operating system we named 'Farseer'. As with HenBox, Farseer also has infrastructure ties to other malware, such as Poison Ivy and Zupdax.

We named this malware Farseer malware due to a string found in the PDB path embedded within the executable files. For example:

e:\WorkSpace\A1\coding**Farseer**\RemoteShellsRemote\Release\RemoteShellsRemote.pdb.

Tracking-back, we've seen over 30 unique samples throughout the past two and half years, with the majority in 2017 and a handful in 2018, the most recent of which were seen, at least from our visibility, during the last two months indicating a relatively low-volume yet steady flow of Farseer samples. Figure 1 below shows the trend of malicious sessions for these samples according to AutoFocus.

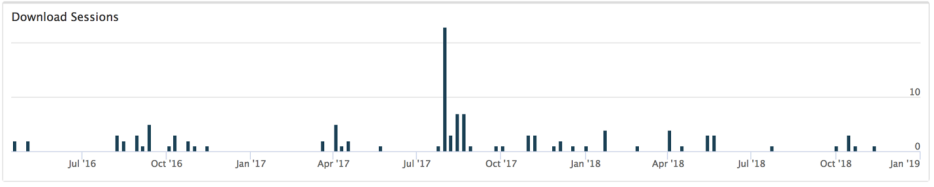


*Figure 1 AutoFocus session trends for Farseer samples over time*

### Ties to HenBox Android Malware et al

As previously mentioned, there are ties between Farseer, HenBox, PlugX, Zupdax, 9002, and Poison Ivy malware families. The infrastructure used by the combination of malware families is pretty vast, with plenty of overlaps, however in this blog we focus only on some of the core ties captured in the green rectangle, as shown in Figure 2 below.

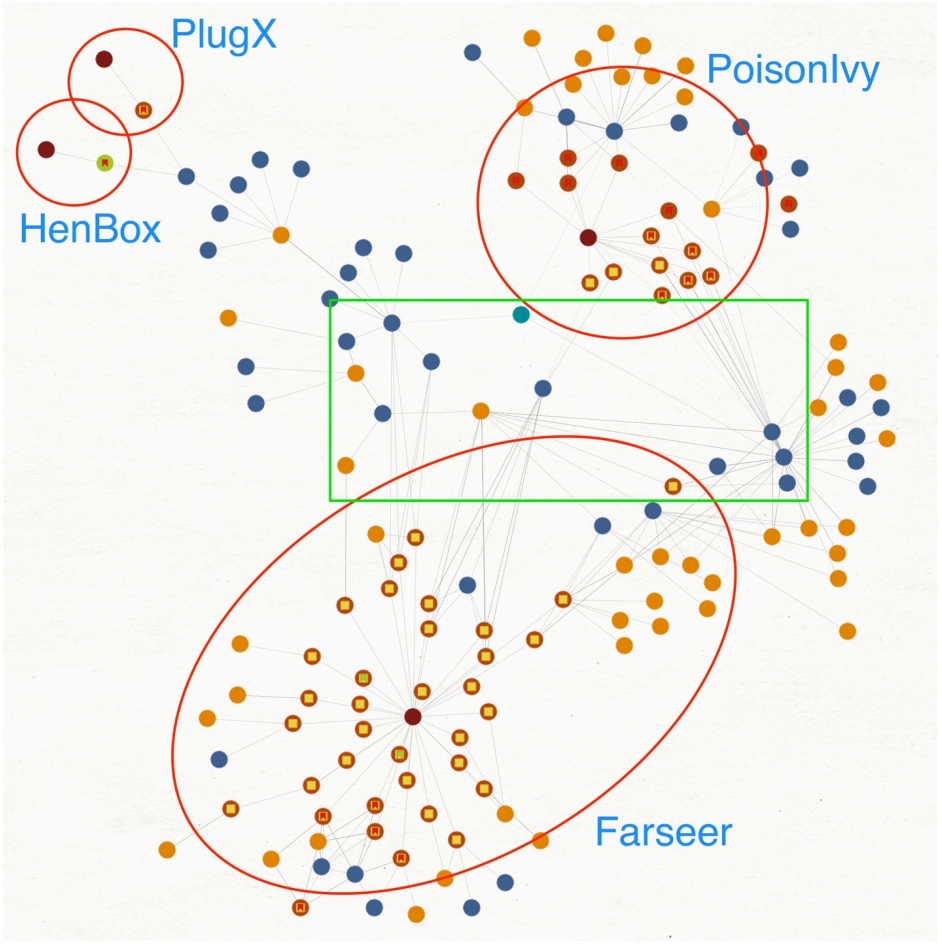


*Figure 2 Maltego chart showing overlaps between Farseer and related threats*

Figure 2 shows a high-level representation of file hashes, IP addresses, and domain names used by some of the various malware families already mentioned, together with their overlaps. Farseer has the largest number of samples in Figure 2 but that's skewed given the focus of this blog.

The green rectangle shows some of the core overlaps between the aforementioned families, which we will discuss in more detail now.

The most recent (at the point of publishing) Farseer sample (SHA256: 271E29FE… detailed in Table 2 below) introduced a new C2 domain – tcpdo[.]net – into the Farseer set, as shown in Figure 3 below.

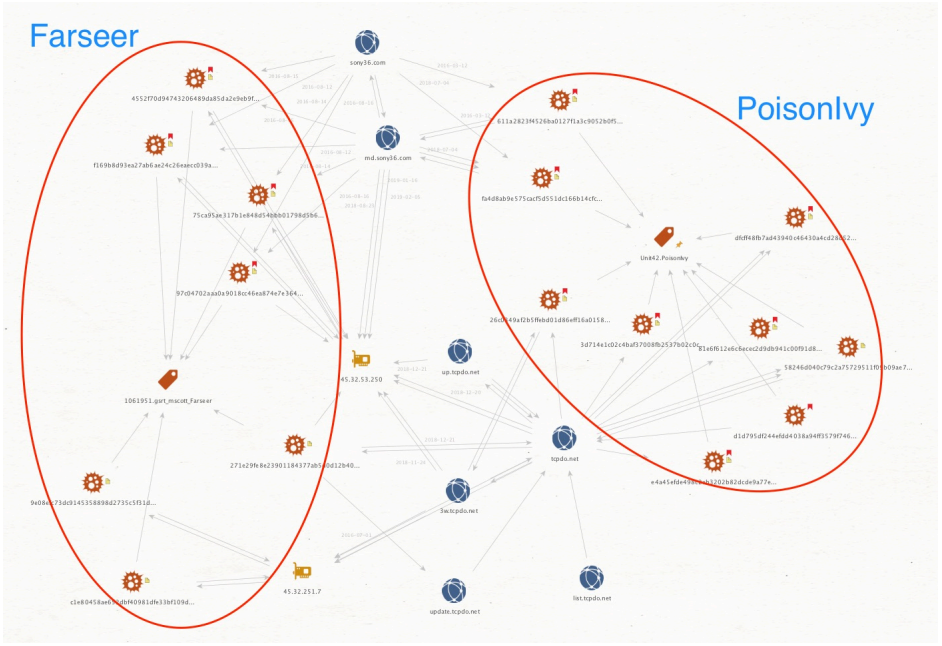


*Figure 3 Maltego diagram showing tcpdo[.]net and other Farseer / PoisonIvy overlaps.*

Figure 3 shows how this new (to Farseer) domain relates both directly to said Farseer sample and indirectly, through third-level domains and IP addresses, to other Farseer samples; a handful of Poison Ivy samples have also used this domain as their C2, mostly before this Farseer sample – as early as mid-2015 – but also more recently, one month after, on December 17th, 2018 indicating it's a domain in fairly active use. Third-level domains of tcpdo[.]net, together with all other indicators are listed at the end of this blog.

The overlaps between Farseer and Poison Ivy don't end with tcpdo[.]net. Much like with HenBox, other infrastructure ties exist: directly through sony36[.]com and md.son36[.]com; indirectly through third-level domains of tcpdo[.]net and IP addresses 45.32.251[.]7 and 45.32.53[.]250.

Farseer also overlaps with HenBox and PlugX samples through multiple C2 domains and IP address resolutions:

- outhmail[.]com (and third-levels of this domain)
- cdncool[.]com (and third-levels of this domain)
- www3.mefound[.]com
- w3.changeip[.]org
- www5.zyns[.]com
- 45.32.53[.]250
- 45.32.44[.]52
- 45.32.45[.]77
- 59.188.196[.]162
- 59.188.196[.]172

Domain outhmail[.]com was documented as part of research into a 9002 Trojan delivered through Google Drive back in 2016 further expanding the capabilities of this group and its tools.

### Ghost Dragon Overlaps

Before we detail the Farseer malware itself, it's worth noting another overlap we encountered during this research. Third-level domain 3w.tcpdo[.]net, as shown towards the bottom of Figure 4 below, resolved to IP 175.45.192[.]234 in 2015. This IP address relates to domains and custom Gh0st RAT malware samples, some of which are documented in this Ghost Dragon campaign report. Considering the time that's passed since this publication, it's harder to investigate how strong the ties are, however, the two domains used by Poison Ivy (md5c[.]net) and Farseer (3w.tcpdo[.]net) have resolved to that IP address more recently than documented in the Ghost Dragon report. Specifically, June 2015, and between July and August 2015, respectively for Poison Ivy and Farseer; these two domains and five others - adminloader[.]com, csip6[.]biz, cdncool[.]com, linkdatax[.]com and adminsysteminfo[.]com - have a common registrant, 46313@QQ[.]COM but no such commonality exists within the set of known Ghost Dragon domains.

It's possible the infrastructure relates to the same group, or multiple groups, conducting various attacks against different operating systems using the various malware families described in this, and related, reports. The possible ties require further investigation.

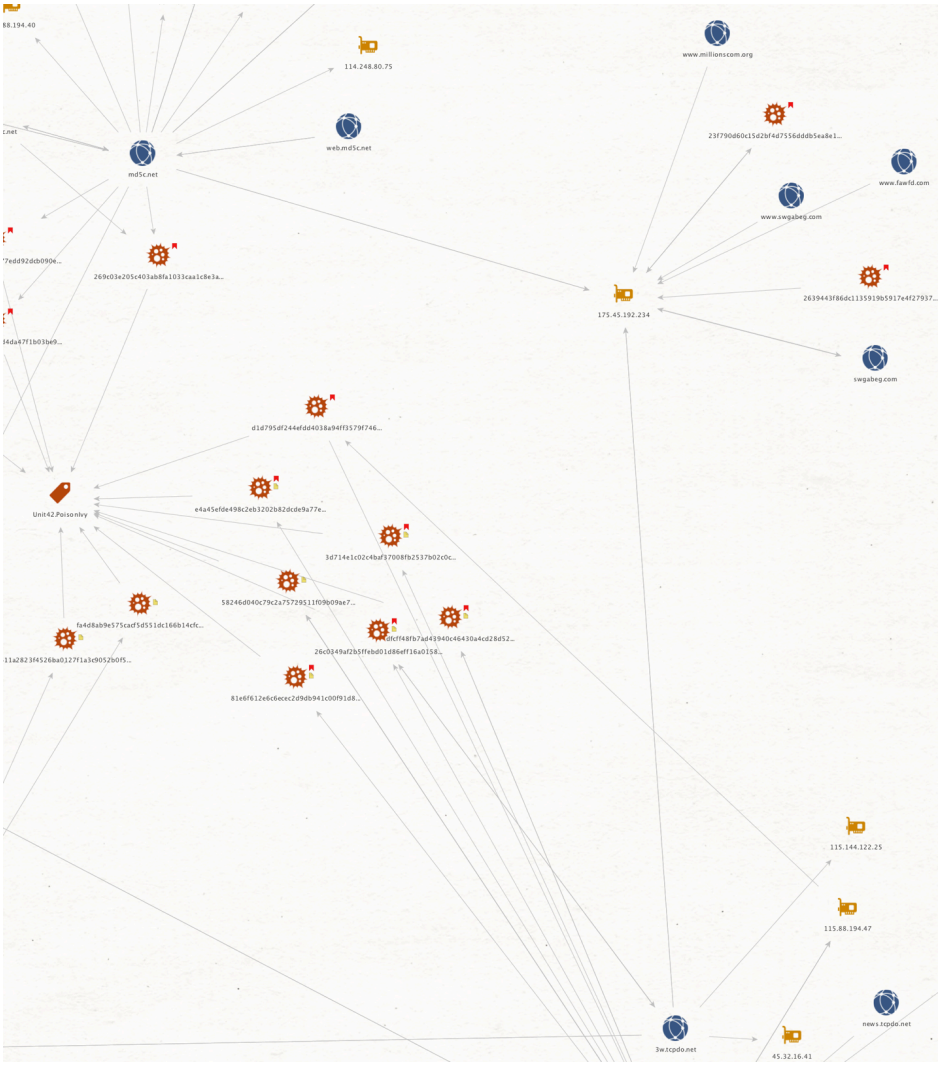


*Figure 4 Maltego chart showing overlaps to Ghost Dragon campaign*

### C2 Server Structure

As previously mentioned in the first HenBox blog, a common registrant registered seven known domains, four of which had malicious activity related to Poison Ivy and Zupdax malware families. Interestingly, all of the domains share at least one third-level domain in common, perhaps indicating a template being used for the infrastructure setup or based on the requirements of the malware's C2 communication. Table 1 below lists the commonalities, aside from other domains such as www, mail and dns.

| Domain / Third-level Domain | info. | re. | update. | up. |
|---|---|---|---|---|
| tcpdo[.]net | • | | • | • |
| adminsysteminfo[.]com | • | • | • | |
| md5c[.]net | | | | |
| linkdatax[.]com | • | • | • | |
| csip6[.]biz | • | • | • | |

| | | | | |
|---|---|---|---|---|
| adminloader[.]com | | • | • | |
| cdncool[.]com | • | • | • | • |
| newfacebk[.]com | | | • | |

*Table 1 Common third-level domain names*

### Farseer Malware

Now that we have introduced Farseer, and how it relates to other known malware families, let's dive into how the malware works. This section aims to provide a description of the general behavior for this malware based on a small subset of total set of samples; a more detailed description exists in the technical appendix.

Figure 5, below, describes at a high-level the post-installation execution flow of a typical Farseer sample.

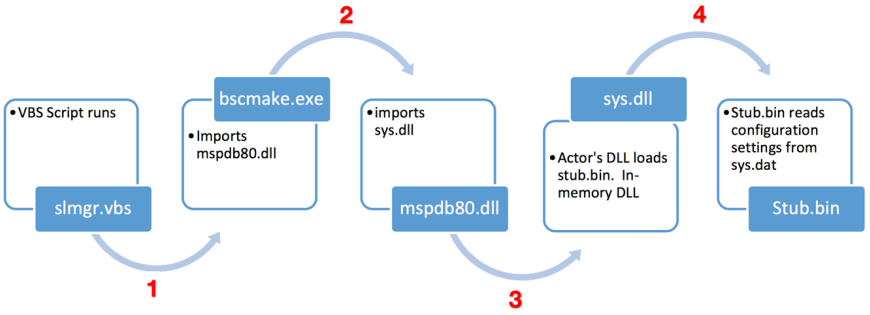


*Figure 5 Farseer Execution Flow*

Farseer employs the known technique of DLL sideloading - the use of trusted binaries to load malicious code – to load its payload, see Figure 5. To achieve this, the malware begins by dropping known, legitimate, signed binaries to the host. These binaries, signed by Microsoft or other vendors, are typically trusted applications when checked by antivirus software or the operating system and thus do not raise any suspicious alerts. Figure 6 below shows the import library list for both the benign PE files highlighting how the nested imports work to ultimately load sys.dll – the malicious payload.

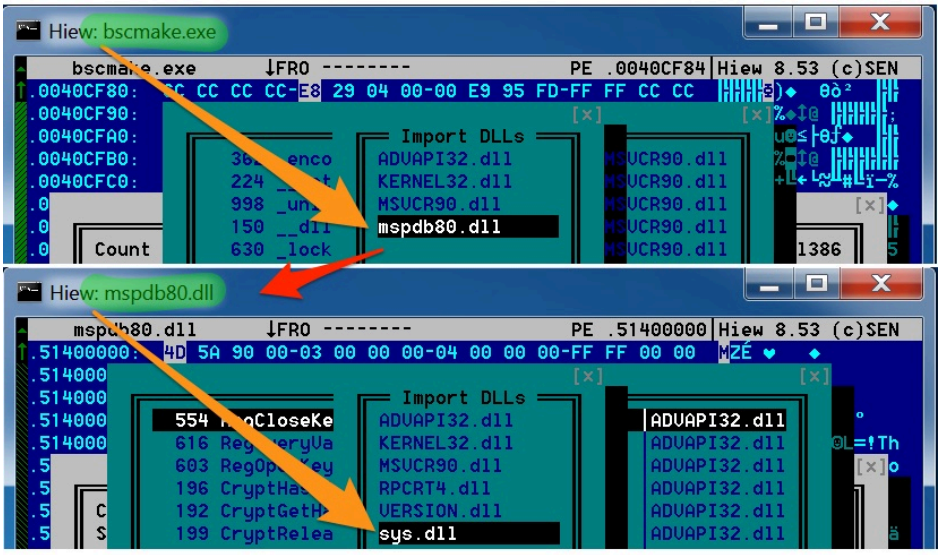


*Figure 6 bscmake.exe importing mspdb80.dll importing Farseer's sys.dll*

The payload on disk is an encrypted and compressed file that most antivirus software will not flag as malicious since the underlying code is hidden. More information about how the decompression and decryption can be found in the appendix.

Once sys.dll is running, it locates a file named stub.bin located in the same folder, and in-turn loads the Farseer config file, sys.dat, on disk. The config relates to C2 communications, amongst other things.

The following two code excerpts show the obfuscated and deobfuscated versions of this variant's configuration file. The obfuscation routine used in this case – and many others – is simply ASCII encoding where characters are replaced with their ASCII value; other variants have used stronger, custom encryption algorithms to hide configuration data. Details are in the appendix.

```
[StudentInfo]

p1=117,112,46,111,117,116,104,109,97,105,108,46,99,111,109,

p2=56,48,

p4=116,101,115,116,45,48,52,45,49,49,

p5=67,58,92,85,115,101,114,115,92,65,68,77,73,78,73,126,49,46,87,73,78,92,65,112,112,68,97,116,97,92,76,111,99,97,108,92,84,101,109,112,92,1

p1=up.outhmail[.]com

p2=80

p4=test-04-11

p5=C:\Users\[<i>username</i>]\AppData\Local\Temp\main.exe
```

The line items in the second code excerpt above are represented as follows:

- p1 relates to the C2 FQDN;
- p2 is the TCP port used for C2 – many variants use non-standard TCP ports;
- p3 is missing;
- p4 appears to be a version string of some sort, which is sent as part of the C2 communication – other variants have used strings, such as "mark";
- p5 is the full file path from where the malware was launched.

Farseer config files share some similarities with those of HenBox, as documented here and shown in Figure 7 below for convenience.

HenBox's config file, setting.txt, is decoded using XOR with a single-byte key, 0x88; filenames and XOR keys differ occasionally between variants. Once de-obfuscated, the config file's contents resembles something like the following text:

```
a1=wd.w3.ezua[.]com
a2=80
a3=crash_report@21cn[.]com
a4=smtp.21cn[.]com
a5=crash_report
a6=lxy.cn@163[.]com
a7=
a8=0914D1D428914B09A5372866B39524B9
a9=
b1=0
b2=0
b3=1
b4=http://www3.mefound[.]com/aa.txt
```

*Figure 7 Screenshot of HenBox configuration file, setting.txt*

Both are text files, read and parsed at run-time; more often than not, the ASCII data is obfuscated using encoding methods of varying sophistication. Perhaps the most notable similarity is the notation of the content, which in both malware families:

- is delimited by an '=' equals character;
- uses a single character followed by a single digit starting from 1 to begin each line;
- has the C2 host/FQDN on the first line;
- has the TCP port to use to connect the C2 on the second line;

For persistence on the host, the Farseer malware creates a registry entry named *sys* under:

HKCU\Software\Microsoft\Windows\CurrentVersion\Run

The entry runs the VBS script *slmgr.vbs* shown below, which executes bscmake.exe, and thus Farseer, each time a user logs on to their PC.

createobject("wscript.shell").run "C:\Users\[username]\AppData\Roaming\windows\bscmake.exe"

One of the earliest Farseer samples Unit 42 analysed also used a decoy PDF document during execution. The PDF content included a copied news article from a Myanmar website that reports on news in the Southeast Asia region. The file properties of said PDF, as shown below, describe the language setting of the application that created it, together with the creation date – eight days prior to the Farseer sample that used the document.

Language : zh-CN

Author : Administrator

Creator : Microsoft® Word 2013

Create Date : 2016:04:11 11:06:30+08:00

Modify Date : 2016:04:11 11:06:30+08:00

More information about this variant of Farseer, and the decoy PDF, can be found in the appendix section.

### Targeting

In this case, we do not have great visibility into the targets of the Farseer malware. However, given our existing knowledge based on previous research, and around malware with closely-related infrastructure, together with certain targeting themes seen in some Farseer samples, it is highly likely that victims lay in and around the South East Asia region.

### ATT&CK Techniques Observed

| ID | Technique |
|---|---|
| T1140 | Deobfuscate / Decode Files or Information |
| T1071 | Standard Application Layer Protocol |
| T1060 | Registry Run Keys / Startup Folder |
| T1045 | Software Packing |
| T1073 | DLL Side-Loading |
| T1065 | Uncommonly Used Port |
| T1043 | Commonly Used Port |
| T1328 | Buy domain name |
| T1319 | Obfuscate or encrypt code |

## Conclusion

The threat actors behind Farseer, and related malware including HenBox, continue to grow their armoury with the addition of this previously-unknown malware family. The overlapping infrastructure, shared TTPs and similarities in malicious code and configurations highlights the web of threats used to target victims in and around the South East Asia region and perhaps beyond.

Farseer payloads are backdoors that beacon to pre-configured C2 servers for instructions. The malware uses various techniques to evade detection and inhibit analysis. For example, DLL sideloading using trusted, signed executables allows the malware to execute rather seamlessly; some payloads are encrypted on disk preventing analysis, especially as decompression and decryption occurs at runtime, in-memory, where code is further altered to thwart forensic analysis.

Whereas HenBox posed a threat for devices running Android, Farseer is built to target Windows, which appears to be more typical given previous threats seen from the group or groups behind this, and related malware.

Palo Alto Networks customers are already protected via:

- All samples in this report have a malicious verdict in WildFire.
- Traps advanced endpoint protection detects Farseer malware.
- Domains have been classified as malicious.
- AutoFocus tags are available for additional context: Farseer.

Update 18th September 2019: This blog has changed to remove two references to PKPLUG as a malware family.

## Appendix

The technical analysis of Farseer malware is described in this section. Table 2 below lists the samples we have chosen for our investigation. The list includes a couple of recent samples and the first Farseer sample seen, according to our data, to highlight key differences in the threat's evolution.

| # | SHA256 | First Seen (Pacific Time) | Key Indicator / Domain |
|---|---|---|---|
| 1 | 271E29FE8E23901184377AB5D0D12B40<br>D485F8C404AEF0BDCC4A4148CCBB1A1A | 11/17/2018 10:11:16 pm | tcpdo[.]net:158 |
| 2 | 4AB41A025624F342DEB85D798C6D6264<br>A9FB88B8B3D9037CF8D5248A9F730339 | 04/02/2018<br>7:18:07 pm | honor2020[.]ga:993 |
| 3 | 9E08EFC73DC9145358898D2735C5F31D<br>45A2571663C7F4963ABD217AE979C7CA | 04/19/2016<br>6:26:15 pm | outhmail[.]com:80 |

*Table 2 Samples discussed in this blog*

Farseer employs the known technique of DLL sideloading - the use of trusted binaries to load malicious code – to load its payload, see Figure 5. To achieve this, the malware begins by dropping known, legitimate, signed binaries to the host. These binaries, signed by Microsoft or other vendors, are typically trusted applications when checked by antivirus software or the operating system and thus do not raise any suspicious alerts. This technique takes advantage of the Windows search order for loading dependencies when a program launches. By default, the Windows loader will first look for any dependency files of the executable in its current working directory. If found, the executable will then load them into memory. With this in mind, the actors place their malicious DLL's in the same directory as the signed executable that was dropped on disk. By naming them as dependency files of that executable, the malicious code will run whenever the executable is started.

Now that the actor has found a way to execute malicious code on the host, they use it to load their final payload, which contains the core functionality of the Farseer malware. The payload on disk is an encrypted and compressed file that most antivirus software will not flag as malicious since the underlying code is hidden. To avoid detection from users and blend with the Windows file system, the payload files themselves have innocuous or common Windows file names and extensions.

Decompression and decryption of the payload occurs only at runtime, in-memory, and the in-memory code is altered to thwart forensic analysis. This is achieved by deconstructing the import address table (IAT) and resolving necessary API calls manually versus relying on the Windows loader. In addition, it further avoids IAT reconstruction by using what is known as stolen code technique, wherein some of the instructions in the beginning of an API subroutine are emulated somewhere else in an allocated memory region. This can cause unexpected results during memory analysis as the IAT API's cannot be resolved. We determined that the in-memory payloads are backdoors that beacon to a pre-configured command and control server (C2) for instructions.

First, bscmake.exe runs and imports mspdb80.dll, one of its dependency files. Bscmake.exe is an older Microsoft executable that is part of Visual Studio. When mspdb80.dll is loaded, it will import its dependency files, one of which is sys.dll. It should be stated that both bscmake.exe and mspdb80.dll are known, trusted files signed by Microsoft Corporation and have not been modified. Sys.dll however is the Farseer malware and is responsible for loading the encrypted file stub.bin file in-memory and begins code execution.

```
.text:100015F2                push    edx             ; lpString1
.text:100015F3                call    esi ; lstrcpyA
.text:100015F5                push    offset aStubBin ; "\\stub.bin"
.text:100015FA                lea     eax, [esp+458h+FileName]
.text:100015FE                push    eax             ; lpString1
.text:100015FF                call    edi ; lstrcatA
.text:10001601                push    0               ; hTemplateFile
.text:10001603                push    80h ; '€'       ; dwFlagsAndAttributes
.text:10001608                push    3               ; dwCreationDisposition
.text:1000160A                push    0               ; lpSecurityAttributes
.text:1000160C                push    1               ; dwShareMode
.text:1000160E                push    80000000h       ; dwDesiredAccess
.text:10001613                lea     ecx, [esp+46Ch+FileName]
.text:10001617                push    ecx             ; lpFileName
.text:10001618                call    ds:CreateFileA
```

*Figure 8 Sys.dll loading stub.bin*

Figure 8 illustrates the connection between sys.dll and stub.bin. When sys.dll is loaded it will look for stub.bin in the current working directory.

### C2

The most recent Farseer sample (#1, as per Table 2 above) communicates with update.tcpdo[.]net over TCP port 158. The contents of the network communications are encoded, unlike the earlier Farseer samples that used no encoding, highlighting one of many changes in the evolution of this malware. Figure 10 below highlights some of the key differences between the three samples used in the analysis for this appendix section.

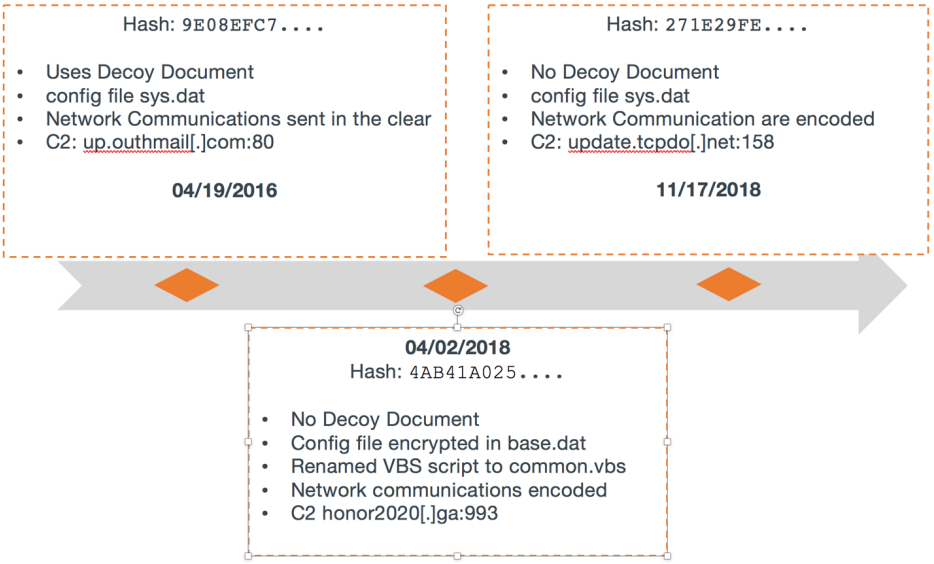


*Figure 9 Timeline for 3 Farseer samples in analysis; comparing notable differences*

Sample #2 (SHA256: 4AB41A025...) behaves almost identically as the others but with the following differences:

- Persistent VBS script renamed to common.vbs
- Encoded network communications
- Configuration file renamed to base.dat
- Encrypted and compressed configuration file
- Does not employ the use of any decoy documents

This sample, seen in early April 2018, communicates with honor2020[.]ga, which started resolving to 199.247.25[.]110 in August 2018, according to Passive Total.

Domain honor2020[.]ga bucks the trend when compared to others' third-level domains, as per Table 1, above. From what we can tell, it has no such subdomains.

Other Farseer samples fall into the same bucket as honor2020[.]ga. That is, they have no third-level domains, or don't match the pattern of others, and they share no overlaps to existing infrastructure whether used by Farseer or other malware families. Examples include windowsnetwork[.]org and newfacebk[.]com. The latter does share one third-level domain with the others in Table 1 but that's where the commonality ends.

Reviewing the dozen or so domains resolving to 199.247.25[.]110, most also make use free ccTLDs from Freenom, including .tk and .ml as per the .ga in honor2020[.]ga. At this point, these domains and others resolving to this IP appear unrelated to Farseer, except for honor2020[.]ga that is connected to Farseer sample 4AB41A025.... It's possible honor2020[.]ga was simply chosen during testing for this more recent Farseer sample but whatever the reason, it's a change from the typically-used .com, .net and .org TLDs used by other samples.

The final sample to discuss (9E08EFC73…) as per Table 2 above, is the oldest sample we have record of in AutoFocus, seen on April 19th, 2016. In this case, a decoy PDF file is dropped and executed from the victim's %TEMP% folder as the malware continues to execute – a behavior not seen again in other Farseer samples. The PDF has filename "Dateline Irrawaddy "Corruption Is Still Rampant Despite The Anti-Corruption Law.pdf" and file properties as shown below, describing the language setting of the application that created it, together with the creation date – eight days prior to us seeing the sample.

Language : zh-CN

Author : Administrator

Creator : Microsoft® Word 2013

Create Date : 2016:04:11 11:06:30+08:00

Modify Date : 2016:04:11 11:06:30+08:00

The content of the benign PDF (shown in Figure 10 below) appears to be a direct copy / paste from old content once posted on the Irrawaddy[.]com news website; their mission "to cover the news in Burma/Myanmar and Southeast Asia accurately and impartially." From what we can tell, the article shown in the PDF was published on the news website sometime in early April 2016, and used as a timely and potentially very topical, social engineering theme for the attack.


*Figure 10 Decoy PDF dropped by earliest version of Farseer malware*

Whilst the decoy PDF is shown to the victim, Farseer continues with the execution process by first creating a Windows sub-folder within the victims C:\Users\[username]\AppData\Roaming folder and drops into it the files listed in Table 3 below.

| Filename | Size in bytes | File Type / Comment |
|---|---|---|
| bscmake.exe | 77,312 | Application signed by Microsoft; used in DLL sideloading technique |
| mspdb80.dll | 193,024 | Microsoft-signed file imported by bscmake.exe |
| slmgr.vbs | 260 | Shell-executes bscmake.exe |
| stub.bin | 71,767 | Encrypted in-memory payload |
| sys.dat | 297 | Config file read by stub.bin |
| sys.dll | 85,504 | Malicious DLL loaded by benign mspdb80.dll file. |

*Table 3 Farseer dropped files*

Palo Alto Networks has shared our findings, including file samples and indicators of compromise, in this report with our fellow Cyber Threat Alliance members. CTA members use this intelligence to rapidly deploy protections to their customers and to systematically disrupt malicious cyber actors. For more information on the Cyber Threat Alliance, visit cyberthreatalliance.org.

## Indicators of Compromise:

### Samples:

271e29fe8e23901184377ab5d0d12b40d485f8c404aef0bdcc4a4148ccbb1a1a

4ab41a025624f342deb85d798c6d6264a9fb88b8b3d9037cf8d5248a9f730339

8ff03c13d0a78003840b7a612e372242c7def123b4fbf5ea1780f2d70eb806a1

5a461104a2b6e313d3d0ee08c26e90db965139b1bff4a785ec297047d570340c

a999489d95e5a94f75de4695c9579ffc88bae02048838e3523f089d970a35abb

0c7e35ca1312204063319a3455ec14bc4b701de205503e63de584f28d99f0291

10bd4507eb12bebc17e216e16950bf77e56c2aad01be7033bf0d5c235f2ad6e5

d44f388842d93807c0b56399c8b7eae5b3dd76871e4908ef3d7d8a559f014fe6

24b52403ff652416c84afed7e12ece11dc59b07f7dba5f007e117a4cfc67c1ab

8890a06d3233ecf661c040ca5c03393c3afd620ccce49fbe08477bbf6b7d9b04

542b2ca4fe2d7d13fa317c58f46942cdf6eb33771bb898d7be773f8ccb50b13c

b782b4c5f8fe2ee318e50ddf985c9132bff6d48b01ea36d6825967bf89e5d0c2

c8b2232360d5d6f56cd6b1076e5e21f0d501f5cb725e0a9b32a0ab661b4c38dd

b82caa5087c6fd8ac79019185c6f8884f5dd9d0266bb7ad635277f3c7ca5c615

da02edf3f33d9801d066c1f93feef33cdedc1bc7b5605498404e8cad8015729f

1e62b7dcb503f47a6330c4dcfc49ea9d921b7d2f8c508769d27df04e61b9471d

0306585900f1b1bddc76149352f90962c365959e44a486ba3547c80d12d56e41

1e46c88420c657c685786bee88f606d494f3d50bcbc616b0f64d2886edd572f2

fd8bb808c7b16cffcb83d7e86d642b5cb6e913e22df69c8dd03ce4e7498f5fdc

f46f162ef279cc6e9c022cffe3a6685d001524e312e7a5f23bd24d76fed1fa99

6e367e10f9c0fb818394e9517ab13c1da00b2545602c23bf6ab83e93063076b8

3d47b99d34e169a8283062937c373264829cf6fe1c7fa0bacee135c392ca24bb

d11d871b07520f43437183fa44bd118c01a3c4c86cffe0cc7343ae9038565cf1

2e84de3408283423ed58764139eed4dd7e343115b943b58a46e2dc25ca2ef3c8

7d5386253d403b74e86658699f9a6d683b7ac3065c4e2cdae192b32b9ac54edb

2085fca368af15a1bd54f7809dfee7cdd4d73df7af88fa53fe5341f0523ca7ea

97c04702aaa0a9018cc46ea874e7e3644146ba4d6b3b30c78a6a6430172b13c7

4552f70d94743206489da85da2e9eb9f1eb3ad017a42edb7a60edb69e5c15a32

75ca95ae317b1e848d54bbb01798d5b61ebcaf4328b3940b5d5f644a01f1943a

f169b8d93ea27ab6ae24c26eaecc039a838bd7e74aef18c1e7a953283c418c30

c1e80458ae652dbf40981dfe33bf109d1b4c85d0affbd16c8d1df6be9e233e05

9e08efc73dc9145358898d2735c5f31d45a2571663c7f4963abd217ae979c7ca

**C2s**

cdncool[.]com

dns.cdncool[.]com

outhmail[.]com

up.outhmail[.]com

tcpdo[.]net

sony36[.]com

md.sony36[.]com

newfacebk[.]com

app.newfacebk[.]com

windowsnetwork[.]org

update.newfacebk[.]com

netvovo.windowsnetwork[.]org

honor2020[.]ga

update.tcpdo[.]net

adminsysteminfo[.]com

md5c[.]net

linkdatax[.]com

csip6[.]biz

adminloader[.]com

outhmail[.]com

cdncool[.]com

www3.mefound[.]com

w3.changeip[.]org

www5.zyns[.]com

108.61.197[.]172

175.45.192[.]234

199.247.25[.]110

208.115.125[.]43

43.224.33[.]130

45.125.33[.]219

45.32.108[.]11

45.32.159[.]168

45.32.24[.]39

45.32.25[.]107

45.32.251[.]7

45.32.44[.]52

45.32.53[.]250

45.76.92[.]113

**Farseer Decoy Docs**

06C091BB0630539DEC0D26EB6BFBF9108152E4C5AF27FF649CE84238CD88F81E - Dateline Irrawaddy “Corruption Is Still Rampant Despite The Anti-Corruption Law.pdf

7F091DA89C4412D71AE583481F91A471751A3C0E8DB0037CF31FFD00F4245B5B –New Microsoft Word 文档.doc

---