Tropic Trooper Targets Taiwanese Government and Fossil Fuel Provider With Poison Ivy

researchcenter.paloaltonetworks.com/2016/11/unit42-tropic-trooper-targets-taiwanese-government-and-fossil-fuel-provider-with-poison-ivy/

Vicky Ray, Robert Falcone, Jen Miller-Osborn, Tom Lancaster

November 22, 2016

By Vicky Ray, Robert Falcone, Jen Miller-Osborn and Tom Lancaster

November 22, 2016 at 3:30 AM

Category: Unit 42

Tags: APAC, Poison Ivy, Taiwan, threat research, Tropic Trooper

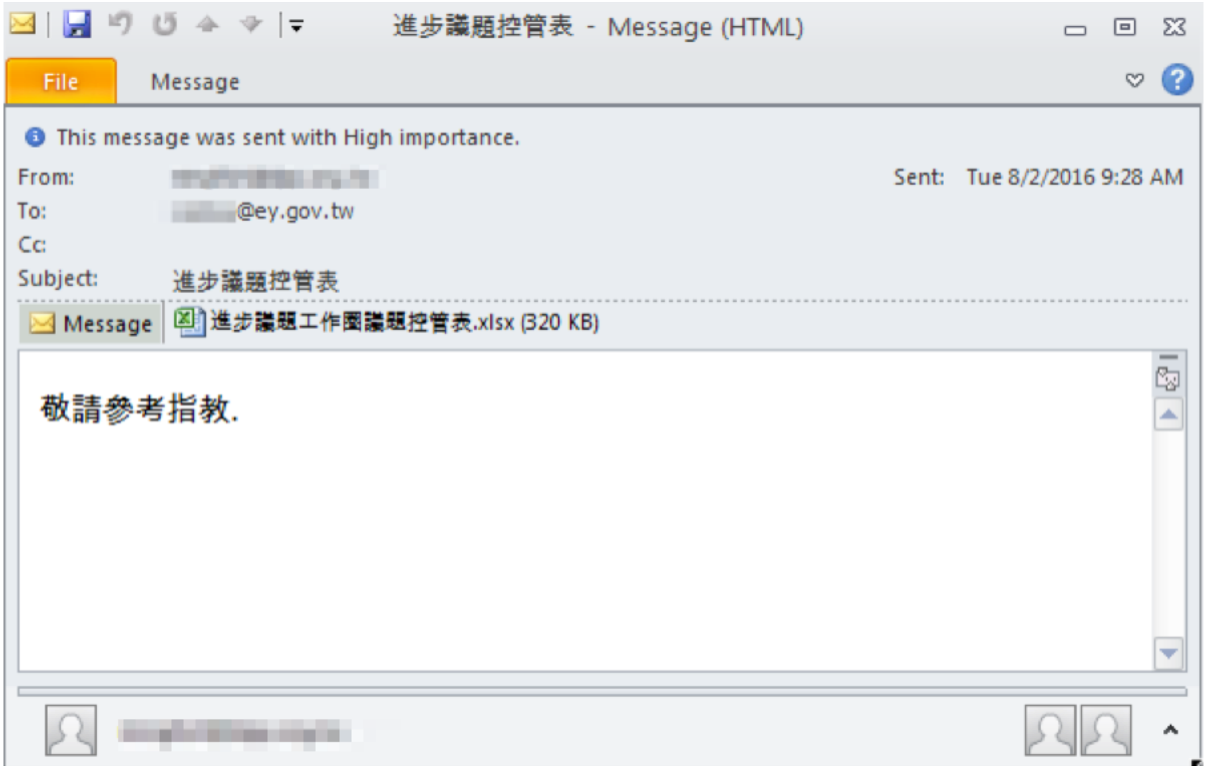
This post is also available in: 日本語 (Japanese)

Taiwan has been a regular target of cyber espionage threat actors for a number of years. Reasons for Taiwan being targeted range from being one of the sovereign states of the disputed South China Sea region to its emerging economy and growth with Taiwan being one of the most innovative countries in the High-Tech industry in Asia.

In early August, Unit 42 identified two attacks using similar techniques. The more interesting one was a targeted attack towards the Secretary General of Taiwan's Government office – Executive Yuan. The Executive Yuan has several individual boards which are formed to enforce different executing functions of the government. The Executive Yuan Council evaluates statutory and budgetary bills and bills concerning martial law, amnesty, declaration of war, conclusion of peace and treaties, and other important affairs. Given the important functions undertaken by the Executive Yuan office, it is not a surprise that they were targeted. The second attack was against an energy sector company also located in Taiwan.

The attacks in this case are associated with a campaign called Tropic Trooper, which has been active since at least 2011 and is known for heavily targeting Taiwan. One of the attacks used their known Yahoyah malware, but the other attack deployed the widely available Poison Ivy RAT. This confirms the actors are using Poison Ivy as part of their toolkit, something speculated in the original Trend Micro report but not confirmed by them. Further analysis uncovered a handful of ties indicating the actors may also be using the PCShare malware family, which has not been previously tied to the group.

Figure 1 shows the spear phishing email which was sent to the Secretary General of Executive Yuan. The email is spoofed so that it appears as though it was sent from a staff member at the Democratic Progressive Party (DPP).



This message was sent with High importance.

From: [redacted]

Sent: Tue 8/2/2016 9:28 AM

To: [redacted]@ey.gov.tw

Cc:

Subject: 進步議題控管表

Message 進步議題工作圈議題控管表.xlsx (320 KB)

敬請參考指教.

[redacted]

Figure 1. Spear-phishing email with malicious attachment.

The document attached to this e-mail exploits CVE-2012-0158, a Microsoft Office vulnerability. This process is described in the Malware Analysis section later in this report, but one interesting aspect of this malicious was the decoy document the attacker chose to deploy.

Decoy Document

As we have noted in many earlier reports, attackers commonly use decoy files to trick victims into thinking a malicious document is actually legitimate. After infecting the computer, the display a clean document to the victim that contains content that is relevant to them.

The decoy document used in this case is a spreadsheet with four tabs, respectively titled “example,” “0720,” “0721,” and “1041109 full update”. All of the text uses Traditional Chinese, in contrast to Simplified Chinese, which is the official written language of the People's Republic of China. Traditional Chinese is used in Taiwan, Hong Kong, Macau, and many overseas Chinese communities. The overarching theme of the spreadsheet is documenting protestor activity and/or progressive reform attempts in progress across Taiwan and the tone of the spreadsheet suggests it was compiled by progressive supporters. Because we were unable to find the spreadsheet online, and there is specific persona data included related to these movements and protests, we are not including any screen shots except for the one below.

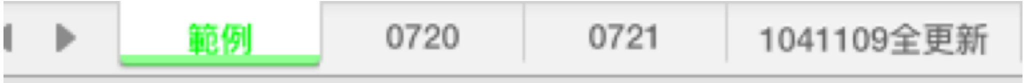


Figure 2. The four tabs in the decoy spreadsheet.

The “example” spreadsheet tab is exactly as described – it contains the headers and suggested information within two of the remaining three tabs. The headers themselves translate, from left to right, to “responsible department,” “issue,” “developments this week,” “political situation judgment,” and “related information.” The tab labeled 0721 only has the matching headers and no additional information. None of the information in the spreadsheet relates to activities past 2015, and there are references made to the then upcoming January 16, 2016 elections in Taiwan. In that election the DPP won, displacing the Chinese Nationalist Party (KMT) for only the second time in history, and with Taiwan’s first female President.

The spreadsheet labeled 0720 refers to the Anti-Black Box Movement, which was a protest by Taiwanese high school students against certain proposed curriculum changes. The use of “black box” by the protestors is in reference to former Taiwanese President Ma Ying-Jeou’s government and its lack of transparency concerning government decisions. Protestors occupied Taiwan’s Ministry of Education last July. A resolution passed by Taiwan’s legislature and approved by the Executive Yuan in May of this year delayed implementing that curriculum until 2020 to allow time for the act to be amended.

The Anti-Black Box Movement is related to the Sunflower Student Movement, a coalition of both student groups and other civic organizations that protested the Cross-Strait Trade Agreement between Taiwan and the PRC, feeling it would hurt Taiwan’s economy and increase the PRC’s sway over the island. On March 17 2014, the KMT, the ruling party at the time, tried to force a vote without a previously agreed clause by clause review with the DPP. The following evening protesters occupied the Legislative Yuan, the first time that had occurred Taiwan's history. On March 23 of the same year, after then President Ma re-affirmed he supported the pact and would not alter or drop it, protestors occupied the Executive Yuan where over 150 were injured and 61 arrested.

The final tab contains the most information of the three and has different headers. From left to right, the headers are titled "responsible person(s)," "summary of issues and major groups," "crisis simulation, political judgment, and recommendations," "degree of tension," and "participating members."

- Information related to the November 2015 "Autumn Struggle" protest, which is an annual protest first done in 2013.
- Information on a Taichung City government development proposal being protested largely on environmental impact grounds, and protestor demands.
- Army 1st Special Forces veterans attempt to receive compensation for alleged illegal extension of forced military service
- The recently settled case where toll workers forced into unemployment by the Taiwanese government's agreement with the Far Eastern Electronic Toll Collection Company to create a national electronic toll collection system ended up resulting in the 2013 layoffs of hundreds, who have since protested for new jobs as well as lost severance and pension.
- Kaohsiung refinery closing and protestor demands, also largely related to environmental effects and necessary cleanup; the refinery officially closed at the end of December 2015
- Closely watching any trade agreements between the Malaysian government and Taiwan
- Potential environmental and current residential issues related to the development of the Aerotropolis around Taoyuan International Airport, which is intended to create a major transportation hub and industry center for Asia with infrastructure for corporate research and development, conference centers, and other facilities.
- The Puyu Development Plan, which is part of Taiwan's Knowledge-based Economy plan
- Taiwan's 12-year compulsory education plan
- Anti-Black Box Movement demands and recent activity
- Improving working conditions for Taiwanese firefighters
- Pension reforms
- The Nest Movement, which started in 2014 and is related to the older "Shell-less Snail Movement," focused on affordable housing, neighborhood and urban development, ending forced demolition and relocation, property tax reform, and related housing issues
- The Environmental Impact Assessment (EIA) voted on by the Environmental Protection Bureau (EPB) for the Dongshi-Fengyuan Expressway, part of the National Highway #4 Project and anti-eviction efforts
- Kaohsiung water quality issues and related projects
- Same sex marriage legalization
- Protecting old trees in Kaohsiung amidst construction for a new "green" library; most of the designated "precious trees" are rare exotic species
- Indigenous peoples in Kaohsiung land return
- Activities against the Miramar Resort Village, including the revocation of the EIA, forcing development to halt
- Lowering the voting age in Taiwan from 20 to 18

Malware Analysis

The documents attached to spear-phishing e-mails used in both attacks contain code that exploits CVE-2012-0158, which despite its age remains one of the most common Microsoft Word vulnerabilities being exploited by multiple threat actors. This matches with known Tactics, Techniques, and Procedures (TTPs) for Tropic Trooper, targeting both government institutions and also the energy industry in Taiwan.

The delivery document uses the XLSX extension typically used by OpenXML documents, but the file itself is actually an OLE (XLS) document. The file extension to file type discrepancy was caused by the actor using Excel's built-in encryption capability, which stores XLSX ciphertext and the information needed for decryption in an OLE document.

Filename: 進步議題工作圈議題控管表.xlsx
MD5: a89b1ce793f41f3c35396b054dbdb749
SHA1: f45e2342e40100b770d73dd06f5d9b79bfce4a72
SHA256: 2baa76c9aa3834548d82a36e150d329e3268417b3f12b8f72d209d51bbacf671
Type: CDF V2 Document, No summary info
Size: 327128 bytes

Table 1. Details of the malicious document attached to the e-mail.

The embedded shellcode enumerates open handles for a file with a size greater than 0xa6f0 (Decimal - 42736) bytes. It will then set the file pointer to 0xa6e8 (Decimal - 42728) and starts looking for the following delimiter:

GfCv\xef\xfe\xec\xce

If it finds this delimiter, the shellcode knows it is working with the correct file and continues by reading 0x600 (decimal 1536) bytes following this delimiter. The shellcode then decrypts the first 0xc0 (decimal 192) DWORDs of the data read from the file using an XOR algorithm that decrypts one DWORD of ciphertext at a time with 0x29f7c592. The resulting cleartext is a second piece of shellcode that continues carrying out further functionality.

The secondary shellcode starts by resolving the following API functions using a ROT13 hashing algorithm:

kernel32.dll!CreateFileA
kernel32.dll!ReadFile
kernel32.dll!WriteFile
kernel32.dll!SetFilePointer
kernel32.dll!CopyFileA
kernel32.dll!MoveFileExA
kernel32.dll!CreateToolhelp32Snapshot
kernel32.dll!Process32Next
kernel32.dll!CloseHandle
kernel32.dll!VirtualAlloc
kernel32.dll!WinExec
kernel32.dll!TerminateProcess
kernel32.dll!LoadLibraryA
kernel32.dll!lstrlenA
kernel32.dll!lstrcpyA
kernel32.dll!lstrcatA
kernel32.dll!GetTempPathA
kernel32.dll!WideCharToMultiByte
kernel32.dll!QueryDosDeviceA
ntdll.dll!NtQueryObject
advapi32.dll!RegOpenKeyA
advapi32.dll!RegSetValueExA
advapi32.dll!RegCloseKey

Immediately following these API functions there are three DWORDS; one used to locate the payload embedded within the exploit file, one for the size of the payload, and one for the size of decoy document. The two size values are added together to get the length of the ciphertext that the shellcode will decrypt. In the sample we analyzed, the following values were present, showing that the payload is at offset 0xabc0 and has a size of 0x45218:

DWORD offset_toPayload; (0ABC0h)
DWORD payload_Size; (1C600h)
DWORD decoy_Size; (28C18h)

The shellcode then creates a string that it uses to create a registry key to automatically run the final payload each time the system starts. It then opens the registry key 'Software\Microsoft\Windows NT\CurrentVersion\Winlogon' and sets the value to the "Shell" subkey to the previously created string. Ultimately, the following registry key is created for persistence:

HKCU\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\Shell:
"explorer.exe,rundll32.exe "C:\Documents and Settings\Administrator\Application
Data\Identities\Identities.ocx" SSSS"

It then uses the "offset_toPayload" value as an offset that it will read 283160 (45218h) bytes from the XLS file. The shellcode then enters a decryption loop to convert the embedded payload from ciphertext to cleartext. The algorithm uses the length of the ciphertext negated as the initial encryption key, which it bit rotates right by 1 to adjust the key for each of decryption. It will use this key to decrypt four bytes of the ciphertext with the XOR operation until all the ciphertext is decrypted. During each iteration of the decryption process, the algorithm will check to make sure the four bytes of ciphertext are not equal to the key or equal to zero before decrypting the ciphertext. The following table contains the first five rounds of the algorithm to explain the decryption process:

	Key	Ciphertext	Cleartext
0	~0x45218 = 0xFFFBADE8 >> 1 = 0x7FFDD6F4	0x7F6D8CB9	0x00905a4d = MZ\x90\x00
1	0x7FFDD6F4 >> 1 = 0x3FFEEB7A	0x3FFEEB79	0x03 = \x03\x00\x00\x00
2	0x3FFEEB7A >> 1 = 0x1FFF75BD	0x1FFF75B9	0x04 = \x04\x00\x00\x00
3	0x1FFF75BD >> 1 = 0x8FFFBADE	0x8FFF4521	0xFFFF = \xff\xff\x00\x00
4	0x8FFFBADE >> 1 = 0x47FFDD6F	0x47FFDDD7	0xB8 = \xb8\x00\x00\x00
5	0x47FFDD6F >> 1 = 0xA3FFEEB7	0x00000000	0x00000000 = \x00\x00\x00\x00
6	0xA3FFEEB7 >> 1 = 0xD1FFF75B	0xD1FFF71B	0x40 = \x40\x00\x00\x00
7	0xD1FFF75B >> 1 = 0xE8FFFBAD	0x00000000	0x00000000 = \x00\x00\x00\x00
8	0xE8FFFBAD >> 1 = 0xF47FFDD6	0x00000000	0x00000000 = \x00\x00\x00\x00

9	0xF47FFDD6 >> 1 = 0x7A3FFEEB	0x00000000	0x00000000 = \x00\x00\x00\x00
10	0x7A3FFEEB >> 1 = 0xBD1FFF75	0x00000000	0x00000000 = \x00\x00\x00\x00
11	0xBD1FFF75 >> 1 = 0xDE8FFFBA	0x00000000	0x00000000 = \x00\x00\x00\x00
12	0xDE8FFFBA >> 1 = 0x6F47FFDD	0x00000000	0x00000000 = \x00\x00\x00\x00
13	0x6F47FFDD >> 1 = 0xB7A3FFEE	0x00000000	0x00000000 = \x00\x00\x00\x00
14	0xB7A3FFEE >> 1 = 0x5BD1FFF7	0x00000000	0x00000000 = \x00\x00\x00\x00
15	0x5BD1FFF7 >> 1 = 0xADE8FFFB	0xADE8FEF3	0x108 = \x08\x01\x00\x00
16	0xADE8FFFB >> 1 = 0xD6F47FFD	0xD84E60F3	0xEBA1F0E = \x0e\x1f\xba\x0e
17	0xD6F47FFD >> 1 = 0xEB7A3FFE	0x26738BFE	0xCD09B400 = \x00\xb4\x09\xcd
18	0xEB7A3FFE >> 1 = 0x75BD1FFF	0x39BCA7DE	0x4C01B821 = \x21\xb8\x01\x4c
19	0x75BD1FFF >> 1 = 0xBADE8FFF	0xD28AAE32	0x685421CD = \xcd!Th
20	0xBADE8FFF >> 1 = 0xDD6F47FF	0xAD4F3496	0x70207369 = is p
21	0xDD6F47FF >> 1 = 0xEEB7A3FF	0x9CD0CC8D	0x72676F72 = rogr
22	0xEEB7A3FF >> 1 = 0xF75BD1FF	0x947BBC9E	0x63206D61 = am c
23	0xF75BD1FF >> 1 = 0xFBADE8FF	0x94C3869E	0x6F6E6E61 = anno
24	0xFBADE8FF >> 1 = 0xFDD6F47F	0x98B4D40B	0x65622074 = t be
25	0xFDD6F47F >> 1 = 0xFEEB7A3F	0x909E081F	0x6E757220 = run

Table 2. Decrypting the payload

As you can see from the table above, the algorithm decrypts what is an embedded portable executable that acts as the payload in this attack. The embedded payload is written to %APPDATA\Identities\Identities.ocx and has the following attributes:

```
MD5: 53f5b9d9e81612804ddaf15e71d983c7
SHA1: aa32739c1b5c23274bfbdc24b882a53c868d1e04
SHA256: c098235a43d9788661490d2c7b09b1b2b3544d22ee8d9ae6cd5d16a977fd1155
Type: PE32 executable for MS Windows (DLL) (GUI) Intel 80386 32-bit
Size: 116224 bytes
Imphash: 58089f7df19ceafda8af75236cb1852a
Compiled: 2016-05-23 07:00:51
Architecture: x86
Exports:
(0x1a90) OnUserModel
(0x1a90) SSSS
```

The decoy document, described in the section above, is saved to %TEMP%\進步議題工作圈議題控管表.xlsx and has the following attributes:

```
MD5: 7ba4837be46ed1d9b58721a2c103a523
SHA1: bb5fa41034bfe16a06ac95fbc504e2e779b3219b
SHA256: 9dc5ecf4235841d91dd90c5410251b3dafee5c8dee598fd934018a1c62452a3a
Type: Zip archive data, at least v2.0 to extract
Size: 166936 bytes
Meta:
  Author: Read64
  Last Modified By: Windows 用户
  Created: 2016:07:21 03:15:34Z
  Modified: 2016:07:21 07:30:17Z
```

The shellcode will move the decoy document to the location of the originally executed XLSX file and will create the following command:

```
cmd /c start excel /e  “<path to original XLSX file, now decoy
document>”
```

Before running the above command to open the decoy document, the shellcode enumerates the running processes on the system, specifically looking for processes created for an executable with a filename that starts with “avp.”, presumably in an attempt to find Kaspersky’s antivirus process. If the process is found, the shellcode will not open the decoy document and exits.

The shellcode does not launch the payload, rather it relies on the registry key it created for persistence to execute the payload when the user reboots the system, meaning during dynamic analysis the execution of the payload may be missed.

Delivered Payload – Poison Ivy

When the system starts up, the persistence registry key will launch the Identities.ocx payload and call its “SSSS” exported function. The “SSSS” function checks to make sure that the DLL is running within the context of a “rundll32.exe” process and then begins piecing 0x141B bytes of data together in the correct order to build the shellcode of the Poison Ivy Trojan.

We found and parsed the following configuration from the Poison Ivy shellcode:

```
Campaign ID: MyUser
Group ID: MyGroup
C2 Cnt: 3
- C2 #0: 202.133.236.177:443
- C2 #1: news.hpc.tw:53
- C2 #2: account.sino.tw:80
Comm Key: twone
Mutex: (V!hex67)
Auto-remove Dropper Flag: 1
Active Setup value name: StubPath
Default browser path reg key: SOFTWARE\Classes\http\shell\open\command
Active Key registry key: Software\Microsoft\Active Setup\Installed Components\
```

Looking for more samples which exhibited the same file structure, encryption and obfuscation to deliver the above Poison Ivy sample yielded only two additional samples. In the other two instances the delivered payloads were respectively PCShare and Yahoyah. PCShare has not been previously associated with Tropic Trooper, but in addition to the aforementioned overlaps, the two samples have passive DNS overlap with some known Tropic Trooper infrastructure. For those reasons, we assess with limited confidence the group is also using this malware family.

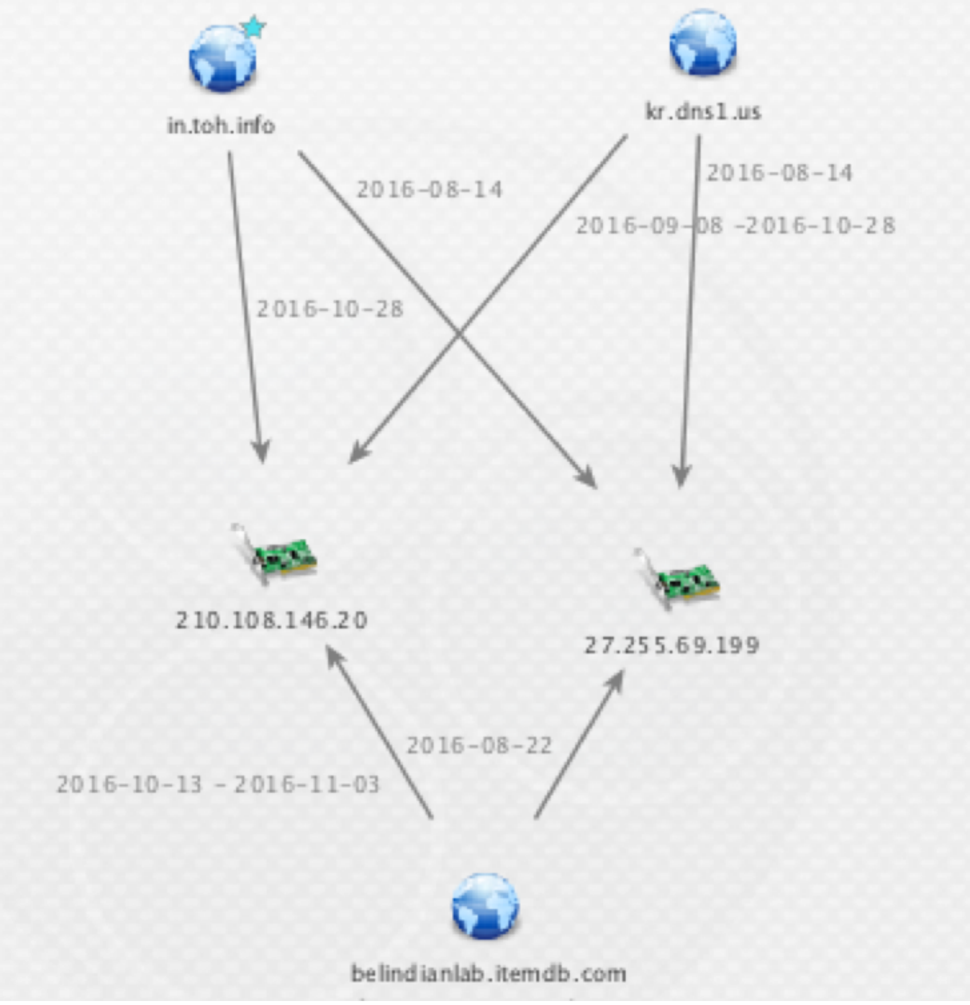


Figure 3. The limited ties between C2 infrastructure used by Yahoyah samples (top) and PCShare malware samples (bottom).

The below table shows the details of the documents, payload delivered and the C2 servers used for communications.

SHA256	a3becf3639fa82bfbf01740ce5a8335f291fb83b544e02a6cc9f1e9c96fb3764

Filename	CTC Statement.xlsx
Payload	d76d7d64c941713d4faaedd5c972558c5136cd1b7de237280faaae89143e7d94
Tool	PCShare
C2	belindianlab[.]itemdb[.]com
C2 IP	210.108.146[.]20

SHA256	ca10489091b71b14f2c3dc0b5201825e63a1f64c0a859ba2bd95900f45580fc4
Filename	全台餐廳更新版餐廳_.xlsx
Payload	bff5f2f84efc450b10f1a66064ed3afaf740c844c15af88a927c46a0b2146498
Tool	Yahoyah
C2	www[.]dpponline[.]trickip[.]org
C2	www[.]myinfo[.]ocry[.]com
C2 IP	223.27.35[.]244

It is interesting to see that the exploit documents we found had either low or no detections on most popular antivirus engines, showing that the threat actors behind this campaign have been having considerable success in bypassing static analysis undertaken by traditional antivirus solutions with this technique.

We further expanded our search using the AutoFocus Threat Intelligence platform on the IOCs extracted from the PIVY, PCShare and Yahoyah payloads and found 42 samples which either matched unique behaviors, the unique PIVY mutex or had common C2 infrastructure. The hashes of all the samples found are given in the appendix section at the end of this blog.

Figure 4 below shows the compilation timestamps of the payload samples found using AutoFocus. Given some of the payloads that were used in recent attacks, which were compiled months before, it shows that the threat actor group continues to reuse the payload within their exploit documents.

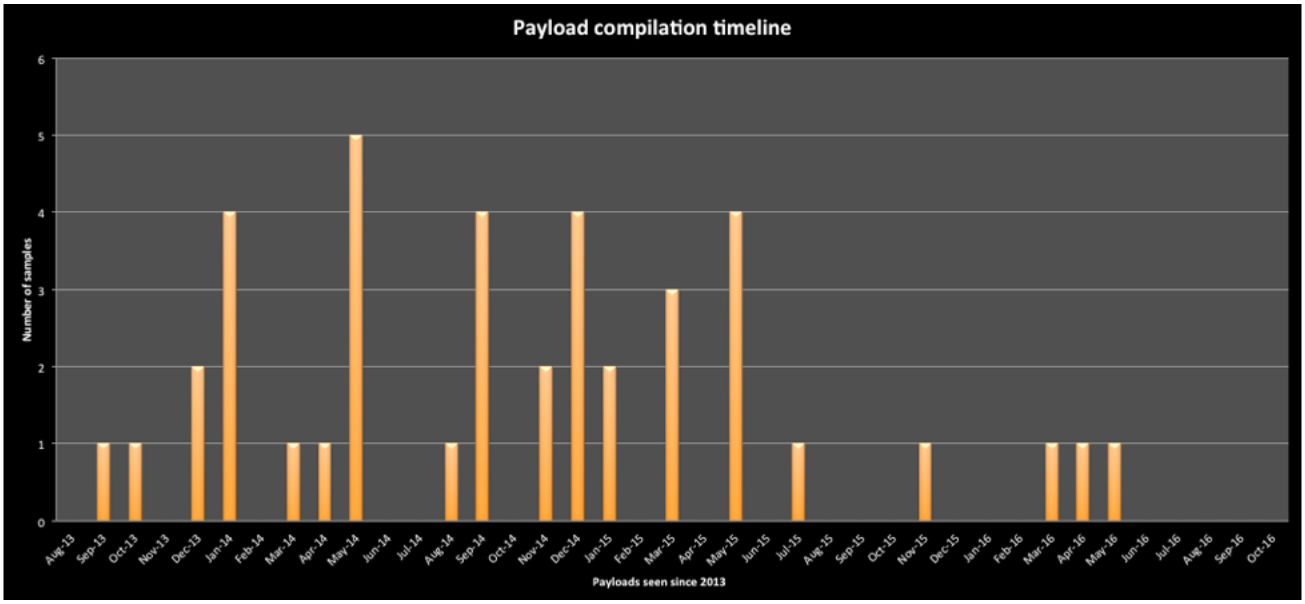


Figure 4. Payload Compilation Timelines

The below Maltego graph shows some of the shared infrastructure which have been used by Tropic Trooper. The complete list of indicators on the graph can also be found in the appendix section of this report.

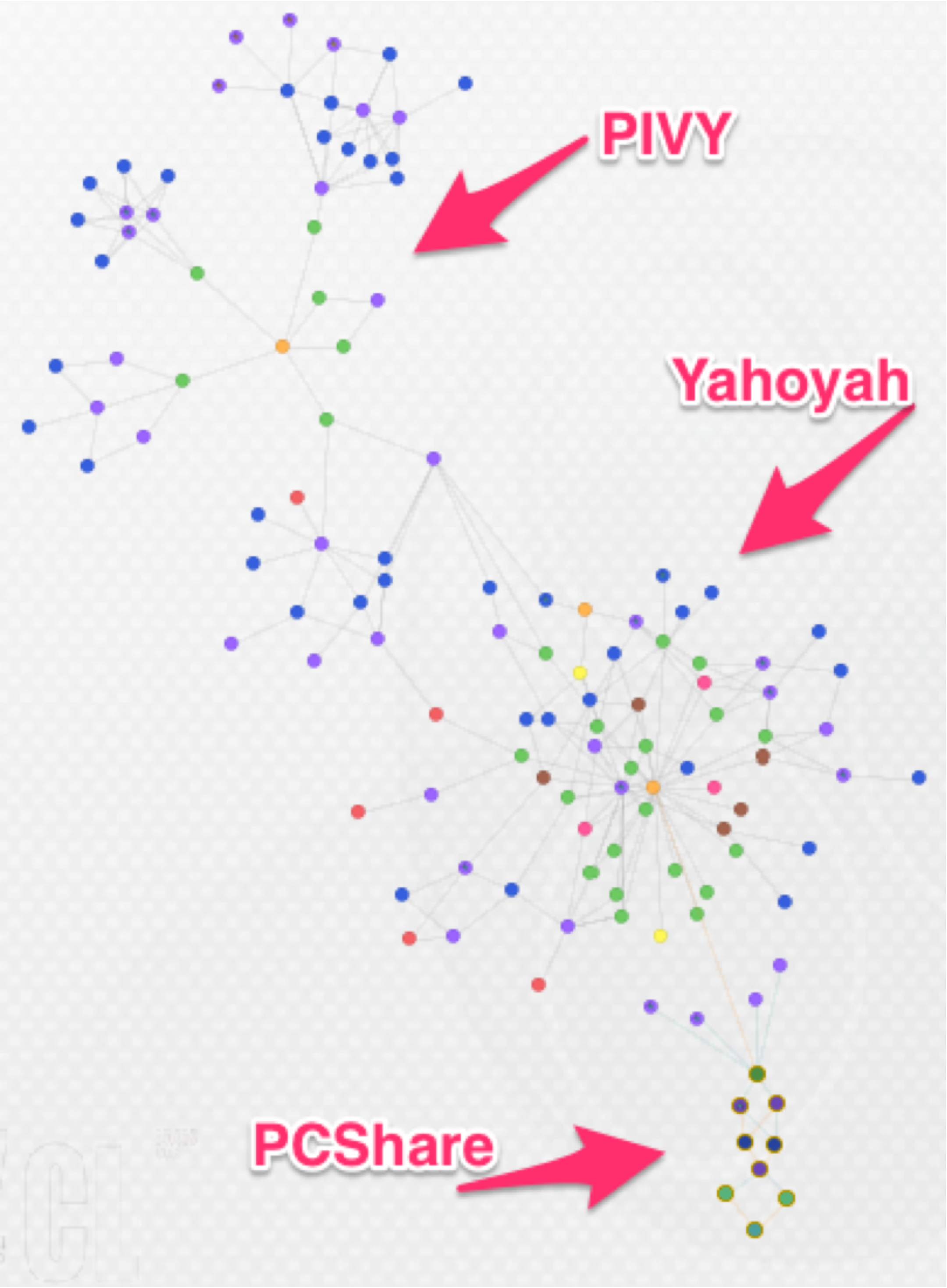


Figure 5 Maltego graph of Tropic Trooper infrastructure

Conclusion

The Tropic Trooper threat actor group has been known to target governments and organizations in the Asia Pacific region for at least six years. In addition to using Yahoyah malware, we were able to confirm they are also using Poison Ivy and possibly PCShare malware families. They are also still exploiting CVE 2012-0158, as are many threat actors. Palo Alto Networks customers are protected from Tropic Trooper's malicious activities by:

- WildFire correctly identifies all related malware as malicious
- The C2 infrastructure are classified as malicious in PAN-DB
- Traps prevents exploitation of CVE-2012-0158

Autofocus customers can discover additional information on Tropic Trooper via the following AutoFocus tags:

Appendix

Samples matching unique indicators, behaviors and C2 infrastructure from the payload extracted out of the malicious documents:

SHA256 hashes
Winsloader
c098235a43d9788661490d2c7b09b1b2b3544d22ee8d9ae6cd5d16a977fd1155
e81bc530075d6d31358aea5784d977d1ac2932a13a615cd1319d01d6e39c2995
cf32fb6371cc751b852c2e2e607c813e0de71cd7bcf3892a9a23b57dfd38d6fc
07663f8bca3c2118f3f77221c35873fd8dd61d9afa30e566fe4b51bcfb000834
92da05bae1d9694a1f63b854e86b5b17ef27d5fc2551318e49e17677c7c90042
e267ecfd37f3af55e8b02b081e7c9d8c0bf633e1d5acb0228be694eae4660eee
PCShare
d76d7d64c941713d4faaedd5c972558c5136cd1b7de237280faaae89143e7d94
66d672a94f21e86655f243877ee04d7e67a515a7153891563f1aeedb2edbe579
Yahoyah
85904e7b88b5049fb99b4b8456d9f01bdbf8f6fcf0f77943aed1ce7e6f7127c2
2fce75daea5fdaafba376a86c59d5bc3e32f7fe5e735ec1e1811971910bc4009
aa812b1c0b24435b8e01100760bc4fef44032b4b0d787a8cf9aef83abd9d5dbd
9623d6f3a3952280f3e83f8dbb29942694bb682296d36c4f4d1d7414a7493db0
f0aa64c1646d91b0decbe4d4e6a7cc53bfd770c86ded9a7408034fa14d2bad83
73bba13d1c7b6794be485a5eeb7b79a62f109c27c4c698601945702303dbcd6c
25809242472a9e1f08ff83c00fae943a630867604ff95c7a57313187287384d2
72d14f0a7ecb04eb2962bc9d8491194deb856ceebf30e7ecd644620932f3d4b0
2172cc228760d6e4fa297bc485637a2b17103ae88237b30df39babe548cefaa5
fdeb384ff68b99514f329eeffb05692c4c1580ca52e43e6dcbb5d760c2a78aa4
1432a8a6ae6faa5d9f441b918ddc3edddb9c133458853ad356756835fe7b3291
a4334a33e4a87cfa52e9e24f6b4d3da0b686f71b25e5cc9a6f144485ea63108a
7f8abefcc4598c643dff1ebf570677fd5c2a4f3d08bc8ddabbfbef1eed097fb3
8e1a0d93ae644ac80048e5c3485bc6282a69d52cf26f94d2be1ce634851ac3aa
c2ad0204ff90c113f7984a9db6006c9f09631c4983098803591170be62cdfaa7
8ccaade84c9c7d5955e8aa1a0d36542beeaed5b8f619aedf82f74e8fd5a5283b
03e9c25fe979f149f6dafb0398cdf3d2223b26f24009ef0f83825b60e961d111
bee4cc2c3c393953f9247eab45767e01cd26d40037fb00bd69441e026d860a63
626f65d4d638437aaa8352fe06589165d52a91e0963c988348b00734b0a3419f
5395f709ef1ca64c57be367f9795b66b5775b6e73f57089386a85925cc0ec596
72cc8c41008310024e9339b9e45bec7815b7fa8a0c3b6a56769d22bc4ced10ed
fefd9bfb0f984590b54908c6868b39ca587a3e0d8198b795ff58f67adee4b9e9
4ee115734733dae0705e5b2cb6789a1cdb877bc53e2fdb6e18ab845c0522d43b
6b6ec318ede71baf79004fe22c46a8d7a500dc6ba6dd40b2641fe9a1c2b3dbd5
78eda231bf494c7008a4ad49e982f2470597199829d46b166a75f654e3cb8d59

21857cdd794649d72ab1bf90acfa8a57767a2a176b46cdb930025cf9242303bb
bff5f2f84efc450b10f1a66064ed3afaf740c844c15af88a927c46a0b2146498
6597c49bedf3fb1964e7f6ccbb03db9e38a5903a671209ae4d3fb4f9f4db4c95

Poison Ivy
6966e511a45e42a9cfa32799dd3ecf9ec1c2cf62ed491f872210334a26e8a533
84f9d3c0895fbcc3148ec77b967eb9cdf33eb90915937b91a61664d36eed7464
c4b73d2102c25e31e3b73a8547a0120e1d3706eed96392acb174ecbf1218fa37
c9d0d7e3ba9a1369b670511966f2c3b5fa3618d3b8ac99cbc3a732bd13501b99
ee3f29d2a68217825666dae6a56ae7ee96297ea7f88ae4fd78819983ae67a3ce
edfedfad21bd37b890d0e21c3c832ff9493612f9959a32d6406750b2d4a93697

C2 domains
news[.]hpc[.]tw
www[.]dpponline[.]trickip[.]org
www[.]forensic[.]zyns[.]com
www[.]bannered[.]4dq[.]com
www[.]forensic611[.]3-a[.]net
bbs[.]zzbooks[.]net
bbs[.]ccdog[.]net
wallstreet[.]1dumb[.]com
www[.]cham[.]com[.]tw
pinkker[.]zzux[.]com
www[.]amberisic611[.]4dq[.]com
www[.]metacu[.]ygto[.]com
bbs[.]zzbook[.]net
www[.]myinfo[.]ocry[.]com
www[.]gmal1[.]com
news[.]hpc[.]tw
www[.]dpponline[.]trickip[.]org
pinkker[.]zzux[.]com
wallstreet[.]1dumb[.]com
redpeach[.]youdontcare[.]com
redapple[.]justdied[.]com
stone[.]mypop3[.]org
zeus[.]jkub[.]com
sniper[.]mynumber[.]org
unclesam[.]jungleheart[.]com
arora[.]x24hr[.]com
flanando[.]fartit[.]com
www[.]dpponline[.]trickip[.]org

www[.]myinfo[.]ocry[.]com
belindianlab[.]itemdb[.]com
kr[.]dns1[.]us

C2 HTTP requests
hxxp://www[.]dpponline[.]trickip[.]org/images/D2015_id[.]jpg
hxxp://223[.]27[.]35[.]244/images/D2015_id[.]jpg
hxxp://www[.]myinfo[.]ocry[.]com/images/D2015_id[.]jpg
hxxp://belindianlab[.]itemdb[.]com/1613986301|C7A5398FBD8214C92F6596CC39B8866B0121E53422D6B8378E5D1F5F63844D693810BDED3
hxxp://202[.]153[.]193[.]73/images/kong[.]24[.]jpg
hxxp://113[.]10[.]221[.]89/images/kong[.]24[.]jpg
hxxp://61[.]221[.]169[.]31/images/kongj[.]24[.]jpg
hxxp://www[.]forensic611[.]3-a[.]net/monitor/images/Smarp140102[.]24[.]gif
hxxp://www[.]bannered[.]4dq[.]com/monitor/images/Smarp140102[.]24[.]gif
hxxp://www[.]forensic[.]zyns[.]com/monitor/images/Smarp140102[.]24[.]gif
hxxp://113[.]10[.]221[.]89/Pictures/sbsb_0620[.]24[.]jpg
hxxp://bbs[.]ccdog[.]net/Pictures/sbsb_0620[.]24[.]jpg
hxxp://www[.]forensic611[.]3-a[.]net/monitor/images/Smartzh131225[.]24[.]gif
hxxp://www[.]bannered[.]4dq[.]com/monitor/images/Smartzh131225[.]24[.]gif
hxxp://www[.]forensic[.]zyns[.]com/monitor/images/Smartzh131225[.]24[.]gif
hxxp://bbs[.]zzbooks[.]net/Pictures/lclc_0523[.]24[.]jpg
hxxp://bbs[.]ccdog[.]net/Pictures/lclc_0523[.]24[.]jpg
hxxp://113[.]10[.]221[.]89/Pictures/lclc_0523[.]24[.]jpg
hxxp://50[.]117[.]38[.]164/Pictures/dzh_0925[.]24[.]jpg
hxxp://www[.]cham[.]com[.]tw/images/dzh_0925[.]24[.]jpg
hxxp://113[.]10[.]221[.]89/Pictures/dzh_0925[.]24[.]jpg
hxxp://bbs[.]ccdog[.]net/Pictures/jpg_140430[.]24[.]jpg
hxxp://198[.]100[.]122[.]66/Pictures/jpg_140430[.]24[.]jpg
hxxp://192[.]69[.]221[.]92/Pictures/jpg_140430[.]24[.]jpg
hxxp://www[.]bannered[.]4dq[.]com/monitor/images/SmartNav141216[.]64[.]gif
hxxp://www[.]amberisic611[.]4dq[.]com/monitor/images/SmartNav141216[.]64[.]gif
hxxp://www[.]metacu[.]ygto[.]com/monitor/images/SmartNav141216[.]64[.]gif
hxxp://www[.]metacu[.]ygto[.]com/monitor/images/SmartNav141216[.]32[.]gif
hxxp://www[.]amberisic611[.]4dq[.]com/monitor/images/SmartNav141216[.]32[.]gif
hxxp://www[.]bannered[.]4dq[.]com/monitor/images/SmartNav141216[.]32[.]gif
hxxp://bbs[.]ccdog[.]net/Pictures/20150120-hex[.]64[.]jpg
hxxp://23[.]27[.]112[.]216/Pictures/20150120-hex[.]64[.]jpg
hxxp://bbs[.]zzbook[.]net/Pictures/20150120-hex[.]64[.]jpg
hxxp://bbs[.]zzbook[.]net/Pictures/20150120-hex[.]32[.]jpg
hxxp://23[.]27[.]112[.]216/Pictures/20150120-hex[.]32[.]jpg
hxxp://bbs[.]ccdog[.]net/Pictures/20150120-hex[.]32[.]jpg

hxxp://bbs[.]ccdog[.]net/Pictures/h20141212012[.]64[.]jpg
hxxp://23[.]27[.]112[.]216/Pictures/h20141212012[.]32[.]jpg
hxxp://113[.]10[.]221[.]89/Pictures/h20141212012[.]32[.]jpg
hxxp://bbs[.]ccdog[.]net/Pictures/h20141212012[.]32[.]jpg
hxxp://113[.]10[.]221[.]89/Pictures/ooba_0823[.]24[.]jpg
hxxp://198[.]100[.]122[.]66/Pictures/ooba_0823[.]24[.]jpg
hxxp://50[.]117[.]38[.]164/Pictures/ooba_0823[.]24[.]jpg
hxxp://www[.]metacu[.]ygto[.]com/monitor/images/SmartNav0120[.]64[.]gif
hxxp://www[.]amberisic611[.]4dq[.]com/monitor/images/SmartNav0120[.]64[.]gif
hxxp://www[.]bannered[.]4dq[.]com/moitor/images/SmartNav0120[.]64[.]gif
hxxp://www[.]bannered[.]4dq[.]com/moitor/images/SmartNav0120[.]32[.]gif
hxxp://www[.]metacu[.]ygto[.]com/monitor/images/SmartNav0120[.]32[.]gif
hxxp://www[.]amberisic611[.]4dq[.]com/monitor/images/SmartNav0120[.]32[.]gif
hxxp://www[.]dpponline[.]trickip[.]org/images/D2015_id[.]jpg
hxxp://223[.]27[.]35[.]244/images/D2015_id[.]jpg
hxxp://www[.]myinfo[.]ocry[.]com/images/D2015_id[.]jpg
hxxp://49[.]254[.]211[.]75//tedws/1[.]64[.]jpg
hxxp://107[.]183[.]183[.]235/public/1[.]64[.]jpg
hxxp://49[.]254[.]211[.]75//tedws/1[.]32[.]jpg
hxxp://107[.]183[.]183[.]235/public/1[.]32[.]jpg
hxxp://flanando[.]fartit[.]com/2015/p1[.]64[.]jpg
hxxp://flanando[.]fartit[.]com/2015/p1[.]32[.]jpg