How to perform long term monitoring of careless threat actors

Daniel Lunghi (@thehellu)
June 03, 2020 - SSTIC conference, (Cyber) Rennes, France
Outline

• Introduction
• Malware analysis and classification
• Pivoting on the samples
• Pivoting on the infrastructure
• Telemetry and links with known threat actors
• Bonus
• Conclusion
Introduction
Introduction

• This talk focuses on the methodology of long term threat actor monitoring.

• Examples are based on a Trend Micro investigation published on February 18, 2020.

  Operation DRBControl - Uncovering a Cyberespionage Campaign Targeting Gambling Companies in Southeast Asia

• Goals:
  • Establish Tactics, Technics and Procedures (TTP) of a threat actor
  • Help incident response/detection
  • Get as much context as possible
Introduction

• Investigation started on July 2019, after Talent-Jump technologies brought interesting samples to us

• The samples were found in a gambling company in Philippines

• No obvious link to a known threat actor
Malware analysis and classification
Malware analysis and classification

• **Goals:**
  - Extract IOCs (domain names, IP addresses, file names, registry keys...)
  - List the malware features
  - Find the malware family, if known

• **How:**
  - Pick your favorite disassembler
  - Classification: Yara, TLSH, search engines...
Malware analysis and classification

- Initial triaging result:
  - 4 different families, of which 3 are unknown
  - Only known family was found in October 2019

- Let’s focus on “Type 1” malware, but the methodology is the same for other families
Malware analysis and classification

- Malware is packed and uses DLL side-loading

![Diagram showing the process of Malware analysis and classification]

- MsMpEng.exe: Legitimate Microsoft file
- Mpsvc.dll: Malicious DLL
- Mpsvc.mui: Encrypted payload

Legitimate file loads Mpsvc.dll, which decrypts and loads MsMpEng.exe.
Malware analysis and classification

- Malware is written using C++, it support plugins, class names can be extracted from RTTI information and are self-explanatory
  - CHPKeylog
  - CHPScreen
  - CHPAvi
  - CHPCmd
  - CHPEXplorer
  - CHPREgedit
  - Complete list on our paper
Malware analysis and classification

• Samples contain a version number

<table>
<thead>
<tr>
<th>Version number</th>
<th>Compilation date</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>May 2019</td>
</tr>
<tr>
<td>8.0</td>
<td>July 2019</td>
</tr>
<tr>
<td>9.0</td>
<td>August 2019</td>
</tr>
</tbody>
</table>

• Shows fast development pace of the threat actor
Pivoting from samples
Pivoting from samples

- “Easy” pivoting: unique strings
  - Query on search engine (sandbox results)
  - “content” modifier on VirusTotal or similar malware repositories
  - Yara rules for more complex queries
  - RetroHunt for past malwares

⇒ Fail, malware is packed
Pivoting from samples

• Algorithm for network communication encryption uses a substitution table of 256 bytes
• 256 bytes hardcoded in a specific order
• Yara rule written, alerting added and RetroHunt launched

⇒ New samples found, all relevant
Pivoting from samples

• On March 23th, an alert matching this substitution table is raised
• The related sample is not a malware
⇒ The Yara rule is prone to false positives
Pivoting from samples

• We found source code posted on February 27, 2015 on CodeProject.com matching the assembly code

Packet encryption/decryption function

See more: C++

Good day to you all!
I have a quick question for the pro-coders around here:
I have a function to encrypt/decrypt my packets in my online game using defined keys.4
Here are the keys, generated random:

```
BYTE server_keys[2][256] = {
    { 0xFC, 0x77, 0xA1, 0x85, 0x1F, 0x30, 0x51, 0x26, 0x93, 0x4A, 0xE3, 0x10,
    0x0E, 0x32, 0x58, 0x64, 0x36, 0x8C, 0x19, 0xF8, 0x61, 0xE0, 0xDF, 0x9E, 0xF9,
    0x90, 0xD0, 0x05, 0xFA, 0xE8, 0x3D, 0x4B, 0xA5, 0xF1, 0x72, 0x73, 0x04, 0xB5, 0x76,
    0xD7, 0xCD, 0x9A,
```
Pivoting from samples

• Metadata in different file formats is also useful
  • VERSIONINFO structure from the PE format contains information on filename, description, version, etc
  • Documents contain metadata (title, author name, …)

• In this particular investigation, we could find several related samples by leveraging metadata
  • 2 malware samples had “HaoZipUpdate” as original filename
  • 4 malicious documents had “Dell_20170514745” as author
Pivoting from samples

- Legitimate HaoZipUpdate was patched
Pivoting from samples

• Mutexes might be used for correlation
  • SFX archive dropping Trochilus malware named “diskshawin.exe” uses mutexes with unique names (“cc5d64b344700e403e2sse”, “cc5d6b4700e403e2sse” and “cc5d6b4700032eSS”)
  • A BbsRAT sample named “diskwinshadow.exe” found in a public sandbox report also uses these mutexes
  • That BbsRAT sample has “bot.googlerenewals.net” as C&C, which is listed in a report from ClearSky on Winnti threat actor
Pivoting from infrastructure
Pivoting from infrastructure

• Passive DNS: database of historical links between IP addresses and domain names
• Some threat actors reuse their servers or domain names for multiple campaigns
• Needs to be handled with caution, it is prone to false positives and false negatives
Pivoting from infrastructure

- IP addresses history for domain name update.microsoftdefender.com as seen on PassiveTotal

<table>
<thead>
<tr>
<th>Resolve</th>
<th>Location</th>
<th>Network</th>
<th>ASN</th>
<th>First</th>
<th>Last</th>
</tr>
</thead>
<tbody>
<tr>
<td>45.32.13.143</td>
<td>JP</td>
<td>45.32.8.0/21</td>
<td>20473</td>
<td>2020-03-31</td>
<td>2020-04-21</td>
</tr>
<tr>
<td>43.228.126.172</td>
<td>SG</td>
<td>43.228.126.0/24</td>
<td>133905</td>
<td>2019-07-19</td>
<td>2020-03-20</td>
</tr>
</tbody>
</table>
Pivoting from infrastructure

- Truncated list of domain names history for IP address 43.228.126.172 as seen on PassiveTotal

<table>
<thead>
<tr>
<th>Resolve</th>
<th>First</th>
<th>Last</th>
</tr>
</thead>
<tbody>
<tr>
<td>update.microsoftdnsdown.com</td>
<td>2019-11-17</td>
<td>2020-03-31</td>
</tr>
<tr>
<td>support.microsoftdnsdown.com</td>
<td>2019-10-21</td>
<td>2020-03-31</td>
</tr>
<tr>
<td>update.microsoftdefender.com</td>
<td>2019-07-19</td>
<td>2020-03-20</td>
</tr>
<tr>
<td>rollback.us</td>
<td>2018-05-22</td>
<td>2019-04-27</td>
</tr>
<tr>
<td>photon-sg-1.sakay.ph</td>
<td>2018-06-04</td>
<td>2018-10-06</td>
</tr>
</tbody>
</table>
Pivoting from infrastructure

• Some threat actors register their domain names in bulk
  ⇒ Creation Date timestamp for those domains is close
• mircosoftdefender.com created on 2018-08-09 at 08:40:27
• By filtering on registrar and name server, we find 3 additional domains created on same date between 08:40 and 08:41
  • dinohonevice.com
  • luxespiremag.com
  • googleusermessage.com
Pivoting from infrastructure

- Many more techniques
  - TLS certificate tracking
  - Correlation through metadata (web server version, hosting provider, HTTP headers ...)
  - Search of domain names/IP addresses on public sandboxes results
  - HTTP static content tracking
Pivoting

• All those techniques need to be reiterated when new IOCs are found

![Diagram showing domain name, sample, and IP address relationships.]

© 2020 Trend Micro Inc.
Telemetry and further links
Telemetry

• As an AV, we have telemetry from our customers (if enabled)

• Spear-phishing emails sent on May 2019
  • Different company, also in South-East Asia
  • Also in gambling/betting industry

⇒ Confirmation of the targeted industry and location
Links with known threat actors

- **Links with Winnti**
  - Shared mutexes, which means probably code sharing for a dropper
  - We noticed a binary being downloaded from an IP address by the threat actor: Passive DNS for that IP address showed domains related to Winnti

- **Links with EmissaryPanda/LuckyMouse**
  - We found a sample from the HyperBro family, which is used exclusively by this threat actor
Using malware features to our advantage
Using malware features to our advantage

- Type 1 malware has a secondary C&C channel

  - Standard C&C channel
    - Uploads computer info in own dir
    - Looks for “bin.asc” file in own dir and loads it
  - Send commands
  - Puts “bin.asc” file in chosen dir
  - Dropbox
Using malware features to our advantage

- To read and write to the repository, the malware uses a hardcoded API key

**Dropbox Python SDK**

A Python SDK for integrating with the Dropbox API v2. Compatible with Python 2.7 and 3.4+. [Doc](https://www.dropbox.com/developers/documentation/python)

**Link an account**

In order to make calls to the API, you'll need an instance of the access token for your own account through the [App Console](https://www.dropbox.com/developers/documentation/python).

```python
dbx = dropbox.Dropbox('YOUR_ACCESS_TOKEN')
```

**Try some API requests**

You can use the Dropbox object you instantiated above to make API calls.

List all of the contents in the user's root directory:

```python
for entry in dbx.files_list_folder('').entries:
    print(entry.name)
```
Using malware features to our advantage

- “bin.asc” is a new malware family using Dropbox as C&C (analysis is available in our paper)
- 142 different directories, of which 129 contain a “bin.asc” file
- ~50 post-exploitation tools found in the repository
  - Mimikatz, Quarks PwDump
  - Nbtscan
  - Privilege escalation tools
  - UAC bypass
Using malware features to our advantage

Bar chart showing the frequency of various malware features:
- List files: High frequency
- Run command: Medium frequency
- Upload file: Lower frequency
- Download file: Lower frequency
- List drives: Very low frequency
- Delete file: Very low frequency
- Execute file
- Rename file
- Copy file

Table showing the number of occurrences of commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Number of occurrences</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat -ano</td>
<td>24</td>
</tr>
<tr>
<td>tasklist</td>
<td>19</td>
</tr>
<tr>
<td>systeminfo</td>
<td>19</td>
</tr>
<tr>
<td>query user</td>
<td>18</td>
</tr>
<tr>
<td>ipconfig /all</td>
<td>16</td>
</tr>
<tr>
<td>whoami</td>
<td>15</td>
</tr>
<tr>
<td>reg query “HKEY_CURRENT_USER\Software\Microsoft\Terminal Server Client\Default”</td>
<td>12</td>
</tr>
<tr>
<td>dir wlbsctrl.dll</td>
<td>11</td>
</tr>
<tr>
<td>type log.txt</td>
<td>10</td>
</tr>
<tr>
<td>set</td>
<td>10</td>
</tr>
</tbody>
</table>
Using malware features to our advantage

• On March 2020, we noticed a new campaign using Type 1 malware family
• After extracting Dropbox API key, we noticed permissions had been modified
• Token was not allowed to list directories

⇒ Threat actor reacted to our publication
Conclusion
Conclusion

• Started from ~20 samples of 4 different malware families, 5 domain names and 3 IP addresses

• After the investigation:
  • 8 different malware families
  • 19 domain names, 9 IP addresses
  • Tens of different samples
  • Infection vector found
  • List of post exploitation tools
  • Victimology confirmed
  • Links with two known threat actors
Conclusion

• Threat intelligence enrich knowledge of a threat actor
• It needs access to big amount of data
• It requires diverse skills
• Each security vendor has its own perspective of the attack

⇒ Collaboration is welcome
Acknowledgements

• Cédric Pernet and Kenney Lu, my dear colleagues
• Our boss Ziv for giving us enough time to dig
• Researchers at Talent-Jump technologies for sharing samples
• Bernard Pivot
The Art of Cybersecurity

Threats detected and blocked globally by Trend Micro in 2018. Created with real data by artist Daniel Beauchamp.