ZEIT8028: Digital Forensics

Lab 4: Network Forensics

Background

Seeing the office has been a bit quiet of late, the boss has suggested that you knock off some of the smaller network forensic jobs that keep getting bumped down the priority list. It's Friday afternoon and you figure it's an easy way to transition into the weekend, so you roll up your sleeves and fire up Wireshark.

You take the first three jobs from the top of the work priority queue; that should be enough to keep you busy until home time.

BEWARE: THERE IS MALWARE IN THESE IMAGES!

Exercise 1: Incident Report 01

The brief attached to the first investigation simply reads:

> *Someone at Dynaccountic has infected their Windows computer. Please provide an appropriate incident report.*

A single network capture file has been provided to you (you'll find it in the `Lab 4 - Network Forensics.7z` lab bundle):

```
PATH:   ./01/network.pcap
SIZE:   1,406,038 bytes
SHA1:   b35aa493edc4075cadd74e63dd54c545b1786a55
MD5:    9a7d22c0be0262fd951060dd27abcf7b
```

To complete this task, write an incident report that includes:

1. The datetime of the infection

2. Who was infected (IP address, hostname, MAC address, and user account name)
3. What malware was involved
4. The likely source of the infection
5. Any indicators associated with the infection (IP addresses, domains, URLs, and file hashes, if any)

> **▼ Hint**
>
> You can open Wireshark from the `favourite apps` menu or the terminal:
>
> ```
> analyst@forensics~$ wireshark network.pcap &
> ```
>
> Try starting with either the `Statistics -> Protocol Hierarchy Statistics` or `Statistics -> Conversations` pane.

> **▼ Hint**
>
> It's probably worth honing in on certain protocols typically used be adversaries (i.e. SMB, DNS, HTTP...).
>
> In Wireshark, `Right Click -> Follow Stream` can be pretty handy for this.

> **▼ Hint**
>
> To export files, go to `File -> Export Objects`
>
> *Again, beware there's malware in these samples!*
>
> If you export interesting files, you could try submitting the hash(es) to VirusTotal.
>
> ***Remember:*** *you can generate the hash of files using* `md5sum` *or* `shasum`

> **▼ Hint**

> Remember our discussions about pivots from module 2.
>
> Don't forget to look at the activity ±10, ±20, ±30+ around the time of the malicious happenings.

> ▼ **Solution**
>
> For more information and the solution, check the solution video.

> ▼ **Spoilers!**
>
> 1. **The datetime of the infection:** 10/04/2018 20:14:26 (most likely - packet 674)
> 2. **Who was infected:**
>   - **IP:** 10.10.10[.]209
>   - **MAC:** 00:30:67:F1:2D:63
>   - **Hostname:** Bastiste-PC
>   - **User:** winford.batiste
> 3. **What malware was involved:** Trickbot
> 4. **The likely source of the infection:** Sample downloaded from http://caveaudelteatro.it/ser0410[.]bin
> 5. **Any indicators associated with the infection:**
>   - https://www.virustotal[.]com/gui/file/c2c1e2c22f67dda6553cbcc173694b68677b77319243684925e8dc3f78b3dbf8
>   - **MD5:** 65c0910b047c11038ea5b723b43a6647
>   - http://caveaudelteatro.it/ser0410[.]bin
>   - ser0410.bin
>   - myexternalip.com - 78.47.139[.]102; 82.214.141[.]134, 86.61.160[.]50 (non-exhaustive)

Exercise 2:

Incident Report 02

The brief attached to the second investigation reads:

> *Figure out which email attachment infected the computer at* `192.168.1[.]95` *. Please provide an appropriate incident report.*

You've been provided with a network capture file and three (3) potentially malicious emails:

```
PATH:   ./02/network.pcap
SIZE:   30,004,191 bytes
SHA1:   911992c14bd721aa72962d89cea1bd58bbfff1e4
MD5:    6f333d25376ed6962128d122eabbfef2
```

```
PATH:   ./02/email_01.eml
SIZE:   2,354 bytes
SHA1:   6c1df5d70e9460fa5f98c3b3c990a450774b39d5
MD5:    70594131076a658157484cee0c441594
```

```
PATH:   ./02/email_02.eml
SIZE:   14,321 bytes
SHA1:   13fe642585defdfafec3fe155e0bf8772773ffc2
MD5:    b35989d66f56aca662ef924e4a8a22b2
```

```
PATH:   ./02/email_03.eml
SIZE:   429,682 bytes
SHA1:   10170bb99676329058524310fd25b79843221de7
MD5:    3f73339cb7f71bdd546767efea1592f5
```

To complete this task, write an incident report that includes:

- The datetime of the infection
- Which malware was involved
- The likely source of the infection
- Any indicators associated with the infection (IP addresses, domains, URLs, and file hashes, if any)

> ▼ **Hint**
>
> Start by looking at the contents of the emails:
>
> ```
> analyst@forensics~$ less email_01.eml
> ```
>
> See anything suspicious? If you find anything, you can extract it using `munpack`:
>
> ```
> analyst@forensics~$ munpack email_01.eml
> ```
>
> Use what you learned in the previous exercise to discover more information.

> ▼ **Hint**
>
> If you identified malware in `email_03.eml` (for example), you should search for its hash in VirusTotal.
>
> Then, check for interesting IPs/URLs. You could use those indicators to pivot and search within the `network.pcap` file.

(They're not actually in there, which is good, but they could be.) Maybe repeat this for the other emails?

▼ **Solution**

For more information and the solution, check the solution video.

▼ **Spoilers!**

1. **The datetime of the infection:** 11/08/2018 05:20:49 (most likely - packet 7256)
2. **Which malware was involved:** Marap
3. **The likely source of the infection:** infected email attachment - PIC35793.iqy
4. **Any indicators associated with the infection:**
   - http://r53x.com/1[.]zip
   - http://r53x.com/a3[.]dat
   - **MD5:** edca225c56f4ee6c46a0ea6463740472 AWBXRECEIPTS_PDF.rar
   - **MD5:** e96b1418314fe28dd5423144f756b7a3 PIC35793.iqy
   - 131.186.113[.]70, 162.88.193[.]70, 94.108.81[.]71, 185.68.93[.]18, 89.223.92[.]202 (non-exhaustive)

Exercise 3: Incident Report 03

The brief attached to the final investigation reads:

> *You've received alerts of bittorrent traffic from* `10.0.0[.]201` *on your organisation's network. Torrent traffic is often associated with file sharing of copyright-protected content; however, many cases of torrent traffic are perfectly legal. Please provide an appropriate incident report.*

You've been provided with a single network capture file:

```
PATH:   ./03/network.pcap
SIZE:   10,135,179 bytes
SHA1:   785739b4582650b3071da687b59e45edfdc952a6
MD5:    5dc1f1b22d3f20d7768692dfe54cb218
```

To complete this task, write an incident report that includes:

- The MAC address of the computer at `10.0.0[.]201`
- The hostname of the computer at `10.0.0[.]201`
- The Windows user account name for the computer at `10.0.0[.]201`
- The Microsoft Windows version of the computer at `10.0.0[.]201`
- The time the torrent activity started (in UTC)
- The file the user downloaded
- The name of the torrent client used
- The file that was being seeded (shared) by the torrent client

> ▼ **Hint**
>
> Take what you learned from the previous exercises and use Wireshark to identify the details of the computer.
>
> Think of the protocols that might be of use here: NBNS, DNS, Kerberos, etc.

▼ **Hint**

One place the Windows version is exposed is in HTTP.

Look for user-agent strings in the HTTP requests from `10.0.0[.]201`

▼ **Hint**

You can easily review BitTorrent traffic by using the `Statistics -> Protocol Hierarchy Statistics` pane again and filtering on BitTorrent.

BitTorrent works by connecting to a `tracker` which serves files by a torrent swarm (torrent file/client users). You can filter on the `scrape` requests in Wireshark to find the file of interest.

This is also likely where you'll uncover the remaining evidence.

▼ **Solution**

For more information and the solution, check the solution video.

▼ **Spoilers!**

1. **MAC address:** 00:16:17:18:66:C8
2. **Hostname:** blanco-desktop
3. **User:** elmer.blanco
4. **Windows version:** Windows 10
5. **Time the torrent activity started (in UTC):** 15/07/2018 04:17:37
6. **The file the user downloaded:** Betty_Boop_Rhythm_on_the_Reservation.avi
7. **Torrent client used:** Deluge 1.3.15
8. **File being seeded by the torrent client:** Betty_Boop_Rhythm_on_the_Reservation.avi.torrent

The "Friday Afternoon Special"

It’s only a few minutes until knock off time. You’re pretty pleased with yourself knowing that you took a decent chunk out of the backlog of work that nobody ever wants to do. However, as you start to pack up your desk and get ready to leave, you notice a potential kerfuffle brewing in your boss’s office. Unable to help yourself, you decide to see what all the commotion is about. Even worse, once you arrive at the office door you open your mouth and ask: *“Is there something I can do to help?”*

Now that you have some runs on the board, your boss has decided to entrust you with a real cybersecurity investigation.

Your company has been contracted to perform an investigation into the compromise of a host from a middle-tier technology company. The customer informs you that they were not aware that they had a problem until they received a heavily redacted tipper from a government Computer Emergency Response Team (CERT). The government CERT informed the customer that they observed one of the customer’s IP addresses connect to a known malicious command and control (C2) operational relay box (ORB). Unfortunately, the CERT was not forthcoming with any additional context or intelligence (go figure).

It's very early on in the investigation and little data is available for

analysis. Fortunately, the customer taps network traffic at certain points in their network and has provided you with a forty-five-minute network capture. Hopefully the malicious activity is in there, fingers crossed.

The customer has no idea which host was compromised, so they want you to find out the following as quickly as possible so they can locate the affected host and inform their SOC.

The customer wants you to answer the following questions in a written report:

1. Which computer was compromised?
2. Where did the malicious activity originate from?
3. What malicious network activity was observed?

Evidence

For this exercise, you've been provided with one (1) network capture file, which is all the evidence you require to complete your investigation.

Before you start, verify that your evidence is not corrupt. This ensures that you don't waste your time and effort troubleshooting data that is not working as expected.

The pertinent evidence metadata is as follows:

```
PATH:   ./victim/traffic.pcap
SIZE:   1,016,208,216 bytes
SHA1:   9ce9aca62c39706671b050b15e75eae9b7d233ac
MD5:    81a14a4c014afb0e1b98f48614709092
```

Things to Remember

During your investigation, remember to take lots of notes and document everything that you find. Doing so will make your life

significantly easier as you start to bring together your smaller analytical discoveries into a larger picture and will prevent you from questioning or repeating analysis. Most importantly, doing so will make your task of compiling the final report much easier.

> **▼ Hint**
>
> Given the size of the data (more than 400,000 packets!), try starting with either the `Statistics -> Protocol Hierarchy Statistics` or `Statistics -> Conversations` pane. Triage the data like we talked about in module 2, identifying the traffic most likely to be of interest, then dig into that.

> **▼ Hint**
>
> You may have noticed there seems to be a lot of DNS and HTTP traffic. That could be a good place to start.
>
> What's of particular interest, though, is the random `Data` protocol that shows up in the `Protocol Hierarchy Statistics` pane. It's not categorised, and makes up about half of the data. That's worthy of investigation.

> **▼ Hint**
>
> Looking at the `Data` packets, they seem like they should be encrypted SSL, because they're going to/from port 443. However, if you `Follow TCP Stream`, the data is actually plaintext.
>
> Reviewing the TCP stream, it's clear this is some kind of C2 (command & control) data. We can observe the adversary sending bash commands like `whoami /all` between `10.2.0[.]2` and `10.1.0[.]2`

> **▼ Hint**
>
> As in the previous exercises, you can further analyse the data by exporting malicious scripts from the network traffic from Wireshark with `File -> Export Objects` and using VirusTotal to gather additional information.

> ▼ **Hint**
>
> If you discover any base64 encoded strings, you can decode them using: `echo '**insert string here**' | base64 -d`

> ▼ **Solution**
>
> For more information and the solution, check the solution video.

> ▼ **Spoilers!**
>
> 1. **Which computer was compromised?** 10.2.0[.]2
> 2. **Where did the malicious activity originate from?** 10.1.0[.]2
> 3. **What malicious network activity was observed?** User enumeration, privilege escalation, account creation, PowerShell use, file download from http://pastebin[.]com/ and https://the.earth[.]li/, reverse shell