



WHY CARBON BLACK ▾

PRODUCTS ▾

SOLUTIONS ▾

PARTNERS ▾

RESOURCES ▾

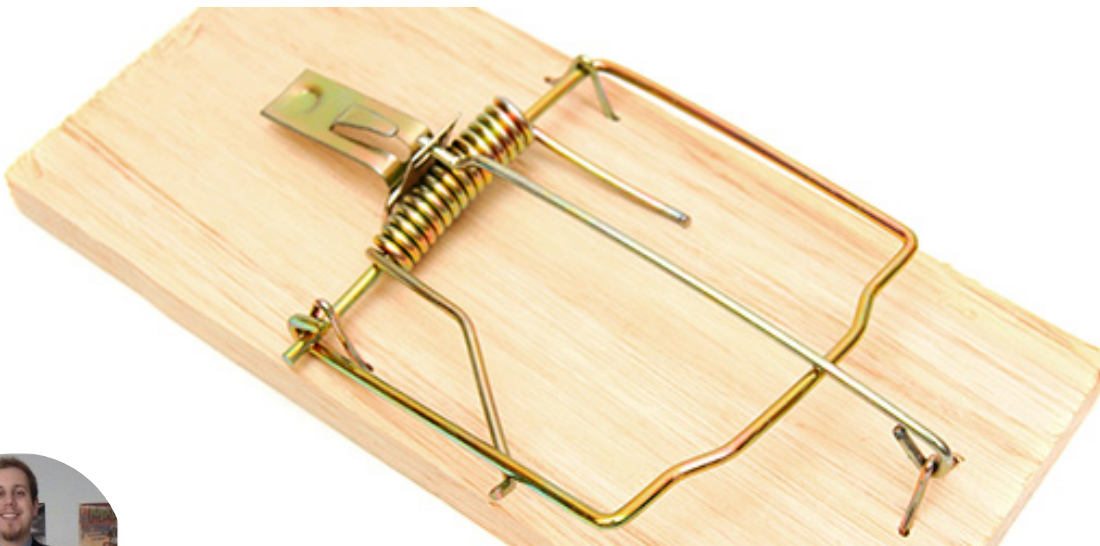
COMPANY ▾

BLOG



Defuse Booby-Trapped Malware by Catching it, Stopping it and Killing it with Fire

26
AUG

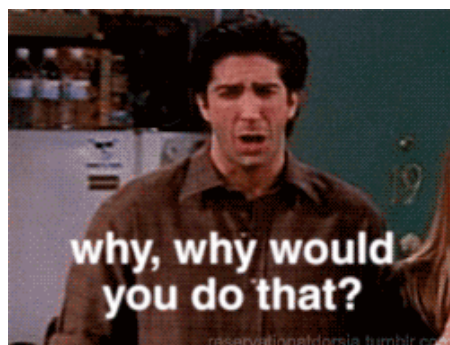


Defuse Booby-Trapped Malware by Catching it, Stopping it and Killing it with Fire

August 26, 2015 / Ryan Nolette / Advanced Threat Protection, Detection and Response, Endpoint and Server Security, Prevention, Response, Tech Toolbox

Recently, a coworker and I were discussing customer concerns. During this conversation they brought up

that a potential customer had been hit with self-destructive malware multiple times. This malware, when found via their intrusion prevention system (IPS), had its connection to the command and control servers (C&C) blocked, and would reformat the system's hard drive. That's not good.



But this got me thinking: "What could I do to stop that from happening?"

I started making a list:

1. Pretend I never heard this and walk away.
– *Yeah, right. Now I'm curious.*
2. Tell them not to block C&C traffic at their perimeter.
– *I'm not sure the company's executive team would totally agree with "let's just wait and see what happens."*
3. Catch it, stop it, and kill it with fire.
– *Yes, this option will do nicely.*



So how would I do this?

My first answer would be to run Bit9 on the system in high-enforcement mode. This would prevent

unapproved binaries from running in the first place, which would avoid this whole issue.

However, let's go down the road of the company that is not running Bitg:

The questions we need to answer are:

1. How do I detect this behavior?
2. How do I remediate the threat?
3. How do I do this without the malware reformatting the system?
4. How do I make this repeatable and easy to modify for new threats?

The quick answer to all of these questions is to run Carbon Black. Carbon Black will give us the visibility we need to detect this behavior and with Carbon Black adapting some of Bitg's key features, we can remediate the threat within this tool without the malware reformatting the system. We can also automate these steps to be easily repeatable and modularly adaptable for new threats.

Detecting the behavior

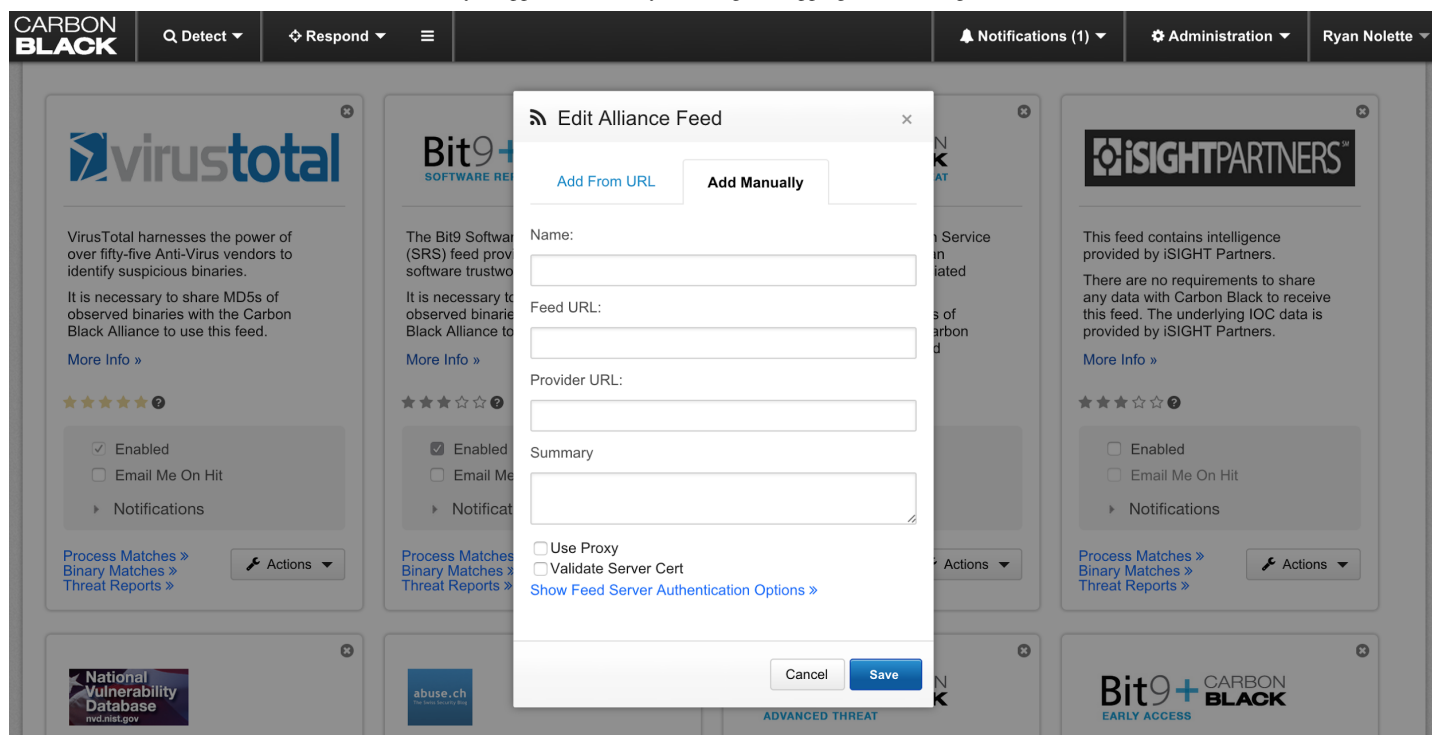
How do I get a list of known C&C servers?

- <https://zeustracker.abuse.ch/blocklist.php>
- <http://www.spamhaus.org/>
- <https://www.shadowserver.org/wiki/pmwiki.php/Stats/BotnetCharts>
- <https://mxtoolbox.com/problem/blacklist/>


These are a few examples of open-source intelligence that you can use to create your own feeds in Carbon Black to alert you of any systems in your environment that are reaching out. The primary data you are pulling from these sources is a list of IP addresses, but you could also use many different kinds of input for feeds. (Here are some [details about creating feeds tailored to your environment.](#))

MORE: [How do I use this information?](#)

The link above is an example script that generates a CB feed from a list of data. This feed can then be imported into the console.



You can see that in the bottom left corner of the "add feed" dialog box that I have already added the Zeus tracker that I referenced above. Now, let's make this data into actionable information.



abuse.ch tracks C&C servers for Zeus, SpyEye and Palevo malware. This feed combines the three domain names blocklists.

There are no requirements to share any data to receive this feed.

[More Info »](#)

★★★★☆?

☒ Enabled
☐ Email Me On Hit
▼ Notifications

On Hit

☐ Create Alert
☐ Log to Syslog

First, I enable the feed. I also select the “log to syslog” option because my Carbon Black data is already going to my SIEM so I can create alerts in there and use my pre-existing workflow. SIEM technology provides real-time analysis of security alerts generated by network hardware, endpoints and applications. If I don't have a SIEM or don't want to use one, I also could create alerts in the CB console (requires

checking of console for alerts) or I can have CB send me an email for every hit. I would personally choose the email option if I didn't have a SIEM so I can be proactively alerted on hits regardless of the time or my location.

Sample malware used:

<https://www.virustotal.com/en/file/2f1d465471ab8a40fa14465f9b3aadf3b58a6968ada2c787cb188ca83f526dab/analysis/>

This file was downloaded via a malicious email I received to an email server I have that only exists to catch stuff like this. The email has a zip attachment that could be decompressed into a scr (the SCR file type is primarily associated with 'Script') file. When I opened this file up on my VM to infect it, the sample started beaconing home.

After a few minutes of beaconing home, the malware pulled down a new binary called "sdf2wd.exe" along with a few other binaries and text files that were probably also malware, but I didn't bother checking since I had what I wanted. This binary ended up being our suicidal malware. To confirm this, I turned off the VMware network adapter and waited. Nothing happened. So I went home for the night and when I came in the following morning, the image had been nuked by the malware. Looking at my logs it looked like the malware waited about 10 hours to format the hard drive.

This confirms that my feed works. It alerted me to a binary on a system that was beaconing home to a domain the list knew about. I could duplicate these results with different samples and lists and maintain similar results each time.

Now how do I remediate without getting nuked?

The simple answer is: Carbon Black

Process:

1. I get an email alert from CB telling me that a binary on a system in my environment is beaconing out to a known C&C server.
2. I open up the email and click the link to take me to the report of the event.
3. The report shows me both the binary information and its parents and children files and processes.
 - a. Grandparent
 - i. outlook.exe
 - b. Parent
 - i. invoice.scr
 - c. Binary
 - i. sdf2wd.exe
 - d. Children
 - i. sdf2wd.txt
 - ii. A copy of sdf2wd.exe in the startup programs folder
 - iii. A registry modifications to the run and runonce keys for sdf2wd.exe
 - iv. A new binary s8df3q.exe created in the roaming file
4. Now that I have this infection scoped I can start on remediation
5. I ban each of the hashes of these files using the CB ban function

**94735AE578B1B35378AF3AFD99598C4C****Seen as:** invoice-186591275-481264.scr**First seen at:** 2014-06-26T20:10:29.117Z (about 1 year)**Status:** **Unsigned****Publisher Name:**[Ban this hash](#)

6. Log into host via CB live response and get some process information for the know files we know are related to the malware.

Cb Live Commands

```

archive      Get an archive (gzip tarball) of all the session data for this session.
argparse     Test parsing of CLI arguments.
cd           Change the current working directory.
clear        Clear the console screen. The "cls" command can also be used for this purpose.
delete       Delete a specific file.
detach       Detach from the current Cb Live session.
dir          Return a list of files in the specified directory.
drives       List available drives on the current remote host (Windows hosts only).
exec         Execute a background process on the current remote host.
execfg       Execute a process on the current remote host and return stdout/stderr.
files        Perform actions on cache-stored session files.
get          Download the specified file from the remote host to the local host.
help         View Cb Live command reference.
hexdump      Output the first 50 bytes of a file, in hexdump format
kill         Terminate the specified process on the current remote host.
memdump      Store the contents of the sensor machine's memory in a file at the specified location.
mkdir        Make a remote directory.
ps           Get a list of active processes from the current remote host.
put          Upload a local file to a specified path on the sensor machine. User will be prompted
to select a file using a dialog box.
pwd          Print the current working directory.
reg          View / modify Windows registry settings.

Use "help [cmd]" to get more details for any command.

```

Below is a sample of that output:

```

[MASTER] C:\Windows\CarbonBlack> ps
4      Analyze NT AUTHORITY\SYSTEM c:\windows\system32\ntoskrnl.exe
328    Analyze NT AUTHORITY\SYSTEM c:\windows\system32\smss.exe
432    Analyze NT AUTHORITY\SYSTEM c:\windows\system32\csrss.exe
484    Analyze NT AUTHORITY\SYSTEM c:\windows\system32\wininit.exe
496    Analyze NT AUTHORITY\SYSTEM c:\windows\system32\csrss.exe
544    Analyze NT AUTHORITY\SYSTEM c:\windows\system32\services.exe
568    Analyze NT AUTHORITY\SYSTEM c:\windows\system32\winlogon.exe
580    Analyze NT AUTHORITY\SYSTEM c:\windows\system32\lsass.exe
1440   Analyze DOMAIN1\master      c:\users\master\appdata\roaming\sdf2wd.exe

```

7. Kill the active processes that I don't want running
 - a. kill 1440 (kill command and PID of the process we want to kill)
 - b. repeat for other files
8. Remove known bad files from system
 - a. delete c:\users\master\appdata\roaming\sdf2wd.exe (delete command and path to file we want to delete)
 - b. repeat for other files
9. Remove the added registry values with the CBLR reg command
10. Get on with my day

NOTE: With the most recent release of Carbon Black, once you complete step five of banning the hashes – you can skip to step seven as it is not needed. What happens under the hood is:

1. At the next check-in the sensor will receive the updated list of banned hashes

2. It will then compare the hash to all running processes and kill any process that was spawned by that hash
3. It will then prevent that executable from MD5 from running again



Until next time, remember my motto: "Flag it, Tag it and Bag it."

**Tags:**

bit9

booby trapped malware

Carbon Black

endpoint

endpoint security

malware

More Posts

