Global Information Assurance Certification Paper

Automated Analysis of “abuse” mailbox for employees with the help of Malzoo

GIAC (GREM) Gold Certification

Author: Niels Heijmans, niels_heijmans@icloud.com

Advisor: Richard Carbone

Accepted: August 20, 2016

Abstract

For most companies, e-mail is still the main form of communication, both internally and with customers. Unfortunately, e-mail is also used heavily by cyber criminals in the form of spam, phishing, spear-phishing, fraud or to deliver malicious software. Employees receive these kinds of messages on a daily basis, even though strict security measures are implemented. Sometimes an employee will fall for the scam but often they will know when it is a false e-mail, especially after good awareness programs. Instead of letting them delete the e-mail, let them share it with you to learn and see what is coming through your security measures or what employees see as “fishy”. But what should you do with the e-mails that are forwarded to this special “abuse” mailbox? Malzoo can be used to analyze this mailbox by picking up the e-mails, parsing them and sharing the results with the CERT team. By using the collected data, you can find new spam runs, update spam filters, receive new malware and learn in what parts of the company awareness is highest (and lowest). This paper explains the benefits and drawbacks of letting employees have a central point to report suspicious e-mail and how Malzoo can be used to automate the analysis.

1. Introduction

1.1. Context

E-mail has been around for thirty-eight years (Peter, 2004) and has since grown to be the predominant method of communication on the Internet. Research by The Radicati Group shows that in 2015 nearly 2.6 billion users were using e-mail and that this number will grow to 2.9 billion in 2019 (THE RADICATI GROUP, INC., 2015). This is also the case for businesses and their employees. A study from 2010 on workplace communication by Paytronics found that 83% of U.S. knowledge workers felt e-mail was critical to their success and productivity at work (Malik, 2010). This is because e-mail is easy to use, has low costs to setup and maintain and allows employees to communicate with other employees, customers or suppliers at any time of the day, around the globe. Unfortunately, it is also misused for spam, (spear) phishing and delivering malware.

The yearly Internet Security Threat Report of 2016 by Symantec reveals that "more than half of inbound business e-mail traffic was spam in 2015" (Symantec, 2016). In the same report, Symantec has seen an increase in phishing e-mails that try to lure the victim into performing actions. According to the report, 1 in 1846 e-mails is a phishing e-mail (spear-phishing not included). These have led to the loss of millions of dollars for companies. Phishing e-mails are also increasingly sophisticated and specialized for specific groups of users. This technique is used for delivering malware via e-mail. The attacker lures the victim into opening an attachment or link that downloads the next phase in the infection chain. Attachments are disguised as fake invoices, resumes or taxes. They also use different file types, with different next phase steps. According to the same report of Symantec, the top three file types used are Office Word (DOC+DOCX), Excel (XLS-XLSX) and ZIP archives, and 1 of every 220 e-mails contains malware.

Many businesses are a victim of digital attacks and it is a false assumption that technology can provide automatic protection for these problems (KPMG, 2014). This is why it is important that employees learn the difference between a legitimate and malicious e-mail, but even more important for the CERT team to learn what the employees consider suspicious e-mail. A centralized mailbox for employees where they can report suspicious e-mails can help to solve this. These e-mails can provide valuable

insights for the CERT team, but they all have to be analyzed accordingly. We will do this by using Malzoo.

1.2. Malzoo

Malzoo was initially built during a graduation project to analyze malware samples of the Portable Executable (Microsoft, 1994) 32-bits file type. The goal was to extract information from samples including packer, compile time, compile language and imphash[1] so that these values can be used to correlate samples.

As input, the samples of the VirusShare project[2] were used to serve as a starting point. These samples sets were analyzed in approximately 6-7 hours, depending on hardware resources. After the graduation, the project was put online at GitHub and has been developed over time to analyze office files and e-mails.

With the same approach as the Portable Executable files, the extracted information of other file types can be used to correlate and serve as input for Malzoo. The development of Malzoo will bring more file type support so that these can be analyzed and extract specific information, resulting in better correlations. Currently, the following file types are analyzed: Portable Executable 32-bits, Word, Excel and PowerPoint files (DOC,DOCX,XLS,XLSX,PPT,PPTX), ZIP archives and e-mails that can be retrieved in the format specified by RFC822 (Resnick, 2001)

Malzoo can be of great value to any CERT team that wants to analyze a lot of samples and e-mails automatically and collect information about these samples, perform dynamic malware analysis with the help of Cuckoo and store and manage the collected samples in the Viper framework. See Section 1.4 for a description of Viper.

1.3. Why build a new tool?

The reason Malzoo was developed following the initial research during the internship was because there were no open source solutions scalable enough to handle the amount of samples that was envisioned (50 GB+). There were projects that inspired the Malzoo project that are open source. Malzoo works by having different processes that

[1] https://www.fireeye.com/blog/threat-research/2014/01/tracking-malware-import-hashing.html.

[2] http://www.virusshare.com.

have a specific role and executes separately from the rest. The setup of Malzoo also gives flexibility to quickly add specific features, for example, new file types or another input source. The only thing that needs to be set correctly is the queue, which is used by the new feature and the communication is fixed.

1.4. Previous work and solutions

CuckooMX[3] submits attachments to Cuckoo, but no analysis of static e-mail information is collected. Malzoo supports the submission of malware samples to the Cuckoo sandbox for dynamic analysis, as the focus of Malzoo is static malware analysis.

The Viper Framework[4] has been an inspiration source of Malzoo, but designed for manual analysis by an analyst. Viper does support an API that can be used by Malzoo to submit samples to Viper. The Viper Framework and Malzoo work great together, where Malzoo can be used as the automated collector, preliminary analysis tool and for distributing samples or checking for duplicates in Viper.

Mastiff [5]is a static analysis framework that automates the extraction of files when manually running the tool. This tool comes the closest to Malzoo in comparison as it focusses on static malware analysis. Differences are that Mastiff does not run as a service and does not have an API or e-mail support

There are great tools available in the open source community that supports the analysis of samples by an analyst. The goal with Malzoo is to contribute to the community and automate this process for analysts, using an e-mail mailbox as a source for samples and extra context, powered by users.

[3] https://tribalchicken.com.au/technical/automated-mail-server-cuckoo-analysis-v2-0/.

[4] http://www.viper.li.

[5] https://git.korelogic.com.

2. Analyzing e-mails

2.1. Prerequisites

To start analyzing e-mails from employees, the first step to success is getting permission by management to setup this process. After that the "abuse" mailbox will need to be setup within the company. The technical aspects for this mailbox are discussed in detail in Section 3.1. Once the mailbox is operational, it needs to be promoted to the business. The focus of this promotion is on the user. It clarifies why it is important to report all suspicious e-mails and what the CERT team does with them. Another key item to add is that the analysis process is automated and that the CERT team will be contacted only if follow up is needed. This can be the case when the user sends a malicious file attached and the analyst wants to check if the user has opened the file or had indicators that the user might be infected.

This promotion and communication helps get employees reporting e-mails they do not trust. To be able to analyze the e-mails automatically with Malzoo, the mailbox needs to be accessible via the IMAP protocol so that Malzoo can retrieve the e-mails that are new. The last pre-requisite for this solution is a Mongo database or an event logging system for Malzoo to send its results to for the analyst to analyze. In this paper, we will do so with Splunk[6].

2.2. Advantages

With this solution in place, the malware analyst(s) can receive great intelligence regarding the business' IT security. The e-mails that are received reveal insights into what employees find suspicious. This can help to improve the awareness trainings that are provided within the company. If a test is performed with fake phishing e-mails, it will show how many people will report this, against the people that may fall for the fake scam. The e-mails that are reported can also show what gets through the digital defenses that are in place, like the mail gateway, antivirus and firewall. Based on the gathered intelligence, these security measures can be tweaked and updated to stop future attacks.

[6] http://www.splunk.com.

And lastly, the analysis of received e-mails can reveal new malware specimens and Indicators of Compromise (IOC).

2.3. Challenges

As with every automated process, there are challenges to overcome to make sure it is and stays a success. One of these challenges is the other side of the coin when discussing promotion of the abuse mailbox. The promotion should not scare employees and make them paranoid, resulting in sending every e-mail to the mailbox, which results in a lot of overhead.

Another challenge is to let the employees report their suspicious e-mails as an attachment, rather than forwarding the e-mail which results in loss of valuable information. Something that can aid with this is to add a "Report" button to the mail client used by the business. This button can be configured to automatically attach the selected e-mail to a new e-mail, with the receiving e-mail address already filled in. A good example on how this is implemented in Outlook is explained on Nerdosaur (Nerdosaur, 2015). This will not be discussed further as it outside the scope of this paper.

The last challenge is management. They have to be convinced that an "abuse" mailbox will help the security of the business. Once convinced, questions may rise about statistics and results. The analyst should be ready to answer these questions beforehand, to be able to deliver the answer to such requests.

3. Technical setup

3.1. Abuse mailbox

Setting up the mailbox begins with a big step, which is coming up with a conventional name for it. The name of this mailbox will be the reference in promoting campaigns and the e-mail address that users will use when reporting e-mails. A few examples of mailbox names are abuse@example.org, false-e-mail@example.org or suspicious-e-mail@example.org. Be careful with names that have specific names of attacks, e.g. phishing@example.org or spam@example.org. Users may think that they

are only allowed to send that specific e-mail to the mailbox and will not share the potentially malicious e-mail.

Malzoo connects to the e-mail server via the IMAP protocol and will need a username and password for authentication. Make sure to have these credentials when starting the process. Because there are many different e-mail servers and ways of hosting, we will not go in-depth on how to configure this.

3.2. Malzoo

3.2.1. Architecture

Malzoo is written in the Python programming language and is designed with the object-oriented style. The architecture is visualized in Figure 1.

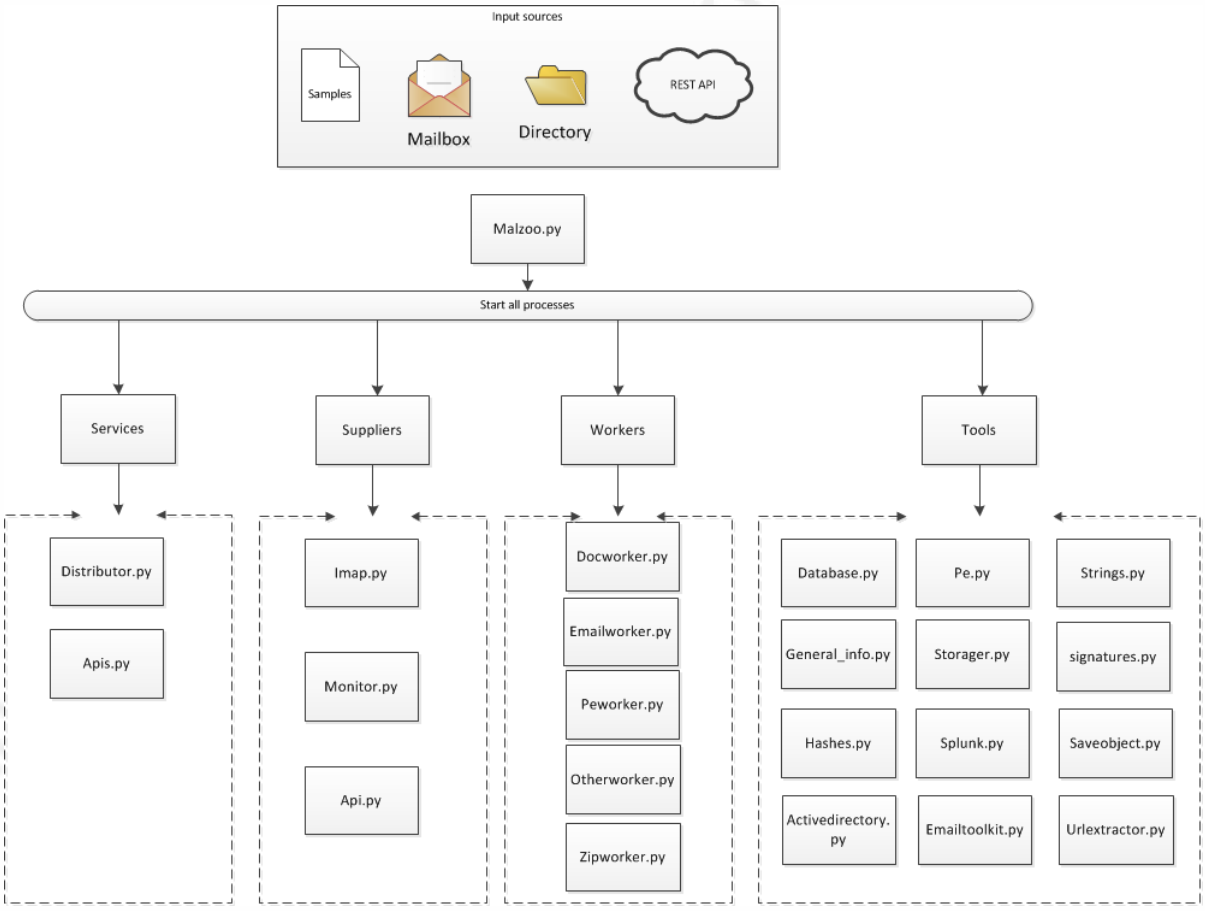

Figure 1: Malzoo architecture (created with Visio).

Malzoo runs on the Linux/UNIX operating systems that support Python 2.7 and the Mongo database. The operating system (OS) Malzoo is developed and tested on is Ubuntu Server 14.04 (LTS).

The communication between the suppliers, services and workers is handled via queues. Each worker has a queue that it checks for new "assignments". The distributor receives new samples in the distribution queue by the suppliers and workers then extract sub files (attachments and files in ZIP archive). The communication approach is shown in Figure 2.

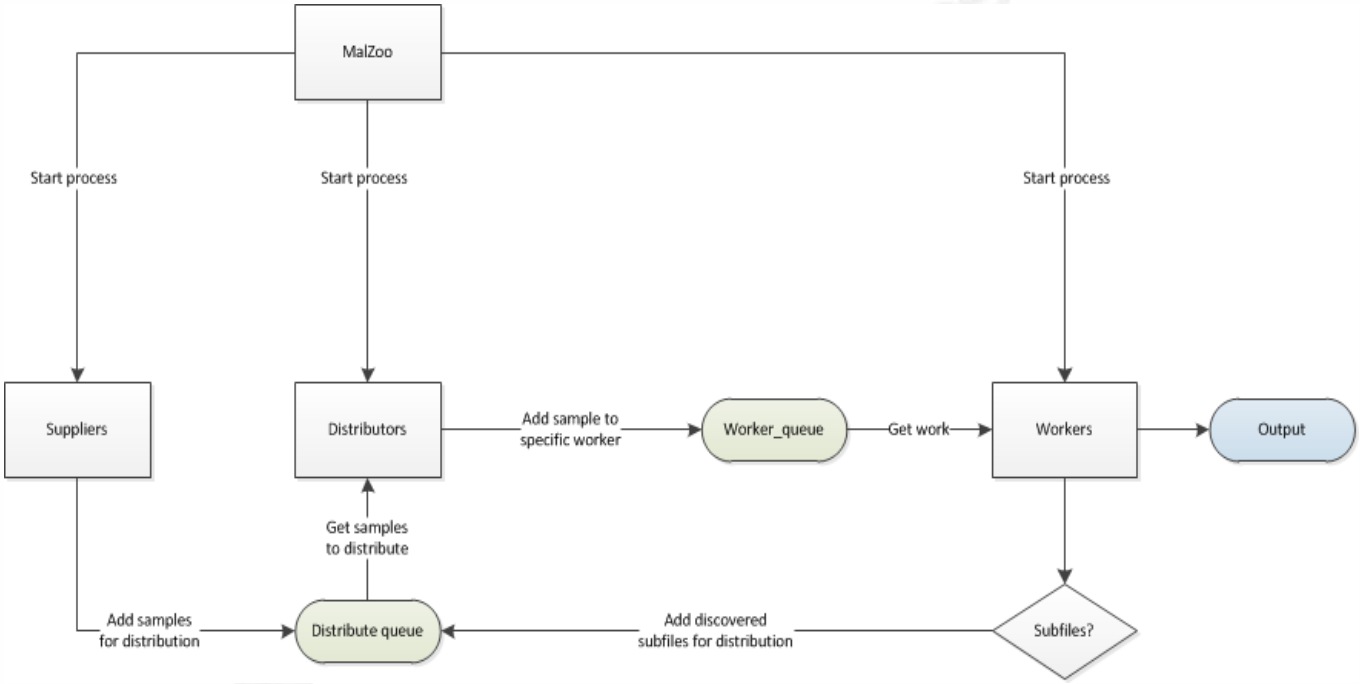

Figure 2: Malzoo communication between processes.

3.2.2. Installation

For Malzoo to run, it will need some dependencies fulfilled in order to run properly. See Table 1 for an overview of requirements.

Table 1: Malzoo requirements.

Software	Description
Mongo database	Database for storing the collected information of the samples.

YARA	A tool to identify and classify malware based on textual or binary patterns. Each description (rule), consists of a set of strings and a Boolean expression which determine its logic. (YARA, 2016).
Python-magic	Python script used to identify the filetype of a sample.
Pydeep	Used to generate the fuzzy hash of a sample with the SSDeep library.
Requirements.txt	A list of Python dependencies for Malzoo.

To make the installation faster or to set up a quick test environment, Vagrant can be used to deploy a virtual machine with Malzoo installed. The Vagrant file can be found on the GitHub page of the project. Another option is to use a bootstrap file to install Malzoo automatically on an existing Linux system.

3.2.3. Configuration

User specific settings, like usernames and passwords, can be set in the configuration file *malzoo.conf*, which can be found in the *config/* folder. The configuration file helps to have all user specific settings in one place. It has a number of sections that help configure Malzoo. The sections are

- settings
- suppliers
- mongo
- imap
- splunk
- malzoo
- cuckoo
- viper

The section settings can be used to decide if the sample should be saved in the storage folder and what tool should be used to check if a sample was already submitted

(Mongo or Viper). It also contains the whitelist of extensions that Malzoo should not submit to Cuckoo and/or Viper. The location of the YARA rules, packer signatures, samples and the directory that should be monitored for new samples can be configured here.

The suppliers section lets the user decide which sample suppliers should be started. Multiple suppliers can be enabled, if desired.

The Mongo section can be used to enable storing results in the defined database and collection.

The IMAP section stores the credentials for the abuse mailbox, the e-mail server and the folder to look in for the e-mails.

The Splunk section is used to enable or disable the storing of results and to set the address, which can be a domain name or IP address. It also is used to set the token for the HTTP Event Collector of Splunk[7].

The Malzoo section is used to configure the host and port Malzoo should start the API supplier.

The last two sections can be used to configure Cuckoo and Viper. It lets the user configure if the sample should be submitted to Cuckoo and/or Viper and on what destination IP address and port Cuckoo and/or Viper are running.

3.2.4. Data collected

Malzoo collects information from all samples it receives. For files that have been designed workers for with Malzoo, it extracts specific data from that file type. The file types with a worker designed for it at this moment are Portable Executable 32-bit, Microsoft Office Documents and ZIP archives. From other file types Malzoo collects a general set of data. It is possible to create new workers for new file types by creating a worker of the superclass *Worker* in the folder *malzoo/modules/worker/<worker>.py*. The worker must be imported in the main script *malzoo.py* and started in a separate process, receiving the distribute queue.

[7] http://dev.splunk.com/view/event-collector/SP-CAAAE6M.

The data that will always be collected is:

- Md5 hash
- Sha1 hash
- File type
- File size
- File name
- YARA rules
- Submit date in epoch time
- ID tag

From Portable Executable files the following data is collected:

- PE hash
- Import hash
- Fuzzy hash
- Compile time
- Imported DLL's
- Packer used (if applicable)
- PE Language
- Original filename

From Microsoft office documents the indicators are extracted with *oletools*[8]. This extracts three types of information: suspicious keywords, Indicators of Compromise (IOC) and AutoExec. For each type a keyword that triggered the categorization will be given, as well a small description why the keyword was extracted.

[8] http://www.decalage.info/python/oletools.

4. Analyzing results

For this research, a data set of one month was selected to demonstrate how the results can help the operational team. The month of May of 2016 (1-31) will be used to demonstrate the insights an abuse mailbox can provide and how it can be made actionable. To put the numbers in context, the mailbox was promoted within the company mainly via e-learnings as the final process is being currently set up.

During this period, a total of 900 e-mails were reported to the mailbox. Most e-mails came in on business days (Monday until Friday), as shown in Figure 3.

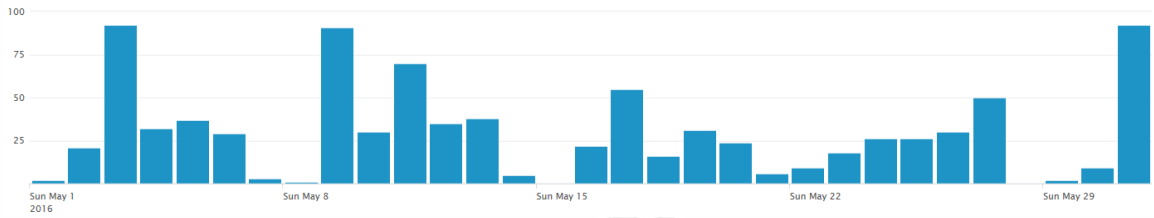

Figure 3: E-mails for May 2016 on a per day basis.

In these e-mails, 277 attachments were delivered, from which 163 were unique. Documents were shared the most (Microsoft Office files and PDF) with 133, followed by 132 images. A complete overview of the total number of samples and unique count by file types are shown in Table 2.

Table 2: Attachment file types, total and unique count.

filetype	TotalCount	UniqueCount
image/jpeg	100	26
application/msword	60	56
application/pdf	38	27
application/vnd.openxmlformats-officedocument.wordprocessingml.document	27	25
image/png	23	11
image/gif	9	4
text/html	7	4
application/vnd.openxmlformats-officedocument.presentationml.presentation	5	3
application/octet-stream	3	3
application/vnd.openxmlformats-officedocument.spreadsheetml.sheet	2	1
application/vnd.ms-excel	1	1
application/x-rar	1	1
application/zip	1	1

As seen in table 2, documents are submitted the most to the mailbox. Malicious documents have seen an increase since the end of 2015 (Proofpoint, 2016). Since they

are very popular with cybercriminals (Rapoza, 2016) this file type will be further investigated.

The hashes were checked by VirusTotal using the mass hash search Intelligence service to give a high level overview of what was shared with the mailbox. See Table 3 for a summary of the results.

Table 3: VirusTotal results.

File type	Total (unique)	Found in VT	Lowest detection	Highest detection
Msword	56	20	0	39
PDF	27	16	0	2
Word 2007+	25	4	1	37
Powerpoint 2007+	3	0	N/A	N/A
Excel 2007+	1	0	N/A	N/A
Msexcel	1	1	36	36

Malzoo collects data from Word, Excel and PowerPoint documents with the help of the *oletools* library. This library collects valuable information for a malware analyst to get a quick triage of a sample. *Oletools* can be used to determine if the sample has macros and what the metadata of the sample contains. It also has a built-in IOC algorithm that can check for suspicious elements that are commonly used by attackers.

A few examples of IOC collected are URL's, executable filenames and suspicious actions like using the Shell library to execute code. Figure 4 shows the results of *oletools*, filtered by the IOC type. The figure shows only the unique values to demonstrate how many different IOC were found.

description	keyword
Executable file name	h.exe
Executable file name	svnhost.exe
Executable file name	kernel32.dll
Executable file name	user32.dll
Executable file name	hs.exe
Executable file name	shell32.dll
Executable file name	urlmon.dll
URL	http://stock.nesthouz.com/system/logs/h.exe
URL	http://papirus.kiev.ua/system/logs/h.exe
URL	http://33.pay-work.ru/system/logs/h.exe
URL	http://123.pay-work.ru/system/logs/h.exe
URL	http://39.pay-work.ru/system/logs/h.exe
URL	http://www.starski.com.ua/system/logs/h.exe
URL	http://www.sweet-puff-pipe.com/system/logs/hs.exe
URL	http://www.ancientherbsfoodproducts.net/system/logs/hs.exe
URL	http://www.distributorhelm.co.id/system/logs/hs.exe
URL	http://natureshavenshop.com/system/logs/hs.exe
URL	http://www.everybitbaby.com/system/logs/hs.exe
URL	http://publimediabajio.com.mx/wp/logo.gif
URL	http://aesthetic-prof.com.ua/system/logs/hs.exe
URL	http://beyoudubai.com/system/logs/hs.exe
URL	http://www.kctw.net/system/logs/hs.exe
URL	http://iwebcart.sourceforge.net/system/logs/hs.exe
URL	http://vikont24.ru/system/logs/hs.exe
URL	http://shlif.kh.ua/system/logs/hs.exe

Figure 4: IOC collected from Microsoft Office documents by Malzoo with oletools.

The above listed IOC can be used in the hunting process, by matching them in proxy logs and identifying what attachments were opened. It also provides a source creating a blacklist so that employees that receive the malicious attachment are not exposed to the next step in the infection chain of the attacker when opening the e-mail later.

Niels Heijmans, niels_heijmans@icloud.com

These results can also be used to correlate samples. In Figure 5, a URL IOC was correlated between all office document samples. Five other samples were found with this IOC.

description	keyword	sha1
URL	http://39.pay-work.ru/system/logs/h.exe	3be69272ea4240512dbc622118bf414d68c115a3 47767abce4349ab4fc15a8af31d4a71eba02d0f8 5a8f061fc6e3760dc46df10a7a9697d1ff682042 5e0b79c14239821a207dd6057a388ea32fee698b d0914978e4f370e45dfea128065e37712a1a4878

Figure 5: Correlated samples based on URL IOC.

A lookup in VirusTotal by SHA-1 hash shows that the most used signature to describe the samples is "W97M.Downloader". When matching this information with the information extracted from the emails by using a join on the message ID of the email, it shows the "from,subject" and filename of the related e-mails as shown in Figure 6. The first part of the filename has been blurred out to preserve the privacy of of the reporting employees, as it contained information that could identify them.

from	subject	filename	keyword
jobs@marksfeedstore.com	Re: formal complaint	[redacted]_complaint521.doc	http://39.pay-work.ru/system/logs/h.exe
jobs@marksfeedstore.com	Re: formal complaint	[redacted]_complaint108.doc	http://39.pay-work.ru/system/logs/h.exe
jobs@marksfeedstore.com	Re: formal complaint	[redacted]_complaint725.doc	http://39.pay-work.ru/system/logs/h.exe
jobs@marksfeedstore.com	Re: formal complaint	[redacted]_complaint157.doc	http://39.pay-work.ru/system/logs/h.exe
jobs@marksfeedstore.com	Re: formal complaint	[redacted]_complaint547.doc	http://39.pay-work.ru/system/logs/h.exe

Figure 6: Correlated emails with the analyzed IOC keyword.

Based on this information, the CERT team can request a new rule in the mail gateway to block all e-mails from a given email address and its subject, optionally with a Word attachment.

5. Conclusion

Giving the opportunity for employees to report e-mails that they deem suspicious gives new insights to the CERT team on some important items. For one, the maturity of the organization in detecting suspicious e-mails can be measured by grading what kind of e-mails is being shared. If only the simple spam e-mails are shared but the phishing e-mails with malware attached keep being opened, then the focus can be placed on this particular topic.

Niels Heijmans, niels_heijmans@icloud.com

Technically, the automated processing of e-mail and attachments aids the CERT team is assessing new trends in spam runs or malware by correlating samples and act on them accordingly by searching against the gathered IOC in different logs or by actively adding the IOC to the security systems and the hunting process.

6. References

KPMG. (2014). *Cyber security: it's not just about technology.* KPMG. Retrieved from https://assets.kpmg.com/content/dam/kpmg/pdf/2014/05/cyber-security-not-just-technology.pdf

Malik, O. (2010, September 22). *Is Email a Curse or a Boon?* Retrieved April 16, 2016, from Gigaom: https://gigaom.com/2010/09/22/is-email-a-curse-or-a-boon/

Microsoft. (1994, March 1). *Peering Inside the PE: A Tour of the Win32 Portable Executable File Format*. Retrieved from Peering Inside the PE: A Tour of the Win32 Portable Executable File Format: https://msdn.microsoft.com/en-us/library/ms809762.aspx

Nerdosaur. (2015, July 16). *Add a Report Phishing Button in Outlook*. Retrieved Augustus 4, 2016, from Nerdosaur: http://www.nerdosaur.com/network-security/add-a-report-phishing-button-in-outlook/.

Peter, I. (2004, 00 00). *The history of email*. Retrieved April 16, 2016, from NetHistory: http://www.nethistory.info/History%20of%20the%20Internet/email.html

Proofpoint. (2016). *The Human Factor.* Proofpoint.

Rapoza, J. (2016, July 22). *The Rise of Document-based Malware - Data Threat Detection and Prevention*. Retrieved from SOPHOS: https://www.sophos.com/en-us/security-news-trends/security-trends/the-rise-of-document-based-malware.aspx

Resnick, P. (2001, April 1). *Internet Message Format*. Retrieved from RFC 2822 - Internet Message Format: https://tools.ietf.org/html/rfc2822

Symantec. (2016, April 27). Internet Security Threat Report. *21*. Symantec.

THE RADICATI GROUP, INC. (2015). *Email Statistics Report, 2015-2019.* THE RADICATI GROUP, A TECHNOLOGY MARKET RESEARCH FIRM. PALO ALTO, CA, USA: Radicati.

Niels Heijmans, niels_heijmans@icloud.com

YARA. (2016, May 4). *Yara - The pattern matching swiss knife for malware researchers*. Retrieved May 7, 2016, from YARA: http://plusvic.github.io/yara/