Exploring Current E-mail Cyber Threats using Authenticated SMTP Honeypot

Lukáš Zobal[1], Dušan Kolář[2] and Jakub Křoustek[3]

[1]*Faculty of Information Technology, Brno University of Technology, Božetěchova 1/2, 612 00 Brno, Czech Republic*

[2]*Faculty of Information Technology, IT4Innovations Centre of Excellence, Brno University of Technology, Božetěchova 1/2, 612 00 Brno, Czech Republic*

[3]*Avast Software s.r.o., Pikrtova 1737/1A, 140 00 Prague, Czech Republic*

Keywords: Spam, Honeypot, SMTP, E-mail, Malware, Cyber Threat Intelligence.

Abstract: Today, spam is a major attack vector hackers use to cause harm. Let it be through phishing or direct malicious attachments, e-mail can be used to steal credentials, distribute malware, or cause other illegal activities. Even nowadays, most users are unaware of such danger, and it is the responsibility of the cybersecurity community to protect them. To do that, we need tools to gain proper threat intelligence in the e-mail cyber landscape. In this work, we show how an e-mail honeypot requiring authentication can be used to monitor current e-mail threats. We study how such honeypot performs in place of an open relay server. The results show this kind of solution provides a powerful tool to collect fresh malicious samples spreading in the wild. We present a framework we built around this solution and show how its users are automatically notified about unknown threats. Further, we perform analysis of the data collected and present a view on the threats spreading in the recent months as captured by this authentication-requiring e-mail honeypot.

1 INTRODUCTION

E-mail has become one of the primary means of communication for people on the Internet. It is used massively both for commercial and personal purposes because of the ease of use and zero cost. Consequently, e-mail has become prone to misuse in the form of spam, also called unsolicited bulk e-mail. According to Symantec, more than 50% of e-mails received in 2018 were spam (O'Gorman et al., 2019). Additionally, according to Verizon's *Data Breach Investigations Report 2018*, more than 92% of the attack vectors use e-mail (Verizon, 2019).

The malicious content in spam ranges from unwanted advertisements, phishing, to Denial-of-Service (DoS) attacks, and spreading malware through URLs and attachments. The impact is also indirect, filling up space and bandwidth of mailing servers and decreasing employees' efficiency while dealing with unwanted e-mails.

Apparently, there is a good reason to monitor and mitigate the current e-mail threats. To do it efficiently, one has to collect data and intelligence about emerging threats. While it's possible to gather data about spam from the real user inboxes directly, this is connected with the complex problem of spam filtering as well as privacy issues.

Honeypots are much more convenient for this purpose. Using a deception strategy, we can gather spam only, without the need to filter through legitimate traffic or to worry about privacy. There are generally two approaches. The first idea is spreading fake e-mail addresses – honeytokens, across the Internet and monitoring incoming e-mails. The second scenario is operating a fake open relay SMTP server, offering adversaries to relay their spam, while collecting the e-mails instead.

1.1 Contribution

In this paper, we offer an alternative to a classical open relay SMTP server. We show that using an authenticated SMTP server, we can collect remarkable threat intelligence data. An open relay server usually captures advertisements and phishing only. With our approach, a significant part of the traffic contains malware-spreading campaigns, including previously unseen families and samples.

Further, we provide an analysis of the collected data using different metrics. Because of the signif-

Zobal, L., Kolář, D. and Křoustek, J.
Exploring Current E-mail Cyber Threats using Authenticated SMTP Honeypot.
DOI: 10.5220/0009591002530262
In *Proceedings of the 17th International Joint Conference on e-Business and Telecommunications (ICETE 2020) - SECRYPT*, pages 253-262
ISBN: 978-989-758-446-6

icant portion of malware in the traffic, we offer detailed insights into the malware families using automatic dynamic analysis sandbox system. Thanks to it, we can correctly label all incoming attachments with their corresponding names and provide a holistic view of the e-mail threats spreading during recent months. We also show how a single malicious campaign can be monitored in real-time using the implemented framework.

1.2 Structure of the Paper

In section 2, works related to e-mail threats analysis as well as honeypots in general, are presented. Section 3 describes architecture of our system. In section 4, we present our findings and results gathered from running the honeypot. These include an analysis of malware families captured as well as an example case study of a single campaign in detail. Section 5 concludes this paper.

2 RELATED WORK

Honeypot is a computing resource value of which lies in being probed, attacked, or compromised (Spitzner, 2003). While the idea of studying attackers instead of removing them from the system came in the late 1980s (Stoll, 2005), the first appearance of the term honeypot was used at the beginning of millennia (Spitzner, 2001).

Since then, hundreds of honeypot tools were created, many of which in collaboration with Honeynet Project[1]. In one of the latest surveys, Nawrocki et al. (2016) provides an extensive list of honeypots in different fields, both historical and current. He also discusses dozens of data categories to analyze from honeypot deployments.

Honeypots have a number of advantages and disadvantages when compared to other cybersecurity tools. As defined by Spitzner (2003), one of the most significant advantages is the unique information honeypots offer – the possibility to watch adversaries in action. Flexibility and simple detection abilities are also key features. On the other hand, the most significant drawback of honeypots is their limited visibility. When the honeypot is not targeted in the attack, it turns out useless. What's worse, it can be used by the adversaries to mislead the defenders.

There are many different approaches to classify honeypots. The most widely used method uses a level of interactivity as a measurement (low/medium/high interaction) (Spitzner, 2003). Other methods include honeypot purpose (research/production), behavior (client/server), form (physical/virtual), or the service they are mimicking (Nawrocki et al., 2016).

Methods of detecting honeypot presence appeared together with honeypots. Krawetz (2004) discussed a simple tool to check the functionality of an e-mail open relay. Zou and Cunningham (2006) presented the idea of detecting honeypots in botnet attacks by testing their ability to spread unmodified malicious payload. Other honeypot detection principles may cover honeypot fingerprinting, e.g. using underlying communication protocol[2], or testing for specific honeypot behaviour[3]. In one of the latest studies, Uitto et al. (2017) surveys anti-honeypot research to date and proposes used detection vectors taxonomy. Honeyscore[4] project makes honeypot detection as easy as looking up its IP address.

On the other hand, honeypots are improving to bypass the detection. Issues are getting fixed and the signatures become dynamic. In a recent study, Tsikerdekis et al. (2018) surveys preventive measures for honeypot detection and discusses possible improvements for the future.

2.1 Spam and E-mail Honeypots

The problem of e-mail misuse in the form of spam is as old as e-mail itself because the issue of spam classification is complex (Brunton, 2013). Solutions for it have evolved throughout the years with the current trend of artificial intelligence and machine learning. In a recent study, Bhowmick and Hazarika (2016) surveys current technologies and trends in this domain. At the same time, adversaries come with new ways to get around spam detection. A botnet is a very powerful tool to surpass any kind of blacklist. Khan et al. (2015) provides a comprehensive study on the problem of spam botnets – a phenomenon we see in our honeypot as well.

However, with honeypots, the problem of spam detection becomes irrelevant, as all the incoming e-mails are considered spam. Therefore, honeypots have become a great tool for spam harvesting and threat intelligence collecting. Oudot (2003) describes how honeypots can be used as open proxies to collect spam. He also mentions the possibility of spreading fake e-mail addresses as honeytokens. Two years

[1] https://www.honeynet.org/

[2] Example of a fingerprinting tool capable of honeypot detection: https://github.com/0x4D31/fatt

[3] Example of an issue used for Cowrie honeypot detection: https://github.com/cowrie/cowrie/issues/512

[4] https://honeyscore.shodan.io/

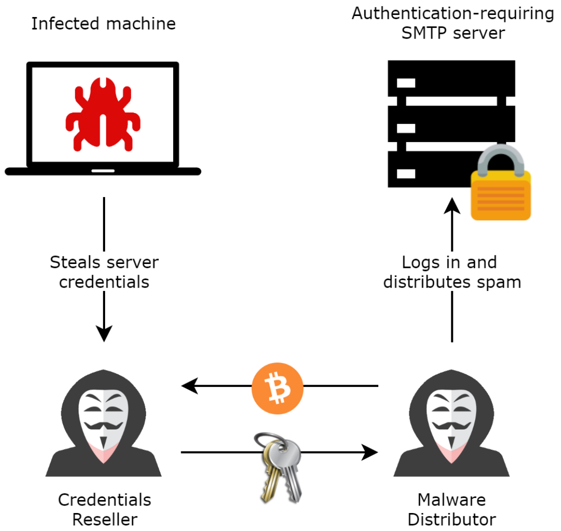
Infected machine
Authentication-requiring SMTP server
Steals server credentials
Logs in and distributes spam
Credentials Reseller
Malware Distributor

Figure 1: Distributing the SMTP server credentials.

later, Andreolini et al. (2005) describes the architecture of a fake open relay honeypot called *HoneySpam*.

Another take on fighting the spam is using greylisting or tarpitting techniques. Practical example is the original tarpitting honeypot LaBrea (Liston, 2003) or a spam deferral daemon spamd[5]. However, these techniques do not provide any additional threat intelligence, as they do not collect any e-mails.

Considering honeypots that collect data, there have been several practical implementations in the past. Jackpot, SMTPot, and Proxypot[6] were all implemented as fake SMTP open relays, which in fact collected incoming e-mails. Neither of these projects is maintained anymore. One of the more recent e-mail honeypots is SHIVA[7], developed as part of Honeynet Project. On top of open relay functionality, it offers the possibility for authentication, which is required by our solution. The main added value of this honeypot is a clustering of incoming e-mails to campaigns using Context Triggered Piecewise Hashing (Kornblum, 2006), also known as *fuzzy hashing*.

Indeed, clustering of incoming spam to campaigns has become an important issue for every tool collecting spam. The *fuzzy hashing* used by SHIVA represents the most straightforward way to cluster e-mails, depending on e-mail body similarity. Calais et al. (2008) proposed a hierarchical way to cluster e-mails to campaigns using FP-Trees (Han et al., 2004). A very similar approach using the Categorical Clustering Tree was introduced by Alishahi et al. (2015). Such an approach proved useful as Dinh et al. (2015) presented a framework for on-the-fly analysis of incoming e-mails, using the FP-Tree approach with slightly modified features. This framework is similar to our architecture, as described in the following section. In his recent Master's Thesis, Smirnov (2018) brings a detailed survey and classification of different spam clustering methods.

Studies mentioned in the previous paragraphs, however, always use open relay servers or "trusted third-party" to obtain the spam data set.

3 ARCHITECTURE

When developing the e-mail honeypot, we aimed to obtain as many malicious e-mails as possible to gain proper cyber threat intelligence. Another target was to gather fresh malicious samples in the form of e-mail attachments and links. However, when operating the open relay SMTP honeypot, we realized the content we are receiving is not representing the real threats. While there were several advertising campaigns, together with general phishing campaigns present, we haven't seen many campaigns with malicious attachments.

There might be several reasons why open relay servers don't relay the threats we were looking for. Arguably, hackers are aware that many of the open relays are, in fact, honeypots, capturing the traffic. Additionally, they might have more reliable ways to distribute spam, e.g. using already infected victims.

In order to gain adversaries' trust, we decided to create an authentication-requiring SMTP server and distribute the credentials directly to the users we were looking for – distributors of the malware. We achieved this by using existing malware samples, in particular, password-stealers and similar spyware, that is able to steal credentials from a machine. This data is then delivered to the malware distributor. Using deception, we were able to trick hackers into believing they acquired a valuable resource to distribute spam.

In practice, the situation is usually more complicated. Typically, there are two different entities – one collecting the credentials and offering them for sale on hidden services[8], and second entity distributing the malware. Also, the malware distributor usually does not use the credentials himself/herself but uses owned botnet. This behavior can be seen in an example study in section 4.2. The described solution of distributing the credentials is depicted in Figure 1.

[5] https://man.openbsd.org/spamd.8

[6] https://www.symantec.com/connect/articles/guide-different-kinds-honeypots

[7] https://github.com/shiva-spampot/shiva

[8] https://krebsonsecurity.com/2017/12/the-market-for-stolen-account-credentials/

In the rest of this chapter, we briefly describe the architecture of our honeypot and solution for collecting and visualizing the extracted knowledge.

3.1 E-mail Gathering

To set up an SMTP server, we used SHIVA honeypot. As described in section 2.1, it is one of the more recently developed e-mail honeypots from the Honeynet Project. It supports both open relay and authenticated mode of relaying. Its advanced feature is the ability to cluster incoming spam to campaigns using body similarity.

While we found this feature useful, we soon realized the lightweight cloud server this honeypot was running on is not powerful enough to both handle incoming traffic and intelligently parse the data and maintain the database. Therefore, we decided to use SHIVA for collecting the spam only and moved the parsing infrastructure to our own servers. This also has the advantage of keeping the public honeypot separated from the rest of the infrastructure. Another advantage is the possibility to run multiple instances of SHIVA while collecting all the data to a centralized server.

The e-mail collecting is achieved with a general infrastructure using MQTT[9] protocol. This enables us to have a heterogeneous infrastructure, possibly with more different honeypots, publishing different payloads, as well as multiple consumers. All this without duplicating the transferred data and the possibility to employ anonymous communication channels easily.

3.2 Data Extraction and Storage

After the batch of spam arrives, each e-mail is parsed and features are extracted. We store a number of features, including the following.

- E-mail subject
- E-mail body, both plain text and HTML
- Headers, including date and recipients
- Attachment information
- URLs and their payload
- Source IP address with geo-location
- Detected language
- Timestamps of campaigns, attachments, URLs, IPs
- SHA256 and ssdeep hash of the file

Before the information is saved to a PostgreSQL database, we compare the ssdeep hash of each e-mail body, used for body similarity clustering, to the rest of the database to check for already existing campaigns. Because this process is performance-intensive and the number of comparisons is significant, we introduced massive parallelization to the process, including several optimizations. The most notable is the process of “pre-clustering”, where we detect the clusters in the batch of incoming e-mails before searching the database. Using this approach, we limit the number of comparisons greatly, and the solution scales well, even with millions of campaigns present in the database.

Additionally, we upload all the malicious attachments to third-party storage from Avast Software for later analysis and labeling. The data, therefore, serves to create better detections and improve user security in the future.

Also, the contents of all the URLs are downloaded. In the case of non-HTML payloads, they are treated as attachments – uploaded to a third-party storage and labeled as well. The URLs are also sent to automatic regular expression blocker, which has an immediate impact on improved user security. As a result, the malicious URLs might be blocked before the campaign arrives to users from a different source.

3.3 Data Analysis and Visualization

There is a number of third-party systems in Avast Software consuming the incoming samples, of which the most interesting are dynamic analysis and labeling tools. Using these, we can improve the threat intelligence and correctly label campaigns with corresponding tags. What’s more, we can detect unknown samples spreading through the honeypot and alert the malware analysts of an unknown threat within minutes of the campaign commencing.

Considering visualization, we present two different views on the current state of the honeypot. First, there is a Grafana dashboard displaying e-mails received per minute, IP geo-location, the most prevalent attachments, and honeypot users in the selected time frame. This tool serves to gain better insights into the overall state of the honeypot.

On the other hand, when analysts want to explore content spread through the honeypot in detail, we developed a custom Threat Intelligence platform that offers several views.

- **Attachments.** Users are presented with a list of attachments received. The list can be sorted using time or sample count, and filtered using file type or date delivered. All the attachments are tagged with a corresponding malware family.

[9] http://mqtt.org/

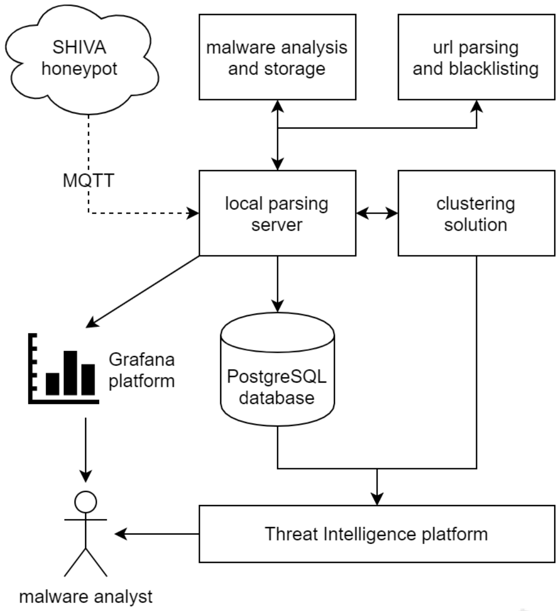

Figure 2: Data flow diagram overview of the honeypot infrastructure.

This is achieved with YARA[10] rules created in Avast. What is more, the users can discover what campaigns delivered given attachment and create a more in-depth connection between campaigns as well as URLs.

- **Campaigns.** Users are presented with a list of spam clustered to campaigns. A list of attachments and URLs is present in each campaign. Sorting and filtering options are similar to the Attachments view.
- **URLs.** Users are presented with a list of URLs, labeled by the payload they contained while the campaign was running.

In addition, users can view more information about each attachment or campaign using the link to different Avast Software third-party services. This includes the possibility to rerun the dynamic analysis or to check the results of the third-party clustering solution.

Using the described infrastructure, we are able to provide the honeypot users a thorough view of the e-mail cyber threats as they are being captured. Even without browsing through the data, the users are automatically notified about unknown threats, e.g. using Slack or e-mail messages.

The complete solution is illustrated in Figure 2.

4 RESULTS

In this chapter, we provide results from running our authentication-requiring SMTP honeypot for the 14-month period, from July 2018 to August 2019. During this interval, we captured over 31,386,000 malicious e-mails. From Figure 3, it can be seen the lack of e-mails received in the late months. We partly attribute this to the fact we were not distributing any new SMTP credentials recently. However, in the time of writing this paper, October 2019, the honeypot is currently very active again, as described in the case study in section 4.2 below.

When looking at the source of the spam, it was delivered from a total of 126,766 IP addresses. When geo-located in the time of delivery, using the latest MaxMind database[11], they originate in a total of 213 countries, according to ISO 3166-1. The most spam was delivered from the USA, nearly 7,450,000 unique e-mails, followed by South Korea with 2,322,000 e-mails and Mexico with 1,850,000 e-mails sent. Interestingly, Colombia and South Africa followed with 1.4 million e-mails sent from each. More than 1 million e-mails were also delivered from India, China, and Argentina. All the countries can be seen in map Figure 4.

English was the most common language, with more than 27.5 million e-mails. Next, there were 1.6 million e-mails written in German. In the third place, we could find nearly 800,000 e-mails in Spanish. Other most used languages in the collected spam can be found in Figure 5.

Considering attachments, we captured a total of 74,071 unique attachments. The most common malicious format was PDF file with nearly 30,000 samples, followed by an MS Word document with more than 17,000 samples. However, while the MS Word files were reused quite often for different victims and the total e-mail count containing MS Word attachments were many millions, the PDF files were often unique, e.g. mentioning the name of the victim in the text. The distribution of all the file types can be seen in the pie chart Figure 6.

When looking at the targeted victims' e-mail addresses, more than 4.6 million different domain names were present. It is not surprising that the most prevalent were generic domains, like gmail.com with 2.5 million addresses, and hotmail.com, and yahoo.com, with 1.1 million addresses each. We provide the top ten attacked top-level domains (TLD) as well as full domain names in Table 1.

[10] https://virustotal.github.io/yara/

[11] https://www.maxmind.com/en/home

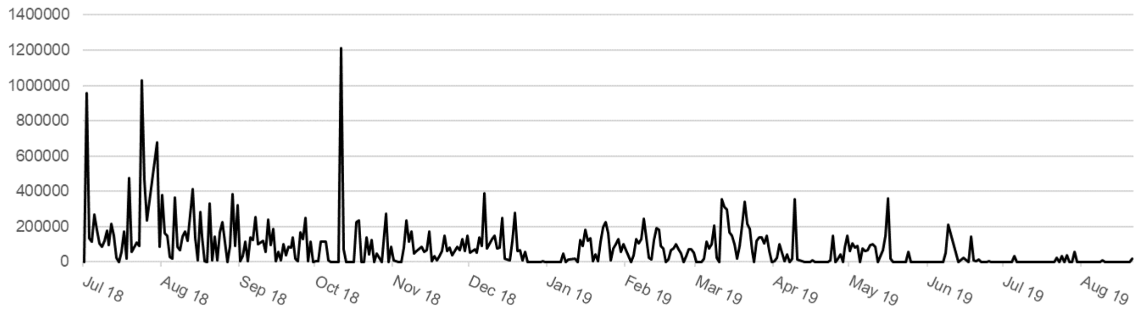

Figure 3: Number of unique e-mails captured each day from July 2018 to August 2019.

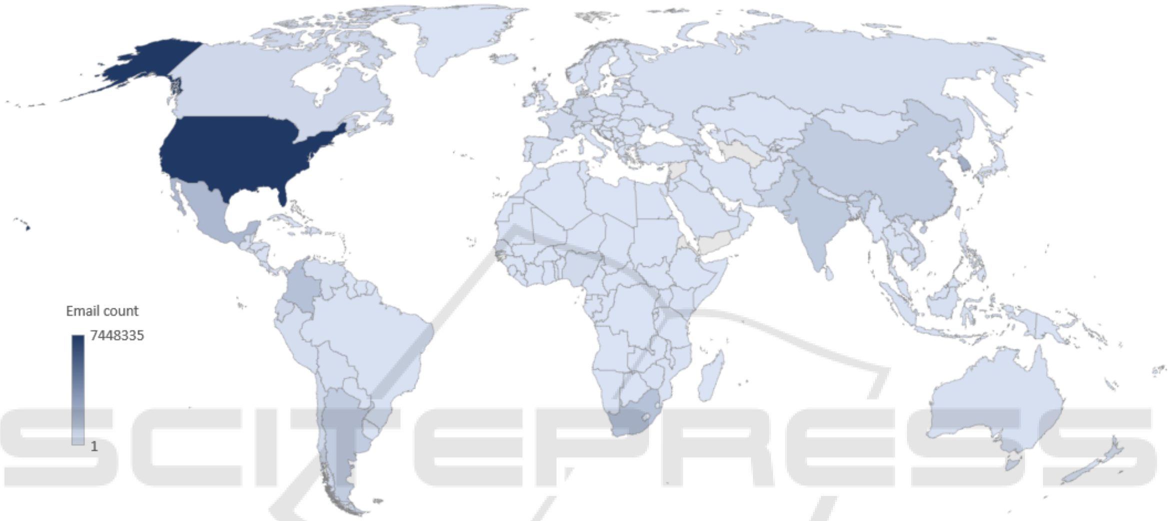

Figure 4: Number of unique e-mails received by country.

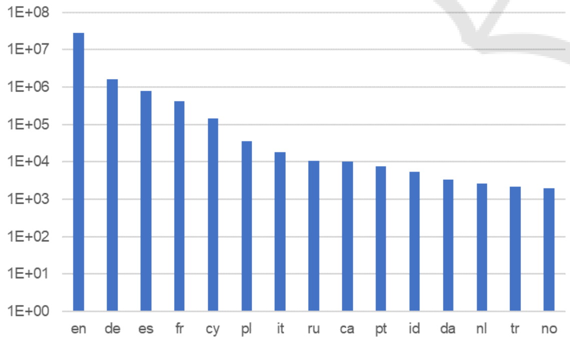

Figure 5: Distribution of languages used in spam.

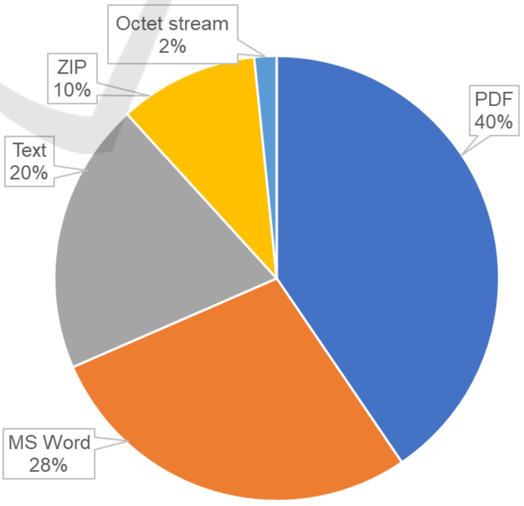

Figure 6: Distribution of attachment file types.

4.1 Analysis of Captured Attachments

Using the third-party dynamic analysis and labeling system, we were able to obtain knowledge about each attachment received during the monitored period. In this section, we present statistics on the malware families spreading through our honeypot, as well as different malware families' specifics.

In the period of 14 months, from July 2018 to August 2019, we collected over 74,000 unique attachments, which were analyzed and labeled. The most prevalent malware family spreading through our hon-

Table 1: Top spam recipients' e-mail domains.

Domain	Count	TLD	Count
gmail.com	2,520,188	com	16,593,050
hotmail.com	1,178,475	de	1,778,513
yahoo.com	1,165,326	net	1,756,130
aol.com	626,561	uk	939,508
t-online.de	192,105	org	923,466
indeedemail.com	169,215	ca	349,499
comcast.net	129,500	ch	313,862
web.de	125,372	mx	311,592
gmx.de	121,178	edu	311,347
msn.com	108,563	fr	289,616

eypot was **Emotet**[12] banking trojan. This advanced and highly destructive malware was spread mostly with malicious MS Word documents, but also URLs to download such a file, which were placed directly inside the e-mail or attached PDF. This malware family was spreading throughout the whole monitoring period, with several weeks pauses and many millions of samples delivered. We are unable to state the exact number of Emotet samples captured due to its massive spreading through PDF files, which we were unable to tag reliably, as well as through URLs, which did not respond to our analysis server, arguably because of blacklisting. However, a detailed look at one such campaign is provided in the next section 4.2.

Another banking trojan **RTM**[13] was less prevalent with only 1,241 unique samples delivered, using more than 10,000 e-mails totally. This family was spreading in the late months of the monitored period, from June 18th, 2019, in the form of MS Word documents as well as executable binaries.

Another prevalent malware family was credentials-stealing trojan **Fareit**[14]. Even though only 35 unique samples were used to spread this malware, more than 700,000 e-mails were captured. Unusual attachment formats, ACE and ARJ archives, were used to pack the binary. Similar to Emotet, the Fareit family was repeatedly spreading throughout both years, with several months pause after each campaign.

Another credentials stealing malware called **FormBook**[15] was observed in two campaigns. The first observed campaign spreading this malware, containing only eight e-mails, appeared on February 19th, 2019. Several months later, on May 2nd, 2019, a much larger campaign containing more than 220,000 e-mails spread another sample of this spyware.

The similar behaviour was observed with infostealer **AgentTesla**[16]. Only five unique samples, either ZIP or ARJ were used in two campaigns on April 24th and July 26th, 2019. However, nearly 95,000 malicious e-mails were captured, spreading this malware. Yet another infostealer, **LokiBot**[17], was spread throughout the monitoring period in 4 different campaigns with the total of 160,000 samples delivered. While Fortinet originally reported[18] spreading through PDF files and 7-Zip archives, in our honeypot, we captured ARJ archive, MS Word document, as well as executable binaries.

There were also smaller one-time campaigns, spreading different types of malware. On April 4th 2019, spreading of another banking trojan **Qakbot**[19] was detected. MS Word attachments were used to send about 500 e-mails throughout a single hour. On April 26th, a ransomware delivering downloader **Nemucod**[20] was captured in a 2000 e-mails large campaign. On June 18th, 2019, we captured several samples of **SmokeLoader**[21] downloader, which correlated with the worldwide resurgence of this old malware family. On July 22nd, 2019, at that time, a brand new ransomware **Sodinokibi**[22] was captured in a small campaign of 30 e-mails.

We also captured several backdoor and remote access trojans (RAT), including **Valyria**[23] and **REMCOS**[24], spreading in February 2019 using MS Word documents, and ZIP files respecitvely. Another campaign of REMCOS was captured on July 3rd 2019 using executable binary attachments. **NanoCore** RAT was observed on September 4th 2019 in small numbers.

Many of the malware samples were identified as

[12] https://www.us-cert.gov/ncas/alerts/TA18-201A

[13] https://usa.kaspersky.com/about/press-releases/2019_rtm-banking-trojan

[14] https://blog.talosintelligence.com/2015/09/down-rabbit-hole-botnet-analysis-for.html

[15] https://www.sentinelone.com/blog/formbook-yet-another-stealer-malware/

[16] https://krebsonsecurity.com/2018/10/who-is-agent-tesla/

[17] https://www.fortinet.com/blog/threat-research/new-infostealer-attack-uses-lokibot.html

[18] https://www.fortinet.com/blog/threat-research/new-loki-variant-being-spread-via-pdf-file.html

[19] https://blog.talosintelligence.com/2016/04/qbot-on-the-rise.html

[20] https://www.cisecurity.org/blog/malware-analysis-report-nemucod-ransomware/

[21] https://research.checkpoint.com/2019-resurgence-of-smokeloader/

[22] https://www.cybereason.com/blog/the-sodinokibi-ransomware-attack

[23] https://threatpoint.checkpoint.com/ThreatPortal/threat?threatType=malwarefamily&threatId=164669

[24] https://www.fortinet.com/blog/threat-research/remcos-a-new-rat-in-the-wild-2.html

different kinds of **Crypter**[25] malware. These families employ obfuscation techniques in order to make automatic analysis hard or even impossible.

Some of the attached MS Word documents failed to download the malicious payload and were therefore identified generally as a downloader family, such as **Donoff**[26] and **SAgent**[27]. We also identified exploiting several MS Word vulnerabilities to run malicious code, namely **CVE-2017-11882** and **CVE-2012-0158**.

Additionally, a significant part of the attachments (33,280 samples) received no label. When looking at the data, one may divide them into several categories and discover further insights. Most of them, about twelve thousand, were benign plain text and image attachments, like icons. Therefore, a missing label is appropriate.

Nearly nine thousand unique unlabeled samples were PDF files. From our analysis, these files were either a document containing phishing content (e.g. lottery winner) or were part of the campaign spreading Emotet. Although missed by the detections, the malicious payload, downloaded from the URL inside PDF, would be caught correctly.

More than five thousand unique unlabeled attachments were ZIP archives. These files were encrypted. Therefore, the automatic analysis was unable to extract and label their payload. However, the archive is easily opened by humans as the password was supplied inside the e-mail. From the manual analysis, we discovered these encrypted attachments were used to spread Emotet. As further work, we plan to implement a dictionary attack against these files, with the dictionary containing words inside the e-mail, therefore including the correct password. This way, it will be possible to analyze the encrypted archives automatically.

The last significant category of unlabelled samples, about four thousand, contained MS Word documents. These files were either corrupted and couldn't be opened or didn't include any malicious payload at all, arguably by mistake. The missing malware label is, therefore, in place.

4.2 Studying Malicious Campaign in Detail

Using our honeypot, one can also watch a single malicious campaign in detail. As an example, we study one of the latest Emotet-spreading campaigns which launched during the writing of this paper, on Wednesday, October 9th, 2019, and ran for 78 hours. During the mentioned period, we captured 83,134 e-mails, most of them spreading Emotet. Detail can be seen in Figure 7, which shows e-mails received per minute.

Even though the campaign continued after the 2-day weekend break, for the sake of this study, we present statistics for the first week only. It is interesting to observe, however, the users of our honeypot generally mimic the workdays, not distributing spam during weekends. We argue it is not for their convenience but to better target their victims who are not checking their e-mail folders during weekends. Spam distribution is, therefore, inefficient in such times.

The payload was delivered in one of three ways – direct malicious MS Word e-mail attachment, URL to download the malicious file, and PDF file, which contained the URL. During the campaign period, we captured 72,728 attachments, from which 66,108 (859 unique) were MS Word files and 6,619 (511 unique) were PDF files. This demonstrates the observation above, PDF files being more distinctive than MS Word documents. While a single PDF was delivered to approximately 13 recipients, an MS Word document is sent to 77 recipients, on average.

Additionally, 31,305 e-mails contained a total of 41,541 unique URLs. From those, only 70% were responding in the time of e-mail relaying. Further, 16,336 (406 unique) URLs were distributing MS Word files, and 329 (293 unique) URLs were distributing a PDF file. Other URLs contained HTML or images.

One could also observe a strategy called *thread hijacking*, which was first observed in 2017 used by North Korean hackers, adopted by Emotet in April 2019[28]. The authors of Emotet use stolen e-mail conversations to create highly customized and targeted e-mails with the malicious payload. We also noticed the language of the malicious text often corresponds to the original conversation. Thanks to this method, every e-mail received is unique and the clustering techniques based of body similarity fail to detect the campaign.

If we look at the source of this campaign, it was delivered from a total of 3,726 unique IP addresses. These originate in 132 different countries. However,

[25] https://blog.malwarebytes.com/cybercrime/malware/2017/03/explained-packer-crypter-and-protector/

[26] https://blog.talosintelligence.com/2018/11/threat-roundup-1123-1130.html

[27] https://blog.talosintelligence.com/2018/09/threat-roundup-0921-0928.html

[28] https://www.zdnet.com/article/emotet-hijacks-email-conversation-threads-to-insert-links-to-malware/

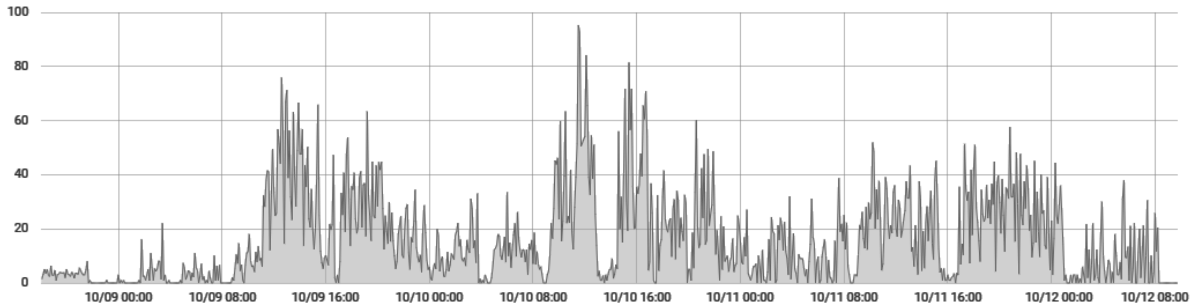

Figure 7: E-mails received per minute during the Emotet-spreading campaign, from Oct 9th to Oct 12th 2019.

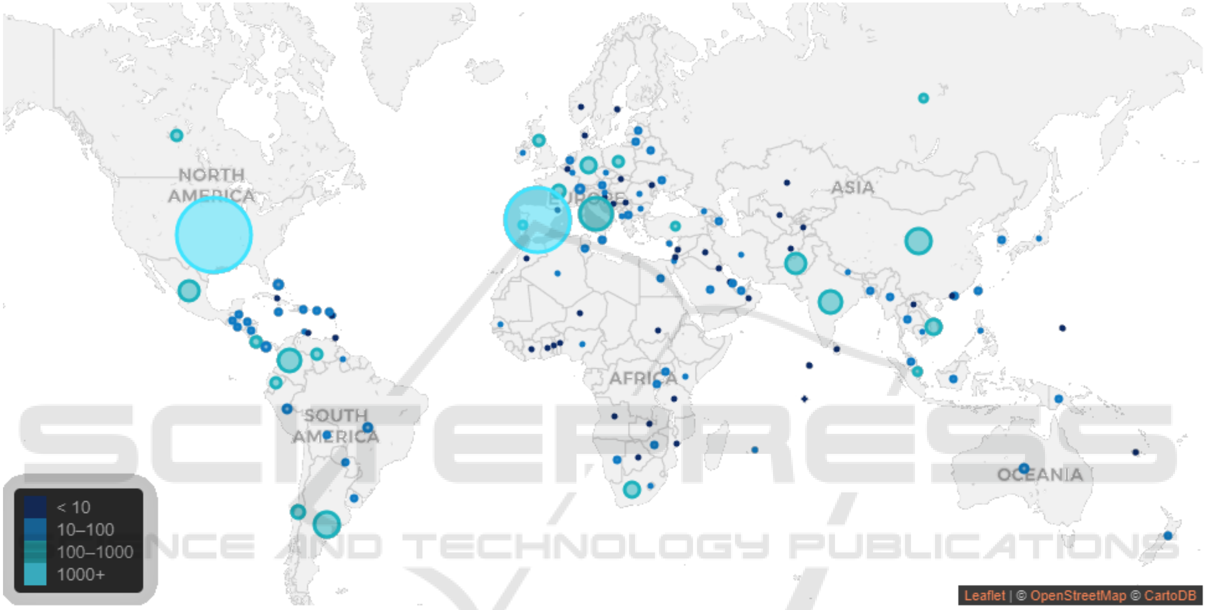

Figure 8: Country of origin of IP addresses spreading Emotet from Oct 9th to Oct 12th 2019, as depicted in the Grafana platform.

a large portion of the spam came from the US, Spain, and Italy, as seen in map Figure 8. Arguably, adversaries use their botnet, existing victims, to spread the infection.

When comparing the language used in this campaign to global statistics, English is still the most prevalent, however, only about four times as much as other languages like Italian, German, and Spanish. Similar data can be seen in victims' domains. While *gmail.com* remains the top domain, *libero.it*, popular Italian portal occupies the second place. Also, domains like *yahoo.it*, *hotmail.it*, and *tiscalli.it* are present in the top 10 domains.

5 CONCLUSION

E-mail plays a crucial role in today's cyber criminal activities. As presented in this work, when credentials are distributed accordingly, authentication-requiring e-mail honeypot provides a powerful tool to collect intelligence about emerging e-mail threats. With the information gathered, one can stay informed about malicious campaigns spreading throughout the world. Samples collected from the honeypot can serve an excellent value for any cybersecurity entity, e.g. for creating updated detections to protect their users, as well as discovering previously unseen malware families.

Furthermore, within our architecture, all the e-mails are parsed, analyzed, and submitted to central storage automatically. The knowledge from this honeypot is then visualized in several ways. Grafana platform serves a high-level overview of the honeypot activity, while the internal Threat Intelligence platform provides malware analysts an in-depth look into each of the samples gathered. Thanks to the third-party real-time dynamic analysis of the malicious samples, we can alert the analysts about unknown sample spreading in the wild within minutes of capturing.

5.1 Future Work

In further research, we would like to focus on the way honeypot credentials are distributed. Indeed, this has a vital role in the kind of content we are capturing. When we gain more insight into this field, it will be possible to connect different malicious samples stealing the credentials, with various malicious campaigns spreading through the honeypot. We could improve the knowledge of the credentials reselling ecosystem or even reveal adversaries' identity. Cooperation with law enforcement agencies could bring them to justice.

ACKNOWLEDGEMENTS

This work was supported by the "Avast Scholarship" project, granted by Avast Software s.r.o., and by Ministry of Education, Youth and Sports of the Czech Republic from the National Programme of Sustainability (NPU II) project "IT4Innovations excellence in science" LQ1602.

REFERENCES

Alishahi, M. S., Mejri, M., and Tawbi, N. (2015). Clustering Spam Emails into Campaigns. In *2015 International Conference on Information Systems Security and Privacy (ICISSP)*, pages 90–97. IEEE.

Andreolini, M., Bulgarelli, A., Colajanni, M., and Mazzoni, F. (2005). HoneySpam: Honeypots Fighting Spam at the Source. *SRUTI*, 5.

Bhowmick, A. and Hazarika, S. M. (2016). Machine Learning for E-mail Spam Filtering: Review, Techniques and Trends. *arXiv preprint arXiv:1606.01042*.

Brunton, F. (2013). *Spam: A Shadow History of the Internet*. The MIT Press.

Calais, P. H., Pires, D. E., et al. (2008). A Campaign-Based Characterization of Spamming Strategies. In *Proceedings of the 5th Conference on E-mail and Anti-Spam (CEAS)*.

Dinh, S., Azeb, T., Fortin, F., Mouheb, D., and Debbabi, M. (2015). Spam Campaign Detection, Analysis, and Investigation. *Digital Investigation*, 12:12 – 21. DFRWS 2015 Europe.

Han, J., Pei, J., Yin, Y., and Mao, R. (2004). Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach. *Data mining and knowledge discovery*, 8(1):53–87.

Khan, W. Z., Khan, M. K., Bin Muhaya, F. T., Aalsalem, M. Y., and Chao, H. (2015). A Comprehensive Study of Email Spam Botnet Detection. *IEEE Communications Surveys Tutorials*, 17(4):2271–2295.

Kornblum, J. (2006). Identifying Almost Identical Files Using Context Triggered Piecewise Hashing. *Digital investigation*, 3:91–97.

Krawetz, N. (2004). Anti-honeypot technology. *IEEE Security Privacy*, 2(1):76–79.

Liston, T. (2003). Tom Liston talks about LaBrea. http://labrea.sourceforge.net/Intro-History.html. Accessed in April 2020.

Nawrocki, M., Wählisch, M., Schmidt, T. C., Keil, C., and Schönfelder, J. (2016). A Survey on Honeypot Software and Data Analysis. *arXiv preprint arXiv:1608.06249*.

O'Gorman, B. et al. (2019). 2019 Internet Security Threat Report. Technical report, Symantec.

Oudot, L. (2003). Fighting Spammers With Honeypots. https://www.symantec.com/connect/articles/fighting-spammers-honeypots-part-1. Accessed in April 2020.

Smirnov, M. (2018). Clustering and Classification Methods for Spam Analysis. Master's thesis, Aalto University, 02150 Espoo, Finland.

Spitzner, L. (2001). The Value of Honeypots, Part One: Definitions and Values of Honeypots. https://www.symantec.com/connect/articles/value-honeypots-part-one-definitions-and-values-honeypots. Accessed in April 2020.

Spitzner, L. (2003). *Honeypots: Tracking Hackers*, volume 1. Addison Wesley.

Stoll, C. (2005). *The Cuckoo's Egg: Tracking a Spy Through the Maze of Computer Espionage*. Simon and Schuster.

Tsikerdekis, M., Zeadally, S., Schlesener, A., and Sklavos, N. (2018). Approaches for Preventing Honeypot Detection and Compromise. In *2018 Global Information Infrastructure and Networking Symposium (GIIS)*, pages 1–6. IEEE.

Uitto, J., Rauti, S., Laurén, S., and Leppänen, V. (2017). A survey on anti-honeypot and anti-introspection methods. In *Recent Advances in Information Systems and Technologies*, pages 125–134, Cham. Springer International Publishing.

Verizon (2019). 2018 Data Breach Investigations Report. Technical report, Verizon.

Zou, C. C. and Cunningham, R. (2006). Honeypot-Aware Advanced Botnet Construction and Maintenance. In *International Conference on Dependable Systems and Networks (DSN'06)*, pages 199–208.