WordyThief: A Malicious Spammer

Dr. Renée Burton
Cyber Intelligence
Infoblox
Sykesville, MD, USA
rburton@infoblox.com

Vadym Tymchenko
Cyber Intelligence
Infoblox
Tacoma, Washington, USA
vtymchenko@infoblox.com

Nicholas Sundvall
Cyber Intelligence
Infoblox
Tacoma, Washington, USA
nsundvall@infoblox.com

Minh Hoang
Cyber Intelligence
Infoblox
Tacoma, Washington, USA
minhhoang@infoblox.com

Jim Mozley
Cyber Intelligence
Infoblox
London, England
jmozley@infoblox.com

Michael Josten
Cyber Intelligence
Infoblox
Tacoma, Washington, USA
mjosten@infoblox.com

***Abstract*—We detail the tradecraft used to discover and exploit a prolific Russian-affiliated malicious spam actor. To the best of our knowledge, this paper is the first description of the actor, whom we call WordyThief, and the first publication demonstrating the application of graph techniques to the identification of malicious spam campaigns. This work contributes to the threat intelligence community both as a technique that can be utilized in daily practice, and as a thorough account of WordyThief, who continues to spread malware in October 2020. We initially discovered isolated malware campaigns using large-scale bipartite graphs created from email metadata. These graphs and related campaign specifics revealed the use of domain names within the spammer's infrastructure devised through dictionary domain generation algorithms (DDGAs). Using a second graph-based technique and time series analysis, we recovered the underlying dictionaries and temporal behavior of the actor. A retrospective review of spam collection and correlation with other Domain Name System (DNS) information led us to conclude that the campaigns were all the work of a single actor. We tracked their activity and substantiated our methods retrospectively, through December 2019. We also leveraged open source intelligence (OSINT) to verify our findings. We found that WordyThief operates a large spam infrastructure and distributes malware that steals personal and financial information from victims. This paper includes not only the scientific methods used to detect the actor, but also detailed descriptions and analyses of several elements of their tactics, techniques, and procedures (TTP). We include an analysis of the actor's tendency to use of aged domains, a text analysis of their emails, use of embedded IP tracking in their campaigns, harvesting of open source images, and an exposition of their evolving exploitation techniques.**

***Index Terms*—network security, computer security, statistical learning, machine learning algorithms, computer crime, computer hacking**

I. Introduction

Often thought of as a nuisance, email spam is the source of a great deal of cyber crime [20]. Malicious email campaigns, commonly referred to as **malspam**, continue at very large scales in spite of modern spam filtering practices. Targeting vulnerable readers through lures that range from social and political campaigns, promises of rewards, or threats of legal action, the malicious actor is only a click away from stealing personal credentials or holding the user's files for ransom [17]. For threat actors to be successful, they only need a small number of victims to fall into t heir trap, whereas to stay safe, the recipients need to be vigilant 100% of the time. Malspam, like so much of commodity cyber crime, is an unfair playing field for the average email user. It is a major criminal industry targeting individuals, corporations, and governments alike, and even feared to be used to disrupt the US Presidential election in 2020. [34, 23, 49]

To protect the vast array of vulnerable users globally, the cyber security community, both commercial and hobbyist, invests heavily in identifying and blocking threats delivered via spam. Open source intelligence (OSINT) plays a pivotal role in both distributing threat information and validating hypotheses. Sources to disseminate findings are numerous, including Twitter (`twitter.com`), dedicated sharing websites such as `abuse.ch` and the newer `threatshare.io`, as well as personal websites. Public sandboxes, such as Virus Total, any.run, CAPE sandbox, and Joe's sandbox, enable citizen threat hunters.[1] Validated threat indicators, typically referred to as indicators of compromise (ioc), are included in commercial and public threat feeds, also known as block lists, and utilized in the full spectrum of security products, from anti-virus software to Domain Name System (DNS) firewalls, to protect individual users.

In spite of the concerted effort, malicious email remains a formidable threat and likely will persist as such [37, 11, 45, 9]. Threat actors are not only able to slightly change their techniques to temporarily evade detection, but they also have the luxury of a black market to assist with all aspects of conducting a cyber campaign. Malware-as-a-Service (MaaS) is a growing industry in which components of a successful malspam campaign can be leased in a supply chain model [18, 16]. In addition, many threat actors take advantage of compromised machines to create botnets that distribute

[1]The respective domains for these resources are virustotal.com, any.run, capesandbox.com, and joesandbox.com

malware [26]. Once successful in infecting a host, modern malware may download other types onto the machine. In the current malspam environment, Emotet is likely the largest and most notorious player [24], but numerous other botnets, malware, and actors are ever present.

In this paper we use the context of a specific spam actor to describe some of our tradecraft for discovering and tracking malspam actors. This actor, whom we call WordyThief, is somewhat different from Emotet, Necurs, and others [47], in that they own and operate their spamming infrastructure rather than using a botnet of compromised machines (spambot) for distribution. Unlike the stereotype of MaaS operations [44], WordyThief not only delivers spam, but also controls the malware distribution, as well as the command and control (C&C). Over an ten month study of WordyThief, we have seen them distribute just two types of malware, Predator the Thief [7] and Taurus Project [14], both of which share the same code base. We will discuss our discovery process, the retrospective review of their activities, elements of their Tactics, Techniques, and Procedures (TTPs), the distributed malware, as well as some elements to track and predict their behavior. While this paper focuses on the particulars of WordyThief, we have used these techniques to uncover other actors, and we hope our experience may help the greater cyber security community deny, disrupt, and degrade these criminals.

In the sections that follow, we will describe a single actor who:

- owns and operates a very large scale spam distribution system,
- hosts the spamming infrastructure in Russian IP space,
- utilizes dictionary domain generation algorithms (DDGA) to create domain names for their infrastructure, and registers them in batches,
- appears to age domains prior to use,[2]
- uses hidden images in emails to track recipient IP addresses and location,
- utilizes an evolving set of Microsoft Office-based vulnerabilities to infect victims, and
- controls the information stealing malware they distribute.

We use proprietary data sets in this research, however we understand lack of data access hinders others, particularly academics, from reproducing the work exactly. Therefore, wherever possible, we have provided commercial and open source references that can validate the specific claims of the paper, and may be used to create similar capabilities in other environments. This work also relies on a broad base of skills, including reverse engineering, open source intelligence, and data science. The intention of the paper is to contribute methods for use by others without access to a full spectrum of researchers.

II. Discovery

Our discovery of WordyThief began with a malspam campaign investigation in mid-May 2020. A malspam campaign is a set of related spam messages typically centered on a single topic, and generally occurs over a period of a few days. During a campaign, the malicious actor will use a lure to entice the mail recipient to take some sort of action that will download malware. Lures come in many forms, from promises of great wealth to threats of exposure [17]. The user generally will need to click on a link, open a document, or enable macros in order for the attack to proceed. Malware infections of this type occur in stages, and security products in a victim's network serve to protect them from the final stage. To provide this protection, security companies first need to find, or predict, the threat.

The workflow that uncovered WordyThief, and their activities, began with the large-scale creation of relational graphs from email metadata. A **graph** is a mathematical concept consisting of nodes and edges that describe the connectivity of objects in a given context. Graphs of this kind can be created in a number of ways, to address different use cases, as we describe in [13]. A primary application is the identification of individual campaigns, and we have found that a bipartite graph, in which the nodes are divided into two disjoint sets based on email header attributes, is very effective for this purpose. In this case, our initial graphs were built from subject lines and the sha256 values of attached files in an email. Considered over several days, we find that campaigns generally lie in connected components, which are sub-graphs for which all the nodes are connected. We select components of an unusual size or structure for further analysis.

The WordyThief campaigns in mid-May created large connected components. An example is shown in Figure 1. In this campaign, a single subject line is associated with a number of different attachments, on the order of 10-15 distinct hash values. However of these files, a small number, typically 1-2, will also be associated with a second subject line. This behavior of the spammer causes the subject lines to all connect through some path and created a large, loosely connected component.

In the campaign shown in Figure 1 there are 17 distinct SMTP IP addresses observed for thousands of emails. Another way in which we analyze the campaign is to connect the IP addresses with subject lines. This is shown in Figure 2.

In mid-May 2020, we observed an unusually large campaign, consistent with sizes we typically see with Emotet, an actor known for large-scale campaigns [48]. However, we also knew that Emotet was inactive [4] at that time. Closer inspection of the email attachments found characteristics reminiscent of Predator the Thief campaigns we had previously reported on [7] [10] but with some unique features. Our analysts determined that the malware was not Predator the Thief, but a new variant called Taurus Project. As reported by Infoblox in [14], Taurus Project is an information stealer released in April 2020 based on Predator the Thief.[3] It is available for purchase in Russian forums and may be regarded as part of the MasS

[2]Aging domains refers to the practice of delaying their use in operational scenarios for some time to avoid detection by security products.

[3]An information stealer is a generic class of malware that exfiltrates user information including keystrokes, financial information, social media contacts, login credentials, installed applications and screenshots.

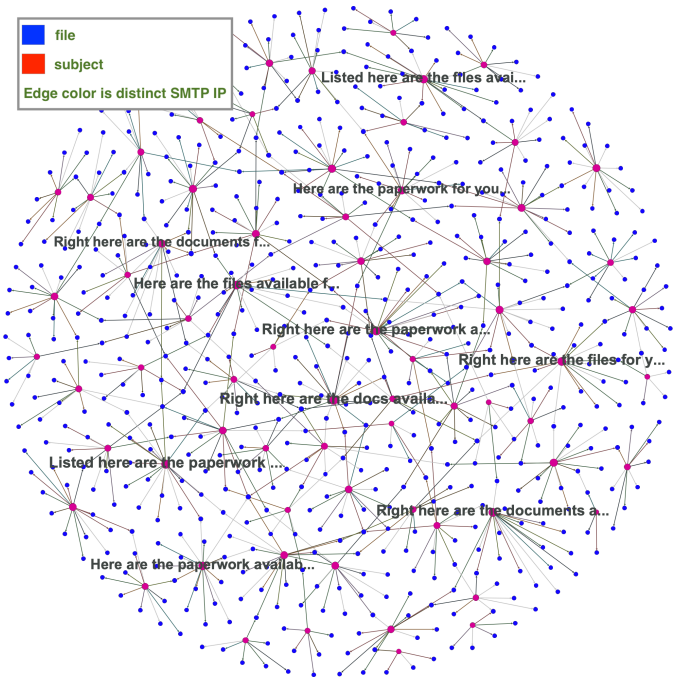

Fig. 1. A WordyThief campaign using DocuSign-related subjects. The bipartite graph is generated with subject lines and attached file sha256 values as nodes. Edges are colored by unique SMTP IP addresses associated with the email.

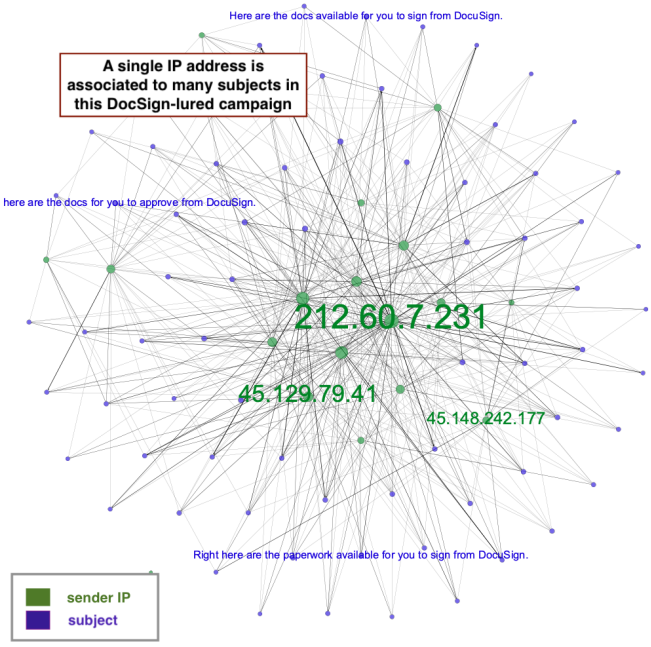

Fig. 2. An example WordyThief campaign graph containing sender IP and subject line nodes.

ecosystem, although we do not know the exact details of how the sale and operations work. Having found one Taurus Project campaign, we began actively looking for others.

In reviewing the Taurus Project campaigns, we realized that one of the C&C domains was previously used in the infrastructure of a spam campaign. This signaled that this spam distributor may also operate the information stealer. Analyzing the data, it became clear that the email was not distributed via a botnet, but through an actor owned-and-operated network. By this, we mean that the actor registered domains and fully configured them to send mail. The IP addresses used for transmitting mail were not associated with shared hosting services, nor were they spoofed, a common technique for hiding the original source. In the event that a recipient's mail system performed routine checks to validate the mail sender, they would pass. This extra care increased the odds that spam would be delivered, but the reuse of the domain also tied the spammer to malware distribution itself.

Furthermore this C&C domain was not a random domain, but generated as part of a large dictionary domain generation algorithm (DDGA). That DDGA was used to determine part of the spam infrastructure, allowing us to unravel the campaigns retroactively.[4] For this paper, we analyzed the actor's activities since late November 2019. We were able to definitively tie all of the campaigns to a single spamming network, hosted in Russian IP space, which initially distributed Predator the Thief, and later Taurus Project. The size and rhythm of this spamming actor is vast and widespread. We coined them WordyThief due to their use of DDGAs and information stealing malware. In the remainder of this paper, we'll detail our findings and methods for discovering and tracking this actor.

[4]This DDGA algorithm is still being used in July 2020 to register new domains.

III. The Retrospective Hunt

Once we realized that we were not observing an isolated Taurus Project campaign, but rather a part of a large established network, we set about uncovering the scope and impact of the threat actor's activities. In this section, we'll describe the methods and sources for our retrospective look, and in Sections IV and V we will detail features of the infrastructure, spam contents, and the distributed malware. For this review, we used historical malspam and domain registration (Whois) sources, multiple DNS sources, and a wide array of open source intelligence, including public sandboxes, threat sharing sites, and blogs. While we are sharing these results holistically, the steps were actually iterative, as is typical in threat research; we used multiple pivots through different types of data to build out a picture of the actor's network. Eventually we were able to gain strong enough signatures to predict their campaigns.

A. Spam Infrastructure

In mid-May 2020, WordyThief made a mistake. We recognized that the two C&C domains `cogihold.site` and `babbleabode.site`, cited in several OSINT sources [50], were part of the spamming infrastructure exposed by our graph-based techniques described in Section II. While the entire graph is difficult to display, Figure 3 shows part of the spamming infrastructure as of June 2020, in which nodes

are represented by the simple mail transfer protocol (SMTP) IP addresses from which spam is transmitted, as well as the spammer domains to which the original SMTP IP address is resolved.[5] We use this particular example, as it demonstrates the use of both Russian IP space, which dominated our historical review, and dedicated virtual private servers (VPS).

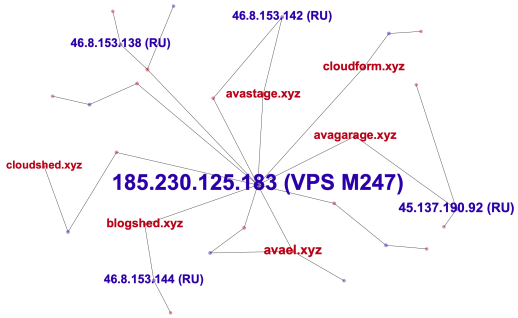

Fig. 3. A depiction of the connectivity between WordyThief spamming domains and their resolution IPs. Note the combination of Russian IP space, Virtual Private Services `vps247.com`, and dictionary-like domain names.

Once the connection between the C&C and spamming domains was made, it was fairly clear that a DDGA was in use, as the domains were seemingly composed of two words, or partial words, used repeatedly, e.g., `blogshed.xyz`, `cloudshed.xyz`, and `blogveranda.xzy`. These features can be observed in the small sample shown in Figure 3. As a result, if we could recover the underlying dictionaries, and their registration patterns, we could unroll the actor's network. We will discuss this portion of the research in Section V.

We used Farsight Channel 202 records [19] and the distributed processing framework Spark [6] to recover the entire connectivity graph of the actor's domains and resolution IPs over a six month period. Those results showed that aside from the expiration of some domains, the spamming infrastructure was consistent with that shown in Figure 3. Specifically, each registered domain the DNS was configured to resolve a mail server (MX) record at the same IP address. While the vast majority of resolutions was in Russian-controlled IP space, some did resolve to virtual private services (VPS). Moreover, domain-IP clusters were relatively small for the most part, meaning one IP address was only associated to a small handful of the more than 500 domains we recovered. It is worthy to note that the actor has fully configured these domains for mail transmission, ensuring their emails will pass through basic integrity checks made by receiving mail servers. This configuration appears to hinder their detection as spam senders, as the IP addresses were included in very few spam blocklists. It also shows that this system is not a spambot utilizing compromised or shared hosting environments as described by others including Adegbola [5] and Mimecast [24].

[5]From a DNS perspective, this is the PTR record for the IP address.

B. Campaign Tempo

We validated malicious behavior through observed malspam. Over the period beginning December 1, 2019 and ending July 28, 2020, we found WordyThief campaigns could be divided into two distinct sets. As shown in Figure 4, WordyThief campaigns have a relatively regular tempo before April 2020, and again from mid-May. It is possible that the actor was active during the time in which we see a lull, either with a different type of malspam campaign or in a manner that is not in our collection. During the first part of the year, WordyThief distributed only the information stealer, Predator the Thief [7]. In mid-May 2020, they began distributing Taurus Project, a new malware evolved from Predator [14].

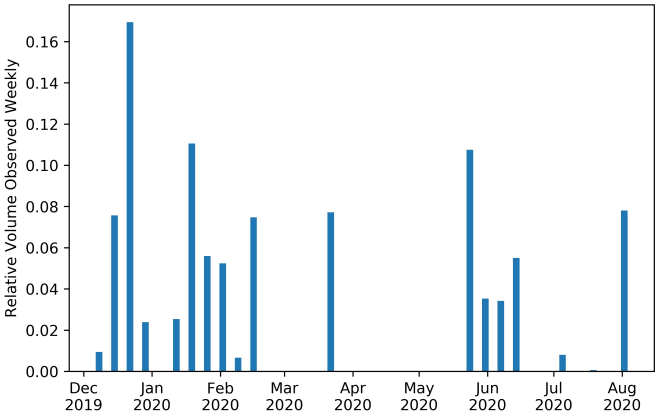

Fig. 4. Relative weekly campaign volume between early December 2019 and mid-July 2020. Early campaigns distribute Predator the Thief, while later campaigns distribute the Taurus Project.

C. Domain Registration and Aging

We considered the registration of the validated WordyThief spamming and C&C domains. In particular, we looked for signs of the actor's TTP regarding preferred domain Registrars, top level domains (TLDs), name server configuration, and the operational timeline from registration to utilization in a malicious campaign. During the period we examined, the vast majority of WordyThief domains were found in the `.site` and `.xyz` TLDs, although some were registered in other TLDs. The registrations we verified were all made via Namecheap [33] or the RU registrar [42]. The domains appear to initially resolve within Russian IP address space, however C&C domains were migrated to shared hosting, predominantly Cloudflare [15], just prior to use in a campaign. Registrations were generally made in bulk, with 35-40 domains registered in a single day, and leveraged a DDGA. We described our work on recovering the dictionaries of these algorithms in Section V.

The final characteristic, sometimes referred to as *domain aging*, is the propensity of an actor to register and use domains immediately, or to hold them for some period of time. In the past, studies of malicious domain activity found that suspect domains were often used shortly after registration. As a result, security products often consider the time since registration

in their algorithms to identify malicious behavior. To assist in this effort, there are multiple sources for newly registered or newly observed domains available, including the Farsight Newly Observed Domain (NOD) and SURBL Fresh [46] lists. These sources are used as *blocklists* in some networks. In recent years, security researchers have noted that many actors will delay using domains for several days to hinder detection; a large study of domain aging was released in 2019 [21]. We used the Farsight NOD and SURBL Fresh historical lists as a proxy for registration in this experiment. Of the over 500 WordyThief domains recovered, we found 259 were in one of these two commercially available sources, and 87 of these were independently identified as spam or malware [25].

As we can see in Figure 5, the number of days between inclusion in a newly observed domain feed and another type of threat feed varies widely. Approximately 77% of the domains are identified as either unwanted or malicious spam more than three days later, the typical period used for blocklisting newly observed domains. This finding is consistent with the trends established by Foremski in [21] and with the intentional aging of domains prior to use. Domain aging may be one factor in the low threat detection rate of these domains.

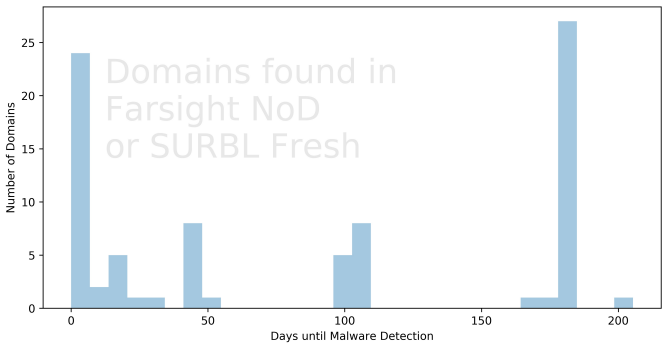

Fig. 5. The time period between newly observed domains and identification in malware campaigns. Approximately 50% of all domains have a time delta of more than three days.

IV. Spam Characteristics

We gathered approximately 20,000 unique emails distributed by WordyThief between December 1, 2019 and July 28, 2020. This activity occurred over 45 unique days, with our daily observation varying from a handful of messages to thousands. Within this set, there are nearly 1800 distinct subject lines and around 13,700 unique attachments. In this section, we will characterize the collection and describe our analytic techniques.

A. Campaign Themes

WordyThief campaign themes over the research period oscillate between encouragement and threats to the reader. While the former is consistent with the 2006 study on genres found in spam [17], the threat of legal or financial action was not recognized in that review. The authors focused on more generic spam, and in the case of malicious spam, the spammer inevitably needs the victim to take some action to enable the malware. To that end, fear of prosecution or financial liability is a common theme, as can be seen by the inclusion of terms like 'notice', 'notification', 'ticket', and 'defendant' in Figure 6, which shows the relative frequency of terms in the subject lines. Positive lures urge the reader to "take a look" or offer opportunities to purchase valuable items at heavily discounted prices. Neutral themes such as a shipping notification or payment receipts leverage natural curiosity as a means to get victims to enable macros in attachments.

Fig. 6. Frequent terms within WordyThief campaign subject lines January-June 2020.

To understand the campaign themes beyond just a visualization of the major keywords within the subjects, we use statistical modeling to identify the similarity of distinct subject lines. In particular, we converted the subjects to sets of tokens, or words, and utilized the frequency of words over time and across documents to create a *Term-Frequency Inverse-Document Frequency model* for the data [41]. Figure 7 shows the resulting pairwise similarity of all distinct subject lines in the data set. In this chart, the degree of similarity between two texts is indicated by the shading of the pixel that represents the pair, with darker shading indicating a greater similarity. We can see the that subjects group into one dominant cluster, as well as several other modest size groupings. That large cluster, representing nearly 50% of the data is related to shipping notifications. These results tell us something about the variation of campaign themes, but due to collection bias, are not necessarily indicative of relative campaign volumes and other features of the activity.[6]

Similar to the findings of Cukier [17], when considering the email subjects and text bodies in depth, we found the emails to contain a mixture of genres and lures. The emails utilized templates for both the subject lines and body. This technique varies the text slightly and hinders detection by security products.

B. Content Generation

Spammers, including those who distribute malware, are known to leverage templates to generate the content of their

[6]Collection bias refers to the limited scope of observation of the actor's activity. Such bias can unduly influence statistical analysis and conclusions should be guarded against this possibility.

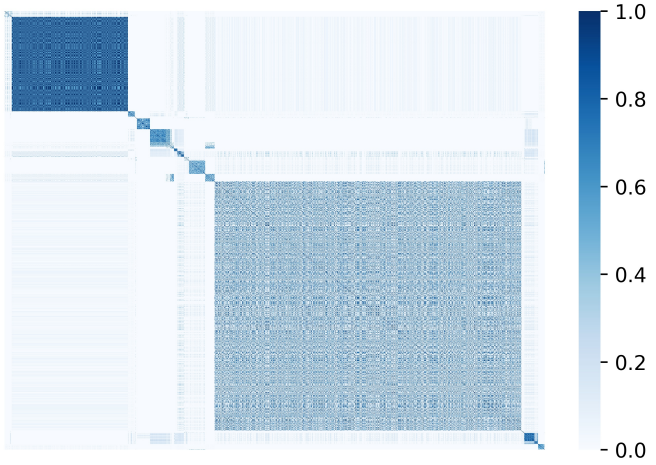

Fig. 7. The similarity of 812 distinct subject lines calculated using a TFIDF model. Highly similar texts have scores near 1.0.

emails. **Templates** are a programmatic means for the actor to generate a large number of emails whose subject lines and body are somewhat similar, but vary enough to pass through many spam detectors. In a study by Sroufe [43] the shape of the email is analyzed using character distributions and other statistical features, for example, to categorize spam templates. It's possible to find examples of such templates on the Internet. A blog from T.V. Raj [40] provides several detailed examples.

WordyThief used templates for their content, as well. Their emails often contained images or logos scraped from the web. We have uniquely tied two images to the website CodePen[7]. CodePen is a website aimed to help front-end developers create code by sharing code templates for a variety of tasks. In the company's words, "CodePen is a social development environment for front-end designers and developers." Unfortunately, it also provides a source of email templates and images that can be used for nefarious purposes. WordyThief extracted images from sample receipts and advertising campaigns hosted on the site.[8]

C. Embedded URLs

WordyThief email content generally contains image urls which are downloaded when opened in an HTML-enabled email client, such as the browser. In some cases, these urls link directly to original images, such as an eBay or Lyft logo.[9] In other cases, the actor scraped images or logos from unknown locations and stored them on an open link in Google Drive or Google Docs. [10] They utilized these type of embedded urls for everything from company logos to Google Maps images.[11]

Starting in May 2020, WordyThief also included hidden (1x1 pixel) images in their HTML email that served to track the recipients IP address. They began by using the service IP Logger, which claims "IP Logger URL Shortener web service helps webmasters to track IP addresses of the visitors on their websites, blogs or forums."[12] In addition to providing tracking for websites, IP Logger offers invisible image trackers. According to their website "This IPLogger image will not be visible to anyone. You can find IP address or even a mobile phone IP address and location (mobile tracker) by sending the generated invisible logger image via messenger". They admonish the customers of their services not to deploy hidden trackers without consent "Consent with T&C and Privacy Policy required!". This free service creates a shortened url that can be distributed via email as an embedded image. When the victim opens the email, the browser, unless configured otherwise, will attempt to download the image and in doing so, convey their IP address. This service was observed in several campaigns in May and July 2020.[13]

In June 2020, it appears that WordyThief may have attempted to replace the IPLogger links with their own software. During multiple campaigns in June, instead of `iplogger.org` embedded images, the emails contained hidden images linking to an actor-controlled domain. We observed multiple iterations of this behavior, with links to urls hosted at `namecount.site` and `brightpatio.site`.[14]

The use of IP logging via the *iplogger.org* service appears quite limited in our malspam sources. In a random sample of over $400,000$ emails that contained hyperlinks observed in the first ten days of July 2020, none contained links to IP Logger. For comparison, that same set of data contains Bitly (*bit.ly*) shortened links in approximately 2% of all emails containing hyperlinks.[15] Our earliest observation of this technique, within the research data set, occurred in mid-March 2020.

D. Malicious Attachments

WordyThief typically distributes malware through Microsoft Office documents and rich text format (RTF) files attached to emails. Enclosed Microsoft Word (DOC) files have leveraged macro features written in Visual Basic for Applications (VBA) to execute malicious scripts. When opened, the user is prompted to enable macros in order to view the file. Once complete, the VBA code uses the AutoOpen() [51] function to run an embedded, encoded, Powershell script [52], a legitimate tool and scripting language built into Windows. The Powershell script then delivered the final malicious payload, which

[7]https://www.codepen.io

[8]One such example is found at this location `https://codepen.io/reallygoodemails/pen/BwqaXL`.

[9]One such example is found at `https://s3.amazonaws.com/uber-static/emails/2017/global/social/footer_social_instagram50x40.png`

[10]Google Drive and Google Docs are two file sharing systems provided by Google and allow for sharing of public links. These urls include the domains drive.google.com or docs.google.com.

[11]One example of a parking company logo used is found at https://docs.google.com/uc?id=1OVOV2buBXx9PgzNFydUzLk55hCRmdOyf

[12]see https://iplogger.org website

[13]An example includes `https://iplogger[.]org/1ePu47`.

[14]An example is `http://brightpatio[.]site/OB1/amsweb.php?DopHSIHe5gdnxIt26xUFiLyKNdncOT0u08XO3NRPAQc4hKELqKNoPqRI1xqHx7ZNTIrjH7SY%2FcInNfpunGHpHWlv66m6DiEhLHlhrMY9iG%2B7vxBfREVoK%2B9skWTkhiCK`.

[15]Bitly is also a url shortener service, but does not offer IP tracking. https://bitly.com/

in our review was either Predator the Thief [38] or Taurus Project [14].

As an example, on June 12, 2020 we observed a campaign distributing the file `order(06.12.20)_1892.doc`[16]. Using the olevba utility from oletools [30] to analyze this file, we were able to extract and decode the obfuscated macro. Inside the VBA was a base64-encoded Powershell script which downloaded three files. The files included a Portable Executable (PE) file compiled by AutoIt [12]. The Powershell script then used Certutil to decode and execute the malicious downloaded payload, in this example Taurus Project.

E. Evolving Methodologies

The malware and downloader that WordyThief delivers has evolved over time. For example, newer samples contained checks to detect if the malware was running in a sandbox. We observed this behavior starting with the June 12 campaign. Two sandbox checks were added to the VBA that halt execution before the Powershell script runs, meaning the final malicious payload never downloads if run in a contained environment. At the beginning of the `AutoOpen()` function, the VBA checks if the file `C:\aaa_TouchMeNot.txt` exists. `C:\aaa_TouchMeNot.txt` is a file present in Windows Defender AV emulator, so the presence of the file could potentially indicate that the malware is running inside a sandbox [32], as a normal user likely would not have this file present on their computer. If the file exists, the VBA calls the function `end`, terminating the execution of the script. As a result, the payload does not get downloaded. In this particular case, adding `C:\aaa_TouchMeNot.txt` to a system would protect it from the malware. Next, the VBA uses Win32_PingStatus to check if a nonexistent Microsoft subdomain, such as `treIeanmascz.microsoft.com`, exists. Under normal circumstances, attempting to contact the subdomain would result in the domain being unable to resolve. If there is no reply, the execution will continue and the Powershell script will run and attempt to download the final malware. However, if it gets a reply from the subdomain then the malware is likely running in an environment set to resolve any DNS request [28], such as a sandbox used for malware analysis. Therefore, the payload does not download.

Recent campaigns sending RTF files have exploited Microsoft Office to execute the malware. To demonstrate the process, the attack chain for one of these campaigns is shown in Figure IV-E. The vulnerability that the downloader exploits is CVE-2017-11882 [35]. It is a stack buffer overflow vulnerability in Microsoft Office's Equation Editor that allows a threat actor to execute arbitrary code on the victims computer [1, 27]. In this case, the threat actor exploits this vulnerability to drop and execute an embedded PE, which leads to AutoIt running the malicious payload and stealing the user's data.

[16]This example file has a sha256 value of 4f054effeac0dd3400e54b93114a117cdd701876028b56d572dd3923fea3999f

F. Malware Delivered

WordyThief delivered only information stealers during the period of our research, specifically Predator the Thief [7], and more recently, Taurus Project [14]. These malware families are well covered by other sources, so we only briefly review them and their capabilities here.

Predator the Thief is an established malware created by Russian cyber criminals. It was made available to purchase online on Russian forums in June 2018 [39] for as little as $80 [7]. Predator the Thief can steal credentials for:

- Chrome and Firefox based browsers as well as Opera;
- email clients such as Outlook and Thunderbird;
- cryptocurrency wallets like Bytecoin, Electrum, and Ethereum;
- the two factor authentication program, Authy;

as well as take screenshots of the desktop and take a list of the installed programs on the computer. In April 2020, the Predator the Thief authors announced that with the advent of the new Taurus Project, Predator the Thief was no longer in operation.

Taurus Project was likely developed by the same group as Predator the Thief, although the Predator authors claim otherwise. It was first observed for sale in April 2020 for $100 [2]. The authors selling Taurus Project advertised that it can steal a wide variety of data including [8, 2]:

- cookies, browsing history, and credit card details from Chrome and Firefox;
- passwords and cookies from Edge;
- credentials for cryptocurrency wallets such as Electrum, Ethereum, and Jaxx;
- credentials for file sharing programs like FileZilla, WinFTP, and WinSCP;
- credentials for Discord, Steam, and Telegram;
- credentials for NordVPN; and
- credentials for Outlook

Purchasing Taurus Project provides the buyer with a convenient online web page that receives updates to help track and keep statistics on who they have infected with the malware [3]. The online portal reportedly allows the attacker to customize the information they would like to target with the malware, as well as enable "Anti virtual machine" and "Self Delete" features [8]. Taurus Project claims to avoid executing in countries of the Commonwealth of Independent States, which includes several Eastern European countries such as Russia, Ukraine, and Belarus [8]. Taurus Project C&C domains are available from a number of sources, including Cyber Crime Tracker [50] and Threat Share [22].

V. Domain Name Analysis

Massive spam actors send a large volume of spam messages to multiple addresses. To be effective, they need to avoid detection by spam detectors. One way of doing this is to use a botnet, as described in [26], to distribute the source IPs of their emails. WordyThief, however, owns and operates their spam infrastructure. If they were to use only a few sending

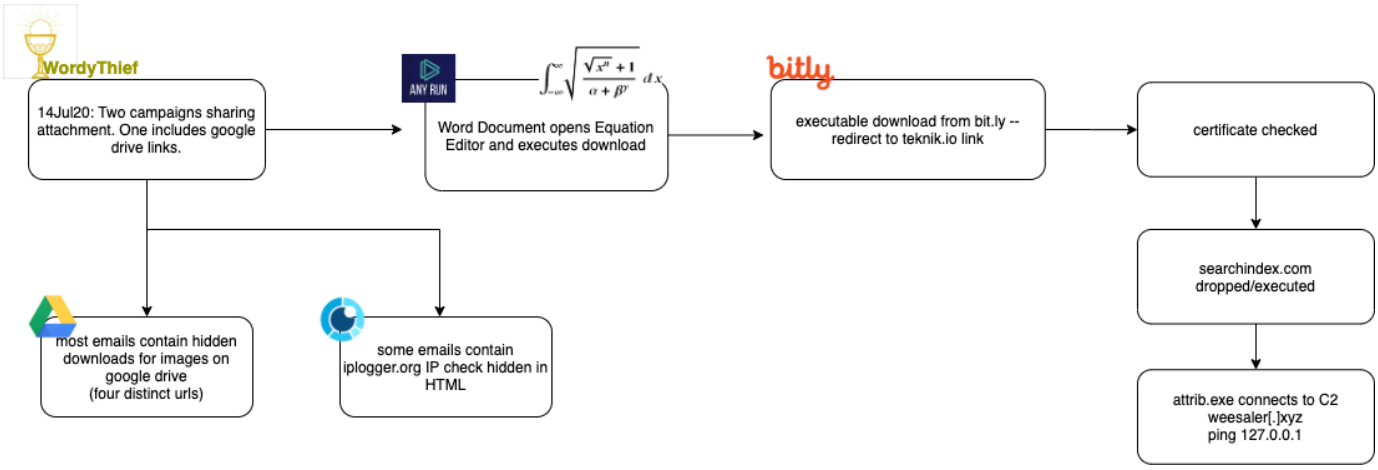

Fig. 8. A sample attack chain from a July 14, 2020 campaign by WordyThief.

IP addresses or domain names, it would make the filtering of such emails a fairly simple operation. However, the actor has to use a significant number of domain names and associated sender IP addresses in order to maintain a reliable delivery of spam messages. As the number of domains involved increases, the actor has to involve some automation for generation and registration of such domain names. We found that WordyThief solved this problem by using dictionary domain generation algorithms (DDGA).

Domain generation algorithms (DGA) are a well-known method of creating domain names frequently used by malware [31]. These algorithms generate a deterministic series of domain names based on some initial settings. In the case of traditional malware, the infected machines attempt to resolve a large number of generated domains to find a C&C location. In this classic scenario, most of the auto-generated domains are not registered and do not resolve.

Spamming actors utilize DGAs in a different way. They use DGAs to create a set of domain names for registration. The registration of a domain name is a one-time process which brings a few significant differences to the traditional use of the DGA. In particular, the spamming actor can tolerate collisions of domain registration much easier than that of automated bots, in that, if a generated domain name already exists, the spamming actor will not include it into the pool of available domain names.

The massive registration of domains is an expensive process (in both associated costs and operations time involved). This forces spamming actors to reuse the registered domains for some period. As they maintain track of domains available, they do not have to rely on the same DGA. They could utilize different algorithms or algorithm settings for each domain batch registration.

It is frequently observed that domains created by spamming actors are natural language friendly, to make addresses look more convincing. A specific group of DGA algorithms that create natural looking domains is a dictionary DGA (DDGA). These algorithms produce a domain name by combining several words drawn from some closed or open dictionary. For example, a two word DDGA might choose “blog” as the prefix word and “shed” as the tail word out of the dictionary to create the domain name “blogshed”.

A. *WordyThief Domain Name Characteristics*

The set of domain names associated with the WordyThief actor has several distinctions. We have identified a few variants of DDGAs that are used by the actor and recovered associated partial dictionaries. All of these variants combine two words into a single domain name, although we observed several different approaches. One maintains a single dictionary and draws words from it without replacement. An optional dash character may be inserted between words. The second variant maintains independent dictionaries for the first and the second word. Very frequently, the words in the same dictionary are strongly associated with some topic (colors, location, etc).

We use a graph-based technique to detect and recover the dictionary as described in [36]. The basic idea is that we build a graph where nodes are words identified as domain parts, and edges show that the words are part of the same domain name. Such representation allows for quick identification of individual dictionaries, as words belonging to the same dictionary will have a stronger connectivity compared to cross-dictionary words. This is clearly visible in fig 9. The task of segmenting domain names into words is nontrivial in a generic case. There are multiple approaches known in literature and there no perfect method. The discussion of the segmentation approaches is outside of the scope of this paper.

As an example, let’s consider a particular DGA that uses two dictionaries for the first and the second words. The word graph is shown in fig 10. The first word dictionary uses color names such as ‘tangerine’, ‘indigo’, and ‘cerule’. The tail word dictionary is a combination of a set of nouns and partial words which when appended to the prefix word, makes the domain seem like one English word. Some examples of noun tail words are ‘sphere’, ‘reach’, and ‘verse’, while some of the partial tail words are ‘tory’, ‘topia’, and ‘nity’. The DDGA

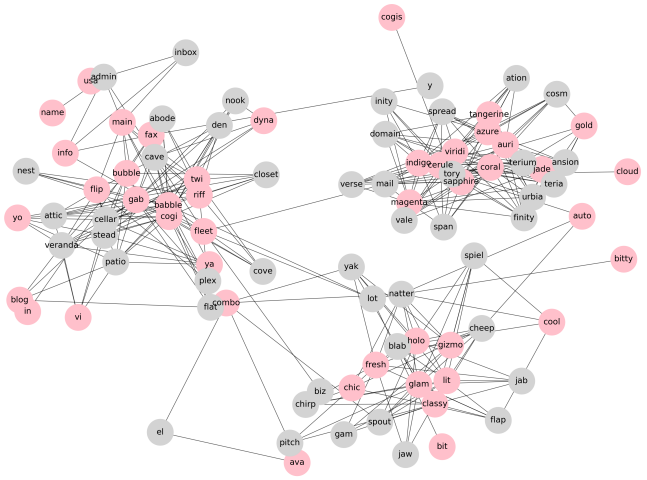

Fig. 9. Graph representation of several dictionaries. Pink words are prefix words and gray words are tail words.

uses a bipartite generational algorithm meaning that the prefix words are not used as tail words or vice versa for domain creation.

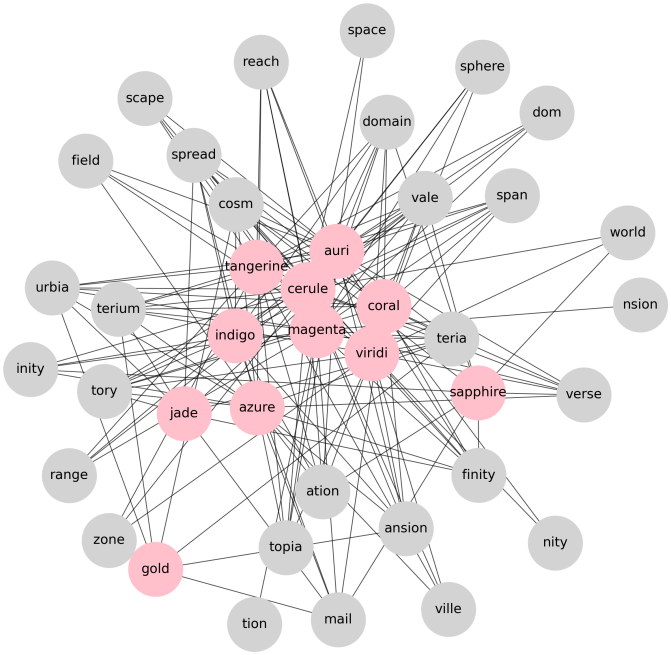

Fig. 10. Graph representation of the "color" dictionary. Pink words are prefix words and gray words are tail words.

B. Domain Resolution Activity

An alternative analysis approach that allows for associating the set of domain name to the same group is a long term activity pattern. As the spam actors reuses registered domains over a significant period of time, the resolution activity of such domains should be similar to each other.

Lets consider the "color" dictionary we have described above. We detected all domains matching with the selected prefix word in the broadband traffic starting from Jan 2020. The resolution activity of a sample of these domains is shows as a heat map in fig 11. We can easily see the pattern of initial domain use and several patterns where these domains became non-resolvable. We also can clearly see that the "goldmail" domain has a distinctly different resolution activity pattern.

VI. Conclusions

The ability to predict threats and attribute them to an actor, even of unknown identity, is a key success measure in cyber intelligence. Predicting the use of certain domains or IP addresses allows us to protect users before the actor has the opportunity to exploit them. Tying malicious behavior to a single actor, and to particular malware types, allows us further to help victims understand the breadth and magnitude of their exposure. We were able to accomplish both of these goals through the use of data science techniques and traditional threat research. Fundamental to our success is our use of "human in the loop" analytic techniques, which allow automation and machine learning to reveal threat leads, but also utilize a researcher's expertise to validate them. We were also fortunate that WordyThief made the error of using the same infrastructure for multiple purposes, allowing us to unroll their activity in a way that might not otherwise be possible.

In this research, we found that there are major actors in the malspam ecosystem who operate their own infrastructure and are involved in multiple aspects of malicious activity, unlike the roles established by Stringhini in [44]. We discovered the WordyThief actor by using graph-based techniques and were able to validate their behavior through passive DNS records, historical malspam collection, domain registration (Whois), malware reverse engineering, and open source intelligence. Consistent with the longitudinal study on abuse in DNS [29], we found that the TLDs `.xyz` and `.site` were exploited for nefarious purposes, and that the actor is able to register a large number of domains all created from a DDGA with impunity. We recovered multiple dictionaries used by this actor, allowing us to further uncover their infrastructure and predict their attacks. Further, we showed that this actor is engaged not only in the distribution of malspam, but in the operation of the malware itself, again in contrast to [44]. Within our data, we can see that the actor utilizes only information stealing malware to exfiltrate credentials and financial information from victims, and only malware associated with the authors of Predator the Thief.

All of these characteristics further our understanding of the malspam ecosystem and provide features that can be leveraged in analysis and the development of machine learning algorithms. Future work includes refining our understanding of this actor's TTP for the purpose of detecting changes in their behavior and further automation of infrastructure recovery. We also intend to compare their behaviors with other actors to determine how well the models of MaaS, particularly those described in [44] hold today. We intend, for example, to compare the topic models [41] generated from WordyThief's emails to other malspam in an effort to characterize the variation and

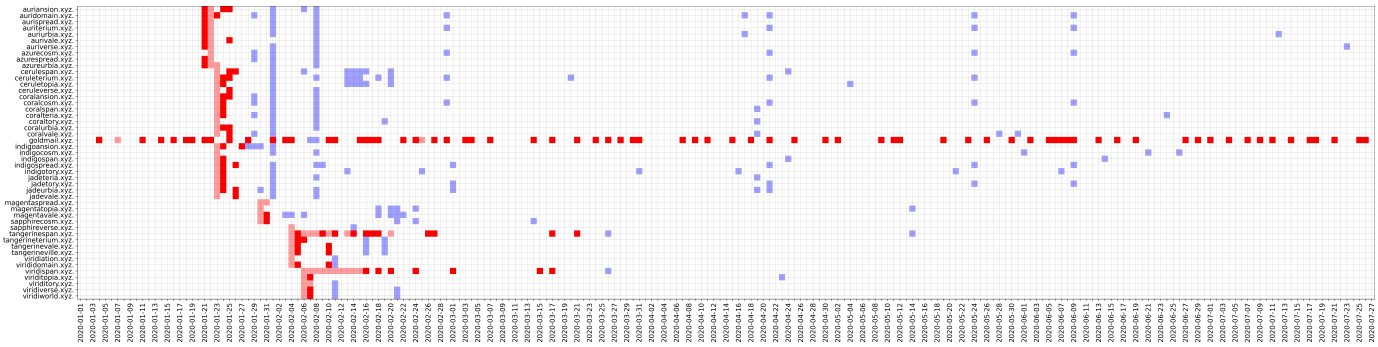

Fig. 11. Domain resolution activity of a sample of of `.xyz` domains with partial match to "colors" dictionary over time. Pink color represents that the domain was successfully resolved less than five times per day. Red color means the domain was resolved five or more times. Blue color means the domain resolution request returned NXDOMAIN response.

creation of lures. Extending our time series analysis, graph-based discovery, and recovery of DDGA structure will further our understanding of the threat landscape and ability to protect users from cyber criminals.

Acknowledgment

The authors would like to thank Jon Armer, who made the initial connection from our malspam sample to Taurus Project, and Darby Wise, for her invaluable technical editing.

References

[1] *17-Year Old MS Office Flaw (CVE-2017-11882) Actively Exploited in the Wild*. Dec. 8, 2017. URL: https://www.trendmicro.com/vinfo/us/security/news/vulnerabilities-and-exploits/17-year-old-ms-office-flaw-cve-2017-11882-actively-exploited-in-the-wild.

[2] 3xp0rtblog. URL: https://twitter.com/3xp0rtblog/status/1254079067810336768.

[3] 3xp0rtblog. URL: https://twitter.com/3xp0rtblog/status/1275746149719252992.

[4] Lawrence Abrams. *Emotet spam trojan surges back to life after 5 months of silence*. URL: https://www.bleepingcomputer.com/news/security/emotet-spam-trojan-surges-back-to-life-after-5-months-of-silence/.

[5] I Adegbola and R Jimoh. "Spambot detection: A review of techniques and trends". In: *network* 6.9 (2014).

[6] *Apache Spark*. URL: https://spark.apache.org/.

[7] J. Armer. *Malspam Campaigns Deliver Predator the Thief InfoStealer*. Jan. 22, 2020. URL: https://insights.infoblox.com/threat-intelligence-reports/threat-intelligence--55.

[8] Uday Pratap Singh Avinash Kumar. *Taurus: The New Stealer in Town*. URL: https://www.zscaler.com/blogs/research/taurus-new-stealer-town.

[9] J. Barnett. *New Ransomware - Avaddon*. June 17, 2020. URL: https://insights.infoblox.com/threat-intelligence-reports/threat-intelligence--79.

[10] J. Barnett. *Rig Exploit Kit Drops Predator The Thief InfoStealer and CrySIS Ransomware*. Nov. 19, 2019. URL: https://insights.infoblox.com/threat-intelligence-reports/threat-intelligence--50.

[11] J. Barnett. *Valak Downloader/InfoStealer Delivers IcedID Banking Trojan*. July 15, 2020. URL: https://insights.infoblox.com/threat-intelligence-reports/threat-intelligence--78.

[12] Jonathan Bennett. *AutoIt*. URL: https://www.autoitscript.com/site/.

[13] R. Burton. *Graph Techniques for Threat Intelligence Workflows*. to appear November 2020. URL: https://infoblox.com/.

[14] R. Burton, J. Armer, M. Hoang, and V. Tymchenko. *New Malware Variant: Project Taurus Infostealer Follows in Predator the Thief's Footprints*. June 8, 2020. URL: https://insights.infoblox.com/threat-intelligence-reports/threat-intelligence--74.

[15] *Cloudflare Website*. URL: https://www.cloudflare.com.

[16] Crowdstrike. *2020 Global Threat Report*. URL: https://www.newcastle.edu.au/__data/assets/pdf_file/0006/616875/2020_Global-Threat-Report.pdf.

[17] W. L. Cukier, S. Cody, and E. J. Nesselroth. "Genres of Spam: Expectations and Deceptions". In: *Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06)*. Vol. 3. 2006, 51a–51a.

[18] National Cybersecurity Department of Homeland Security and Communications Integration Center. *Malware Trends*. URL: https://us-cert.cisa.gov/sites/default/files/documents/NCCIC_ICS-CERT_AAL_Malware_Trends_Paper_S508C.pdf.

[19] *Farsight Security*. URL: https://farsightsecurity.com.

[20] Emilio Ferrara. "The history of digital spam". In: *Communications of the ACM* 62.8 (2019), pp. 82–91.

[21] Pawel Foremski and Paul Vixie. "The Modality of Mortality in Domain Names". In: *Virus* (2018), p. 1.

[22] fr3dhk. URL: https://threatshare.io/malware/.

[23] *Frankfurt shut down IT network following Emotet infection*. URL: https://www.zdnet.com/article/frankfurt-shuts-down-it-network-following-emotet-infection/.

[24] Sam Greengard. *Emotet as a Service: A Serious New Cyber Threat*. Apr. 2020. URL: https://www.mimecast.com/blog/2020/04/emotet-as-a-service-a-serious-new-cyber-threat/.

[25] *Infoblox Threat Intelligence Data Exchange for Active Trust Suite*. SN-0218-03 0318. URL: https://www.infoblox.com/wp-content/uploads/infoblox-solution-note-infoblox-threat-intelligence-data-exchange-for-activetrust.pdf.

[26] Jarosław Jedynak and Maciej Kotowicz. *Peering into spam botnets.* URL: https://www.virusbulletin.com/uploads/pdf/magazine/2017/VB2017-Jedynak-Kotowicz.pdf.

[27] Yanhui Jia. *Analysis of CVE-2017-11882 Exploit in the Wild*. URL: https://unit42.paloaltonetworks.com/unit42-analysis-of-cve-2017-11882-exploit-in-the-wild.

[28] Peter Kacherginsky. *FakeNet-NG: Next Generation Dynamic Network Analysis Tool*. URL: https://www.fireeye.com/blog/threat-research/2016/08/fakenet-ng_next_gen.html.

[29] M Korczy'ski, Maarten Wullink, Samaneh Tajalizadehkhoob, Giovane CM Moura, and Cristian Hesselman. *Statistical Analysis of DNS Abuse in gTLDs Final Report*. Tech. rep. Technical Report. https://www.icann. org/en/system/files/files/sadag-final ..., 2017.

[30] Philippe Lagadec. *Python Oletools*. Version 0.55. Dec. 3, 2019. URL: https://github.com/decalage2/oletools.

[31] Hongliang Liu and Yuriy Yuzifovich. *A Death Match Of Domain Generation Algorithms*. Jan. 9, 2018. URL: https://blogs.akamai.com/2018/01/a-death-match-of-domain-generation-algorithms.html.

[32] *Malware Analysis of Dridex, BitPaymer and DoppelPaymer Campaigns*. URL: https://lifars.com/2019/11/analysis-of-dridex-bitpaymer-and-doppelpaymer-campaign/.

[33] *Namecheap Website*. URL: https://www.namecheap.com.

[34] *New action to combat ransomeware ahead of US elections*. URL: https://blogs.microsoft.com/on-the-issues/2020/10/12/trickbot-ransomware-cyberthreat-us-elections/.

[35] NIST. *CVE-2017-11882 Detail*. Nov. 14, 2017. URL: https://nvd.nist.gov/vuln/detail/CVE-2017-11882.

[36] Mayana Pereira, Shaun Coleman, Bin Yu, Martine DeCock, and Anderson Nascimento. "Dictionary extraction and detection of algorithmically generated domain names in passive DNS traffic". In: *International Symposium on Research in Attacks, Intrusions, and Defenses*. Springer. 2018, pp. 295–314.

[37] Nathan Popper. "Ransomware Attacks Grow, Crippling Cities and Businesses". In: *New York Times* (Feb. 2020). URL: https://www.nytimes.com/2020/02/09/technology/ransomware-attacks.html.

[38] *Predator the Thief*. URL: https://malpedia.caad.fkie.fraunhofer.de/details/win.predator.

[39] *Predator the Thief*. URL: https://research.checkpoint.com/2020/predator-the-thief/.

[40] T.V. Antony Raj. *Want to Spam? Here are a Few Templates You Dan Use*. URL: https://tvaraj.com/2013/05/21/want-to-spam-here-are-a-few-templates-you-can-use/.

[41] Radim Řehůřek and Petr Sojka. "Software Framework for Topic Modelling with Large Corpora". English. In: *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. http://is.muni.cz/publication/884893/en. Valletta, Malta: ELRA, May 2010, pp. 45–50.

[42] *RU Registrar Website*. URL: https://nic.ru.

[43] Paul Sroufe, Santi Phithakkitnukoon, Ram Dantu, and João Cangussu. "Email shape analysis for spam botnet detection". In: *2009 6th IEEE Consumer Communications and Networking Conference*. IEEE. 2009, pp. 1–2.

[44] Gianluca Stringhini, Oliver Hohlfeld, Christopher Kruegel, and Giovanni Vigna. "The harvester, the botmaster, and the spammer: on the relations between the different actors in the spam landscape". In: *Proceedings of the 9th ACM symposium on Information, computer and communications security*. 2014, pp. 353–364.

[45] N. Sundvall. *Vidar InfoStealer*. July 21, 2020. URL: https://insights.infoblox.com/threat-intelligence-reports/threat-intelligence--79.

[46] *SURBL: URI Reputation Data*. last accessed May 22, 2019. URL: www.surbl.org.

[47] Emre Suren and Pelin Angin. "Know Your EK: A Content and Workflow Analysis Approach for Exploit Kits." In: *J. Internet Serv. Inf. Secur.* 9.1 (2019), pp. 24–47.

[48] SophosLabs Research Team et al. "Emotet exposed: looking inside highly destructive malware". In: *Network Security* 2019.6 (2019), pp. 6–11.

[49] *The Rise of Information Stealers*. URL: https://www.webroot.com/blog/2019/01/31/the-rise-of-information-stealers/.

[50] Cyber Crime Tracker. *Taurus Project C2 Domains*. URL: http://cybercrime-tracker.net/index.php?s=40&m=40&search=taurus.

[51] *VbaAutoOpen*. URL: https://docs.microsoft.com/en-us/office/vba/word/concepts/customizing-word/auto-macros.

[52] *What is PowerShell?* May 22, 2020. URL: https://docs.microsoft.com/en-us/powershell/scripting/overview?view=powershell-7.