Marked for Disruption: Tracing the Evolution of Malware Delivery Operations Targeted for Takedown

Colin C. Ife★, Yun Shen†, Steven J. Murdoch★, and Gianluca Stringhini‡
★University College London, †Norton Research Group, ‡Boston University
★†United Kingdom, ‡United States
{colin.ife,s.murdoch}@ucl.ac.uk,yun.shen@nortonlifelock.com,gian@bu.edu

ABSTRACT

The malware and botnet phenomenon is among the most significant threats to cybersecurity today. Consequently, law enforcement agencies, security companies, and researchers are constantly seeking to disrupt these malicious operations through so-called *takedown* counter-operations. Unfortunately, the success of these takedowns is mixed. Furthermore, very little is understood as to how botnets and malware delivery operations respond to takedown attempts. We present a comprehensive study of three malware delivery operations that were targeted for takedown in 2015–16 using global download metadata provided by Symantec. In summary, we found that: (1) Distributed delivery architectures were commonly used, indicating the need for better security hygiene and coordination by the (ab)used service providers. (2) A minority of malware binaries were responsible for the majority of download activity, suggesting that detecting these "super binaries" would yield the most benefit to the security community. (3) The malware operations exhibited *displacing* and *defiant* behaviours following their respective takedown attempts. We argue that these "predictable" behaviours could be factored into future takedown strategies. (4) The malware operations also exhibited previously undocumented behaviours, such as `Dridex` dropping competing brands of malware, or `Dorkbot` and `Upatre` heavily relying on upstream dropper malware. These "unpredictable" behaviours indicate the need for researchers to use better threat-monitoring techniques.

CCS CONCEPTS

• Security and privacy → Malware and its mitigation; **Social aspects of security and privacy**; **• Information systems → World Wide Web**;

ACM Reference Format:
Colin C. Ife★, Yun Shen†, Steven J. Murdoch★, and Gianluca Stringhini‡. 2021. Marked for Disruption: Tracing the Evolution of Malware Delivery Operations Targeted for Takedown. In *24th International Symposium on Research in Attacks, Intrusions and Defenses (RAID '21), October 6–8, 2021, San Sebastian, Spain.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3471621.3471844

1 INTRODUCTION

Malware delivery has evolved into a major business for the cybercriminal economy and a complex problem for the security community. The *botnet* – a network of malware-infected devices that is controlled by a single actor through one or more command and control (C&C) servers – is one phenomenon that has benefited from the malware delivery revolution. Diverse distribution vectors have enabled such malicious networks to expand more quickly and efficiently than ever before. Once established, these botnets can be leveraged to commit a wide array of secondary computer crimes, such as data theft, financial fraud, coercion (ransomware), sending spam messages, distributed denial of service (DDoS) attacks, and unauthorised cryptocurrency mining [1, 14, 17, 47, 48]. Even worse, these botnets could be further monetised as *pay-per-install* services [19], allowing the operator to rent out access of their network to other criminals. Finally, botnet operators employ a myriad of techniques to avoid detection and improve the resiliency of their operations, such as using software polymorphism [16] to beat antivirus engines, Fast Flux Service Networks (FFSNs) [29] to rapidly change the IP addresses of their servers, Domain Generation Algorithms (DGAs) [15] to constantly change the domain names of their C&C servers, and distributed servers spanning multiple regions for redundancy and elusive software delivery [35, 41].

Because of the serious and growing threat that botnet and malware delivery operations pose to society, law enforcement agencies (LEAs), security companies, and researchers constantly seek methods, opportunities, and intervention points to disrupt such malicious operations [25, 36]. Takedown operations are just a subset of some of the disruptive techniques employed: infiltrating botnets for intelligence-gathering and sabotage; re-routing network traffic meant for known C&C servers to disrupt their communication channels (i.e., a DNS sinkhole); forcing Internet service providers (ISPs) to shutdown malicious servers that they host; or physically seizing malicious server infrastructure and assets, and arresting the miscreants involved. The success of these operations is mixed [25].

Although the details of a number of takedown operations have been recorded in the literature, few studies examine how the targeted malware delivery operations actually respond to such interventions. This leaves many important questions unanswered: After a takedown operation, what happens next? Do the malware operations break down? If not, how quickly do they resurface? Do the operators move their infrastructure elsewhere, or perhaps change their modus operandi? Assessing the efficacy of takedown operations, are there more effective intervention points in these malicious infrastructures? Finally, considering the behaviours of these miscreants, could some of their reactions be predicted and taken into account by LEAs and security practitioners?

RAID '21, October 6–8, 2021, San Sebastian, Spain

ACM ISBN 978-1-4503-9058-3/21/10…$15.00
https://doi.org/10.1145/3471621.3471844

In this study, using global download metadata collected in 2015–2016, we devise a novel tracking and analysis methodology to quantitatively assess the global activity of malware delivery operations targeted for takedown, and how this activity evolves over the course of a year in light of these actions. In particular, we focus on three malware delivery operations (botnets) that were targeted in the fall of 2015: the `Dridex`, `Dorkbot`, and `Dyre-Upatre` operations. These botnets were selected as they were among the few known to have been targeted for takedown between October 2015–September 2016, corresponding to the collection period of the dataset used herein. We analyse the activities of two operational components over the course of a year: the upstream server infrastructure (server-side), and the downloaded binaries and their dropper networks (client-side). In summary, this study makes the following contributions:

(1) We devise a novel methodology to track and analyse malware delivery operations over time using download metadata. This methodology could be used to analyse any class of software delivery operation at scale, such as malware, potentially unwanted programs (PUPs), or benignware.

(2) We observe a myriad of behavioural responses to takedown attempts by each malware delivery operation. Specifically, we find that: (1) The use of distributed delivery architectures was common among the studied malware. (2) A minority of malware binaries were responsible for the majority of download activity. (3) The malware operations exhibited some "predictable" behaviours following their respective takedown attempts such as *displacement* [28] and *defiance* [43] behaviours. (4) The malware operations also exhibited previously undocumented behaviours, indicating the need for the research community to use better monitoring techniques.

This study gives the security community deeper insight into the dynamics and complexities within malware delivery operations, particularly in light of a takedown attempt, while also uncovering challenges and further opportunities to disrupting such operations.

2 RELATED WORK

Botnet takedowns are counter-operations to disrupt botnet activities and the malware delivery networks that enable their growth. There are diverse techniques to taking down botnets: botnet infiltrations [46], ISP takedowns [20], DNS sinkholes [46], and arrest and seizure. However, the fundamental problem with botnet takedowns is that if the botnet is not taken down fully or its operators not prosecuted, the operators may simply revive their operations and make them more resilient, making the task of taking down the botnet more difficult the next time round. Because of this, various studies have attempted to quantify the effects of takedowns. Clayton [23] examined email statistics from a medium-sized UK ISP to assess the effects of the 2008 McColo takedown on global spam volume. It was found that significant reductions in spam email volumes around the time of the takedown operation. However, it was also found that particular types of spam detection mechanisms employed by this ISP ceased to be as effective. Dittrich [24] conducted a broader study, qualitatively analysing a set of highly publicised botnet takedown efforts between 2009-2011. It was concluded that, while some takedown strategies are more effective than others, the arms race between security practitioners and cybercriminals will continue to make botnet takedowns more expensive and difficult as cybercriminals will continue to make their infrastructures more resilient. The author called for more coordination and shared knowledge between the security community to make botnet takedowns more efficient and sustainable.

In an attempt to bring measurement and order to botnet takedown analysis, a takedown analysis and recommendation system, *rza*, is proposed by Nadji *et al.* [36]. This system allows researchers to conduct a post-mortem analysis of past botnet takedowns, and provide recommendations on how to execute future ones successfully. This work is motivated with some real case studies. In a second work, Nadji *et al.* [37] propose improvements to the *rza* system by enhancing its risk formula to include botnet population counts. Two additional botnet takedowns are also examined, and the policy ramifications of takedowns are discussed in detail by the authors. Lerner [34] also discusses regulatory and policy solutions to botnet takedowns, particularly arguing the need for more public-private partnerships to achieve this endeavour. Shirazi [44] surveys and taxonomises 19 botnet takedown initiatives between 2008–2014 and proposed a theoretical model to assess the likelihood of success for future botnet takedown initiatives. To the best of our knowledge, the author is still in the process of building this database before releasing it to the security community.

Investigating the effects of takedowns further, a recent historical study by Edwards *et al.* [25] was conducted on the causal effects of botnet takedowns on ISPs that hosted spamming activity. In this work, the authors build an autoregressive model for each ISP to model *wickedness* – a metric defined as total spam released per ISP – as a function of (i) external factors and (ii) each takedown that occurred as represented as a time-lagged step-function. They find that, for most takedowns, the effect of a takedown is minimal after a period of 6 weeks. However, takedowns with a seizure element appear the most effective over the long-term. They also find evidence of a takedown in one region causing a diffusion of benefits and crime in others.

3 DATASETS

3.1 Download Metadata

We use a download activity dataset provided by Symantec. This anonymised dataset consists of download data from 12 million end-users of Symantec's products between October 1st, 2015 and September 29th, 2016. These users explicitly opted into the data-sharing programme, which does not include personally identifiable information. These participating users periodically report metadata information on the binaries that they download, offering rich information regarding the time at which a binary is downloaded, which server it is downloaded from, and which program initiated the download activity. Equation 1 outlines the structure of a download event:

$$\mathbf{d} =< F_f, A_f, U_r, U_f, D, I, F_p, U_p > \tag{1}$$

where F_f is the downloaded file identified by its SHA-2; A_f represents a set of attributes which provides additional information about file F_f, such as its filename, its size (in bytes), and its "reputation" and "prevalence" scores assigned by Symantec's analysis systems (see Section 3.2); U_r is the initial (referrer) URL in an HTTP redirection chain; U_f is the download URL (after removing URL parameters) while I is the IP address of the download server and D its fully qualified domain name (FQDN); F_p is the SHA-2 of the parent file (or *dropper*) and U_p the source URL of the parent file. In total, 81.5 million download events were observed over 53 days, which were sampled one day per week from October 1st, 2015.

Note that this study focuses on malicious file downloads. To this end, we leverage the reputation scores assigned to files by Symantec, discard any file that has a high (benign) reputation score and is not confirmed as malicious by VirusTotal[1], and only keep the files involved in the delivery of malware or PUP. Note that we consider a file malicious if at least one of the top five AV vendors by market share (in no particular order, Avast, AVG, Avira, Microsoft, and Symantec) and a minimum of two other AVs detect it as malicious. Other works have used a similar technique [31, 38, 49]. We also filter out IP addresses that are not valid for public use as well aberrant data (e.g. events with no parent file or network resource).

3.2 Binary Ground Truth

We utilise a variety of ground truth to establish whether files are malware or PUP, and to which families they belong. This allows us to track the evolution of different malware and PUP delivery operations, especially in response to different mitigation strategies.
Reputation and Prevalence Scores. Symantec employs extensive static and dynamic analysis systems to determine the maliciousness of a binary, as well as estimate its prevalence in the wild.
VirusTotal. We query VirusTotal [6] with each file SHA-2 to obtain the number of AV vendors that flag the file as malicious and the malware or PUP family labels designated to it by each vendor. VirusTotal can sometimes take several months (or years) to flag malicious files in the wild due to coverage issues [33, 35, 39]. As such, this analysis is conducted 3–4 years after the data is first collected. This makes sense since we seek maximise our ground-truth data (namely family labels) so as to characterise the evolution of different malware and PUP operations as accurately as possible.
AVClass. In conjunction with VirusTotal, we utilise the AVClass malware labelling tool [42] to remove “noisy” and conflicting family labels for a given file so as to determine a correct and consistent one. For example, multiple AV engines may generate labels of `Adware.Rotator.F`, `Adware.Generic`, and `Adware.Adrotator.Gen!Pac` for the same AdRotator SHA-2 (a PUP). We utilised an updated set of AVClass family labels at the time of this study.[2]
National Software Reference Library. NSRL Reference Data Set (RDS) version 2.67 provides us SHA-2 hashes of known benign and reputable programs involved in malicious file delivery.

4 TARGETED MALWARE OPERATIONS

In this study, we seek to understand how malware operations evolve in light of takedown operations against them. However, it is important to first identify takedowns that occurred within the dataset collection period, i.e., between 1st October, 2015 and 29th September, 2016. We identify three different botnets that were targeted for takedown within the subject period: Dridex, Dorkbot, and the Dyre-Upatre malware delivery operations.

Dridex

The Dridex malware (also known as Bugat, Cridex, Drixed, and Dridexdownloader) is a banking trojan and botnet malware, specifically designed to steal banking credentials and other personal information on a compromised system. Dridex has been known to spread through phishing emails as a malicious attachment, to self-replicate by copying itself from compromised devices to mapped network drives and local storage devices [3], and to be delivered through exploit kits on compromised web servers [5]. Between August and October, 2015, one of the botnet operators was arrested, while the NCA in the UK undertook a DNS sinkhole operation against Dridex servers. Between 9th October and 8th December, 2015, LEAs conducted a second *DNS sinkhole* and *disinfection* campaign against Dridex, though the specifics are unknown [2, 9].

Dorkbot

The Dorkbot malware is a family of worms known to steal data from compromised systems, disable security applications, and form botnets to distribute other types of malware [8, 11]. It has been known to propagate through infected USB flash drives, instant message applications, social networks, spam messages, and exploit kits. Researchers identified Dorkbot diversifying its C&C servers to multiple regions, such as throughout Europe, Asia, and North America [7, 11]. Around 3rd December 2015, security companies and LEAs around the world conducted a swift *DNS sinkhole* and *seizure* operation against the Dorkbot botnet [7, 10, 12, 12].

Dyre and Upatre

The Dyre and Upatre operations provide an interesting case study, not least given the reported LEA operation against the Dyre botnet coincided with a sudden, global drop in malicious download activity, which was observed in an earlier study [31]. Dyre (also known as Dyreza, Dyzap, and Dyranges) is a sophisticated financial fraud trojan that targets Windows computers [50]. However, most notably, security researchers have identified the Dyre-Upatre relationship as being key to its operation, where, after hosts are infected with Upatre malware, it proceeds to install Dyre malware onto these devices [4, 32, 50]. More specifically, Upatre is a dedicated dropper malware: once on a victim machine, its sole purpose is to deliver additional malware components onto it. However, besides delivering Dyre samples, Upatre has been known to distribute other malware families such as GameOver Zeus, Kegotip, Locky, and Dridex [13].

In this study, we focus only on the activities of the Upatre dropper as little to no observable Dyre download activity is found in this dataset. Why exactly this is the case is not known. Since Dyre was known to undergo rapid polymorphism [50], it could be indicative of the inability of antivirus engines to keep up with its high churn of malware binaries, or some form of measurement error with the telemetry sensors used to collect this dataset. Around 18th November 2015, law enforcement officials conducted a *seizure* and *arrest* operation against the Dyre operators in Moscow, Russia [18].

5 METHODOLOGY

Although the principal focus of this study is the dynamics of three specific malware delivery operations targeted for takedown, we devise a methodology that may be adopted to analyse any class of

[1] VirusTotal is a free online service that analyses submitted files and URLs across different antivirus engines and website scanners.
[2] Commit 21806f3 from https://github.com/malicialab/avclass (July 27th, 2018)

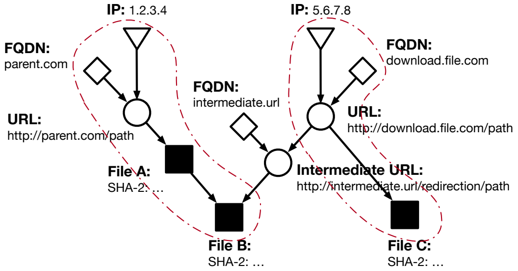

Figure 1: A legend to interpret download graphs, adapted from [31]. Two series of download events are highlighted.

file delivery operation, whether malicious or benign. Therefore, in this section, we detail the steps to (i) build the download graphs for the year-long dataset; (ii) classify the file nodes as either malware, potentially unwanted programs (PUPs), or benign, along with their specific software brands/families; and (iii) aggregate and track each software delivery operation in time, with a particular focus on their evolving use of *delivery infrastructure* and their *dropping behaviours*.

It is pertinent to note that, in this study, we only seek to analyse file delivery *operations* – not file delivery *campaigns*. More precisely, we only analyse aggregate (global) file delivery activity pertaining to a given software family (e.g., all *Zeus* malware delivery activity). This is opposed to the more fine-grained analysis of individual clusters of activity (campaigns) pertaining to a single software family (e.g., individual *Zeus* botnet campaigns, which may involve independent operators by virtue of its crimeware-as-a-service business model [45]). We align with the above distinction between the terms *operation* and *campaign* for the purposes of this study. As such, disentangling individual delivery campaigns (and the respective actors) for a given operation is beyond the scope of this study.

5.1 Building Download Graphs

We adopt the graph-building methodology used in similar work [31]. In summary, we build a directed graph $G = (V, E)$ where V is a vertex list representing different entities (file SHA-2s, URLs, IPs, and FQDNs) and E is an edge list representing relationships between nodes from the same download events. An example of this graph schema is shown in Figure 1, which includes FQDNs. Taking download event 1, for example, `File A` is a dropper that was downloaded from `parent.com/path` with IP `1.2.3.4`. `File A` initiates the download of `File B`, it first makes a request to `intermediate.url`, which redirects to the terminal URL `download.file.com/path`.

5.2 File Classification

Having constructed the download graphs for each observation window, we build on the file classification technique used in previous work [31]. Specifically, we label each file node (based on its SHA-2) as either *malware*, *PUP*, or *benign* using the ground truth sources outlined in Section 3.2, or leave it as *unlabelled*. If known, we also specify the *software family* to which the SHA-2 belongs, whether malicious or benign. Otherwise, we label SHA-2s without known family labels as *singletons*. In total, we classify 1,034,763 malicious file SHA-2s (4.83% of all files), 443,541 (2.07%) of which are classified as malware, and the remainder as PUP. On the other hand, 350,517 SHA-2s (1.64%) are known to be benign, as either VirusTotal flags them as not malicious (349,746 files), and/or the NSRL maintains that they are reputable (9,007 files).

5.2.1 *Aggregating Family Aliases.*

A major part of this study is to analyse the activities of three malware delivery operations: Dridex, Dorkbot, and Upatre. It is common for some antivirus engines to label each malware family differently, which may lead to multiple aliases being observed that refer to the same malware family. Therefore, we configure the AVClass tool to map specific aliases to specific families. Specifically, based on the sources for each malware operation in Section 4, we aggregate the following aliases:

- Dridex, Cridex, Bugat, Drixed, Dridexdownloader ⟶ `Dridex`;
- Dorkbot, Ngrbot ⟶ `Dorkbot`; and
- Upatre ⟶ `Upatre`.

Other known aliases for these families that are ambiguously designated (i.e., used to refer to several, independent malware families) or were not observed in the dataset were omitted.

5.3 Tracking and Analysing Operational Activity

Besides just monitoring malicious file presence, we want to establish how their use of delivery infrastructure and their dropping behaviours evolve alongside them. It is particularly interesting to understand the evolution of malicious file delivery operations in the wake of different, disruptive strategies being applied against them, such as botnet takedowns or coordinated arrests. We achieve this goal in two stages. First, we devise a methodology to identify and track a (malicious) file delivery operation. And second, we derive a set of metrics that describe different aspects of a given file delivery operation, and conduct time series analysis on these metrics.

5.3.1 *Tracking Delivery Operations.*

Our approach to tracking delivery operations is simple: For a (target) software family that we seek to analyse, SF, and for the ith observation period, where $i \in [1..53]$ (i.e., every Thursday for a year), we conduct the following:

1. We compute F_i^{SF}: the set of all file nodes pertaining to software family SF in observation period i.
2. We compute P_i^{SF}: the set of all parent nodes (URLs, IPs, FQDNs, parent files) involved in the download events that deliver the files F_i^{SF} in observation period i. In terms of real-world actors, these parent nodes could be attributed to, for example, upstream hosting services, compromised websites, or pay-per-install network operators and affiliates [19].
3. Likewise, we compute C_i^{SF}: the set of all child nodes (files) that are dropped by the files in F_i^{SF} in observation period i. Being payloads, these child nodes could be attributed to the clients of the SF delivery network.
4. Finally, we compute the node attribute look-up table, A_i^{SF}, which stores the attributes of all nodes forming the delivery network of software family SF in each observation period i (e.g., family, # of downloads/drops, country, domain name).

5.3.2 *Time Series Analysis.*

We seek to generate a set of metrics (or features) which sufficiently describe the different aspects of a file delivery operation. Using the data structures, F^{SF}, P^{SF}, C^{SF},

Metric	Description
	Aggregate Network Activity*
URL count	Total no. of URLs used in file delivery.
FQDN count	Total no. of FQDNs used in file delivery.
E2LD count used	Total no. of e2LDs used in file delivery.
IP count	Total no. of IP addresses used by file delivery servers.
Country count	Total no. of countries associated with file delivery servers.
	Evasion Indicators*
IP count per e2LD used	No. of IPs associated with each e2LD used in file delivery.
E2LD count per IP used	No. of e2LDs associated with each IP used in file delivery.
	Aggregate Download Activity†
Download count	Total no. of times the target family is downloaded.
Drop count	Total no. of times the target family delivers other files.
Download count per SHA-2	No. of times each target family SHA-2 is downloaded.
Drop count per SHA-2	No. of times each target family SHA-2 delivers other files.
	Relational Dynamics†
Parent SHA-2 count	Total no. of SHA-2s used to deliver the target family.
Child SHA-2 count	Total no. of SHA-2s delivered by target family.
	Distributed Delivery Indicators†
URL count per SHA-2	No. of URLs used to deliver each target family SHA-2.
IP count per SHA-2	No. of IPs used to deliver each target family SHA-2.
E2LD count per SHA-2	No. of e2LDs used to deliver each target family SHA-2.
	Polymorphism Indicators†
SHA-2 count	No. of target family SHA-2s observed.
SHA-2 churn	No. of SHA-2s in observation i lost in observation $i + 1$.
File size per SHA-2	File size of each SHA-2 in kilobytes.
Reputation score per SHA-2	Malice score assigned to each SHA-2 by Symantec.
Prevalence score per SHA-2	Prevalence score assigned to each SHA-2 by Symantec. N.B: Prevalence indicates how often a SHA-2 is detected.

Table 1: The network* and downloader† metrics used to analyse each malware delivery operation.

and A^{SF} as defined above, we compute and analyse time series data based on two groups of metrics:

Network dynamics. These metrics capture the dynamics of server-level activity in the file delivery operation. The numbers of URLs, domains, IPs, and countries used to host delivery servers indicate the pervasiveness and extent of resources used for each operation. The numbers of IPs associated with each domain provide indicators of use of the Fast Flux technique (rapidly changing IPs) [29], or the use of content distribution networks (CDNs) and multi-region servers – common methods used by botnet to avoid detection and increase resiliency [41]. On the other hand, the number of domains associated to any given IP could be an indicator of use of shared-hosting clusters, or servers using domain generating algorithms (DGA) – another commonly used technique by C&C servers to avoid detection [40]. Finally, we also quantify the most popular domains, IP addresses, and regions used in each operation.

Downloader dynamics. These metrics capture information relating to the software family in question and the binaries it uses to drive the delivery operation. Specifically, we obtain the total and per-SHA-2 counts of download and dropping events for the software family – key performance indicators of delivery operations. We also keep track of the total and top N families involved in the software family's download activities. Further, we analyse the numbers of URLs, domains, and IPs used to deliver each file SHA-2, which are all indicative of the use of diverse distribution vectors, perhaps to increase outreach to end-users, or to evade detection systems more effectively. We also examine metrics that indicate polymorphism (a malware characteristic to evade detection [16]): the number of SHA-2s observed, their churn rates, and the distributions of their file sizes, malice (reputation) scores, and prevalence scores (as detected and assigned by Symantec). It should be noted that a higher malice score corresponds to a higher likelihood that a file is malicious (see Section 3.2), while a higher prevalence score indicates that a file is detected more frequently.

These metrics are summarised in Table 1. We analyse the time series derived from these metrics in Section 6.

In summary, this methodology and rich dataset grants us an unprecedented insight into the dynamics of malicious file delivery operations, the business relationships between them, and, most importantly, how they each react to disruptive counter-operations.

6 ANALYSIS

In this section, we use the techniques described in Section 5.3 to analyse the evolution of three different malware delivery operations: the Dridex, Dorkbot, and Dyre-Upatre botnets. Each botnet faced law enforcement takedown attempts between October 2015 and September 2016. For each malware delivery operation, we mark the time period of the associated takedown operation and analyse the botnet's activities afterwards. We focus our analysis on two types of metrics: the network dynamics, and the downloader dynamics.

6.1 Network Dynamics

We begin our analysis with the upstream infrastructure of each malware delivery operation. To this end, we compute and analyse the network dynamic metrics described in Section 5.3.2. Figure 2 shows a number of time series denoting aggregate network dynamics, and

Figure 3 evasion indicators for each malware delivery operation. We note some apparent features. For instance, Dridex exhibits consistent growth in all forms of network activity from early October 2015 (despite the DNS sinkhole operation) until the end of February 2016, after which its network activities tail off. On the other hand, the Dorkbot and Upatre operations (which faced "seizure" counter-operations) both exhibit significant drops in overall network activity in the short-term, with varying long-term responses. This is consistent with the findings of other researchers [25] in that, though botnet responses to takedowns are highly variable, takedowns that involve the physical seizure of botnet infrastructure are usually associated with longer-lasting and more significant effects.

6.1.1 Dridex. Looking more deeply into the network dynamics of the Dridex operation, we see two distinct stages of activity. Initially, there is a consistent increase and diversification of its server usage in all respects, specifically over the course of the 60-day DNS sinkhole counter-operation (1st October–3rd March). This is followed by a stage of consistent decrease in network activity from 3rd March–29th September. We find some interesting observations.

The first observation is that the period of consistent growth in malicious server activity seems to align with the 60-day period of the DNS sinkhole. On the other hand, once this sinkhole operation concluded in early December 2015, Dridex server usage appears to fall and rise for a number of weeks. Why this sequence of events occurs is unclear. One would typically expect malicious server activity to decrease or to remain at a controlled level during a DNS sinkhole operation, as observed by other researchers [25]. This is clearly not the case in this instance, where we observe the opposite. Nonetheless, one must consider (at least) two factors regarding this observation. First, the DNS sinkhole operation itself may not have been effected adequately or consistently. It is possible that the Dridex operators switched to back-up or alternative servers that were not tracked by the agencies enforcing the sinkhole counter-operation. At the same time, it is possible that the C&C servers targeted for DNS sinkholing were separate to the servers used to deliver Dridex malware to the end-hosts. If this was the case, this could highlight a significant limitation of DNS sinkholing as a sole countermeasure. Second, it is likely that the Dridex operators were already aware of the impending law enforcement operation, taking into account the earlier arrest of one of their operators in August 2015, the preceding sinkhole operation by the National Crime Agency in September 2015, and the fact that the US authorities had already served four of the other Dridex operators notices of indictment [2]. As a result, and perhaps in retaliation, the botnet operators may have increased their activities and/or moved their operations elsewhere, both of which would lead to an overall increase in network activity during this period.

The second observation of interest is that we see significantly increased usage of download URLs with a `.com` suffix (Figure 2(a)) and an increased usage of servers hosted in the US (Figure 2(d)). Likewise, we see similar (albeit less significant) increases in URLs with `.uk` suffixes and GB-based servers. Given that US law enforcement (along with that of the UK) were the driving force behind the Dridex takedown efforts, this increased usage of US-based (and to a lesser extent, GB-based) servers and domains could again be indicative of a concerted response by the Dridex operators. Specifically, the malware operators could have been targeting US infrastructure and end-users in reaction to their takedown attempts. At the same time, without any additional data, one cannot rule out the possibility that the Dridex operation had a significant dependence on US infrastructure prior to these takedown efforts, which would mean that they could just be attempting to recover lost ground. Nonetheless, it is clear that these malware operators ramped up

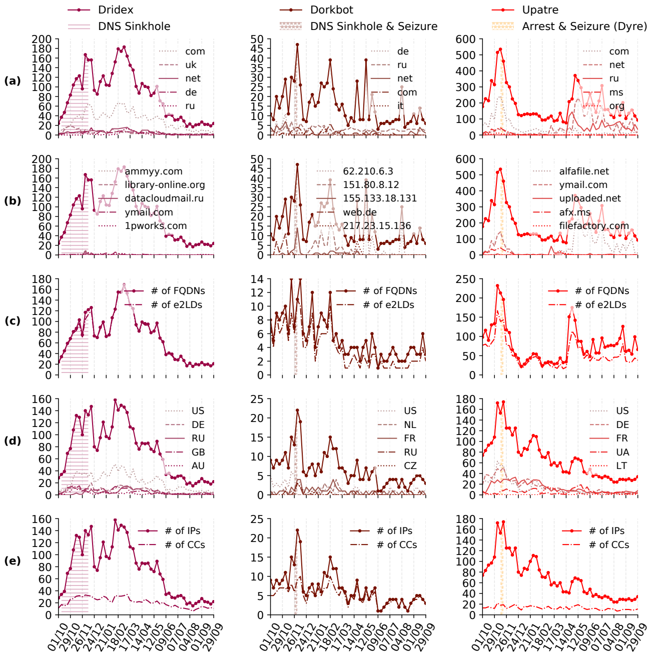

Figure 2: Aggregate network activity: (a) # of URLs used and top 5 TLDs; (b) # of URLs used and top 5 e2LDs/IPs; (c) # of FQDNs and # of e2LDs; (d) # of IPs used and top 5 hosting countries; and (e) # of IPs and # of hosting countries. Dridex exhibits consistent growth in network activity during the DNS sinkhole, while Dorkbot and Upatre both exhibit significant, short-term drops in network activity after their respective takedowns with varying long-term responses.

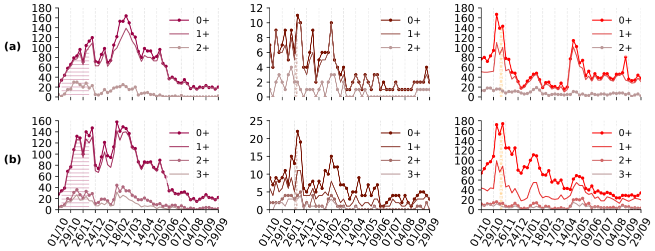

Figure 3: Evasion indicators: (a) # of e2LDs associated with N+ IPs; and (b) # of IPs associated with N+ e2LDs. Dridex was found to use shared-hosting platforms and CDNs often. Upatre increases its use of IPs with 2+ domains from mid-April, most of which were for `.ru` DGA domains.

their operations at the same time that law enforcement were launching a counter-operation against them, culminating in significantly increased network activity over the ensuing months.

It is also interesting to note that the Dridex operation did not rely on any one download server or region. This is shown by the low proportion of download activity via the most commonly used domains (e.g., ammyy.com, library-online.org – see Figure 2(b)). This is also reflected in the approximate 1:1 ratio in # of FQDNs-to-# of e2LDs attributed to its download servers (Figure 2(c)). Similarly, as Figure 2(e) shows, up to 35 different countries are used to host Dridex download servers. Querying the data, we find that the Dridex operation makes significant use of (i) websites on shared-hosting platforms, and (ii) multi-region CDNs (such as dropbox.com or googleusercontent.com) as malware delivery vectors. This accounts for the distributions of domains using multiple IPs and IPs using multiple domains (see Figures 3(a)–(b)). This diversity in distribution channels makes it difficult to identify bottlenecks in the Dridex operation. This could have been implemented by design, or a learned adaptation to previous takedown attempts.

Finally, as we see in the second era of its network activity, the Dridex operation appears to "wind down" its server usage just as quickly as it grew in the preceding months. This reduced server usage seems to stabilise from around 4th August 2016. Without additional data, it is difficult to draw any robust conclusion on what causes this reduction in activity (e.g., whether it was a consequence of a takedown operation, or a conscious decision by the malware operators to reduce operational activity).

6.1.2 Dorkbot. On a general note, the network activity of the Dorkbot operation appears to be varied and highly stochastic in clear contrast to the other malware operations. It also appears that the Dorkbot operation is significantly less diverse in its use of download servers, as indicated by its use of fewer unique URLs, domains, and IPs in the Dorkbot delivery operation. Further, we previously noted the sharp decline in Dorkbot's overall network activity just after the DNS sinkhole and seizure counter-operation. However, due to its stochastic nature, it is difficult to determine the significance of this decline as Dorkbot exhibits erratic changes in network activity, both before and after the takedown operation.

Analysing its network dynamics more closely in Figures 2(a)–(b), Dorkbot's overall use of download/redirection URLs shows some cyclicity. Specifically, we observe peaks in the number of URLs used roughly every 12 weeks. A similar pattern is observed with its use of IPs, as shown in Figure 2(d)–(e), albeit with a more pronounced, downward trend. It must be said that this pattern does not appear in Figure 2(c), which shows Dorkbot's (equally stochastic) use of domains gradually decaying for a few months before oscillating around a reduced baseline. Looking at its use of top e2LDs/IPs in Figure 2(b), it is clear that these peaks in URL and IP activity are linked. Particularly, the Dorkbot operation tends to rotate between specific server IPs to spearhead its network-based delivery activities: initially, it primarily uses web.de (a German TLD) between 1st October–12th November, then it briefly moves to 155.133.18.131 (a server in Poland) between 12th November–17th December, traversing the takedown period. Afterwards, it moves to 151.80.8.12 (a server in France) from 24th December–31st March, before briefly switching to 217.23.15.136 (a server in Netherlands) from 31st March–5th May, before fluctuating in its use of 62.210.6.3 (a server in France) from 5th May–11th August.

This pattern of displacement in Dorkbot's server usage appears to be highly coordinated, although the cause or purpose of this behaviour remains unclear. It could be that the Dorkbot operators were changing servers to beat blacklisting services, or for some financial benefit. However, whatever the cause, it is difficult to attribute this patterned behaviour to the takedown operation. As the data shows, Dorkbot had already begun to rotate between servers just before the takedown occurred. Even if the takedown was a factor, this rotating behaviour could also have been part of Dorkbot's distributed delivery architecture [7], and perhaps the reason for its apparent resilience to the takedown attempt. It should be noted that this (slow) rotation between servers is not the same as Fast Flux, the latter of which involves a single domain rotating between multiple IP addresses in a short period of time (e.g., within minutes).

Notwithstanding, we also observed Dorkbot domains that flux between several IPs per day, such as masterhossting7772.in and superstar7747.pw. Given that online sources have identified these domains as malicious,[3] it is likely that these servers used Fast Flux.

Beyond its heavy use of particular IP addresses, the Dorkbot operation also utilises some domains from a mix of regions, as shown in Figures 2(a) and 2(d). This spread of servers is consistent with other research that identified the Dorkbot C&C infrastructure to be distributed among a number of intercontinental regions [7]. Given that the Dorkbot operation only used a few, particular servers to spearhead its delivery activities, it is probable that these other servers were held in reserve as back-up infrastructure.

6.1.3 Upatre. The Upatre operation also exhibits an interesting progression of network activity, which, like the Dridex operation, can also be divided into a number of distinct stages, depending upon which network characteristic one is focusing.

In general, the Upatre operation experiences a rapid increase in network activity in the first few weeks (1st October–12th November) up until the arrest and seizure is carried out against the Dyre operation. Specifically, looking at Upatre's use of download URLs in Figures 2(a)–(b), we see that during this period the Upatre malware tends to operate through download URLs with .com (and to a lesser extent, .ms) suffixes. The most common effective second-level domains that it uses in this period are ymail.com (Yahoo! Mail) and afx.ms, which is a domain registered by Microsoft Corporation and known to be associated with Outlook Mail.[4] This is consistent with the observation that Upatre is often delivered to victims through malicious email attachments [4, 32, 50]. During the same period, we observe Upatre's varied and progressive use of IPs from different countries, led by its use of servers in the United States, Germany (DE), France, and Ukraine (UA), as shown in Figure 2(d). It is also interesting to note that, as we see in Figure 2(e), the Upatre operators ensure that their delivery servers are distributed among a number of countries. Clearly, the Upatre operation was distributed through servers across multiple geographic regions, such as edge CDN servers for email services. A simple query of the data confirms this as we find Upatre malware being linked to hundreds of region-specific subdomains of various email servers

[3] https://www.malwareurl.com/ns_listing.php?as=AS45945
[4] https://whois.domaintools.com/afx.ms

during this early period, such as {region}{integer}.afx.ms or email{integer}.secureserver.net.

After the takedown operation, Upatre's network activity rapidly decreases over a number of weeks (12th November–24th December). As security researchers have noted [4, 32, 50], the Dyre malware heavily relied on the Upatre dropper malware as its main infection vector. As such, the taking down of the Dyre operation could have plausibly led to some reverberations in the Upatre operation, perhaps due to some infrastructure being shared between the two. However, as we will later see, this drop in Upatre network activity corresponds to a drop in Upatre binaries being downloaded onto victim computers. Therefore, it remains unclear what causal links could exist between the takedown of the Dyre operation (an Upatre payload) and the subsequent drop in Upatre downloads (a dropper predominantly delivered through malicious email attachments). Likewise, the question also remains: what infrastructure could have been shared between the two operations?

As time goes on, we observe contrasting behaviours between Upatre's use of download URLs/domains and its use of IPs. Namely, Upatre's use of IPs has a downward trend over the ensuing months (24th December–29th September), as shown in Figures 2(d)–(e). On the other hand, as Figures 2(a)–(c) show, its use of download URLs and domains is quite stable for the first few months (24th December–14th April), but then suddenly increases and fluctuates at a raised level (14th April–29th September). This behavioural disparity between Upatre's use of IPs and its use of download URLs/domains is interesting. In particular, we observe a transition from the two metrics being quite strongly correlated at one stage (i.e., their correlated peak and trough between 1st October–24th December) to them becoming increasingly incongruent as time goes on.[5] This could indicate a significant change in Upatre's upstream delivery infrastructure some point after the Dyre takedown operation, such as a move from a distributed architecture to a more centralised one. We find some evidence to support this hypothesis. First, in Figures 2(a)–(b), we observe clear displacement in the Upatre operation from one set of domains to another: particularly from sites with .com and .ms TLDs (such as *.ymail.com and *.afx.ms) to those with .net and .ru suffixes (such as *.alfafile.net). Second, as we see in Figure 3(b), from around 14th April we observe an increase in the use of IPs that are associated with 2+ e2LDs, corresponding to Upatre's migration to the .net and .ru domains. Upon further inspection, these new servers (particularly those with .ru suffixes) were most likely generated by a DGA. For instance, on 28th April, we identified 139 domains with a common domain structure: a static keyword for the subdomain, a random sequence of words and numbers for the second-level, and the .ru TLD (e.g., slingto.scene-root85.ru, slingto.robbusymyself.ru, and slingto.hanghandle.ru). Furthermore, these domains were all clustered around the same set of IPs, some of which involved over 10 different e2LDs per cluster. We also note Upatre's heavy use of the alfafile.net (a file-hosting platform) and its various subdomains around this time, apparently replacing the email services and CDNs that it relied on several months before. This marked change in delivery infrastructure by the Upatre operators shows a complete change in their *modus operandi* (i.e., from using compromised email services to using malicious domains with DGA as infection vectors), and could very well be evidence of an adaptation to previous takedowns.

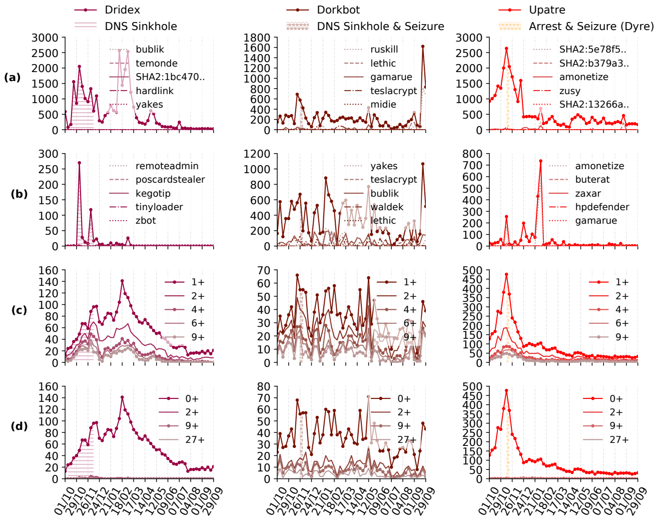

Figure 4: Aggregate download activity: (a) # of times downloaded; (b) # of drops by target malware; (c) # of SHA-2s downloaded N+ times; (d) # of SHA-2s that drop N+ files. Bursts of dropping activity by Dridex (during takedown) and Upatre (after takedown). Dorkbot activity more consistent throughout the year except for the sudden increase at the end. N.B: a few binaries are responsible for the majority of download activity (an approximate Power law relationship).

6.2 Downloader Dynamics

In the last section, we analysed the network-level dynamics pertaining to each of the three malware delivery operations under study. In this section, we move our analysis on to the characteristics and download activities of the malicious binaries themselves, which are fundamental to malware delivery operations. In particular, we juxtapose the aggregate downloader dynamics, familial relationships (parent, children), delivery tactics, and polymorphic behaviours of the three malware operations. Figure 4 shows the aggregate download dynamics of each malware operation, while Figure 5 shows their relational dynamics (i.e., # of parent and child files), Figure 6 shows indicators of distributed delivery tactics, and Figure 7 indicators of polymorphic behaviour by the binaries.

6.2.1 Aggregate download activity. Figures 4(a)–(b) show the aggregate downloads and dropping activities of the Dridex, Dorkbot, and Upatre malware, whereas Figures 4(c)–(d) show the distributions of files that are downloaded N+ times, or drop N+ files. One notices similar download behaviours between the Dridex and Upatre malware, but significantly different behaviours from Dorkbot. This becomes a recurring theme in our analysis of download activities.

For the ***Dridex*** malware, we observe "bursts" of downloads and dropping activity during the takedown counter-operation, and

[5] Pearson's and Spearman's correlation coefficients were computed for the Upatre IP count vs. FQDN count during three periods (inclusive): 1st October–24th December, 31st December–14th April, 21st April–29th September. (r, ρ) as follows: Oct_Dec(0.76, 0.60), Dec_Apr(0.63, 0.45), Apr_Sep(0.41, 0.11).

resurgence of (just) download activity between 11th February–11th March, in correspondence with the peak in its network behaviours around the same time. This supports the notion that the Dridex operators expanded their operation during the law enforcement takedown, perhaps in anticipation of (or in retaliation to) the expected disruptions due to the DNS sinkhole. It is worth noting that that 95.8% of the files dropped by Dridex between 29th October–24th December were unclassified. Nonetheless, we identify a few instances of known malware families being delivered by Dridex, including some backdoor malware (`farfli`, `tinyloader`), financial fraud trojans (`zbot`, `zusy`, `poscardstealer`), among others (`troldesh`, `yakes`, `kegotip`). It is difficult to draw any formidable conclusions on this aberrant behaviour given the lack of ground truth on the files dropped by the Dridex malware. Still, it is interesting to see Dridex - a financial fraud trojan known to operate only as a payload rather than a dropper - suddenly engage in this practice of diversified, downstream malware delivery. Looking at Figure 4(c), it appears (at least, visually) that the Pareto principle applies to the frequency of downloads for each Dridex file, where the majority are only downloaded once while decreasing proportions of files are downloaded more frequently. On the other hand, as we see in Figure 4(d), almost none of the Dridex binaries engage in dropping activities. Rather, through querying the data, we find that only up to 3 binaries are responsible for all dropping activity on any given day. This supports the notion that the Dridex malware was primarily designed to operate as a malicious payload rather than an intermediate dropper. However, specific strains of this malware were clearly modified to drop other malware onto victim systems.

With the ***Upatre*** malware, we observe similarities to that of the Dridex malware. As we see in Figure 4(a), and much like its network activity, we observe a peak in Upatre downloads just before the arrest and seizure counter-operation around 19th November. We also observe several "bursts" of Upatre dropping activity in Figure 4. Of the files that Upatre drops, we find that on 12th November, 60% were PUP (mostly `convertad`) and 23% malware; on 24th December, 98% were unclassified; and between 28th January–4th February, 77% were PUP (mostly `amonetize`) and 3% malware. It is interesting to see that such a high proportion of Upatre payloads are PUP such as `convertad` and `amonetize` (as opposed to other malware), which are families known to bundle and integrate with legitimate software.[6] This case study gives an indication of how convoluted file dependencies and delivery chains between malware, PUP, and benignware can be in the wild. As we look at the bounded frequency plots of downloads per SHA-2 and drops per SHA-2 in Figures 4(c)–(d), we see a similar case as with the Dridex malware: (i) an apparent, inverse relationship between SHA-2 count and the frequency in which each SHA-2 is downloaded; and (ii) a minority of files being responsible for all of the Upatre's dropping activity. The latter observation is more strange in this case, given that Upatre is known to operate mainly as a dropper malware. More generally, we find that Upatre is downloaded more frequently than it downloads other files within this observation window.

Analysing the ***Dorkbot*** malware, we observe significantly different download behaviours than the other malware families. First, as we see in Figures 4(a)–(b), the download and dropping dynamics of the Dorkbot operation do not appear to change significantly over the course of the year (including the takedown period), barring a sudden increase at the end of the observation period. We previously noted that it was difficult to attribute Dorkbot's ever-changing network behaviours to the takedown counter-operation. The lack of any significant change in Dorkbot's overall download activity over the observation period seems to support this position even further. In Figure 4(c), the plots of downloads per SHA-2 for the Dorkbot malware show a generally "flatter" distribution between each group (i.e., more evenly spaced plots for $N = 1, 2, 3, \ldots$). This seems to indicate a weaker Pareto distribution (if any) in comparison to the other operations. The Dorkbot operation is also differentiated by its higher proportion of file SHA-2s that engage in dropping behaviour. Specifically, in Figure 4(d), while most do not engage in any dropping behaviours, up to 40% of Dorkbot SHA-2s deliver 9+ subsequent payloads over the course of the observation period.

6.2.2 Relational dynamics. In Figure 4 we observed the aggregate download activity of the three malware operations under study. It is also important to understand the other software families that contribute to this activity, either as droppers (i.e., parent files that download the target malware), or as payloads (i.e., child files that are dropped by the target malware). In particular, Figures 4(a)–(b) show the top 5 labelled software that either download the target malware (parent files) or are downloaded by the target malware (child files). In most cases, we see that these "top" families account for a very small percentage of the overall download activity of the target families. The exception to this appears to be the case of the ***Dorkbot*** operation, where in Figure 4(a) we see a sharp increase in `ruskill` downloads towards the very end of the observation window, while in Figure 4(b) we see that the `yakes`, `teslacrypt`, and `bublik` malware families account for most of Dorkbot's dropping activities.

Turning to the question of how many families are related to the studied malware, Figure 5 shows the aggregate number of families involved in each malware operation. For the ***Dridex*** operation, Figure 5(a) shows very few upstream malware distributing it during the year. This implies that the Dridex operation relied more on server delivery infrastructure than dropper malware, which is consistent with other observations of this malware being delivered through malicious email attachments and exploit kit downloads [5].

The ***Dorkbot*** behaves very differently. As Figure 5(a) shows, the Dorkbot malware relies consistently (of a cyclic nature) on upstream malware droppers. Particularly up until the takedown, Dorkbot was delivered by malware such as `gamarue`, `kasidet`, and `yakes`. However, after the takedown, the number of upstream malware in the Dorkbot operation dropped significantly, though, as previously noted, it's overall download activities seemed unaffected for the most part. Given the lack of ground-truth in this regard, it is difficult to ascertain whether the takedown only affected a subset of the Dorkbot operation (i.e., upstream dropper networks). In like manner, we see that Dorkbot also distributed a wide range of downstream malware throughout the observation period. Again, one cannot see any sign of diminished activity due to the takedown.

The ***Upatre*** operation also exhibits some interesting relational behaviours. In particular, as Figure 5(a) shows, Upatre relies mostly on a few families in the first half of the observation window, such as the `amonetize` PUP and `gamarue` malware. However, in the second

[6] https://www.shouldiremoveit.com/ConvertAd-88792-program.aspx

half of the observation window, we see a significant change in behaviour: Upatre shifts to a diversified, upstream dropper network, as indicated by (i) a large increase in the total upstream families, and (ii) the "top" families (e.g., `loadmoney`) accounting for only a small proportion of them. Though it is unclear why the cause of this change, we note it occurring from 14th April onwards – the same time Upatre began to use DGA download servers (see Section 6.1.3).

6.2.3 Distributed delivery tactics. Figure 6 shows distributed delivery metrics of each malware operation: the numbers of SHA-2s associated with varying numbers of URLs, e2LDs, and IPs. Again, we observe similarities in the Dridex and Upatre operations, but considerably different characteristics in the Dorkbot operation.

Starting with the bounded frequency plots of ***URLs per SHA-2*** in Figure 6(a), we see that almost all the download activities of the Dridex and Upatre SHA-2s involve network activity, as indicated by the near-total overlap of the plot lines for $N = 0$ and $N = 1$. This is in stark contrast to the Dorkbot malware, which shows significant "gaps" between the $N = 0$ and $N = 1$ plot lines, indicating that some files are not associated with any download URL. This could allude to Dorkbot writing directly to the filesystem from the malicious process as opposed to initiating the download from an external server. This is consistent with reports which identified spreading through USB flash drives as one of Dorkbot's infection vectors [8]. It is still possible, however unlikely, that this discrepancy could be due to some measurement error in the data collection process. Nonetheless, we see that SHA-2s being associated with multiple URLs is a common occurrence for these malware operations (although relatively less common for the Dorkbot operation).

Figure 6(b) shows the bounded frequency plots of ***e2LDs per SHA-2***, while Figure 6(c) shows ***IPs per SHA-2.*** Most of the Dridex malware is associated with at least one e2LD or an IP, while up to 50-60% of its files are associated with 2+ e2LDs/IPs. It is particularly interesting to see that the highest proportion of files associated with multiple e2LDs/IPs occurred during the takedown period. Again, this supports the notion that a concerted effort was made by Dridex operators to ramp up malware activity during the sinkhole operation. The Upatre operation exhibits significantly different characteristics: the proportion of its files that are associated with 1+ e2LDs/IPs is highly variable across the year. For instance, between 1st October–24th December, there is a significant evolution in its delivery patterns: (i) a sudden rise and fall in files associated with 1+ e2LDs, and (ii) at one point, the majority of files having no traceable IP. It remains unclear why Symantec's telemetry could not detect IPs for these download events, or why these files were prominent only in the early part of the observation window. Nonetheless, it is unlikely this was a random occurrence, given these correlated behaviours. Finally, the Dorkbot malware exhibits much of the same delivery patterns as before: a significant (but still minor) proportion of its files are not linked to any server. This alludes to binaries writing directly onto victim filesystems.

6.2.4 Polymorphism. Figure 7 shows the polymorphic characteristics of each malware operation. The ***number of active SHA-2s*** (or malware variants) and ***churn rates*** for each malware are shown in Figure 7(a). Clearly, each of the malware delivery operations makes extensive use of polymorphism during the observation window. Furthermore, we see that the active SHA-2 count of each malware

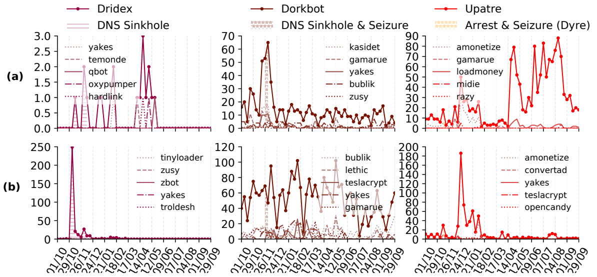

Figure 5: Relational dynamics: (a) # of SHA-2s that download target malware; and (b) # of SHA-2s dropped by target. N.B: the sharp increase in Upatre upstream droppers after mid-April, correlating with its increased use of DGA servers.

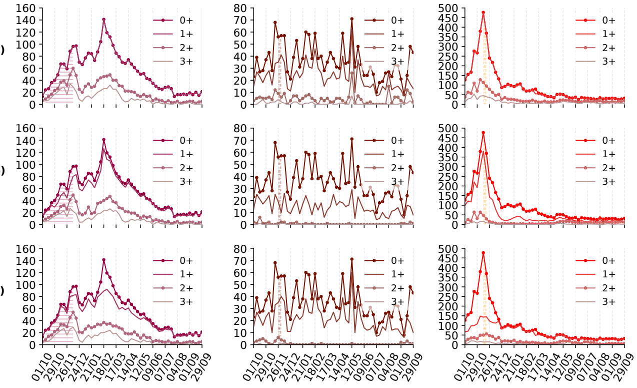

Figure 6: Distributed delivery indicators: (a) # of SHA-2s associated with N+ URLs; (b) # of SHA-2s associated with N+ e2LDs; and (c) # of SHA-2s associated with N+ IPs. Dorkbot downloads often without any traceable network resource, alluding to direct writing to filesystems.

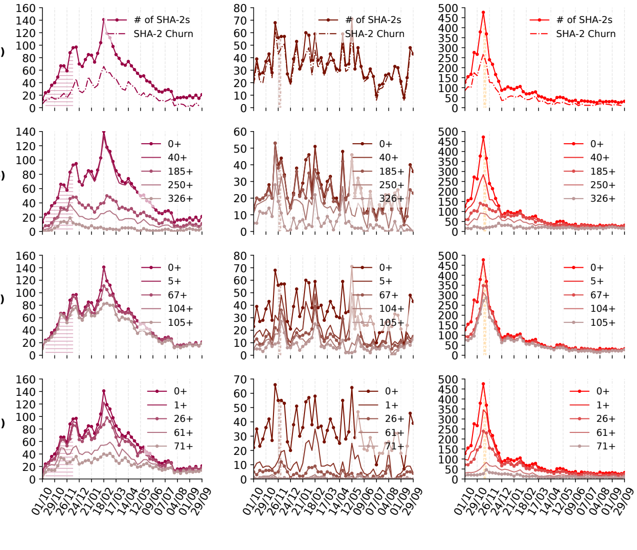

Figure 7: Polymorphic characteristics: (a) # of active SHA-2s and SHA-2 churn; (b) # of SHA-2s of size N+ KB; (c) # of SHA-2s with M+ malice score, where $0 \geq M \geq 128$; and (d) # of SHA-2s with P+ prevalence score, where $0 \geq P \geq 127$. N.B: malice and prevalence scores are assigned by Symantec.

	Dridex	***Dorkbot***	***Upatre***
LEA take-down	60-day DNS Sinkhole and Disinfection.	DNS Sinkhole and Seizure.	Arrest and Seizure.
Malware operation behaviours	• Malware operations increase and diversify during first half of observation window (including LEA takedown). Gradually decreases in second half of window. • Distributed delivery architecture: significant use of shared-hosting platforms and multi-region CDNs. • Sparse bursts of dropping activity: delivered other malware including ransomware, banking trojans, backdoors. Uncharacteristic of Dridex malware. • Minority of files responsible for majority of downloads / all dropping activity. • Few upstream droppers; heavy reliance on upstream network infrastructure. • Up to 60% files delivered by 2+ e2LDs/IPs. • Significant polymorphism and churn rate (up to 60%). High detection rates (prevalence/malice scores).	• Highly cyclic/stochastic operational activity. • Distributed delivery architecture: multi-region servers. • Coordinated rotation between servers in different countries over observation window. Likely use of Fast Flux also. • Sharp but brief drop in network activity after LEA takedown. No observable long-term effects. • Potentially held back-up infrastructure. • Slightly "flatter" distribution of download activity across SHA-2s. • Sharp increase in downloads at end of observation window: mainly delivered by `ruskill`. • Consistent reliance on upstream droppers; mixed reliance on upstream network infrastructure. • Broad range of downstream malware dropped. • Likely use of direct writing to file system (e.g., binary replication). • Extremely high polymorphism and churn rate (almost 100%). Low-to-mild detection rates (prevalence/malice scores).	• Rapid, initial increase in operational activity; sharp drop after takedown. • High use of email services (initially) and IPs in multiple regions. • Apparent shift in delivery infrastructure over observation window: distributed to more centralised. • Displacement in domains used (from `.com` and `.ms` to `.ru` and `.net`). • Increased use of DGA servers in latter half of window; corresponding decreased use of mail servers. • Dropped a range of downstream software in bursts: mainly PUP; some malware and unlabelled families. • Minority of files responsible for majority of downloads / all dropping activity. • Relies on a few upstream droppers in first half of window; sudden change and increase of upstream droppers in second half (correlated with DGA usage). • Significant reliance on upstream network infrastructure. • Significant polymorphism and churn rate (up to 80%). Mild-to-high detection rates (prevalence/malice scores).

Table 2: Summary of LEA takedowns and observed behaviours of the targeted malware delivery operations.

evolves much like the network dynamics of its respective delivery operation. For example, the active SHA-2 count for the Dridex operation increases while the DNS sinkhole takes place, and falls some months after; that of Upatre falls sharply after the arrest and seizure occurs (although its network components behave very differently in the second half of the observation window); that of the Dorkbot operation continues to fluctuate in apparent immunity to its respective takedown. This correlation in SHA-2 count and the number of network components used to deliver them (URLs, domains, IPs)[7] could be the result of campaign IDs being hard-coded into each binary, being unique to each upstream distributor. In this case, the binaries delivered by each distributor would have a different file hash. Looking at the churn rates, we see that all of the operations exhibit high churn. Nonetheless, Dorkbot exhibits exceptionally higher churn rates, where almost all its SHA-2s are replaced weekly.

Figure 7(b) shows the distribution of ***file sizes*** (in KB). We observe significant variability in the sizes of each malware, although most SHA-2s are less than 326KB. It should be noted, however, a few binaries as large as 15MB were observed in the data (particularly Upatre binaries). It is unclear whether this variability in file size (or how much of it) is a result of some polymorphic technique (e.g., binary padding), or if it's simply due to additional functionality being coded into certain versions of these malware.

Figure 7(c) shows the distribution of assigned ***malice scores***, while Figure 7(d) shows the distribution of ***prevalence scores***. It is interesting to see that most Dridex and Upatre SHA-2s are assigned very high malice scores with very low variance, while Dorkbot is assigned much more variable malice scores. This suggests that Dorkbot was much more successful than the other malware at evading detection systems such as Symantec and the other antivirus engines used to generate these scores. Likewise, Dorkbot is generally assigned much lower prevalence scores than the other malware. This indicates that the detection systems did not observe Dorkbot malware as frequently at the time. This is most likely the result of Dorkbot's very high churn rate, which could also be a contributing factor to it being assigned significantly lower malice scores.

Summary of Results

In this section, we presented a comprehensive analysis of the network and download activities of three, different malware delivery operations, and how they evolved over the course of a year in light of law enforcement efforts to disrupt them. A summary of these observations is presented in Table 2.

7 DISCUSSION

We conducted a detailed analysis of the dynamics and behaviours of three malware delivery operations over the course of a year. In this section, we take a step back to consider the implications of these findings. Specifically, we identify what the security community can learn from these observations, and how these findings could be factored into future countermeasures. We also reflect on the limitations of this study, and opportunities for future work.

7.1 Lessons Learned

We observed a diversity of structural designs, behaviours, patterns, and responses to takedown attempts in the studied operations, finding the following commonalities between them.

7.1.1 Distributed delivery architectures. All three operations made significant use of distributed delivery infrastructures: Dridex used

[7] Pearson's and Spearman's correlation coefficients were computed for SHA-2 count vs. URL count over 1st October–14th April – the period for which these relationships are approximately linear. (r, ρ) as follows: Dridex$(0.93, 0.93)$, Dorkbot$(0.74, 0.71)$, Upatre$(0.99, 0.93)$.

shared-hosting services and CDNs in up to 35 different countries; Dorkbot constantly rotated between international servers; and Upatre heavily used multi-region CDNs and cloud services (`ymail.com`, `alfafile.net`). This has been observed of malicious file delivery operations in multiple studies [24, 31, 35, 49]. This makes effective server-based takedowns more difficult, thus requiring greater coordination between LEAs, security companies, and service providers on the Internet. Most especially, given that these service providers have been so commonly abused, it is pertinent that they continue to step up their security hygiene and coordination with other stakeholders to prevent cybercriminals from abusing such platforms.

7.1.2 Polymorphism and Pareto's principle. Polymorphism was rigorously employed by all three malware. However, some malware binaries (Dridex, Upatre) were detected more frequently than others (Dorkbot). One possible explanation for this is that a malware (such as Dorkbot) that churns through binaries more frequently would be more difficult to detect in the short-term. We also observed manifestations of Pareto's principle across all malware operations in that a minority of binaries were responsible for a majority of download activity. Although detecting polymorphic malware will be a continued challenge for the security community, this skewed distribution of activity towards a minority of binaries indicates that detecting these "super binaries" would yield the most benefit.

7.1.3 Takedown resilience. Each malware operation responded differently and showed some degree of resilience to takedown. For instance, Upatre shifted to a more centralised infrastructure over several months; Dridex significantly increased its activity *during* the LEA takedown attempt; Dorkbot showed no significant changes, but continued in it's cyclic/stochastic behaviours and likely use of Fast Flux. In view of this, one may ask the age-old question of whether botnet takedowns are *actually* effective? Researchers have found that, historically, the success of botnet takedowns is highly variable [25, 47]. Perhaps a more pertinent question to ask is whether botnet takedowns are the *only* effective means to controlling malware delivery? Granted, there are alternative takedown techniques that could also be employed, such as infiltrating botnet infrastructure and disrupting them from within [17, 27, 46]. However, by viewing malware delivery as a supply chain problem, for example, the security community may achieve more success by targeting other aspects of the malware economy in parallel, such as by attacking the flow of money around malware delivery (the reliability of Dark markets, the process of monetising stolen data and compromised devices, etc). It has also been argued [30] that the security community could elicit more disruptive techniques from other fields of security research. For example, frameworks such as Situational Crime Prevention [22] could be adapted to derive countermeasures against botnet operations [30].

7.1.4 Predictable responses. Environmental criminology literature recognises several types of offender responses to anti-crime interventions. These include (i) *displacement* – a change in an offender's behaviour to circumvent the intervention or seek out alternative targets or crime types [28]; (ii) *adaptation* – a longer-term process of displacement whereby the offender population as a whole discover new crime vulnerabilities and opportunities after an intervention has been in place for a while [26]; and (iii) *defiance* – an increase in offender activity in retaliation to an intervention, usually when the offender perceives the intervention as unjust or disproportionate [43]. Behaviours such as these are usually expected and factored into interventions supported by environmental criminology.

Similarly, in this study, we observed interesting responses from the malware operators to takedown efforts. For instance, the Dridex operators significantly ramped up botnet activity during the DNS sinkhole counter-operation, with an increased concentration of servers in the US and UK. We noted that this was the second or third LEA counter-operation against the Dridex botnet in as many months. Assuming this is linked to the attempted takedowns, this is characteristic of defiant and displacing behaviours. Likewise, we observed significant changes in the Upatre infrastructure only a few months after the Dyre takedown. Particularly, it shifted in its use of multi-region email services to more centralised clusters of DGA servers and a single CDN (`alfafile.net`). Again, this is characteristic of displacement, potentially to regain more control of the malware delivery process.

As such, the main takeaway here is that, much like crime in the physical world, reactions from the malware operators must be expected and factored into any mitigation strategy against their operations. This highlights the importance of two things: first, the continued monitoring and management of malware operations, before, during, and after any takedown attempt (e.g., assessing the potential for unwanted side-effects [21], implementing action-research models for botnet takedowns [30]); and second, the necessity for security researchers, companies, and LEAs to disseminate information regarding botnet takedown attempts, as this shared body of knowledge would better equip the security community to implement effective countermeasures. Nonetheless, there is the argument that cybercriminals could also learn how to make their operations more resilient through this shared knowledge. This raises the question of how best to implement such knowledge-sharing.

7.1.5 Unpredictable responses. At the same time, we also observed very *aberrant* and previously undocumented behaviours by each malware operation. For instance, though Dridex is a financial fraud trojan and has been known to operate as a payload, we observed it engaging in bursts of dropping activity, delivering downstream ransomware, backdoor malware, and even competing families of financial fraud trojans! Dorkbot exhibited sudden and sharp increases in downloads at the end of the observation period through upstream `ruskill` malware. Upatre suddenly and significantly increased in its use of upstream malware droppers in the latter half of the observation period. Such behaviours could be very difficult to predict, especially when monitoring malware activity from a limited perspective (i.e., download traffic). This highlights the need for the security community to incorporate data sources from multiple ecosystems to monitor botnet activity effectively. For instance, monitoring download traffic (as in this study) could be complemented with other intelligence sources, such as network traffic from ISPs, online discussions in social media and web forums (Twitter, Reddit), as well as discussions and market activity in the Dark Web. Potentially, using multiple perspectives could give researchers more context and clarity regarding some of these observed behaviours.

7.2 Limitations

This work builds on the data and techniques used in a previous measurement study of the malicious file delivery ecosystem [31]. As such, the same data limitations apply, such as the limited view we have on only one stage of the malware supply chain (software download), or VirusTotal's limited coverage in mappings between file hashes and malware families. To mitigate the former issue, we used additional data sources to provide as much context as possible (ground truth on the operations, VirusTotal/AVClass/NSRL software labels, malware aliases, etc). To mitigate the latter issue, we collected VirusTotal labels for a period of three years after the initial observations, maximising positive predictive performance. It is still possible that some files were mislabelled with the wrong malware family, which would mean that the time series analytics is unrepresentative of the given family. However, we suspect such cases would be few given the reported accuracy of the classifier [42].

A major part of this study involved analysing malware delivery operations that were subject (or in the case of Upatre, linked) to a takedown attempt. However, a number of challenges arise. One challenge relates to the fact that ground truth on takedown operations is usually scarce. This was the case with the operations studied herein. As such, this study is limited regarding the specifics of each takedown operation, and finding parallels in the data. More generally, and as a result of this general lack of ground truth data on takedown operations, this study was scoped as a measurement study of global malware activity. This means that we are only able to observe and evaluate the overall structure and activities of each malware operation but cannot do more than speculate why such phenomena occur, nor can we isolate observable effects to the specific parts of each infrastructure that were targeted for takedown. In light of this challenge, one interesting extension to this work could be the use of a causal inference framework to analyse the effects of takedown attempts on different aspects of each malware operation (aggregate network and download activity, distributed delivery, etc), as well as the wider malicious file delivery ecosystem. Alternatively, causal relationships could be uncovered more directly with additional ground truth on the specifics of each takedown operation. Another, more general challenge is the issue of *survivorship bias*. In the context of this work, this refers to the biases that arise out of the fact that certain characteristics of the studied botnets would make them more likely to be targeted for takedown than other botnets. Such biases ultimately threaten the external validity of these findings (i.e., how well they apply to other botnets, particularly those not targeted for takedowns).

Finally, on the topic of understanding the behaviours of the malware operators, it is also worth noting that we could only observe *spatial displacement* in this study (i.e., an operator moving from one set of upstream servers and dropper networks to another). The methodology could be extended to include *ecosystem dynamics* that could allow us to observe *offender displacement* (i.e., a malicious operator replacing another's use of upstream delivery infrastructure).

8 CONCLUSION

In this study, we tracked and analysed three different malware delivery operations over the course of a year, studying the dynamics of their upstream servers and dropper networks. We made a number of key findings – mainly, the tendency of malware operators to move their operations elsewhere after a takedown, or in one case, to openly defy it. We also found the use of distributed delivery architectures (particularly CDNs) and the heavy reliance on a few "super binaries" to be common by the studied malware operators. These observations give the security community deeper insight into the complexities of malware delivery and ought to be factored into future takedown strategies.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their comments on this work. Colin C. Ife was supported in part by EPSRC under grant no. EP/M507970/1. Steven J. Murdoch is supported by The Royal Society under grant no. UF160505. Gianluca Stringhini is supported by NSF under grant no. CNS-1942610.

REFERENCES

[1] [n.d.]. Botnet activity in H1 2018: Multifunctional bots becoming more widespread | Kaspersky Lab. https://www.kaspersky.com/about/press-releases/2018_botnet-activity-in-h1-2018-multifunctional-bots-becoming-more-widespread. Accessed: 2018-12-01.

[2] [n.d.]. Bugat Botnet Administrator Arrested and Malware Disabled — FBI. https://www.fbi.gov/contact-us/field-offices/pittsburgh/news/press-releases/bugat-botnet-administrator-arrested-and-malware-disabled. [Accessed online: 11-September-2020].

[3] [n.d.]. Endpoint Protection - Symantec Enterprise (Dyre). https://community.broadcom.com/symantecenterprise/communities/community-home/librarydocuments/viewdocument?DocumentKey=82c547f6-ce80-4fe6-b055-f64c962158d8&CommunityKey=1ecf5f55-9545-44d6-b0f4-4e4a7f5f5e68&tab=librarydocuments. [Accessed online: 11-September-2020].

[4] [n.d.]. Endpoint Protection - Symantec Enterprise (Dyre). https://community.broadcom.com/symantecenterprise/communities/community-home/librarydocuments/viewdocument?DocumentKey=82c547f6-ce80-4fe6-b055-f64c962158d8&CommunityKey=1ecf5f55-9545-44d6-b0f4-4e4a7f5f5e68&tab=librarydocuments. [Accessed online: 11-September-2020].

[5] [n.d.]. Trojan.Dridex. https://blog.malwarebytes.com/detections/trojan-dridex/. [Accessed online: 11-September-2020].

[6] [n.d.]. VirusTotal. https://www.virustotal.com.

[7] [n.d.]. White hats, FBI and cops team up for Dorkbot botnet takedown. https://www.theregister.com/2015/12/04/dorkbot_botnet_takedown/. [Accessed online: 11-September-2020].

[8] [n.d.]. Win32/Dorkbot threat description - Microsoft Security Intelligence. https://www.microsoft.com/en-us/wdsi/threats/malware-encyclopedia-description?Name=Win32%2fDorkbot. [Accessed online: 11-September-2020].

[9] 2015. Bugat Botnet Administrator Arrested and Malware Disabled. https://www.justice.gov/opa/pr/bugat-botnet-administrator-arrested-and-malware-disabled. [Accessed online: 11-September-2020].

[10] 2015. Cloud power disrupts global malware (Dorkbot). https://blogs.microsoft.com/on-the-issues/2015/12/17/cloud-power-disrupts-global-malware/. Section: Microsoft on the Issues.

[11] 2015. Dorkbot botnets disruption. https://www.cert.pl/en/news/single/dorkbot-botnets-disruption/. [Accessed online: 11-September-2020].

[12] 2015. News from the Dorkside: Dorkbot botnet disrupted. https://www.welivesecurity.com/2015/12/03/news-from-the-dorkside-dorkbot-botnet-disrupted/. [Accessed online: 11-September-2020].

[13] 2018. Upatre Continued to Evolve with new Anti-Analysis Techniques. https://unit42.paloaltonetworks.com/unit42-upatre-continues-evolve-new-anti-analysis-techniques/. [Accessed online: 11-September-2020].

[14] Moheeb Abu Rajab, Jay Zarfoss, Fabian Monrose, and Andreas Terzis. 2006. A Multifaceted Approach to Understanding the Botnet Phenomenon. In *Proceedings of the 6th ACM SIGCOMM Conference on Internet Measurement (IMC '06)*. ACM, New York, NY, USA, 41–52. https://doi.org/10.1145/1177080.1177086

[15] Manos Antonakakis, Roberto Perdisci, Yacin Nadji, Nikolaos Vasiloglou, Saeed Abu-Nimeh, Wenke Lee, and David Dagon. 2012. From throw-away traffic to bots: detecting the rise of DGA-based malware. In *USENIX Security Symposium*.

[16] Ulrich Bayer, Imam Habibi, Davide Balzarotti, Engin Kirda, and Christopher Kruegel. 2009. A View on Current Malware Behaviors.. In *LEET*.

[17] Hamad Binsalleeh, Thomas Ormerod, Amine Boukhtouta, Prosenjit Sinha, Amr Youssef, Mourad Debbabi, and Lingyu Wang. 2010. On the analysis of the zeus

botnet crimeware toolkit. In *Privacy Security and Trust (PST)*.

[18] Thomas Brewster. [n.d.]. Russian Cops Bust Key Members Of World's Busiest Cybercrime Gang: Sources (Dyre). https://www.forbes.com/sites/thomasbrewster/2016/02/08/russia-arrests-dyre-malware-masterminds/. [Accessed online: 11-September-2020].

[19] Juan Caballero, Chris Grier, Christian Kreibich, and Vern Paxson. 2011. Measuring pay-per-install: the commoditization of malware distribution.. In *Usenix security symposium*. 13–13.

[20] Zoe Carpou. 2015. Robots, Pirates, and the Rise of the Automated Takedown Regime: Using the DMCA to Fight Piracy and Protect End-Users. *Colum. JL & Arts* 39 (2015), 551.

[21] Yi Ting Chua, Simon Parkin, Matthew Edwards, Daniela Oliveira, Stefan Schiffner, Gareth Tyson, and Alice Hutchings. 2019. Identifying unintended harms of cybersecurity countermeasures. In *2019 APWG Symposium on Electronic Crime Research (eCrime)*. IEEE, 1–15.

[22] Ronald V. Clarke (Ed.). 1997. *Situational crime prevention: successful case studies* (2. ed ed.). Harrow and Heston, Guilderland, NY. OCLC: 36877499.

[23] Richard Clayton. 2009. How much did shutting down McColo help. *Proc. of 6th CEAS* (2009).

[24] David Dittrich. 2012. So You Want to Take Over a Botnet.... In *Presented as part of the 5th {USENIX} Workshop on Large-Scale Exploits and Emergent Threats*.

[25] Benjamin Edwards, Steven Hofmeyr, Stephanie Forrest, and Michel Van Eeten. 2015. Analyzing and modeling longitudinal security data: Promise and pitfalls. In *Proceedings of the 31st Annual Computer Security Applications Conference*. 391–400.

[26] Paul Ekblom. 1997. Gearing up against crime: A dynamic framework to help designers keep up with the adaptive criminal in a changing world. 2 (Jan. 1997), 249–265.

[27] Birhanu Eshete, Abeer Alhuzali, Maliheh Monshizadeh, Phillip A Porras, Venkat N Venkatakrishnan, and Vinod Yegneswaran. 2015. EKHunter: A Counter-Offensive Toolkit for Exploit Kit Infiltration.. In *Network and Distributed Systems Security Symposium (NDSS)*.

[28] Rene B P Hesseling. [n.d.]. Displacement: A review of the empirical literature. ([n. d.]), 34.

[29] Thorsten Holz, Christian Gorecki, Konrad Rieck, and Felix C Freiling. 2008. Measuring and Detecting Fast-Flux Service Networks. In *Network and Distributed Systems Security Symposium (NDSS)*.

[30] Colin C Ife, Toby Davies, Steven J Murdoch, and Gianluca Stringhini. 2019. Bridging Information Security and Environmental Criminology Research to Better Mitigate Cybercrime. *arXiv preprint arXiv:1910.06380* (2019).

[31] Colin C Ife, Yun Shen, Steven J Murdoch, and Gianluca Stringhini. 2019. Waves of Malice: A Longitudinal Measurement of the Malicious File Delivery Ecosystem on the Web. In *ACM ASIA Conference on Computer and Communications Security*. Association for Computing Machinery.

[32] Brett Stone-Gross and Pallav Khandhar Intelligence, Dell SecureWorks Counter Threat Unit™ Threat. [n.d.]. Dyre Banking Trojan Threat Analysis. https://www.secureworks.com/research/dyre-banking-trojan. [Accessed online: 11-September-2020].

[33] Bum Jun Kwon, Jayanta Mondal, Jiyong Jang, Leyla Bilge, and Tudor Dumitras. 2015. The dropper effect: Insights into malware distribution with downloader graph analytics. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 1118–1129.

[34] Zach Lerner. 2014. Microsoft the botnet hunter: the role of public-private partnerships in mitigating botnets. *Harv. JL & Tech.* 28 (2014), 237.

[35] Chaz Lever, Platon Kotzias, Davide Balzarotti, Juan Caballero, and Manos Antonakakis. 2017. A Lustrum of Malware Network Communication: Evolution and Insights. In *IEEE Symposium on Security and Privacy*.

[36] Yacin Nadji, Manos Antonakakis, Roberto Perdisci, David Dagon, and Wenke Lee. 2013. Beheading hydras: performing effective botnet takedowns. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*. ACM, 121–132.

[37] Yacin Nadji, Roberto Perdisci, and Manos Antonakakis. 2015. Still beheading hydras: Botnet takedowns then and now. *IEEE Transactions on Dependable and Secure Computing* 14, 5 (2015), 535–549.

[38] Terry Nelms, Roberto Perdisci, Manos Antonakakis, and Mustaque Ahamad. 2015. WebWitness: Investigating, Categorizing, and Mitigating Malware Download Paths. In *USENIX Security Symposium*.

[39] Peng Peng, Limin Yang, Linhai Song, and Gang Wang. 2019. Opening the blackbox of virustotal: Analyzing online phishing scan engines. In *Proceedings of the Internet Measurement Conference*. ACM, 478–485.

[40] Daniel Plohmann, Khaled Yakdan, Michael Klatt, Johannes Bader, and Elmar Gerhards-Padilla. 2016. A Comprehensive Measurement Study of Domain Generating Malware. In *USENIX Security Symposium*.

[41] Christian Rossow, Christian Dietrich, and Herbert Bos. 2013. Large-Scale Analysis of Malware Downloaders. In *Detection of Intrusions and Malware, and Vulnerability Assessment (DIMVA)*.

[42] Marcos Sebastián, Richard Rivera, Platon Kotzias, and Juan Caballero. 2016. AV-class: A Tool for Massive Malware Labeling. In *International Symposium on Research in Attacks, Intrusions, and Defenses (RAID)*.

[43] LAWRENCE W. SHERMAN. 1993. Defiance, Deterrence, and Irrelevance: A Theory of the Criminal Sanction. *Journal of Research in Crime and Delinquency* 30, 4 (Nov. 1993), 445–473. https://doi.org/10.1177/0022427893030004006 Publisher: SAGE Publications Inc.

[44] Reza Shirazi. 2015. Botnet takedown initiatives: A taxonomy and performance model. *Technology Innovation Management Review* 5, 1 (2015).

[45] Aditya K. Sood and Richard J. Enbody. 2013. Crimeware-as-a-service—A survey of commoditized crimeware in the underground market. *International Journal of Critical Infrastructure Protection* 6, 1 (March 2013), 28–38. https://doi.org/10.1016/j.ijcip.2013.01.002

[46] Brett Stone-Gross, Marco Cova, Lorenzo Cavallaro, Bob Gilbert, Martin Szydlowski, Richard Kemmerer, Christopher Kruegel, and Giovanni Vigna. 2009. Your botnet is my botnet: analysis of a botnet takeover. In *ACM conference on Computer and communications security (CCS)*.

[47] Brett Stone-Gross, Thorsten Holz, Gianluca Stringhini, and Giovanni Vigna. 2011. The Underground Economy of Spam: A Botmaster's Perspective of Coordinating Large-Scale Spam Campaigns.. In *Workshop on lage-scale exploits and emerging threats (LEET)*.

[48] Gianluca Stringhini, Oliver Hohlfeld, Christopher Kruegel, and Giovanni Vigna. 2014. The harvester, the botmaster, and the spammer: on the relations between the different actors in the spam landscape. In *ACM symposium on Information, computer and communications security (ASIACCS)*.

[49] Gianluca Stringhini, Yun Shen, Yufei Han, and Xiangliang Zhang. 2017. Marmite: Spreading Malicious File Reputation Through Download Graphs. In *Proceedings of the 33rd Annual Computer Security Applications Conference*. ACM, 91–102.

[50] zemana.com. [n.d.]. What is Dyre and does Zemana protect me from it? - Zemana. https://www.zemana.com/removal-guide/dyre-malware-removal. [Accessed online: 11-September-2020].